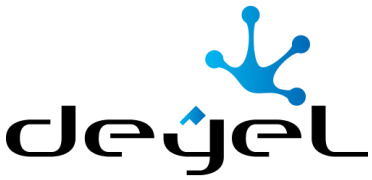


Documentación del Producto

Manual del Usuario

Deyel 7.8

CONTENIDOS

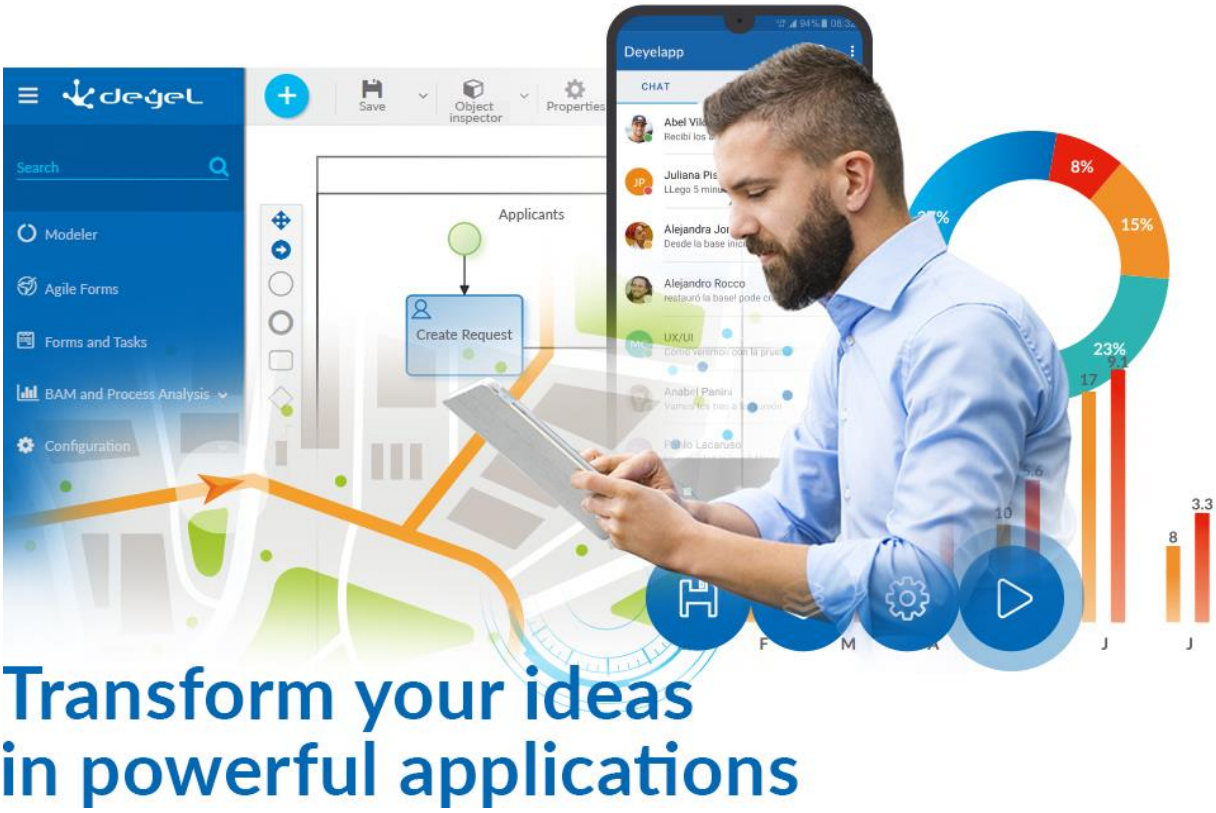
1. Welcome to Deyel 5
2. About Deyel 5
3. User Manual 9
   3.1. Architecture 9
      3.1.1. Plan de Recuperación de Desastre (DRP) 11
   3.2. General Concepts 16
   3.3. Examples of Use 24
   3.4. User Portal 28
      3.4.1. Acceso al Portal 28
      3.4.2. ¿Olvidó su Contraseña? 30
      3.4.3. Menú 30
      3.4.4. Barra de Herramientas Superior 33
      3.4.5. Breadcrumb 39
      3.4.6. Tableros 39
      3.4.7. Formularios y Tareas 43
      3.4.8. Tareas 46
      3.4.9. Casos 58
      3.4.10. Calendarios Gráficos 72
      3.4.11. Grilla de Resultados y Búsqueda 78
      3.4.12. Instancias de Formularios 81
      3.4.13. Elementos de Formularios 91
   3.5. Agile Forms 104
      3.5.1. Conociendo Formularios Agiles 106
      3.5.2. Plantillas 109
      3.5.3. Modelado de Formularios Agiles 125
   3.6. Modeler 156
      3.6.1. Characteristics 158

3.6.2. Object States 164
3.6.3. Processes Modeling 165
3.6.4. Forms Modeling 296
3.6.5. Rules Modeling 351
3.6.6. Modelado de Aplicaciones 449
3.6.7. Widgets Modeling 451
3.6.8. Dashboards Modeling 479
3.6.9. Value Lists Modeling 488
3.6.10. Exportación e Importación 499
3.7. Integration 505
3.7.1. Deyel API Rest 507
3.8. Enterprise Social Network - Tedis 522
3.8.1. Chats 523
3.8.2. Comentarios asociados a Objetos 540
3.8.3. Bots y Comandos 544
3.9. BAM and Process Analysis 548
3.9.1. Process Analysis - Process Behavior 549
3.9.2. Process Analysis - Monthly Duration 549
3.9.3. Process Analysis - Process Activity 550
3.9.4. Process Analysis - Activation Ranking 551
3.9.5. Process Analysis - Case Details 552
3.9.6. Activity Analysis - Activity Behavior 552
3.9.7. Process BAM - Load Analysis 553
3.9.8. Process BAM - Work in Progress 553
3.9.9. Activity BAM - Load Analysis 554
3.9.10. Activity BAM - Work in progress 555
3.10. Configuration 556
3.10.1. Organization 557
3.10.2. Security 562
3.10.3. Localización 617
3.10.4. Ambiente 623

3.10.5. Consolas 673
3.10.6. Herramientas 680
3.10.7. Value Lists 688
3.11. Requirements 690
3.12. On-Premise Installation 692
3.12.1. Estructura de Directorios 695
3.12.2. Instalación en Apache Tomcat 697
3.12.3. Actualización de Versión 700
3.13. Phased Learning 703

1. Welcome to Deyel

Fase 1: Introducción



2. About Deyel

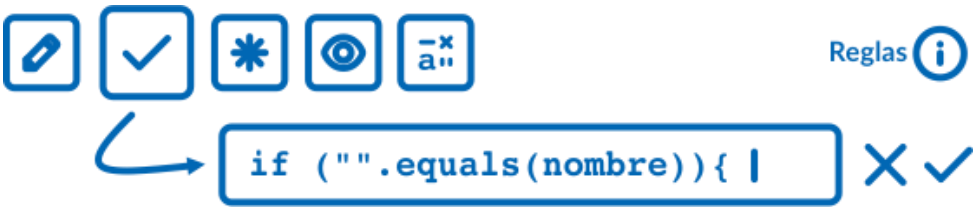
Fase 1: Introducción a Deyel > Low-Code vs No-Code

Deyel is a platform for creating applications. It natively gathers the required components and functionalities required to automate integral processes that impact on the entire company, adapting perfectly to the business journey, reducing the return on investment time and allowing a greater commitment and better experiences.

Do you have a good idea? Dare to make it come true

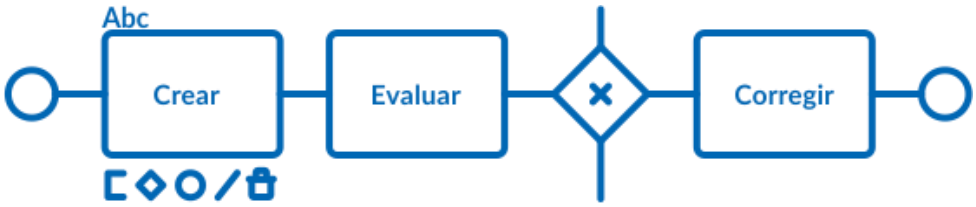
Start creating freely from the form modeler. Model all the forms you use only by selecting, dragging, drop and display. It`s really very simple. A good user experience starts with receptive interfaces that can be used in all type of devices.

Secure the integrity and reliability of your information



We can all write rules and define the behavior of our forms. With the rules wizard of **Deyel**, you can create simple rules for your forms field, validations, obligatory, editability and visibility, even when you don`t have coding skills. If more complexity is needed you count with tools to define advanced rules.

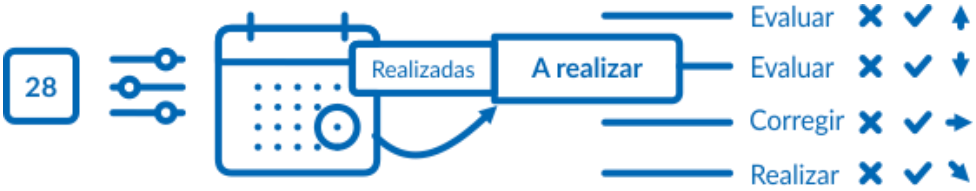
Draw your processes instead of coding them



You bring the experience… the process modeler the tools. No one knows the commercial processes better than who use them and be able to design them freely, automate to the maximum, and integrate them to the rest of the business as well as its protagonists, is the key to digital innovation and for the success of the business itself.

Process modeler is an agile tool for everyone who uses it, not just for IT connoisseurs. Generating an environment of collaborative work among business users, process experts and computer technicians.

The key… teamwork



Take advantage of new technology, simplify your tasks and keep up to day by gaining productivity. Our activities are simplified from **Deyel** portal, where everybody receives their tasks sorted by various criteria and can be seen in list form, calendar or both, facilitating its fulfillment and return.

The calendar view grants the user the simplicity and tranquility to see the whole picture and not just the short term. This is accomplished by the strong integration among business processes, their activities and the interactions between their participants with the personal calendar.

From where you are and with your favorite device

Take decisions faster and in the right time. All the applications developed in **Deyel** can be available on smartphones, tablets, notebooks and other devices as one more app. Downloads from the app store for free and you can use them where you need them with the device you have at the time.

We love to integrate with your applications ... don't worry about this

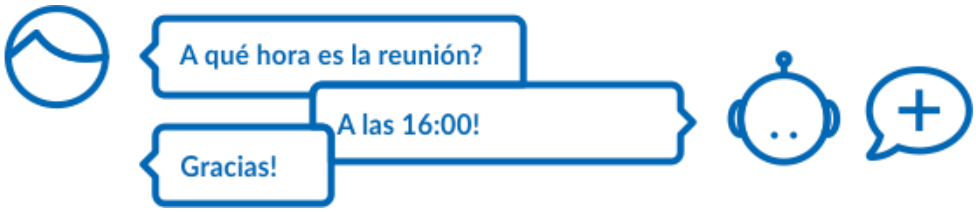
Deyel makes it easy for you for your application to play in the big leagues. Integration rules, together with the adapters already defined or the ones you can create, make the applications developed in Deyel able to interact directly with your corporate systems, web services, robots, artificial intelligence, the internet of things and your favorite apps.

For authentication, **Deyel** has an embedded security system that can be integrated with LDAP, allowing users to be managed in an integrated way.

Visualize and control each detail of your business

Deyel dashboards are the best way to show what is important in one single place. **Deyel** portal provides you with all the information and tools necessary to define dashboards containing business indicators based on multiple types of graphs, without the need for coding. Forget about business intelligence tools.

Improve communication and share information with your whole team

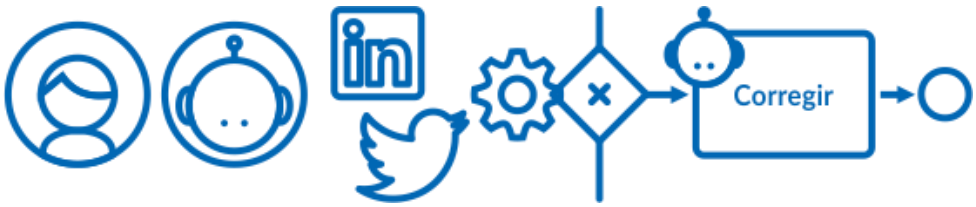


Collaboration is key to improve productivity. **Deyel** has collaboration tools where users interact through messages, share information, state, comments and documentation related to your business applications and each case in particular.

Deyel's corporate social messaging, carried out through private or group chats, allows, unlike others, that all communication between sender and receiver is associated with the case as part of its execution and resolution, becoming such a fundamental piece as the process information itself.

The management indicators, based on the analysis of the execution of the applications, are defined in a very easy way through the **Deyel** widget modeler and with them, you can design your dashboards in the dashboards modeler, with high visual impact, favoring the interpretation of the evolution of your business.

Define who you interact with



Who, when and how they access your application data. Participants in a process are much more than a simple user; they are the ones who give life to the business, and its success depends in part on their performance.

Deyel participants don`t necessarily have to be human. Many tasks are automated through IoT technology and others are executed transparently and efficiently by bots, driving new digital experiences in a wide range of use cases.

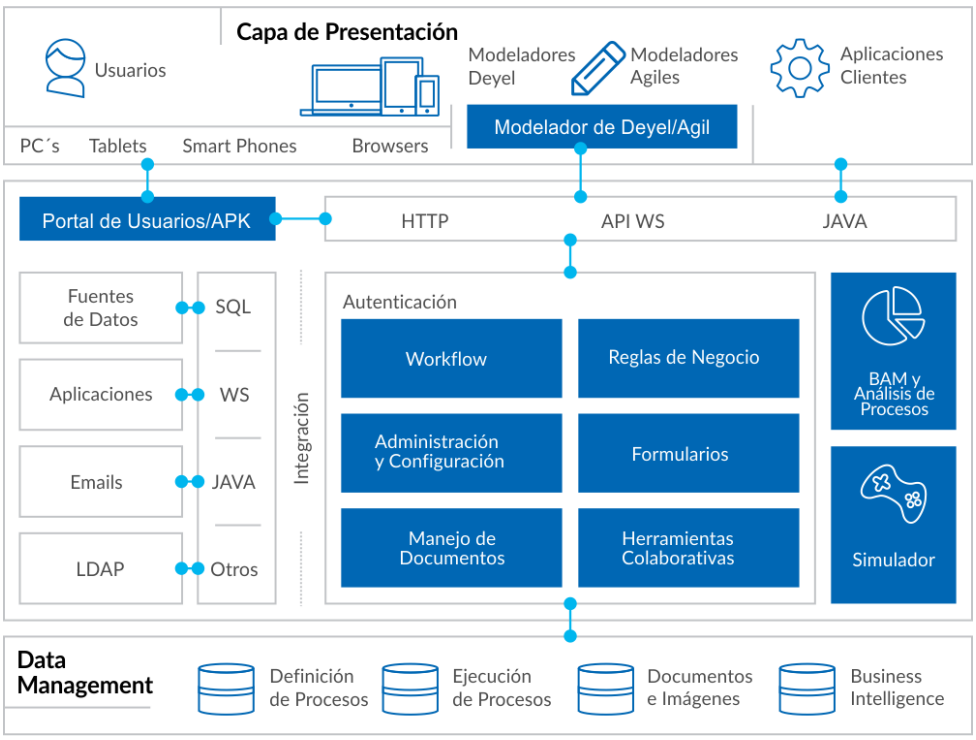
Deyel allows you to observe, through each participant, what tasks are being carried out, which ones have to be carried out and also how they are carried out, including their characteristics, behavior, performance and level of participation in company tasks.

3. User Manual

3.1. Architecture

Deyel Deyel is product based on Java technology, developed with modern object-oriented patterns. It supports several industry standards such as HTML5, CSS3, RWD, web services, multiple database and application servers.

The following chart describes the main components of its architecture.



Graphic Modeler

Using the graphic modeler, business users can graphically design their processes using **BPMN 2.0** (Business Process Model and Notation) symbols in an intuitive and easy-to-use environment. Each process with its activities, interconnections, events, times, alerts, etc. can be defined and also documented. In this same workspace, aspects of processes implementation can be included, such as user interfaces of the activities, integration with other applications, variables and business rules, among others.

User Portal

Through the portal of **Deyel**, users can interact with their list of activities and execute their tasks, start cases of their processes, consult them, access collaborative tools, monitoring reports, etc.The administration and configuration functionality can also be used through the portal. The portal is based on standards such as RWD, HTML5 and CSS3. They can be used on mobile devices or desktop PCs through web browsers. Its appearance is configurable according to the characteristics of each organization.

Workflow Engine

The workflow engine of **Deyel** interprets and executes the definition of business processes that are modeled in the graphic modeler.The modeler stores the graphically modeled processes within the process repository, so that the workflow engine can interpret these definitions and execute them. The execution state remains in the case execution repository.

Form Administrator

Through this **Deyel**tool, forms that represent the user interface of human activities are generated in the modeled processes. Forms contain the process variables that are used by business rules and other related objects.

Forms have an interface rich in components (jQuery and Bootstrap), support RWD (Responsive Web Design) concept and are based on standards such as HTML5 and CSS3.

Business Rules.

The behavior of processes is defined by business rules. Rules can be used in validations, specific business logic, process flow control, integration with other applications, etc.

Deyel offers wizards the business user that allow them to intuitively define the behavior of the rule. These wizards present the user with control structures, access to available parameters and variables, previously defined business rules, executing components, logical operators, etc. Rules persist in the object catalog of **Deyel.**

Collaborative Tools

With the messenger service of **Deyel**, users can communicate easily. Messages can be used among users in private, or they can be public when they are associated with cases or processes definitions. Associating them with cases provides a collaborative tool among the participants of the case, making these messages part of the case. On the other hand, messages related to the definition of processes enable a direct communication channel to improve processes among the participants of the case and process owners.Also, subscriptions enable an option to choose to receive messages about cases or processes of interest.

Integration

Deyel provides tools and services that allow any application to interact in a bi-directional way with the defined processes. Making use of adapters based on Web Services, Java, JDBC and others, allows to integrate existing applications or data sources.

Deyel also has Java APIs and web services that allow any application to interact with processes through them.

BAM and Process Analysis

Deyel offers powerful capabilitiesfor analysis and browsing of multidimensional structures that allow to analyze the operation of processes and their tasks, both historically (Process Analysis) and in real time (BAM - Business Activity Monitoring).

Through predefined reports, it is possible to observe and understand the operation of processes and detect possible improvements. They analyze in brief the behavior of processes and their tasks, being able to identify what was done on time, with delay, deviations against maximum and expected durations, trends, participant performance, bottlenecks, etc.

The behavioral information is displayed as a grid and through charts, always having the reference of the expected trend curves to contrast against reality. In this way, behavior can be graphically monitored. In all cases, it is possible to drill down into the information, reaching the process level, task, executor, date range, case, etc. Multiple views of the information can be generated by navigating it through multiple dimensions, such as time, priorities, participants, initiators, states of cases and tasks, among others.

Administration and Configuration

The administration and configuration of **Deyel** allows to define the security and authentication of process participants with their different profiles and roles, their relation with LDAP repositories, workgroups, their districts, calendars, time zones, user portal appearance and also technical aspects of the solution.

3.1.1. Plan de Recuperación de Desastre (DRP)

Es fundamental para una empresa garantizar la continuidad del negocio. Para ello, necesita un plan sólido de recuperación ante desastres, que le permita mantenerse en funcionamiento frente a un desastre natural o ataques malintencionados.

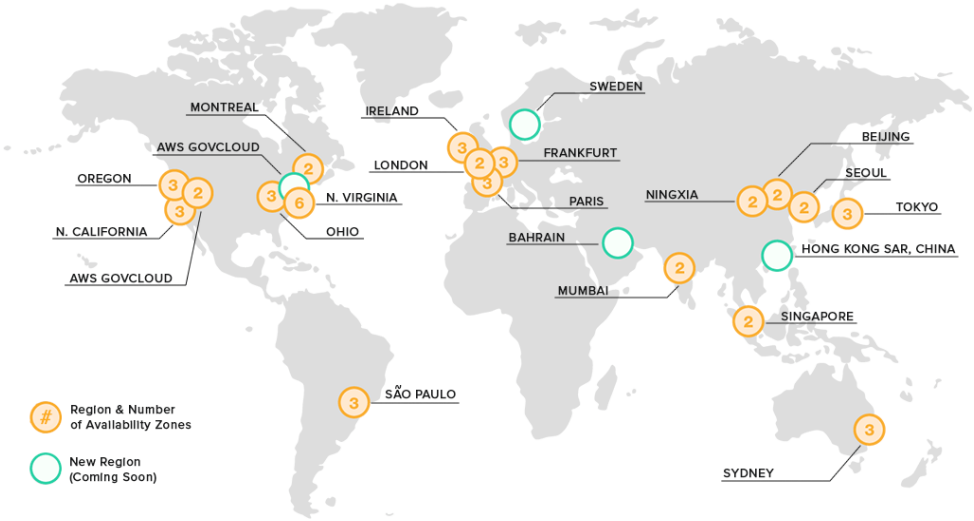
La infraestructura Deyel Cloud en AWS, basa principalmente su estrategia del DRP, en las ventajas que ofrecen los servicios de AWS sobre los que está montada.
Los dos ejes principales de estas ventajas son la utilización de las regiones, zonas de disponibilidad y data centers distribuidos en el mundo, y la configuración de los servicios utilizados para que hagan uso de esta infraestructura global.

Utilización de Regiones, Zonas de Disponibilidad y Data Centers de AWS

La infraestructura Deyel Cloud se basa en regiones y zonas de disponibilidad de AWS.

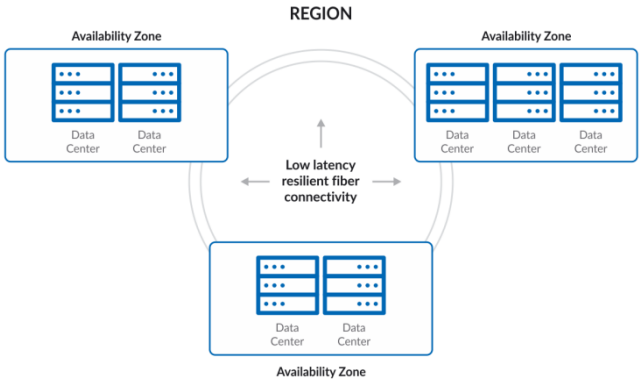
Una región de AWS es una ubicación física en el mundo, que tiene múltiples zonas de disponibilidad (Availability Zones). Estas zonas cuentan con uno o más data centers y cada uno de ellos con alimentación, redes y conectividades redundantes, alojados en instalaciones separadas.

Estas zonas de disponibilidad ofrecen la capacidad de operar aplicaciones de producción y bases de datos con alta disponibilidad, tolerantes a fallas y escalables. AWS cuenta con más de 60 zonas de disponibilidad y con más de 20 regiones geográficas alrededor del mundo.



https://www.infrastructure.aws/

Cada región de AWS está diseñada para estar completamente aislada de las demás regiones. Esto permite mayor tolerancia a fallas y estabilidad.
Cada zona de disponibilidad está aislada, aunque se encuentren conectadas a través de enlaces de baja latencia.



Cada zona de disponibilidad está diseñada como una zona de falla independiente, esto significa que las zonas de disponibilidad están físicamente separadas dentro de una región metropolitana típica y están ubicadas en llanuras con poco riesgo de inundación.

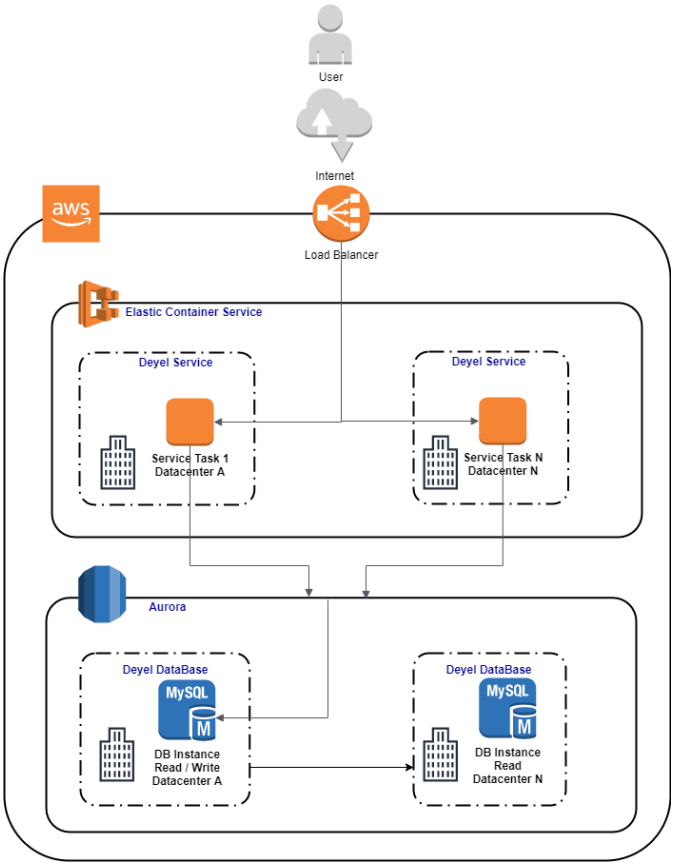
Además cuentan con fuentes de alimentación ininterrumpida (UPS) y backups onsite. Las zonas de disponibilidad se encuentran conectadas con enlaces de alta velocidad (tier-1).

Configuración de los Servicios de AWS Usados por Deyel Cloud para que Utilicen la Infraestructura Global

Los servicios de AWS utilizados en la infraestructura Deyel Cloud están configurados para utilizar regiones y zonas de disponibilidad.

Principales componentes de infraestructura Deyel Cloud

Se detallan a continuación los principales componentes de la infraestructura Deyel Cloud y cómo están configurados los servicios de AWS que los soportan.



- Datos de las aplicaciones
  Los datos de las aplicaciones se almacenan en clusters de Aurora del servicio Amazon Relational Database (RDS). Estos clusters están configurados con instancias de read/write y de read en diferentes zonas de disponibilidad, que pueden intercambiarse inmediatamente ante un eventual failover. Para la creación de los backups se utiliza la opción multiAZ,

- Ejecución de las aplicaciones

Las aplicaciones se ejecutan en clusters del servicio AWS Elastic Container Service (ECS) con instancias EC2 distribuidas en zonas de disponibilidad, con autoescalado. Las aplicaciones se ejecutan en servicios de ECS y pueden ser multitarea distribuidas en instancias EC2 en distintas zonas de disponibilidad.

Los clusters utilizados por la infraestructura Deyel Cloud, pueden ser creados en cualquier región de AWS a nivel global en cuestión de segundos.

- Archivos críticos de la infraestructura replicados entre regiones.
  Todos los archivos necesarios para el armado y creación de la infraestructura y para la recuperación de un desastre, están almacenados con el servicio de AWS S3 y Amazon ECR, usando replicación entre regiones.

- Infraestructura como código (IaC)
  La infraestructura y los recursos necesarios para la ejecución de **Deyel** en AWS se crean utilizando el servicio AWS CloudFormation. Al utilizar este servicio se cumple con las buenas prácticas de estandarizar los componentes de la infraestructura y se permite una rápida solución de problemas.

Suministra los recursos de las aplicaciones de una manera segura y repetible, ya que permite crear y recrear la infraestructura y las aplicaciones, sin tener que realizar acciones manuales ni escribir scripts personalizados.
A través de este servicio, la infraestructura Deyel Cloud y sus aplicaciones pueden ser implementadas en cualquier región de AWS a nivel global.

Eventos de Fallas, RPO y RTO para las Ediciones Standard y Enterprise

Las tareas y la base de datos con su réplica están en varias zonas de disponibilidad (en uno o más data centers) dentro de una región.

Evento de Fallas	Acción	RPO	RTO
El servidor que ejecuta la aplicación sale de servicio. Ejemplo: falla de hardware, placa madre, fuentes, discos, etc.	Si se tienen varias tareas de la aplicación, se deriva la carga al resto. En todos los casos automáticamente se inicia una tarea idéntica en otro servidor del mismo data center.	0	En la edición Enterprise el RTO es 0. En la edición Standard el RTO es menor a 90 segundos.
El data center que contiene los servidores que ejecutan la aplicación sale de servicio. Ejemplo: catástrofe en la ciudad del data center.	Si se tienen varias tareas de la aplicación, se deriva la carga al resto. En todos los casos automáticamente se inicia una tarea idéntica en otro servidor de la misma zona de disponibilidad.	0	En la edición Enterprise el RTO es 0. En la edición Standard el RTO es menor a 90 segundos.
El servidor que ejecuta la base de datos sale de servicio. Ejemplo: falla de hardware, placa madre, fuentes, discos, etc.	Automáticamente la instancia que está solo de lectura se convierte en R/W.	0	En ambas ediciones el RTO es menor a 90 segundos.
La base de datos de R/W sale de servicio. (se rompe la estructura, problemas en el motor).	Automáticamente la instancia que está solo de lectura se convierte en R/W.	0	En ambas ediciones el RTO es menor a 90 segundos.

Instancia de Réplica de Read en una Región Diferente a la Principal

La instancia de réplica de read en una región diferente a la principal es otra opción de mayor disponibilidad y tolerancia a fallas.

En el caso de considerar catástrofes de magnitud mundial en donde toda una región puede dejar de operar, se generan los clusters a través del servicio AWS CloudFormation en otra región y se continúa utilizando la base de datos interregional logrando una sincronización activa y una mayor tolerancia de fallas en base a una mejor dispersión geográfica.

El RTO mínimo indispensable es de 3hs, aunque se recomiendan tareas de verificación por parte del cliente antes de habilitar el uso de la plataforma nuevamente, que pueden elevar el RTO global.
A modo de ejemplo se detallan las tareas a realizar en estos casos.

Tareas	**RTO < 8 Hs**
Evaluación de la situación, verificación de la no operación de las regiones en donde estaba funcionando la plataforma.	2hs
Generación de la infraestructura de la plataforma en la tercera región.	Menor a 30 minutos
Configuración de la nueva infraestructura con la base de datos operativa en esta región.	Menor a 30 minutos
Verificación por parte del cliente de la plataforma.	Recomendado. 3hs

3.2. General Concepts

Se explican a continuación un conjunto de conceptos que introducen al desarrollo con **Deyel**.

Aplicación

Una aplicación es un conjunto de procesos, entidades de negocio basadas en formularios, entidades de parametrización, reglas, adaptadores, roles y permisos, relacionados entre sí, con el objetivo de resolver un requerimiento determinado.

Desde una aplicación pueden utilizarse entidades del sistema, tales como los usuarios, la estructura organizacional, los casos y los procesos, entre otros y además todos aquellos objetos que hayan sido definidos como globales.

Tipos

- Licenciadas
  Una aplicación licenciada requiere de una licencia específica, en cuyo caso se la denomina producto o solución, dependiendo de la funcionalidad de cada una. **Deyel** es un producto, mientras que Contratos, CRM y Autoatención de empleados son soluciones.

- No licenciadas
  Son aplicaciones que pueden ser utilizadas libremente por los usuarios de **Deyel**.

Propiedades

Nombre
Se usa para referenciar la aplicación dentro de reglas o como parámetro. Soporta multi-idioma.
Ejemplos: Contratos, Autoatención.

Nombre Descriptivo
Se usa en la interfaz cuando se la referencia, en la galería de aplicaciones y en el árbol de objetos. Soporta multi-idioma.
Ejemplos: Sistema de Contratos, Autoatención de Empleados.

Descripción
Texto que define a la aplicación. Soporta multi-idioma.

Identificador
Se usa en el código de programación. No es multi-idioma.
Ejemplos: CONTRACTS, SELFSERVICE.

Entidad

Una entidad es un objeto compuesto por atributos que representan los datos del negocio y del ambiente. Los procesos y las aplicaciones tienen interacción con las entidades, las cuales pueden relacionarse entre sí.

Alcance

Puede ser global o de una aplicación.

Tipos

- Del Sistema
  Compuesta por atributos donde persiste la información de ambiente y ejecución.
  Son predefinidas en la aplicación **Deyel** y no pueden ser modificadas por el usuario.
  Ejemplos: Usuario, Unidad Organizacional, Caso.

- De Parametrización
  Compuesta por atributos donde persisten valores, que posteriormente a su carga, pueden ser seleccionados por medio de listas desplegables.
  Pueden ser modificables por el usuario.
  Ejemplos: Ciudad, Género, Tipo de Cliente.

- De Negocio
  Compuesta por atributos donde persiste la información del negocio.
  Se modelan gráficamente a partir de su apariencia, utilizando el modelador gráfico de formularios.
  Ejemplos: Empleado, Contrato, Factura.

Propiedades

Nombre
Se usa para referenciar la entidad dentro de reglas o como parámetro. Soporta multi-idioma.

Ejemplos: Empleado, Cuenta

Nombre Descriptivo
Se usa en la interfaz cuando se la referencia, en la lista de entidades y en el árbol de objetos. Soporta multi-idioma.
Ejemplos: Empleado, Cuenta.

Descripción
Texto que define a la entidad. Soporta multi-idioma.

Aplicación
Aplicación a la que pertenece.

Identificador
Se usa en el código de programación. No es multi-idioma.
Ejemplos: EMPLOYEE, CRM_ACCOUNT

Atributo

Un atributo es la unidad de información dentro de una entidad.

Propiedades

Etiqueta
Se usa en la interfaz cuando se lo referencia, en la lista de atributos de la entidad y en el árbol de objetos. Soporta multi-idioma.
Ejemplo: Legajo

Nombre
Se usa para referenciar el atributo, precedido por el identificador de la entidad, dentro de reglas o como parámetro. Soporta multi-idioma.
Ejemplo: legajo

Identificador
Se usa en el código de programación, precedido por el identificador de la entidad. No es multi-idioma.
Ejemplo: personnel_nr

Descripción
Texto que define al atributo. Soporta multi-idioma.

Proceso

Un proceso es una secuencia de tareas humanas o automáticas (servicios), cuya realización permite cumplir con un objetivo de negocio. La secuencia tiene inicio, finalización y puede contener conectores. Se representa como un diagrama de proceso que responde al estándar de **BPMN 2.0** (Business Process Model and Notation).

Alcance

Puede ser global o de una aplicación.

Propiedades

Nombre
Se usa en la interfaz cuando se lo referencia, en la galería de procesos y en el árbol de objetos. Soporta multi-idioma.
Ejemplos: Solicitud de Reintegro, Solicitud de Vacaciones.

Código
Se usa en el código de programación. No es multi-idioma.

Versión
El número de versión junto con la propiedad Código, identifican unívocamente al proceso.

Responsable
Dueño del proceso, es responsable del mismo de principio a fin.

Descripción
Texto que define al proceso. Soporta multi-idioma.

Formulario

Un formulario es la interfaz de usuario de las entidades de negocio. Puede ser usado en las actividades de usuario de los procesos o en la administración de datos de la entidad en sí misma.

A través del modelador de formularios se modelan las entidades del negocio. De una forma intuitiva y gráfica se define la apariencia de la entidad y paralelamente su estructura de datos y persistencia.

Ejemplos

- El formulario Solicitud de Vacaciones se usa en el proceso Vacaciones.
- El formulario Empleado se utiliza para actualizar los datos almacenados en sus campos durante su ciclo de vida. Además, los campos de dicho formulario pueden utilizarse desde diferentes procesos que lo tengan vinculado y que contengan actividades de usuario.

Alcance

Puede ser global o de una aplicación.

Propiedades

Nombre
Se usa para referenciar el formulario dentro de reglas o como parámetros. Soporta multi-idioma.
Ejemplos: Empleado

Nombre Descriptivo
Se usa en la interfaz cuando se lo referencia, en la galería de formularios y en el árbol de objetos. Soporta multi-idioma.
Ejemplos: Empleado.

Aplicación
Aplicación a la que pertenece.

Descripción
Texto que define al formulario. Soporta multi-idioma.

Identificador
Se usa en el código de programación. No es multi-idioma.
Ejemplos: EMPLEADO

Control

Un control permite definir la apariencia en la interfaz de un campo de un formulario. Además, el control agrega características por defecto de comportamiento y persistencia al campo del formulario, que posteriormente podrán ser configuradas.

Por medio de un conjunto de controles se puede definir una guía de diseño de interfaces basada en patrones con el objeto de resolver situaciones comunes y lograr una interfaz homogénea.

Los formularios se crean a partir de una galería de controles, cada uno de ellos permite definir:

- La apariencia de un campo de un formulario en la interfaz.
- La disposición de los campos y elementos gráficos dentro del formulario.
- Acciones a realizar desde la interfaz.

Ejemplos

- Caja de Texto
- Número
- Botón
- Contenedor

Regla

Una regla de negocio define el comportamiento de los procesos y las aplicaciones.

Las reglas pueden ser usadas en validaciones, lógica de negocio, control de flujos de procesos, integración con otras aplicaciones, tareas programadas y control de visualización de campos en formularios.

Ejemplos

- Definir condiciones en flujos dentro de un proceso.
- Ejecutar acciones automáticas en actividades de un proceso.
- Definir obligatoriedad, visibilidad, edición, cálculos y validaciones en campos de un formulario.

Alcance

Puede ser global o de una aplicación.

Tipos

- Reglas embebidas
  Se definen dentro del objeto cuyo comportamiento afectan y persisten dentro de dicho objeto. Se utilizan para establecer el comportamiento del objeto que las contiene.

- Reglas avanzadas
  Se definen en el modelador de reglas y persisten dentro del catálogo de reglas.

Propiedades

Nombre
Se usa en el código de programación y para referenciar la regla dentro de galería de reglas o como parámetros. No es multi-idioma.
Ejemplos: AEImpGetOficinas

Versión
El número de versión junto con la propiedad Nombre, identifican unívocamente la regla.

Nombre Descriptivo
Se usa en la interfaz cuando se la referencia, en la galería de reglas y en el árbol de objetos. Soporta multi-idioma.
Ejemplos: Leer Oficinas

Descripción
Texto que define a la regla. Soporta multi-idioma.

Aplicación
Aplicación a la que pertenece.

Función

Unidad de software que realiza una tarea particular que se ejecuta al ser invocada desde un código de programación o de los asistentes de modelado. Puede recibir parámetros de entrada y devuelve un resultado.

Propiedades

Nombre
Se usa para referenciar la función dentro de reglas o como parámetros. Soporta multi-idioma.

Ejemplos: emailUsuarioActual, dia.

Nombre Descriptivo
Se usa en la interfaz cuando se la referencia, en la lista de funciones y en el árbol de objetos. Soporta multi-idioma.
Ejemplos: Email del Usuario Ejecutor, Día Actual.

Identificador
Se usa en el código de programación. No es multi-idioma.
Ejemplos: EXEC-USER_MAIL, CURRENT_DAY.

Descripción
Texto que define a la función. Soporta multi-idioma.

Uso de los Conceptos

A continuación se describen algunas consideraciones relativas al uso de los conceptos descriptos en este tópico.

Propiedad
Describen en forma general a aplicaciones, entidades, procesos, formularios, atributos, reglas y funciones.

Control
El término control se utiliza solo en el modelado de la interfaz gráfica de los formularios.

Campo
Los formularios almacenan sus datos en un conjunto de campos.
En el modelador de formularios, un campo del formulario puede definirse a partir de un atributo de otra entidad, en este caso se lo llama atributo relacionado y puede tener persistencia.

Atributo
Es la menor unidad de información de una entidad, corresponde al modelo de datos.

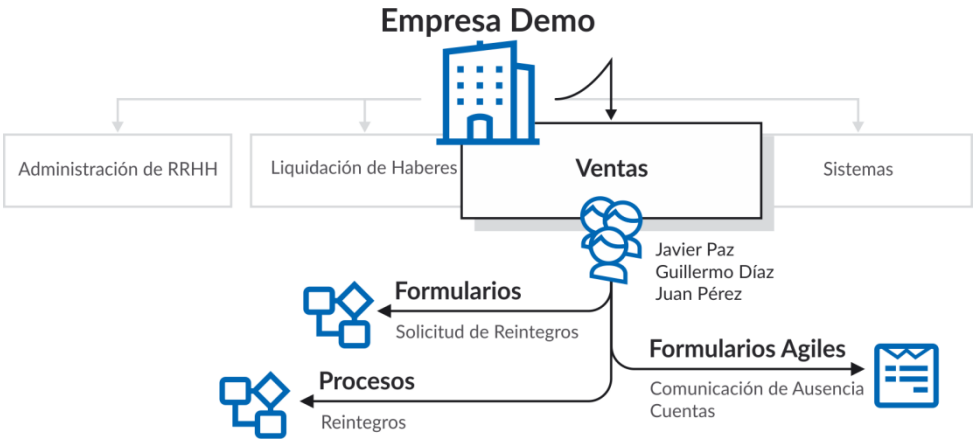
Variable
Una variable corresponde al atributo de una entidad del sistema o de negocio y es utilizada por los procesos de negocio para definir su estado y comportamiento. Puede contener datos de las condiciones temporarias de las instancias de los procesos(casos), de los objetos de negocio o referencias a estos últimos.
Alcance:
- Local: Solamente puede ser accedida y modificada durante la ejecución del caso del proceso que la usa. Las variables de casos finalizados pueden ser accedidas pero no modificadas.
- Global: Puede ser accedida y modificada por las operaciones del formulario correspondiente, por los procesos que la usan o por reglas de negocio.

3.3. Examples of Use

Deyel incluye ejemplos que permiten familiarizarse rápidamente con la herramienta.

Para esto, existe una empresa de demostración con ejemplos de formularios ágiles, unidades organizacionales y usuarios, con los que se puede experimentar cómo **Deyel** permite modelar rápida y fácilmente las aplicaciones, promoviendo la productividad en la empresa.

La empresa de demostración puede eliminarse en cualquier momento que se desee, junto con toda la información relacionada que pudo haberse generado durante las pruebas. También es posible importar estos ejemplos en cualquier momento.

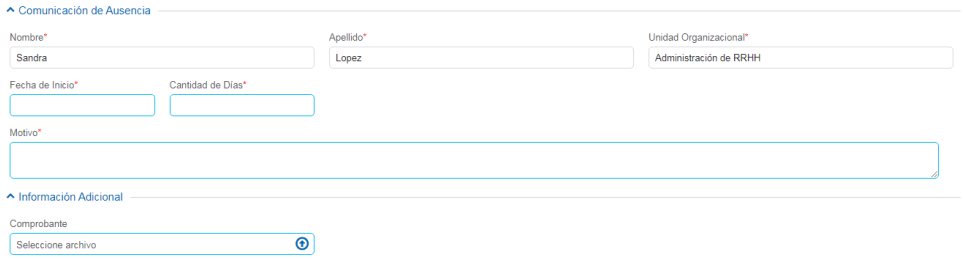


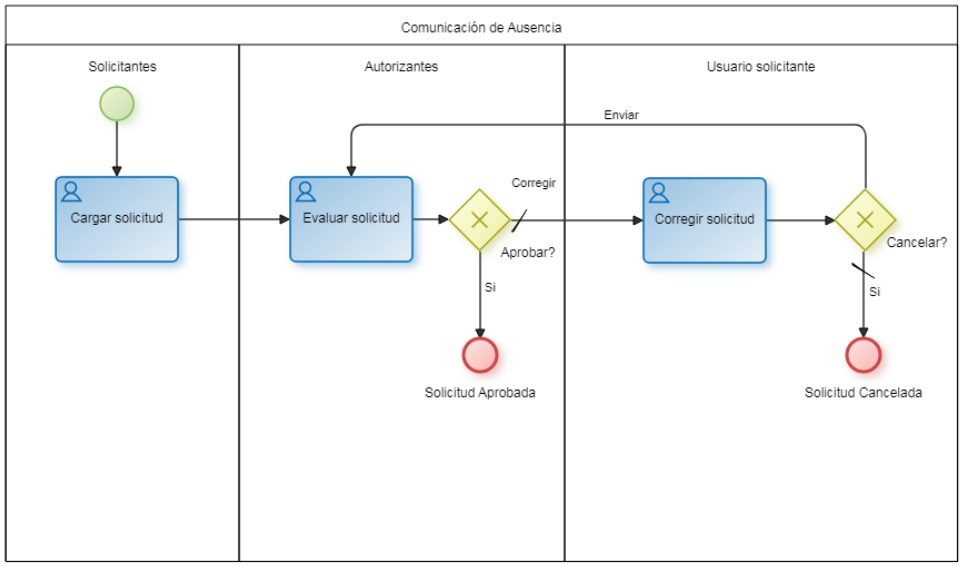
Formularios Agiles

Se incluye un conjunto de formularios ágiles modelados a partir de las plantillas existentes. Cada uno de los formularios contiene un grupo de campos sugeridos, que pueden ser extendidos con nuevos campos definidos por el usuario de negocio de acuerdo a las necesidades de su empresa.
Los procesos provistos corresponden a modelos de uso corriente en todas las organizaciones.

Comunicación de Ausencia

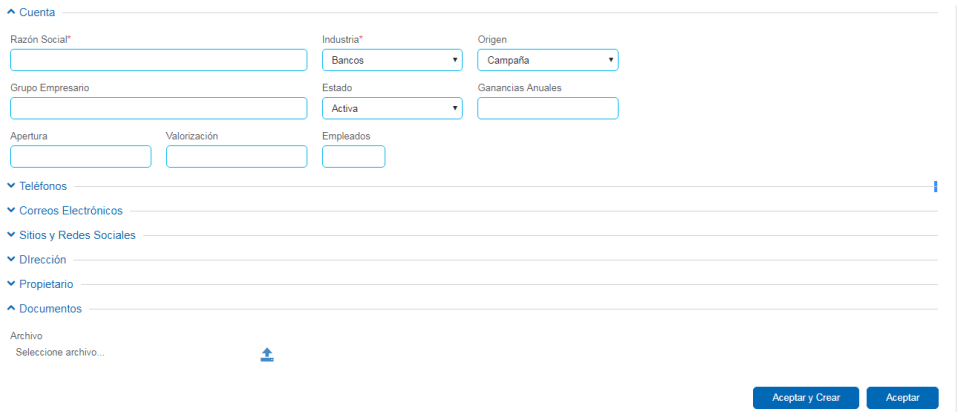
Formulario ágil con proceso asociado para comunicar ausencias, mediante el cual los empleados de la empresa pueden reportar ausencias laborales, junto con los motivos de las mismas y solicitar la autorización a un superior. El solicitante se conecta al portal e inicia un proceso para comunicar a sus colegas que se desempeñan en la misma unidad organizacional, que estará ausente. Completa para ello un formulario donde registra el período de ausencia con la pertinente justificación y envía la solicitud al autorizante correspondiente para que verifique la documentación respaldatoria. Cuando la ausencia está debidamente justificada, con certificación médica u otro tipo de documentación, el jefe del sector autoriza y finaliza la comunicación. De no ser aprobada se envía a revisar por el solicitante, que puede reenviar el pedido o cancelarlo.





Clientes

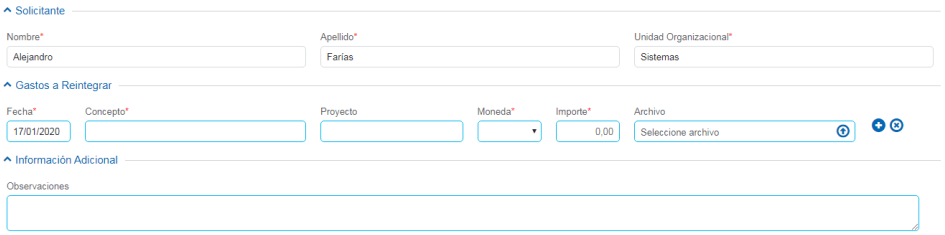
Formulario ágil sin proceso asociado, sobre el que los usuarios de negocio pueden realizar las operaciones de creación, consulta, modificación y eliminación, de acuerdo a los permisos que tengan definidos.

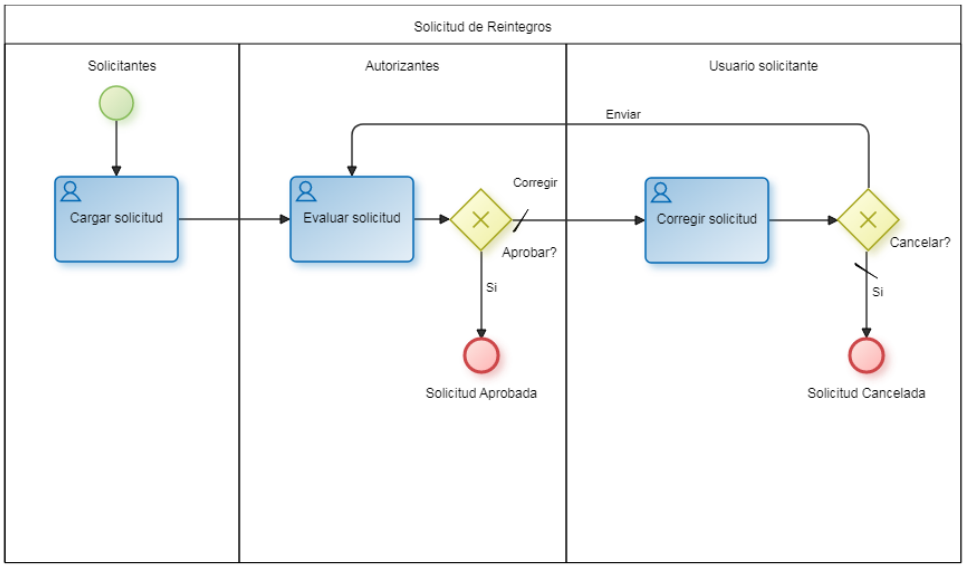


Solicitud de Reintegros

Formulario ágil con proceso asociado para solicitar reintegros de gastos, mediante el cual los empleados de la empresa, completan un formulario de solicitud detallando la lista de gastos, el monto y ad-

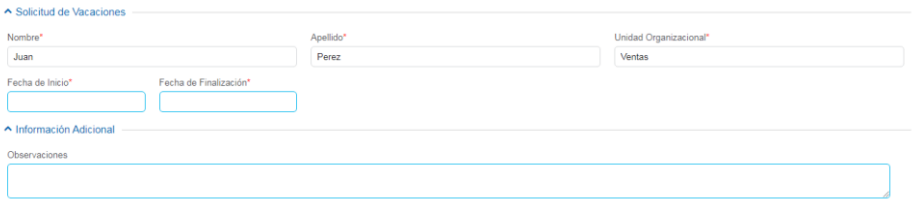
junta sus comprobantes. Una vez ingresada la solicitud, es enviada al autorizante correspondiente para aprobarla o bien enviarla a revisar por el solicitante, que puede reenviar el pedido o cancelarlo.

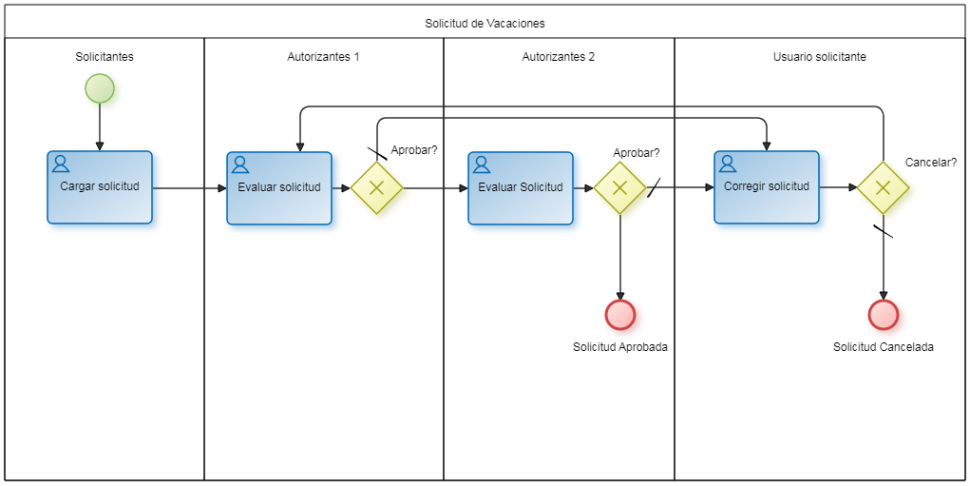




Solicitud de Vacaciones

Formulario ágil con proceso asociado, mediante el cual los empleados de la empresa pueden gestionar sus vacaciones. Para ello completan un formulario con la fecha de inicio y la fecha de finalización, que se envía al autorizante y luego al sector de administración de personal y liquidación de haberes, para que realicen los controles y gestiones correspondientes. A medida que los sectores las autorizan, las solicitudes avanzan hasta su aprobación total. Si alguna solicitud no es aprobada, se envía al solicitante para su revisión, que puede corregirla y reenviarla a autorizar o bien cancelarla.





Estructura Organizacional de "Empresa Demo"

"Empresa Demo" es la empresa de demostración definida con datos con ejemplos de procesos y formularios, En la estructura organizacional se encuentran definidas oficinas o unidades organizacionales, que van a agrupar usuarios de negocio.

> *La Empresa Demo puede ser eliminada por un usuario con permisos de administrador.*
> *Al confirmar esta operación se eliminan los siguientes datos:*
> *- La unidad organizacional Empresa Demo y todas sus unidades subordinadas.*
> *- Los usuarios Javier Paz, Juan Perez, Sandra Lopez, Valentin Pereira, Guillermo Diaz y Alejando Farias, que pertenecen a la unidad organizacional.*
> *- Los formularios ágiles Solicitud de Vacaciones, Solicitud de Reintegros y Comunicación de Ausencia.*

Usuarios

Los usuarios definidos en la Empresa Demo pertenecen a distintas unidades organizacionales y tienen asignados permisos de acuerdo a sus puestos. Todos los usuarios utilizan la contraseña “deyel123” para iniciar la sesión.

Usuario	Unidad Organizacional	Permiso
SLOPEZ - Sandra López	Administración de RRHH	DEYEL- Usuario Final

Usuario	Unidad Organizacional	Permiso
VPEREIRA - Valentin Pereira	Liquidación de Haberes	DEYEL- Usuario Final
AFARIAS -Alejandro Farías	Sistemas	DEYEL - Administrador DEYEL - Administrador de Seguridad DEYEL - Modelador Deyel
GDIAZ - Guillermo Díaz	Ventas	CRM - Vendedor
JPAZ - Javier Paz	Ventas	DEYEL - Modelador Agil CRM - Gerente de Ventas
JPEREZ - Juan Perez	Ventas	DEYEL - Usuario Final CRM - Vendedor

3.4. User Portal

Fase 3: Portal

Utilizando el portal, los usuarios pueden interactuar con su lista de actividades y ejecutar sus tareas, iniciar casos de sus procesos, consultarlos, acceder a las herramientas colaborativas, reportes de monitoreo, etc.
También a través del portal, se tiene acceso a la funcionalidad de Administración y Configuración de **Deyel**.

La interfaz de usuario se basa en el concepto "**Model once, execute anywhere**" permitiendo que las aplicaciones implementadas en **Deyel** puedan ser ejecutadas indistintamente desde cualquier tipo de dispositivo, incluyendo puestos de trabajo PC, notebooks, tablets y smartphones.

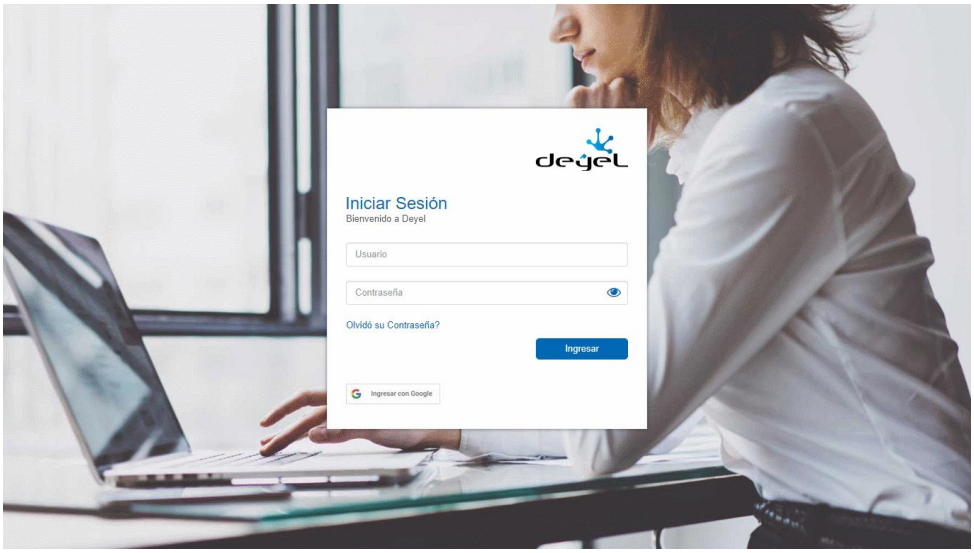
La interfaz de usuario ha sido diseñada considerando "Responsive Web Design" con lo cual se adapta dinámicamente en función del dispositivo utilizado para acceder al Portal.

Utilizando estándares tales como HTML5 y hojas de estilos, se obtiene una interfaz moderna, rica en funcionalidad, que se adapta e incrementa muchísimo la facilidad de uso en los distintos tipos de dispositivos, aprovechando las ventajas de cada uno, tales como cámaras, gps, etc.

Poder utilizar los formularios y ejecutar los procesos de negocio utilizando tecnologías móviles, habilita a una nueva generación de aplicaciones, en donde los usuarios pueden conectarse en cualquier momento y desde cualquier lugar, ejecutando y consultado sus tareas, con la misma familiaridad disponible en las herramientas de uso habitual.

3.4.1. Acceso al Portal

Para poder acceder al portal se debe ingresar el usuario y la contraseña, pudiendo usar tanto el nombre de usuario como la dirección de email que puede definirse como alias.



Se puede acceder desde dispositivos móviles, PCs de escritorio o notebooks, adaptándose la ventana a cada uno de ellos.

El sistema dá seguridad para asegurar la correcta identificación del usuario y controlar que tenga autorización para utilizar el producto seleccionado. En caso que no la tenga o que haya ingresado un usuario o contraseña no válidos se visualiza el mensaje correspondiente, el mismo permanece visible hasta que el usuario intente ingresar nuevamente o hasta que cierre el mensaje.

En caso de haber olvidado la contraseña el usuario el usuario puede seleccionar la opción Olvidó su Contraseña para establecer una nueva.

Mediante un ícono a la derecha del campo contraseña, se puede visualizar la misma.
Permite visualizar la contraseña.
Permite ocultar la contraseña.

Cuando un usuario se autentica correctamente e ingresa al portal desde un determinado dispositivo, su imagen de perfil es recordada en la ventana de acceso. De esta forma, la próxima vez que ingrese al portal desde el mismo dispositivo tendrá su imagen visible sobre el campo de ingreso del usuario y haciendo clic sobre la imagen de perfil se completa dicho campo, debiendo completar solamente la contraseña.

Si el usuario sale de **Deyel** sin cerrar sesión, la misma se resguarda. Por lo tanto la próxima vez que ingrese al portal lo hará directamente con el mismo usuario sin necesidad de volver a autenticarse.

Reingreso de Datos

Al recibir un aviso o un mensaje de error, el usuario puede:

- Presionar el botón “Volver” o “Atrás” del navegador para retornar a la pantalla anterior y corregir cualquier dato incompleto.
- Seleccionar nuevamente la opción desde el menú principal del sistema.

Cierre de Sesión por Inactividad

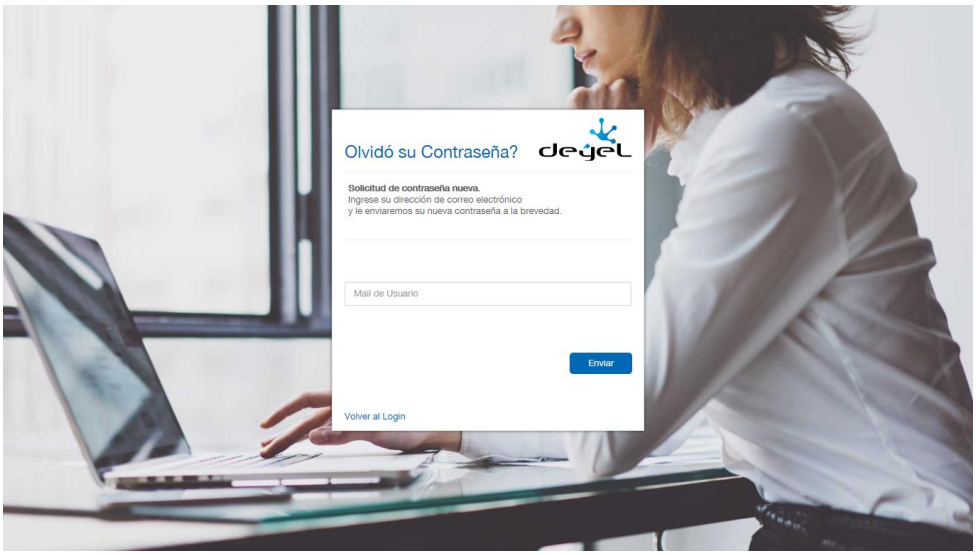
Si durante la sesión de trabajo se supera un cierto tiempo de inactividad, dicha sesión se cierra en forma automática. Si se intenta realizar alguna operación en una sesión que fue cerrada, el usuario es enviado al inicio de sesión y recibe un mensaje informativo.

3.4.2. ¿Olvidó su Contraseña?

Para acceder a la función se debe hacer clic sobre el texto "Olvidó su contraseña?" en la ventana de ingreso al portal.

Esta función permite que un usuario solicite una nueva clave de acceso, luego de realizarse las correspondientes validaciones, una nueva contraseña es enviada por email al usuario.

Al ejecutar la función "Olvidó su contraseña?" se solicita el email de la cuenta registrada en el perfil del usuario, para que pueda ser identificado y de esta forma recupere la contraseña.



Cuando el usuario ingresa su email ocurre lo siguiente:

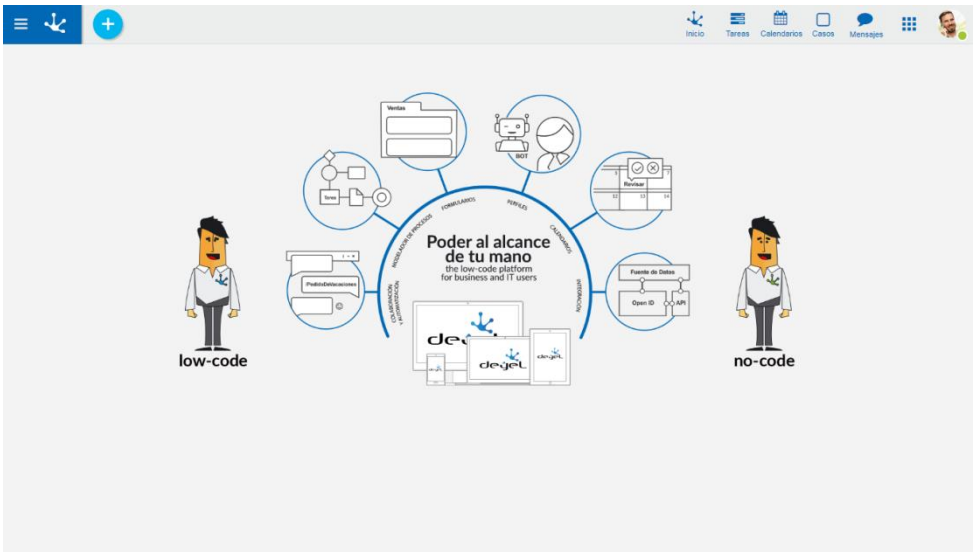
- Si el email ingresado corresponde a un usuario registrado, entonces se genera una nueva clave y se le envía la misma por correo a la cuenta informada.
- Si no pertenece a ningún usuario, recibe el mensaje "El email ingresado no está registrado en el sistema".

> *Solamente está disponible cuando se utiliza la autenticación Nativa, donde **Deyel** administra las contraseñas de acceso.*

3.4.3. Menú

Fase 3: Portal > Menúes

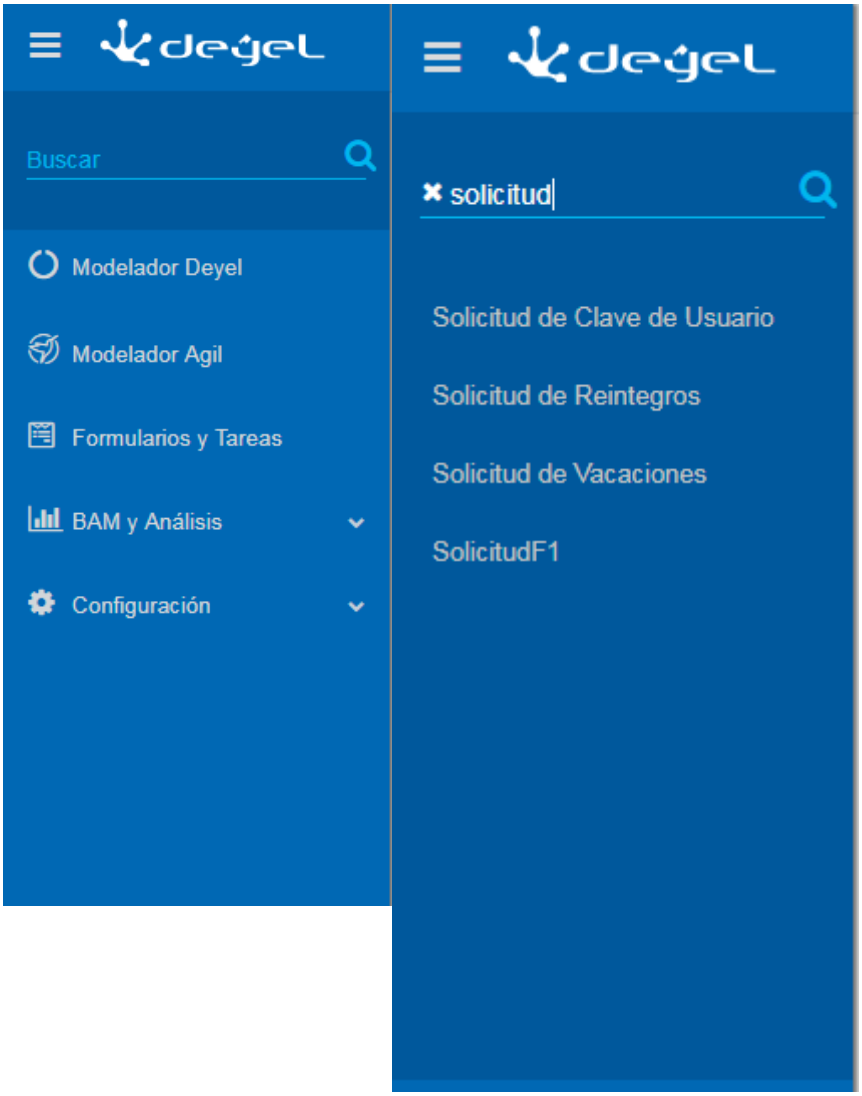
El menú se encuentra en la esquina superior izquierda de la ventana representado por el ícono que al seleccionarse despliega sus opciones.



Características Principales

Buscador

Permite buscar entre todas las opciones disponibles en el menú, formularios y funciones habilitadas para el usuario. Escribiendo un criterio de búsqueda, como una palabra o parte de una palabra, la aplicación busca entre las funciones, formularios y opciones del menú aquellas que coinciden con el patrón de búsqueda ingresado.



Una vez que se muestran los resultados de la búsqueda, se puede acceder a la opción deseada haciendo clic en la lista. Puede eliminar el criterio de búsqueda presionando la X que figura a la izquierda de la palabra buscada, retornando a las opciones del menú.

Soluciones

El menú incluye una entrada para las soluciones a las que el usuario tienen acceso.

BAM (Business Activity Monitoring) y Análisis

En esta opción del menú se accede a un conjunto de reportes que permiten analizar el funcionamiento de los procesos y sus tareas de dos maneras:

- En tiempo real (BAM)
- Histórico (Análisis)

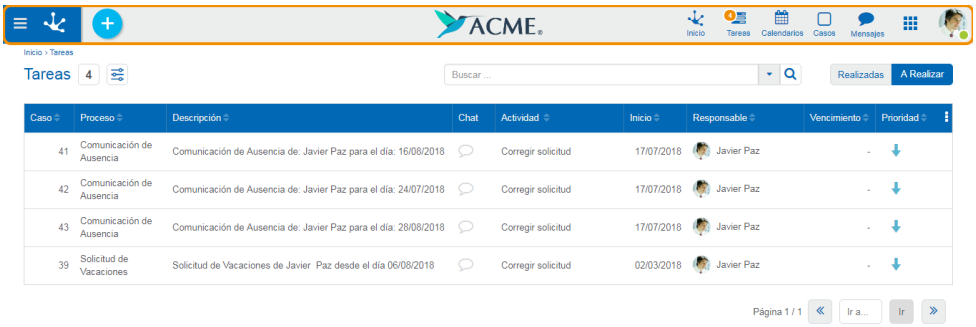
Configuración

En esta opción se puede configurar permisos de usuarios, definir las Unidades Organizacionales (UO) y acceder a herramientas de administración de la aplicación.

3.4.4. Barra de Herramientas Superior

Fase 3: Portal > Barra superior

Contiene los íconos correspondientes a las funcionalidades principales de **Deyel**. Permite el acceso rápido a las mismas al hacer clic sobre ellos.



Elementos de la barra

Inicio

Posiciona al usuario en la ventana inicial de la solución en la que se autenticó, cerrando toda opción del menú abierta en ese momento.

Icono +

Permite dar de alta a una nueva instancia de formulario o iniciar un nuevo caso. Es considerado como sinónimo de alta rápida, de acuerdo a la función que se está ejecutando en ese momento.

Al hacer clic sobre este ícono, se despliega un menú contextual que permite dar de alta información asociada a la actividad que está ejecutando el usuario.

Logo

Deyel permite personalizar la solución incorporando el logo de la empresa usuaria. Para una mejor visualización, se recomienda una imagen de 40 pixeles de altura como máximo.
Para incorporar, modificar o eliminar el logo, hacer clic en el lápiz que aparece a la derecha del ícono o del logo preexistente y se despliega un menú con las opciones disponibles.

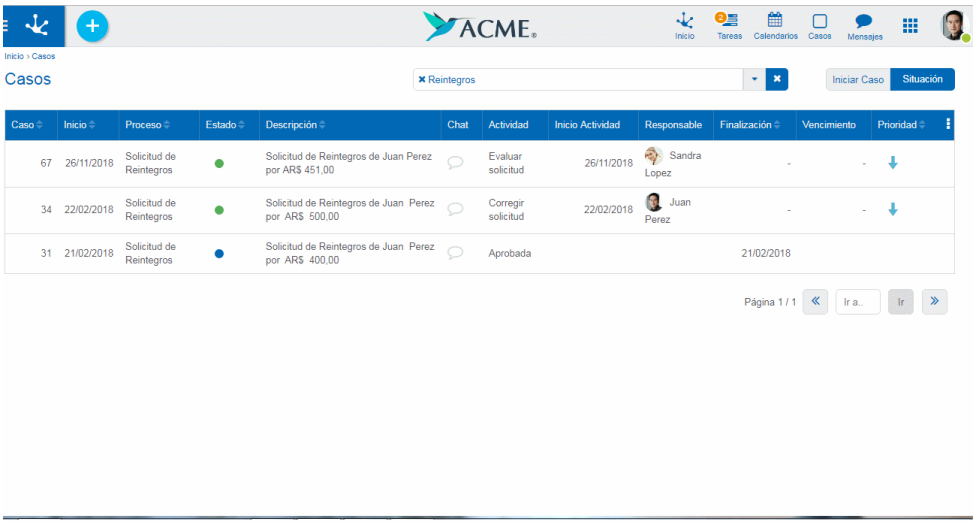
Incorporación / Modificación del logo

- Seleccionar Subir y se abre una ventana con los archivos del equipo del usuario.
- Seleccionar el archivo que contiene la imagen deseada.
- Doble clic para incorporar esa imagen a la barra de herramientas superior.

Eliminación del logo
Seleccionar la opción Eliminar.

Tareas

Muestra la cantidad total de tareas que el usuario tiene pendiente de ejecución. Haciendo clic sobre este ícono se despliega el listado de las últimas tres tareas asignadas al usuario y los correspondientes botones de acción disponibles para cada una de ellas. Al final de la lista figura el enlace todas mis tareas para acceder a una ventana que lista todas las tareas pendientes del usuario.



Calendarios

Se utiliza para consultar y administrar los Calendarios disponibles para el usuario.

Casos

Se utiliza para acceder a los casos iniciados por el propio usuario. Si el usuario es coordinador visualiza también los casos iniciados por los integrantes de su equipo.

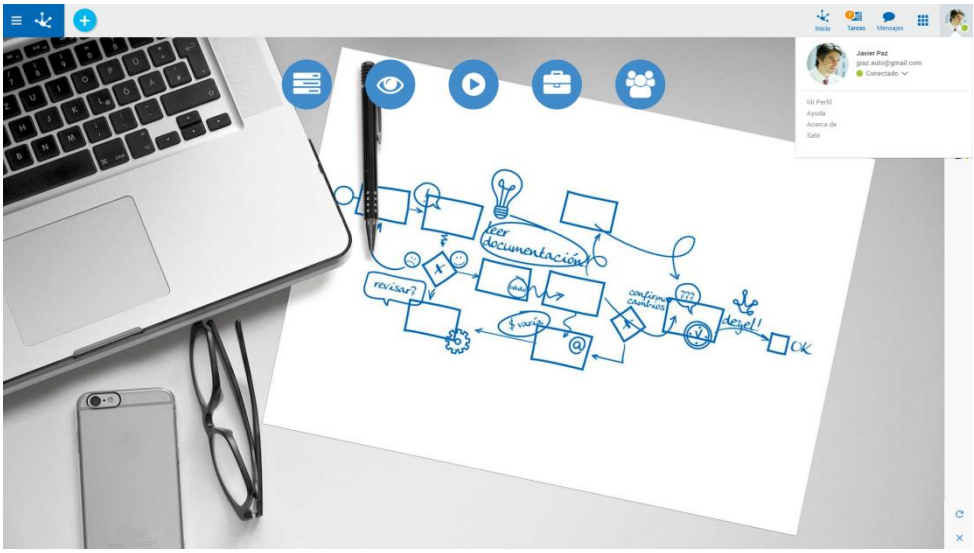
Mensajes

Se utiliza para intercambiar mensajes y participar en las conversaciones relacionadas a las tareas, accediendo a los recursos y beneficios que brinda la Red social empresarial Tedis.

Perfil

Se utiliza para visualizar rápidamente un resumen de los datos del usuario:

- Nombre
- Correo electrónico
- Estado para el servicio de mensajería



Y acceder a las siguientes opciones:

Mi Perfil
Esta opción permite gestionar los datos de perfil del usuario.

Cambio de Contraseña
Permite al usuario cambiar su contraseña.
Esta opción solo está disponible cuando el método de autenticación de usuarios incluye la Autenticación Nativa, donde las palabras clave son verificadas por **Deyel**.
Cuando el usuario es Administrador de Cuenta, la contraseña debe ser gestionada desde el sitio Web de **Deyel**.

Mis Preferencias
Se presentan un conjunto de propiedades para que el usuario pueda establecer el valor que prefiere para cada propiedad.
En la configuración del ambiente se explica con mayor detalle el funcionamiento de esta herramienta.

Ayuda
Acceso a los manuales de usuario.

Acerca de
Presenta información de la instalación y del licenciamiento de las soluciones en uso.

Cerrar Sesión (Logout)
Cierra la sesión en el sistema. Presenta la ventana de Acceso al Portal (Login) para un nuevo inicio de sesión.

3.4.4.1. Menú Contextual

El menú contextual presenta un menú dinámico donde las opciones visualizadas dependen de la funcionalidad que se está utilizando. Permite abrir rápidamente las opciones de menú con diferentes presentaciones.

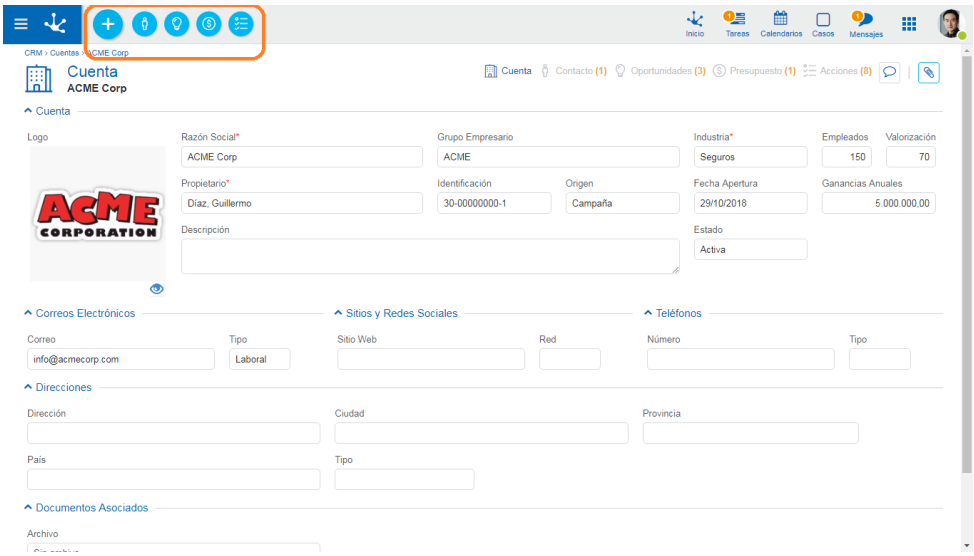
Menú de Iconos

Este menú se habilita al pasar el mouse sobre el ícono y abre sobre la barra de herramientas los íconos que representan los formularios y procesos correspondientes a la funcionalidad utilizada.

Presionando cualquiera de los íconos desplegados se accede a la opción de agregar una instancia del formulario correspondiente o iniciar el proceso que lo gestiona.

Ejemplo

En el siguiente ejemplo se visualizan los formularios y procesos disponibles para el usuario de la solución **Deyel CRM**. Desde la consulta de una instancia seleccionada del formulario Cuenta, el menú contextual de íconos permite iniciar los diferentes formularios asociados a la Cuenta: Contacto, Oportunidad, Presupuesto y Acción.



Menú Desplegado

Este menú se habilita al presionar el ícono , desplegando un panel con las secciones:

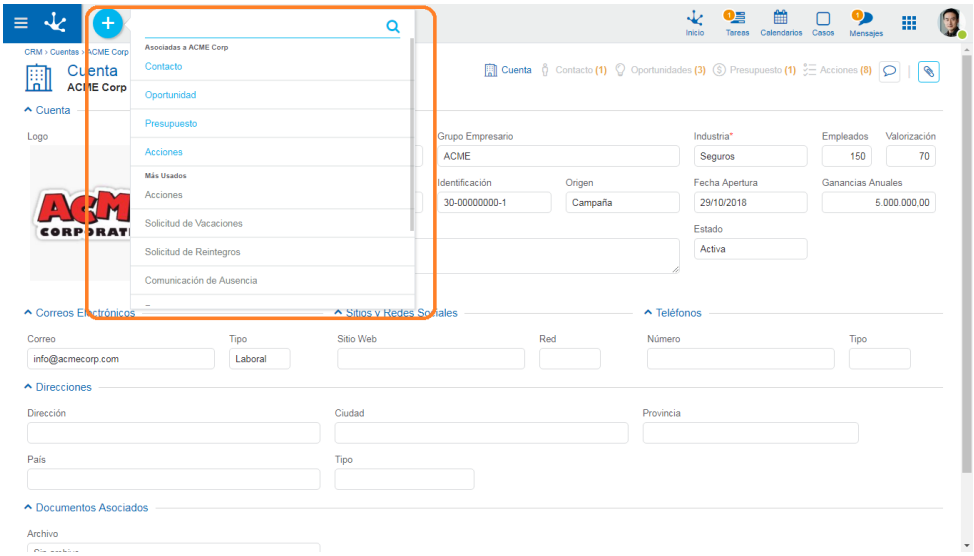
- Lista de formularios y procesos relacionados con la funcionalidad que se está usando (color celeste).
- Lista de formularios y procesos más utilizados por el usuario (color gris).

En ambos casos, presionando una opción, se habilita la funcionalidad para iniciar el proceso o incorporar información al formulario seleccionado.

El ícono permite realizar la acción de búsqueda sobre las opciones listadas.

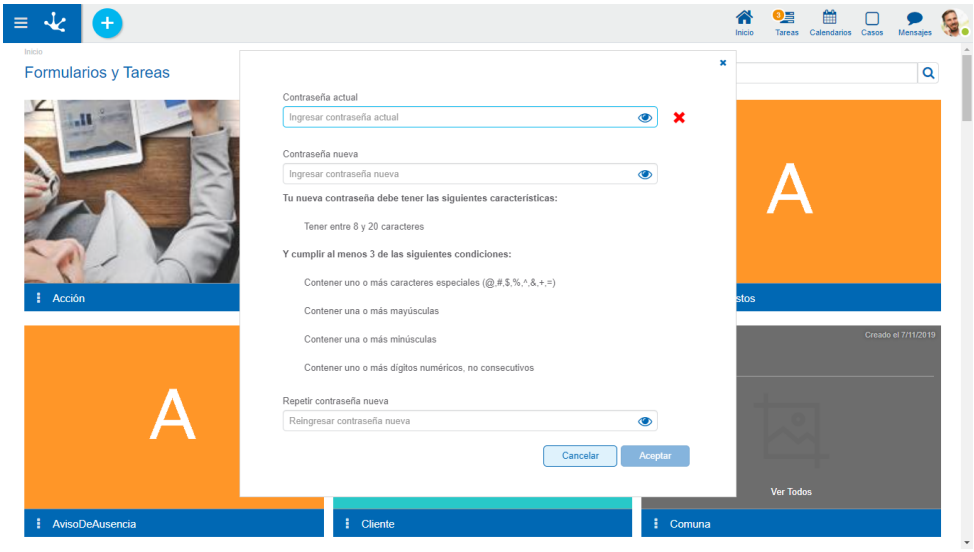
Ejemplo

En el siguiente ejemplo se visualizan los formularios y procesos disponibles para el usuario de la solución **Deyel CRM**. Desde la consulta de una instancia seleccionada del formulario Cuenta, el menú contextual desplegado permite iniciar los diferentes formularios asociados a la Cuenta: Contacto, Oportunidad, Presupuesto y Acción, o bien cualquiera de los objetos que ha utilizado con mayor frecuencia.

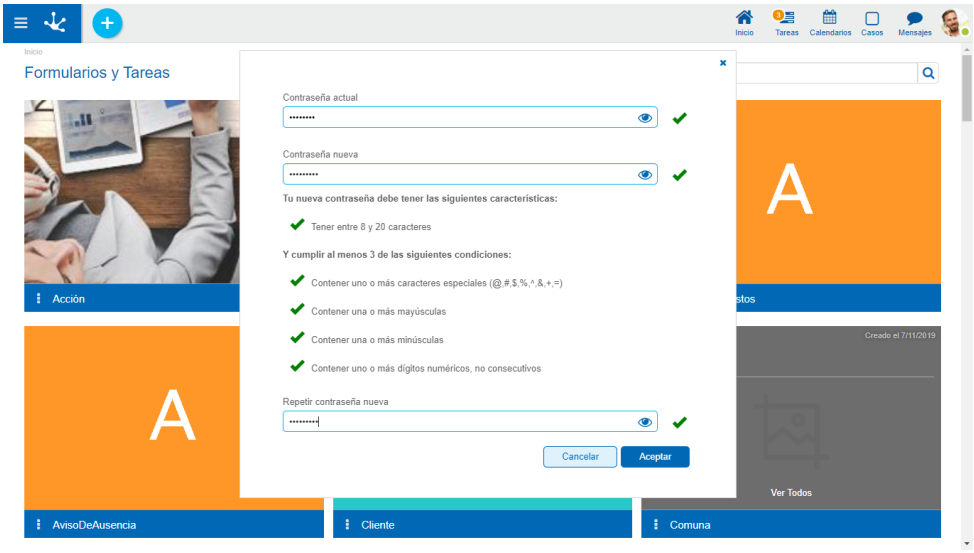


3.4.4.2. Cambio de Contraseña

Se habilita un panel para ingresar la nueva contraseña con las características y condiciones solicitadas.



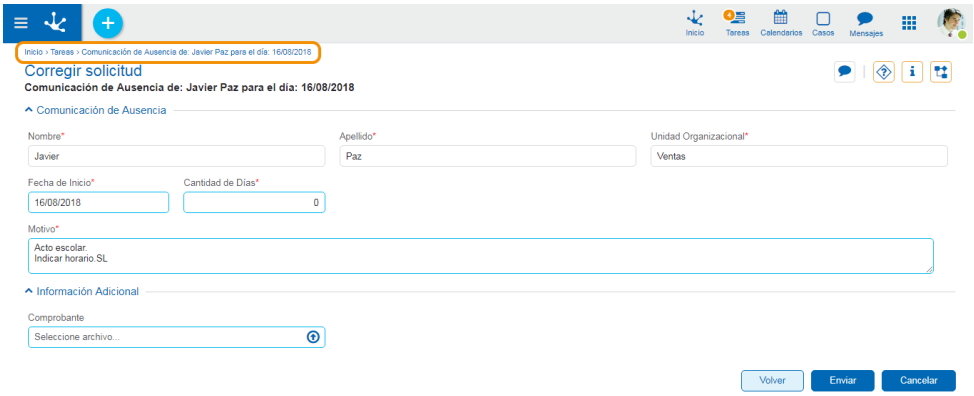
A medida que las características y las condiciones de la nueva contraseña se cumplen se indica con el ícono ✔.



3.4.5. Breadcrumb

Breadcrumb es un recurso disponible que se ubica inmediatamente debajo de la barra de herramientas superior. Presenta la ruta de la navegación realizada y permite retroceder hacia cualquiera de los puntos que informa, simplemente haciendo clic sobre los mismos.

El breadcrumb mejora la experiencia del usuario al indicar en forma resumida los pasos previamente ejecutados, otorgando al mismo tiempo rapidez e intuición en el uso de **Deyel**.



La imagen muestra que se puede navegar por las opciones de menú partiendo de “Formularios Agiles”, pasando por “Formularios y Tareas”, hasta llegar al formulario “Comunicación de Ausencia” para el usuario y fecha seleccionados.

3.4.6. Tableros

Fase 3: Portal > Tableros e indicadores

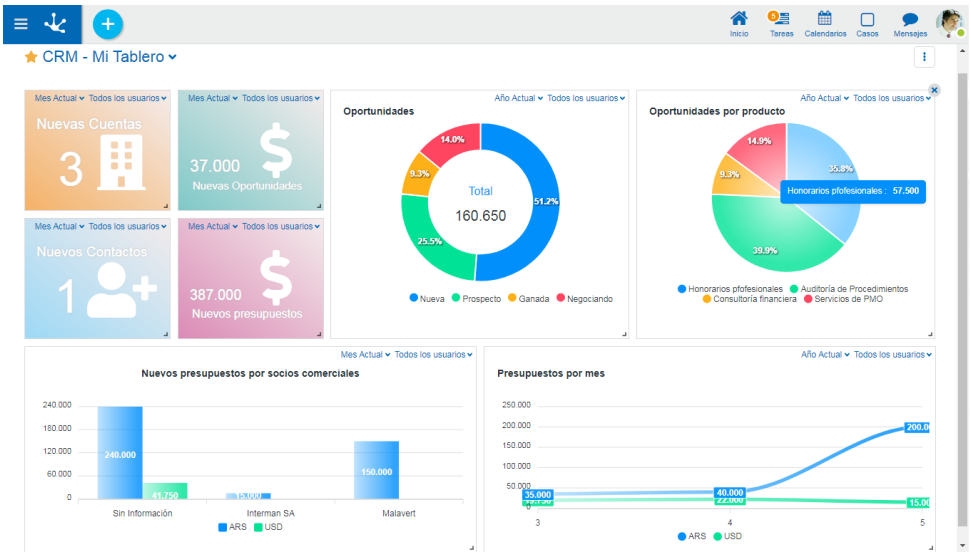
Un tablero es una colección de indicadores ordenados de forma gráfica, que presentan información relevante para el usuario, con alto impacto visual.

Los indicadores se conocen también como KPIs, que corresponden a las iniciales de Key Performance Indicator. Los mismos permiten conocer la evolución del negocio, basándose en el análisis de la ejecución de las aplicaciones.

Los tableros pueden ser:

- De aplicaciones
  Son modelados a través del modelador de tableros, por usuarios modeladores de TI.
- De usuarios
  Todos los usuarios pueden modelar sus tableros propios.

Dependiendo de su perfil, el punto de inicio del usuario, ya sea al iniciar sesión o luego de presionar el ícono , es su tablero favorito. **Deyel** asigna el tablero favorito predeterminado de cada usuario, de acuerdo a sus permisos de **Deyel** para soluciones y de aplicaciones. Cada usuario puede visualizar los tableros sobre los que tiene permiso y puede marcar a uno de ellos como favorito.



> *Si el usuario no tiene permiso para ningún tablero de aplicación o de usuario, el punto de inicio es la galería de formularios y tareas.*

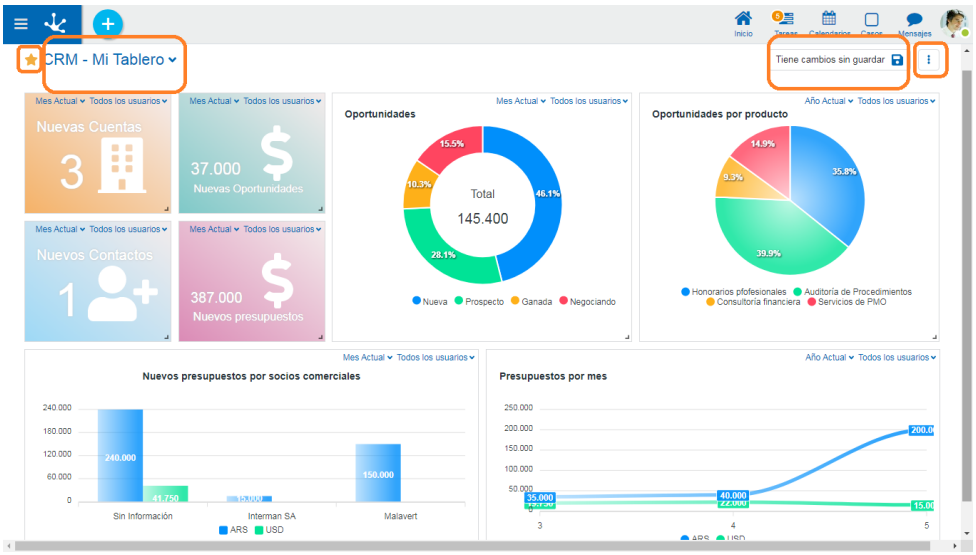
Los usuarios pueden crear sus tableros cuando se encuentran posicionados sobre uno de ellos, sea de usuario o no, accediendo al menú contextual .

- Desde el ícono que se visualiza al pasar el cursor sobre el ícono de menú contextual.
- Desde el menú desplegado, utilizando la opción "Tablero".

Para modificar un tablero, el usuario debe posicionarse en él, seleccionándolo desde el menú de tableros.

3.4.6.1. Facilidades del Uso

Un tablero de usuario presenta recursos visuales y gráficos que facilitan el mantenimiento del mismo en tiempo real.



Secciones

Indicador Favorito

Indica que el tablero está marcado como favorito para el usuario.
Presionando el ícono a la izquierda del nombre del tablero, el usuario puede marcar el tablero como favorito.

Menu de Tableros

A la derecha del nombre del actual tablero, se visualiza una lista desplegable de todos los tableros a los que puede acceder el usuario, pueden ser tableros de aplicaciones o definidos por el usuario.

Area de Información

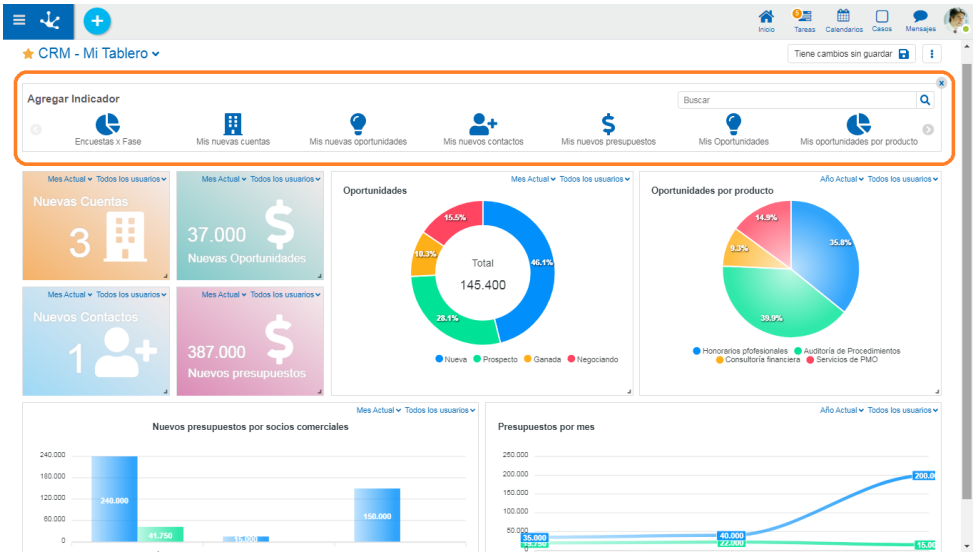
Tiene cambios sin guardar

Cuando el tablero tiene alguna modificación pendiente de guardar, se habilita este botón para guardar los cambios.

Submenú de Operaciones

Permite realizar diferentes operaciones.

- Agregar Indicador
  Permite incorporar al tablero los indicadores sobre los que el usuario tiene permiso de consulta. Para agregar un indicador, se debe arrastrar y soltar el indicador que se desea incorporar, desde la galería de indicadores que se despliega.



- Guardar como
  Guarda el tablero con otro nombre. Este nuevo tablero es de tipo usuario.

- Eliminar Tablero
  Esta operación está permitida solamente para un tablero de usuario. Si el tablero que se elimina está marcado como favorito, el usuario visualiza desde el portal el tablero predeterminado.

Cuerpo del Tablero

En esta sección se ubican los indicadores publicados que fueron agregados al tablero. Sobre estos indicadores, el usuario puede realizar diferentes operaciones.

- Modificar el tamaño
  Se realiza posicionando el cursor en el vértice inferior derecho del indicador y arrastrándolo para ampliar o disminuir el área del indicador, hasta el mínimo permitido.

- Modificar la ubicación
  Por medio de la facilidad "Arrastrar y Soltar" se reubica el indicador en el espacio del tablero.

- Modificar los filtros de usuarios y fechas
  Si el usuario tiene asignado el permiso de modificación para el tablero, puede modificar los filtros que al modelar se definieron como modificables y visibles. Se deben guardar los cambios para conservar la modificación.

- Eliminar
  Un indicador se puede eliminar presionando el ícono ✕ que se visualiza en el ángulo superior derecho, al pasar el cursor sobre el indicador.

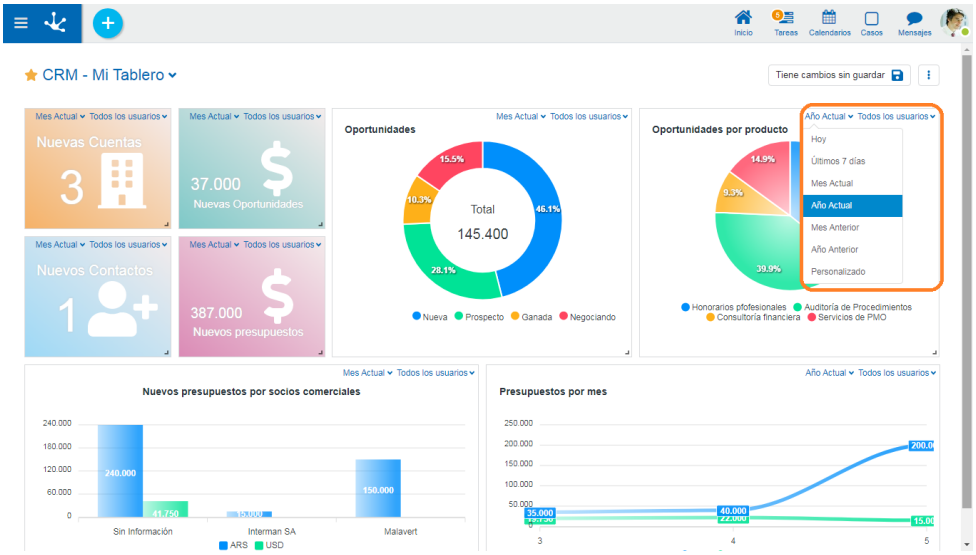
Luego de realizar las modificaciones, al guardar el tablero el resultado puede ser diferente según el tipo de tablero que se modifica.

- Si es de usuario, se guardan los cambios en el mismo tablero.
- Si es de aplicación el usuario no puede guardar los cambios en el mismo tablero. Se visualiza un panel para que el usuario informe el nombre del nuevo tablero que se guarda como de tipo usuario.

Filtros de Indicadores

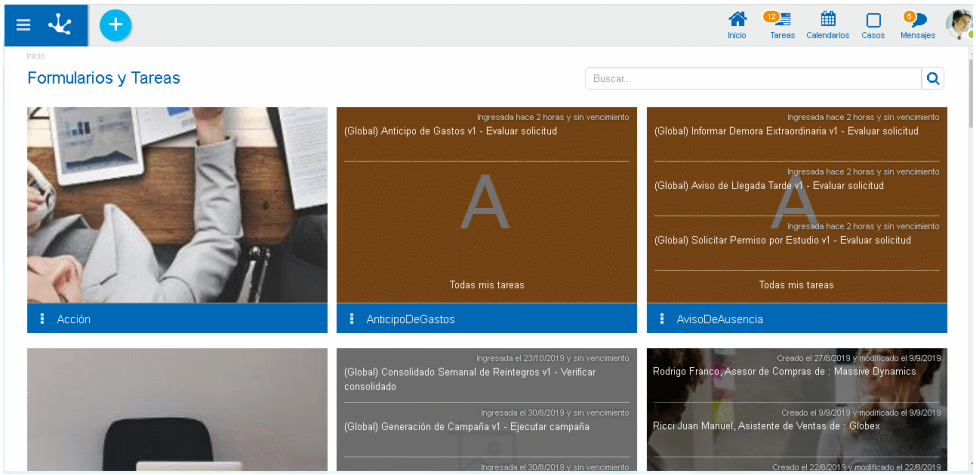
Sobre cada indicador se aplican dinámicamente filtros de fecha y de usuario.

- Filtro de usuarios:
  Si el usuario es responsable de una unidad organizacional o de un rol puede consultar los usuarios pertenecientes a los mismos.
- Filtro de fechas
  El selector de fechas permite informar opciones de períodos.



3.4.7. Formularios y Tareas

Al iniciar sesión, el usuario visualiza la galería de formularios y tareas. Cada elemento representa a un formulario o a un formulario ágil y permite realizar operaciones sobre las instancias de los mismos, con una visión rápida de sus últimas 3 actualizaciones o tareas.



Contenido de los Elementos

Cada elemento de la galería se identifica con el nombre del formulario o formulario ágil previamente modelado. El contenido de cada elemento y las acciones que pueden realizarse dependen si el formulario o el formulario ágil tienen un proceso relacionado.

Ultimas Actualizaciones

Contiene las últimas tres instancias creadas o modificadas del formulario. Al pasar el cursor sobre cada una de ellas, se habilitan los botones correspondientes a las operaciones de consultar, modificar y eliminar, siempre que el usuario tenga los permisos correspondientes.

Al final de la lista, la opción "Ver Todos", permite el acceso directo a la opción "Buscar" del menú de acciones del mismo elemento.

Se visualiza para:

- Un formulario sin proceso relacionado, o con uno o más procesos relacionados y en ninguno de ellos es formulario principal, es decir, no tiene marcada la propiedad Principalen la pestaña "Avanzado" del panel de propiedades del formulario.

- Un formularios ágil sin proceso relacionado.

Cada línea de actualización contiene:

- El texto definido en la propiedad Descripción del formulario.

- La fecha de creación y fecha de última modificación de la instancia.

Ultimas Tareas

Contiene las últimas tres tareas que el usuario conectado tiene pendientes para ejecutar, en cualquiera de los procesos en que el formulario está relacionado y está definido como formulario principal. Al

pasar el cursor sobre cada una de ellas, se habilitan los botones de ejecución y el ícono de acceso al chat del caso.

Al final de la lista, la opción "Todas mis tareas", permite el acceso directo a la opción "Mis Tareas" del menú de acciones del mismo elemento.

Se visualiza para:

- Un formulario con uno o más procesos relacionados, si en alguno de ellos es formulario principal, es decir, tiene marcada la propiedad Principalen la pestaña "Avanzado" del panel de propiedades del formulario.

- Un formularios ágil con proceso relacionado..

Cada línea de tarea contiene:

- El nombre de la tarea, precedido por el nombre y versión del proceso en que se encuentra la misma.

- La fecha de ingreso y la fecha de vencimiento de la tarea.

Menú de Acciones

Se habilita al pasar el cursor sobre el ícono .

Nuevo
Permite crear una instancia de formulario si el elemento contiene últimas actualizaciones, o inciar un nuevo caso si el elemento contiene últimas tareas. Si el formulario está relacionado con más de un proceso como formulario principal, se despliega la lista de procesos para que el usuario seleccione el que se desea iniciar.

Mis Tareas
Abre las tareas que se encuentran bajo responsabilidad del usuario conectado y que corresponden al proceso o a los procesos en que el formulario relacionado se encuentra definido como principal.

Buscar
Abre la grilla de resultados y búsqueda del formulario.

Icono del Formulario
Es la imagen asociada al formulario en la propiedad Icono de la pestaña "General" de propiedades del formulario.

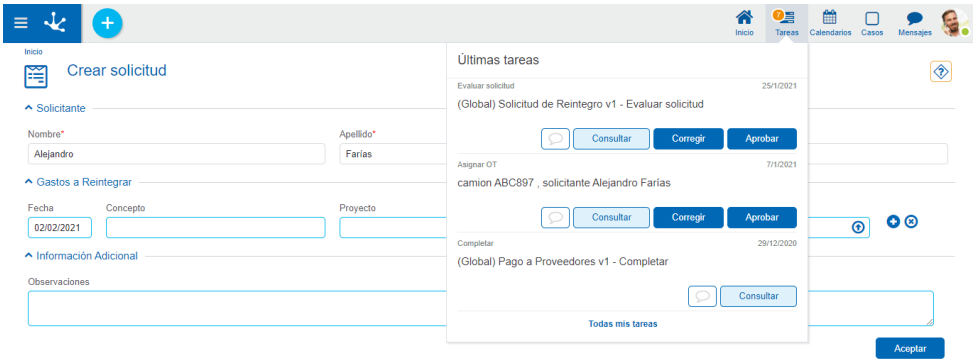
Filtros de Búsqueda

La cantidad de elementos de la galería puede reducirse utilizando un filtro de búsqueda, el cual se aplica al nombre del formulario. A medida que el usuario ingresa caracteres en el campo de búsqueda, la cantidad de elementos en la grilla se reduce a aquellos formularios cuyos nombres contienen los caracteres ingresados.

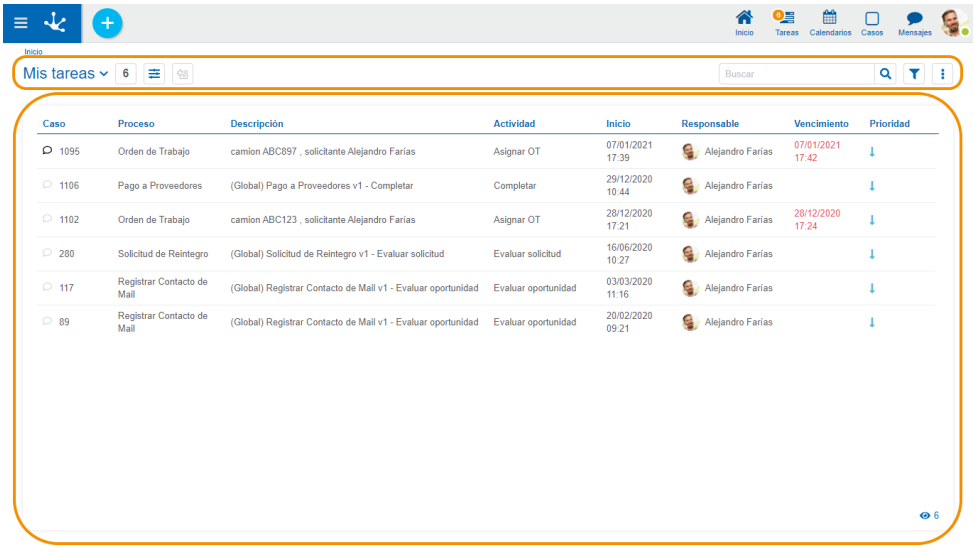
3.4.8. Tareas

Permite visualizar en forma de grilla todas las tareas que se encuentran pendientes de ejecución y están asignadas al usuario o a su equipo, en el caso que se trata de un usuario responsable de una unidad organizacional o rol.

Al presionar el ícono correspondiente a tareas de la barra de herramientas superior se abre un panel donde se visualizan las tres últimas tareas pendientes de ejecución y se puede seleccionar el enlace "Todas mis tareas".



De esta manera se visualizan en forma de grilla todas las tareas que se encuentran pendientes de ejecución por el usuario o su equipo si fuera responsable de una unidad organizacional o rol. Estas tareas corresponden a casos activos.



Secciones

- Barra Superior
- Grilla

3.4.8.1. Barra Superior

En la barra superior se visualiza un conjunto de indicadores que permiten seleccionar el contenido y la presentación de la grilla.

Mis tareas **Mis tareas o Tareas de mi Equipo**

Permite seleccionar una de las siguientes opciones:

- Mis tareas: Lista las tareas pendientes de ejecución del propio usuario.
- Tareas de mi Equipo: Esta opción pueden utilizarla los responsables de unidades organizacionales o de roles para visualizar las tareas que los usuarios a su cargo tienen pendientes de ejecución.

6 **Cantidad de tareas**

Indica la cantidad de tareas pendientes de ejecución.

Totales por Responsabilidad o Totales por Procesos

Si previamente se seleccionó la opción “Mis Tareas”, al presionar este icono se puede acceder al panel de totales por responsabilidad o al de totales por proceso. Visualizar un panel u otro depende de la opción que el usuario seleccione en el título del panel de totales.

Si en cambio se seleccionó la opción “Tareas de mi Equipo”, al presionar este ícono se accede solamente al panel de totales por responsabilidad.

Si se presiona este ícono con el panel de totales abierto, el mismo se cierra y se vuelve a cargar la grilla de tareas.

El panel de totales se puede desplegar u ocultar mediante los íconos correspondientes o .

Versión de la grilla

Permite cambiar a la versión anterior de la grilla o volver a la nueva.

Búsqueda Rápida

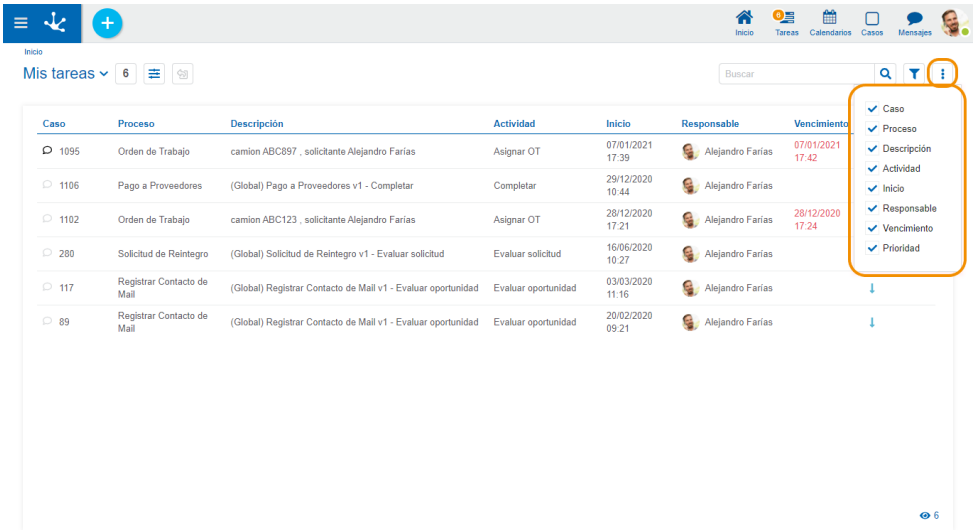
Permite filtrar tareas por las columnas “Caso” y “Descripción” en base a los caracteres ingresados a la izquierda de la lupa.

Busqueda Avanzada

La búsqueda avanzada se habilita al presionar este ícono, mediante la utilización de una barra para ingresar filtros de búsqueda.

Visualización de Columnas

Presionando este ícono se despliega una lista con los nombres de las columnas y un tilde a la izquierda que indica que la misma se visualiza en la grilla. Mediante este tilde el usuario puede activar o desactivar la visualización de cada columna. El conjunto de columnas seleccionadas tiene vigencia hasta tanto el mismo usuario lo vuelva a modificar.



Las columnas de la grilla se pueden configurar mediante parámetros del ambiente.

Al deslizar el mouse sobre el nombre de cada columna se visualizan los iconos que definen el comportamiento de la misma.

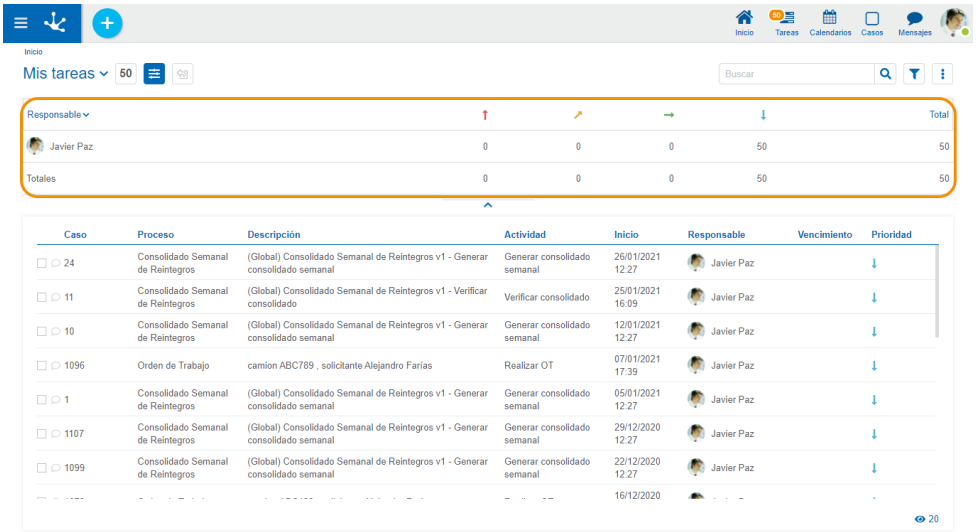
3.4.8.1.1. Totales

Los totales de las tareas se visualizan en un panel que se abre sobre la grilla de tareas.

Totales por Responsabilidad

Muestra la totalidad de las tareas pendientes de ejecución para el usuario o los integrantes de su equipo, si se tratara de un usuario responsable de una unidad organizacional o un rol. Las tareas se visualizan agrupadas por responsabilidad y por la prioridad de cada una de ellas.

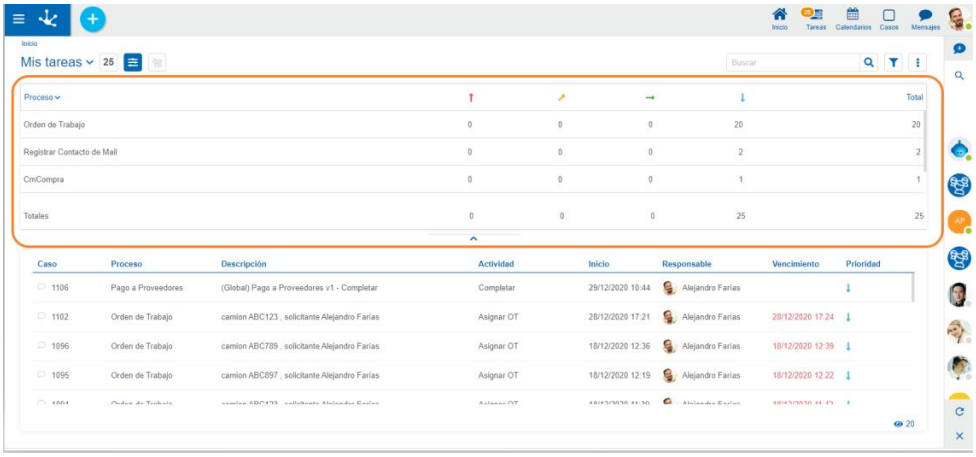
Presionando sobre la cantidad de tareas indicada para cada responsabilidad y prioridad, se puede visualizar la grilla de tareas filtrada por estos criterios.



Totales por Proceso

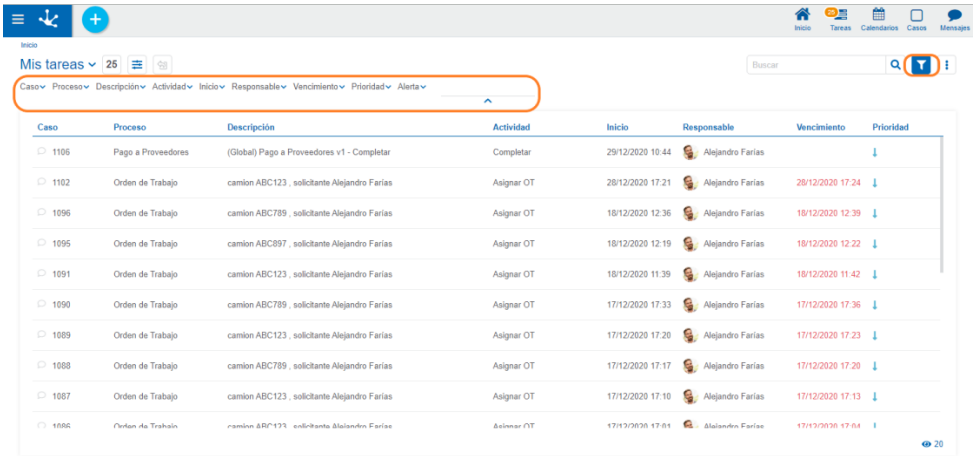
Muestra la totalidad de las tareas pendientes de ejecución para el usuario, agrupando por proceso y prioridad de casos.

Presionando sobre la cantidad de tareas indicada para cada proceso y prioridad, se puede visualizar la grilla de tareas filtrada por estos criterios.



3.4.8.1.2. Filtros de Búsqueda

Al seleccionar el ícono de búsqueda avanzada, se abre una barra que contiene los nombres de todas las columnas, las mismas se pueden utilizar como criterios de búsqueda. Los criterios de búsqueda pueden combinarse.



Por cada criterio de búsqueda se puede ingresar un valor o seleccionarlo de una lista, dependiendo de la opción elegida y aplicar el filtro elegido.

Se pueden agregar todos los filtros que sean necesarios, cada vez que se agrega uno se ejecuta automáticamente una nueva búsqueda que actualiza la grila de tareas.

Los filtros aplicados en la búsqueda quedan resaltados en la barra desde donde se ingresaron, seguido cada uno de ellos con un ícono "x". Presionando dicho ícono se elimina el filtro correspondiente y se ejecuta automáticamente una nueva búsqueda que actualiza la grilla de tareas.

Visualiza la barra de opciones sin modificar la lista de tareas de la grilla.

Oculta la barra de opciones sin modificar la lista de tareas de la grilla. Mantiene los filtros en el caso que los tuviera.

Para eliminar todos los filtros de búsqueda, se debe volver a presionar el ícono de búsqueda avanzada. De esta forma se vuelve a cargar la lista completa de tareas en la grilla.

Filtros

La lista de filtros corresponde a las columnas de la grilla.

Caso

El valor que se ingrese debe ser numérico.
Criterios de búsqueda: "Mayor a", "Menor Igual a".

Proceso, Descripción, Actividad y Responsable

Se ingresa un texto por el que se desea buscar.
Criterio de búsqueda: "Contiene".

Inicio, Vencimiento

Se pueden buscar las actividades que iniciaron su ejecución, o que vencieron en un determinado periodo de tiempo.
Opciones:

- Hoy

- Ultimos 7 días
- Mes Actual
- Año Actual
- Mes Anterior
- Año Anterior
- Rango (Requiere la selección de una fecha inicial y una fecha final)
- Desde (Requiere la selección de una fecha inicial)

Prioridad

Permite filtrar seleccionando los íconos correspondientes al tipo de prioridad.
Criterio de búsqueda: "Igual a".
Opciones:

- Urgente
- Prioridad Alta
- Media
- Prioridad Baja

Alertas

Permiten filtrar seleccionando los íconos correspondientes al tipo de alerta.
Criterio de búsqueda: "Igual a".
Opciones:

- Vencida
- A vencer a corto plazo
- A vencer a largo plazo
- No definida

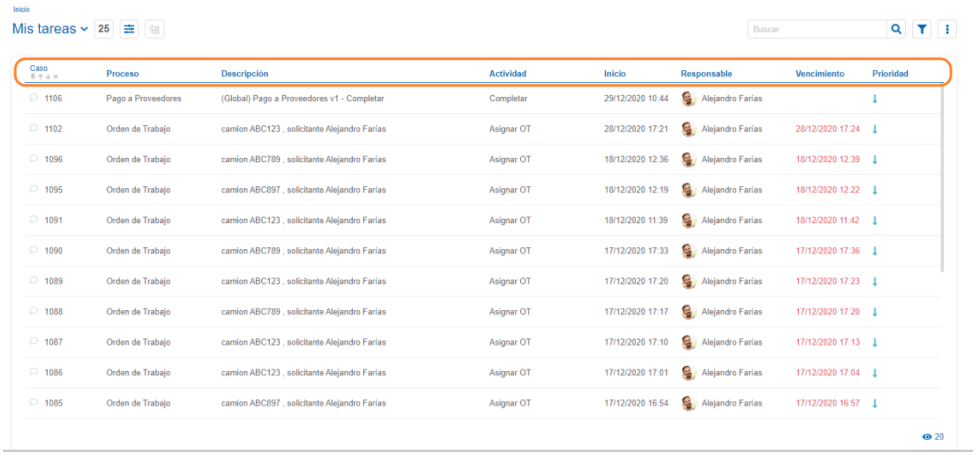
3.4.8.2. Grilla

La grilla de tareas permite visualizar la totalidad de las tareas o bien aquellas que son resultado de haber aplicado filtros de búsqueda. Se compone de columnas y líneas.

Columnas

La grilla de tareas permite visualizar diferentes columnas sobre las que se puede indicar un comportamiento.

Las columnas de la grilla de tareas se visualizan o no según la resolución de la pantalla. A resoluciones más bajas algunas columnas se van ocultando con el objetivo de mantener una lista de tareas legible y sencilla de utilizar.



Chat

Se visualiza un indicador dependiendo que el caso tenga o no un chat asociado. Haciendo clic sobre el ícono se agregan, modifican o consultan las conversaciones asociadas al caso.

El caso no tiene chat asociado.

El caso tiene al menos un chat asociado.

Caso

Es el número del caso al que pertenece la tarea.

Proceso

Es el nombre del proceso al que pertenece la tarea.

Descripción

Breve descripción del caso, definida en la propiedad Descripción del Caso del proceso. Si el proceso correspondiente al caso no tiene definida la descripción, se visualiza el código de proceso, la versión del mismo y el nombre de la actividad actual.

Actividad

Es el nombre de la actividad modelada en el proceso.

Inicio

Indica la fecha y hora de inicio de la actividad.

Responsable

La responsabilidad de una tarea puede estar asignada a un usuario, a un rol o a una unidad organizacional.

- Cuando la tarea es responsabilidad directa del usuario, se visualizan el avatar del usuario y su nombre completo.
- Cuando la tarea es responsabilidad de la unidad organizacional a la cual pertenece el usuario, se visualizan el icono de unidad organizacional y el nombre de la misma.
- Cuando la tarea es responsabilidad de un rol que tiene asignado el usuario, se visualizan el ícono del rol y la descripción.
- Cuando la tarea es responsabilidad de otro usuario, pero ha sido delegada a quien consulta, se visualizan el avatar y el nombre completo del usuario que delega. Esto permite conocer fácilmente quien ha delegado la tarea.

Vencimiento

Indica la fecha y hora en que vence la tarea. Mediante la propiedad Duración de la Actividad<link> se modela el origen de la fecha de vencimiento. La misma se calcula según la duración de la tarea, o si se toma de un campo perteneciente a un formulario asociado al proceso.
Los valores de esta columna se muestran en diferentes colores, logrando una semaforización de las tareas vencidas, las que vencen en el corto plazo o las que vencen con un plazo más largo. Siendo un dato de definición opcional en el modelado de procesos, esta columna puede no tener información.

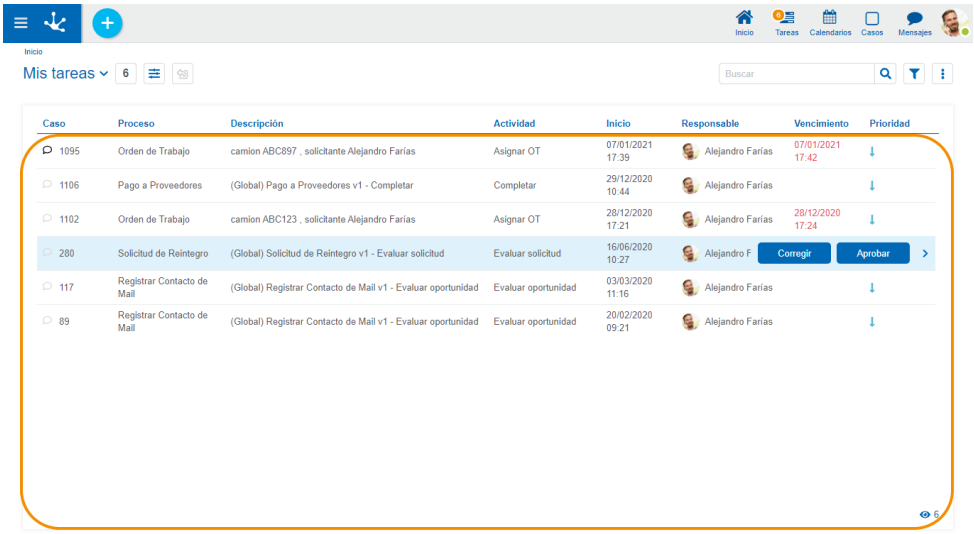
Prioridad

Indica visualmente la prioridad de la tarea.

- Urgente
- Alta
- Mediana
- Baja

Líneas

Una tarea de la grilla puede consultarse haciendo clic sobre la correspondiente línea. Las operaciones disponibles para cada tarea se visualizan al deslizar el mouse sobre la línea.



Visualización de los Botones sobre las Filas

El usuario puede decidir si desea ver los botones con las operaciones de la tarea sobre la línea.

- Visualiza los botones disponibles para cada tarea.
- Oculta los botones disponibles para cada tarea.

Barra de Scroll

Las líneas en la grilla se van completando a medida que se avanza en forma vertical, permitiendo el desplazamiento hacia arriba o hacia abajo entre las tareas visualizadas mediante el uso de una barra de scroll.

En el margen inferior derecho se encuentra un icono y un número que permiten conocer la cantidad de tareas cargadas en la grilla.

3.4.8.2.1. Comportamiento de las Columnas

Si se desliza el mouse sobre cada columna se puede visualizar abajo del título de cada una, un conjunto de íconos que representan su comportamiento.

Indicadores de Comportamiento

Al seleccionar alguno de estos indicadores, se visualiza el icono elegido a la derecha del título de la columna sobre la que se definió el comportamiento. Para revertir la selección se debe hacer clic en este ícono.

El ordenamiento de las líneas se puede realizar solamente por una columna, por lo que sí se selecciona más de un criterio de ordenamiento sólo se toma el último, eliminando siempre el anterior.

Anclaje

Permite dejar la columna fija cuando se desplaza la grilla mediante la barra de scroll horizontal. Se encuentra disponible para la columna “Caso”.

Ordenamiento Ascendente

Permite realizar el ordenamiento de la columna en sentido ascendente. Icono común a todas las columnas.

Ordenamiento Descendente

Permite realizar el ordenamiento de la columna en sentido descendente. Icono común a todas las columnas.

Eliminar la Columna

Permite ocultar la columna.

3.4.8.3. Ejecución de Tareas

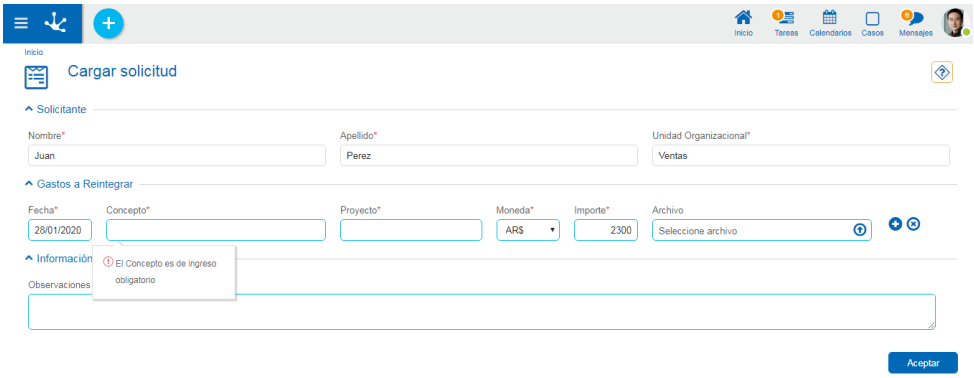
Esta funcionalidad tiene por objetivo ejecutar la una actividad actual y luego evaluar las condiciones para finalizarla. Si se cumplen estas condiciones, permite el pase a la próxima actividad según se haya modelado en el proceso.

Descripción

Es posible acceder a esta función desde diferentes opciones:

- Desde el ícono "Casos" de la barra de herramientas superior del portal, con la opción "Iniciar Caso" se puede solicitar la ejecución de la primera actividad.
- Desde el ícono "Tareas" de la barra de herramientas superior del portal.
- Desde la grilla de tareas.
- Desde la consulta del caso.

En el siguiente ejemplo, se ejecuta la actividad “Cargar solicitud”, que crea una solicitud de reintegro según el proceso modelado a partir de una plantilla de aprobación de un nivel. Al ejecutar la actividad se visualiza el formulario relacionado a la primera actividad del proceso.

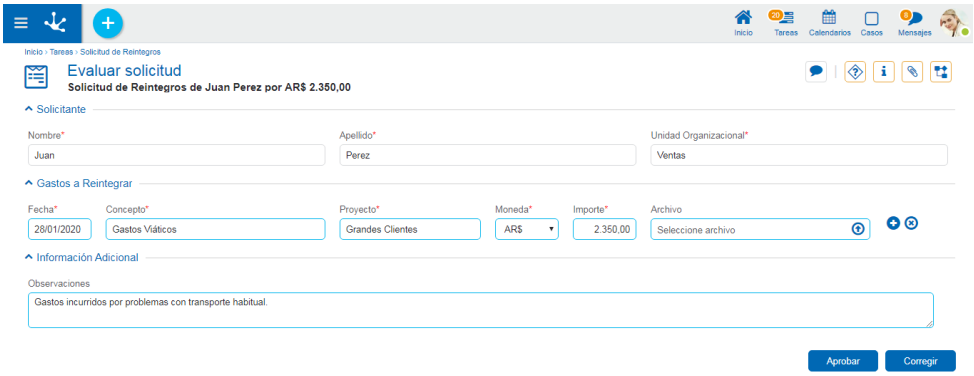


Cuando se ejecuta la primera actividad de un proceso, como en este ejemplo, en el encabezado de la ejecución se muestra el nombre de dicha actividad.

Los datos a ingresar que se muestran con un asterisco en rojo son obligatorios para poder realizar esta actividad según se definió el comportamiento al modelar el formulario y el proceso. En el caso que no se registren dichos datos, se muestra en cada uno de ellos el mensaje indicando que deben ser informados.

Una vez ingresados los datos solicitados, al presionar el botón “Aceptar” se pasa a la siguiente actividad dependiendo de cómo haya sido modelado el proceso.

En las siguientes actividades del proceso, se visualizan ademas los íconos que permiten acceder a información específica del caso.



Comentarios Asociados al Caso

Haciendo clic en este ícono se accede a la Conversación asociada al Caso que está consultando. El título de la conversación es la descripción del caso y los participantes son los usuarios suscriptos.

Ayuda

Haciendo clic en este ícono se muestra la descripción de la actividad. El modelador del proceso puede incluir una descripción detallada de cada actividad. Esta información puede ser consultada al momento de ejecutar la actividad, de manera que el usuario sepa cómo realizar la tarea. Si la actividad no tiene informada descripción, el ícono es deshabilitado.

Información del Caso

Haciendo clic en este ícono se despliega un panel con información del caso. Para cerrar este panel se debe hacer clic nuevamente en el ícono de información.
Para visualizar esta información en una nueva ventana, se debe hacer clic en el ícono que se encuentra en la esquina inferior izquierda del panel.

Adjuntos

Haciendo clic en este ícono se accede una ventana donde se visualiza el detalle de la información asociada: nombre del archivo, usuario, fecha, hora y el origen del adjunto (desde el formulario, chat o caso).

Consulta Gráfica

Esta opción permite visualizar el caso de forma gráfica, con animaciones que muestran cómo fue pasando por las diferentes actividades que componen el proceso al que pertenece y los tiempos de duración de cada tarea.

Finalizar la Actividad Actual

Cuando el usuario termina de ingresar y confirmar los datos, **Deyel** verifica si la actividad actual cumple las condiciones para ser finalizada y entonces pasar a la siguiente actividad. Caso contrario, indica con un mensaje cuál es la condición que no permite finalizar la actividad.

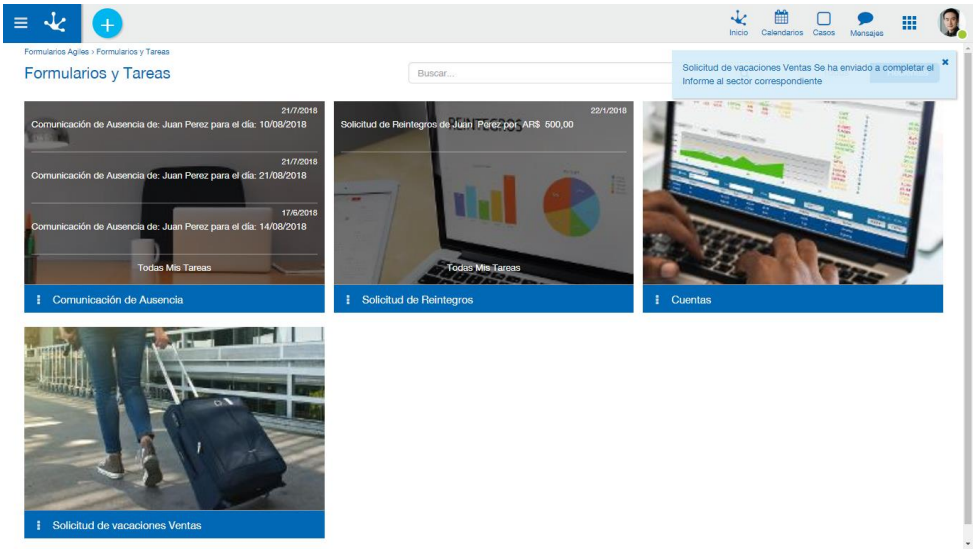
Pasar a la Actividad Siguiente

Cuando se cumplen las condiciones para finalizar la actividad actual, se inicia el procedimiento para pasar a la siguiente actividad. Este procedimiento depende del indicador Ejecución Continua que forma parte de la definición de la ejecución de la actividad actual.

Si este indicador está activado y el pase a la siguiente actividad fue exitoso, **Deyel** analiza la definición de la misma y si corresponde al usuario ejecutarla, entonces se inicia automáticamente dicha ejecución. Este mecanismo permite que el usuario pueda ejecutar una tras otra, de manera continua, las actividades que le competen sobre el caso, sin tener que acceder a la lista de tareas pendientes. Cuando termina de ejecutar una actividad, se le presenta la pantalla para ejecutar la siguiente.

En el caso que el indicador de ejecución continua esté desactivado o bien que la siguiente actividad no corresponde que sea ejecutada por el usuario, entonces se muestra el mensaje de ejecución de la actividad, según como se haya modelado en el proceso.

En la siguiente imagen se visualiza el mensaje de la ejecución:



3.4.9. Casos

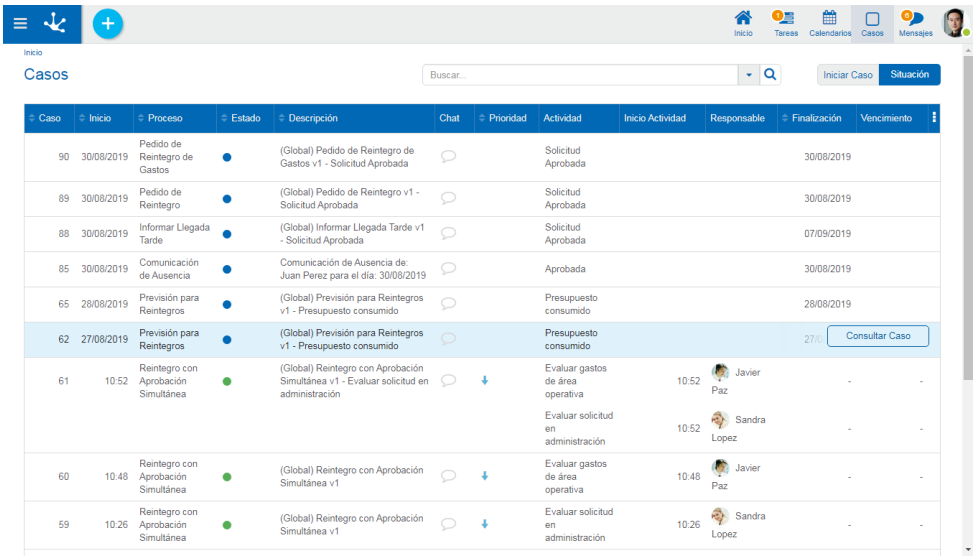
Un caso es un desarrollo particular de un proceso.

Dado un proceso definido, por ejemplo la solicitud de vacaciones o de licencias, una solicitud en particular es lo que se denomina caso.

Un usuario puede consultar sus casos o iniciar nuevos desde la opción "Casos" del menú o ingresando la palabra "Casos" en el buscador del menú y accediendo a la opción.

3.4.9.1. Situación

En la pestaña “Situación” se visualiza la grilla de casos que involucran al usuario. Si el caso tiene más de una actividad en ejecución se visualiza una fila por cada tarea en ejecución.



Columnas

Caso
Número que identifica el caso.

Inicio
Fecha de inicio del caso. Si el caso tiene fecha de inicio en el mismo dia, entonces se visualiza la hora de inicio (incluye horas y minutos). De lo contrario se muestra la fecha y hora de inicio.

Proceso
Nombre del proceso al que pertenece el caso.

Estado
Estado del caso y color que identifica su estado.

- Activo
- Finalizado
- Cancelado
- Suspendido

Descripción
Descripción del caso. Puede ser modificada desde el modelador de procesos. De manera predeterminada la descripción del caso es la conjunción entre el código del proceso, la versión del mismo y el nombre de la actividad que se va a ejecutar.

Chat
Permite visualizar o agregar comentarios en la conversación asociada al caso. Si el ícono se encuentra resaltado existen comentarios para dicho caso. Si el ícono no está resaltado es que aún no se ha agregado ningún comentario. Presionando sobre el ícono se puede acceder al chat para consultar los comentarios existentes o bien para agregar uno nuevo.

Actividad
Nombre de la actividad o actividades en ejecución del caso.

InicioActividad
Por cada actividad en ejecución del caso se visualiza su fecha de inicio. Si la actividad tiene fecha de inicio el mismo día, entonces se visualiza la hora de inicio (incluye horas y minutos). De lo contrario se muestra la fecha de inicio y haciendo clic sobre la misma se despliega una ayuda visual, indicando también la hora.

Responsable
Por cada actividad en ejecución del caso, se visualiza el responsable de su ejecución. No se informa responsable si el caso está finalizado. Puede ser un usuario, un rol o una unidad organizacional y se representa de diferentes formas.

- La tarea es responsabilidad directa del usuario, en la grilla se visualiza la imagen del usuario y su nombre completo.
- La tarea es responsabilidad de la unidad organizacional a la cual pertenece el usuario, se visualiza el icono de la unidad organizacional y el nombre de la misma.
- La tarea es responsabilidad de un rol que tiene asignado el usuario, en el listado se visualiza el ícono del rol y la descripción.
- La tarea es responsabilidad de otro usuario pero ha sido delegada a quien consulta, se visualiza la imagen y el nombre completo del usuario que consulta.

Finalización
Fecha de finalización de la última actividad del caso, su comportamiento es como el de la propiedad Inicio Actividad.

Vencimiento
Por cada actividad en ejecución del caso, indica su fecha de vencimiento. Las fechas son visibles con una semaforización, indicando las vencidas, en riesgo o en tiempo. En el caso que en el modelado de la actividad no se indique que tenga duración o fecha de vencimiento, la actividad no muestra dicha fecha.

Cálculo de la Fecha de Vencimiento y Semaforización
Haciendo clic sobre una fecha se puede verificar el tiempo que lleva vencida la tarea, mediante una ayuda visual, siendo el tiempo especificado diferentes criterios.

- Más de 30 días vencida, cuando el plazo es mayor a 30 días.
- DD HH, días y horas cuando el plazo es mayor a un día.
- HH MM, horas y minutos cuando el plazo es menor que un día y mayor o igual a una hora.
- MM SS, minutos y segundos cuando el plazo es menor que una hora y mayor o igual que un minuto.
- SS, solo segundos, cuando el plazo es menor que un minuto.

Prioridad
Indica la prioridad que tiene la tarea mediante íconos de diferentes colores.

Urgente

Alta

Mediana

Baja

Operaciones

Al pasar el cursor sobre las filas de los casos se habilita el botón "Consultar Caso", para consultar el caso. Si el usuario puede ejecutar la actividad actual, se habilita también el botón "Consultar Tarea", que permite ejecutar la tarea.

3.4.9.1.1. Búsqueda de Casos

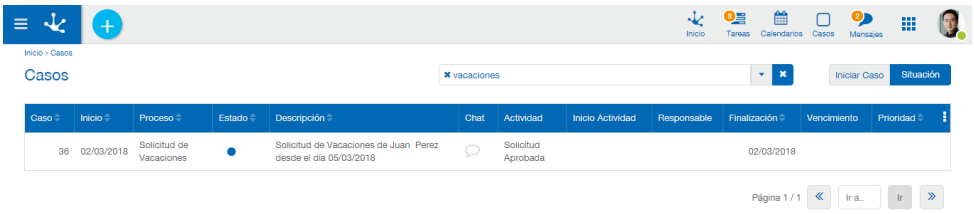
Se pueden realizar dos tipos de búsqueda.

- Rápida
- Avanzada

Búsqueda Rápida

Permite recuperar todos los casos cuyas descripciones contengan una o más palabras elegidas.
Se debe posicionar el cursor en la sección de búsqueda, ingresar el texto que se desea buscar y hacer clic en la lupa.

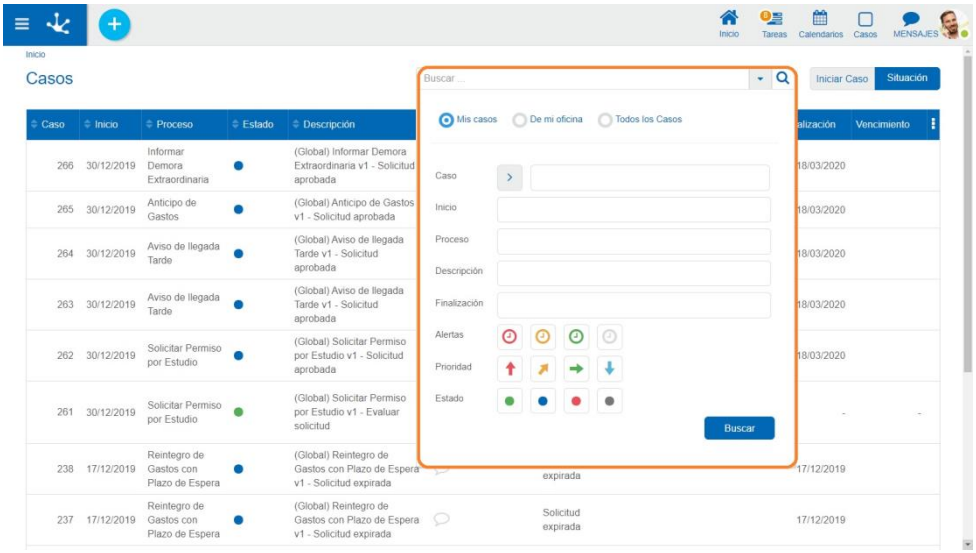
Se filtran de esta forma todos los casos en cuya descripción exista el texto ingresado.



Para eliminar el filtrado se hace clic sobre el ícono ✖ que aparece a la izquierda del texto buscado.

Búsqueda avanzada

Para acceder a la búsqueda avanzada se debe hacer clic sobre el ícono a la derecha del campo Buscar.
Se abre de este modo una ventana que permite indicar diferentes criterios de búsqueda dentro de la grilla de casos, como se muestra en la siguiente imagen:



Como primera opción se detalla si se desea buscar:

- Mis casos: Lista los casos iniciados por el usuario que está consultando.
- De mi oficina: Lista los casos iniciados por el propio usuario o por los usuarios integrantes de algún rol o unidad organizacional donde el usuario es el responsable.
- Todos los casos: Lista todos los casos existentes en Deyel. Esta opción es visible si el usuario tiene la función de seguridad entre sus permisos.

Luego se puede filtrar por:

- En caso se puede buscar por mayor o menor de un determinado número.
- En proceso y descripción se puede buscar por el texto que se desea.
- En Inicio se puede buscar por un rango de fechas en la que se inicia el caso, lo mismo que para Finalización.
- Alertas, Prioridad y Estado permite filtrar seleccionando los íconos correspondientes.

Tipos de Búsqueda

Mis casos

Lista los casos iniciados por el usuario que está consultando.

De mi oficina

Lista los casos iniciados por el propio usuario o por los usuarios integrantes de algún rol o unidad organizacional donde el usuario es el responsable.

Todos los casos

Lista todos los casos existentes en Deyel. Esta opción es visible si el usuario tiene la función de seguridad entre sus permisos.

Filtros

Caso

Se puede buscar por mayor o menor de un determinado número.

Proceso y Descripción

Se puede buscar por el texto que se desea.

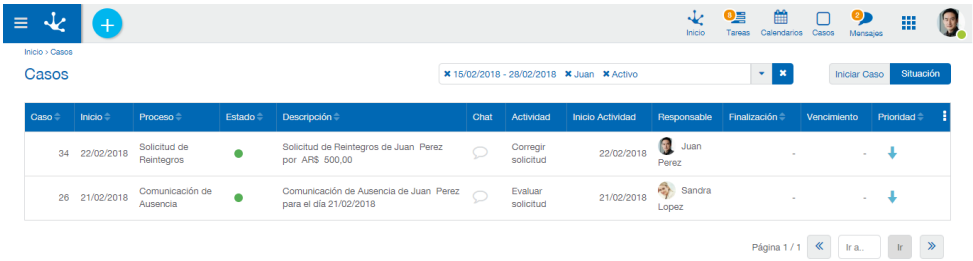
Inicio

Se puede buscar por un rango de fechas en la que se inicia el caso, lo mismo que para Finalización.

Alertas, Prioridad y Estado

Permite filtrar seleccionando los íconos correspondientes.

Los filtros aplicados en la búsqueda quedan visibles en la sección de búsqueda precedidos cada uno de ellos con el ícono ✖. Haciendo clic sobre este ícono se elimina el filtro correspondiente y se ejecuta automáticamente una nueva búsqueda que actualiza la grilla de casos visualizados.



Para eliminar cada uno de los criterios, se hace clic sobre la cruz y ese criterio se elimina de la búsqueda actualizando la lista de casos.

3.4.9.1.2. Ordenamiento y Visibilidad

Las columnas de la grilla de casos son configuradas en forma general al momento de definir los parámetros de la instalación.

Sobre cada columna de la lista y del lado derecho del nombre de la misma, figuran los íconos para ordenar por esa columna.

Cada vez que se hace clic sobre este ícono se cambia el sentido del ordenamiento, a ascendente, a descendente o sin ordenamiento por esa columna (vuelve al ordenamiento estándar).

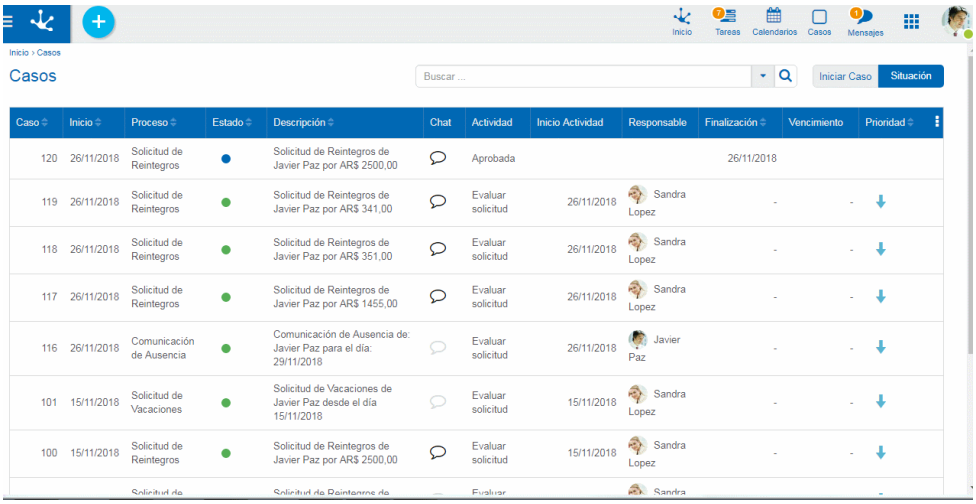
El ordenamiento es siempre por una columna, por lo que si se tilda más de un criterio de ordenamiento sólo se toma el último, limpiando siempre el anterior.

A la derecha del título de cada columna, el ícono permite dejar de visualizar la columna.

Preferencia del Usuario

Las columnas configuradas inicialmente, pueden ser ajustadas según las preferencias y necesidades de cada usuario.
Cada usuario puede activar o desactivar la visualización de cada columna en todo momento.

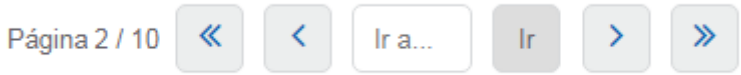
Sobre el margen izquierdo de la lista, en la barra de títulos de la lista, figura el ícono . Presionándolo se despliega la lista de las columnas para seleccionar las que desean visualizar.
Estas modificaciones tienen vigencia hasta tanto el mismo usuario las vuelva a modificar.



Haciendo clic sobre el mismo icono la ventana se cierra.

Paginado

Debajo de la lista de casos se encuentra el paginado:



En el paginado se indica la página que está visualizando y el total de páginas de casos que existen.

Funcionamiento del paginado

Presione para ir a la página anterior.

Presione para ir a la página posterior.

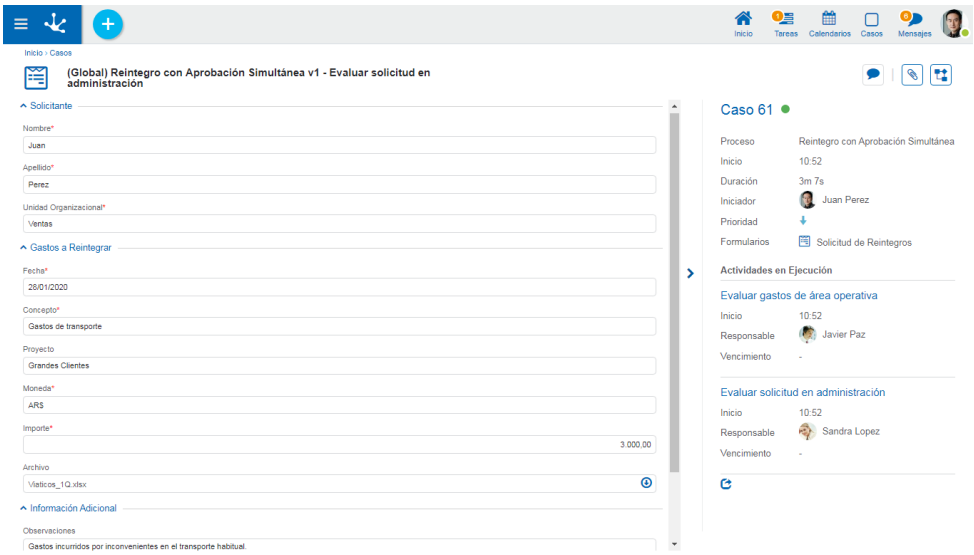
Presione para ir a la primera página.

Presione para ir a la última página.

En "Ir a..." se permitirá poner un nro. de página a la que se desea ir que deberá ser menor o igual al número de página total que tiene la lista, y al presionar el botón Ir, se pasará a la página indicada.

3.4.9.2. Consulta de Casos

En el panel de la consulta de un caso, se visualizan el ícono del proceso y la descripción del caso, en la parte superior izquierda y los íconos que permiten acceder a información específica del caso, en la parte superior derecha.



Información del Caso

Comentarios Asociados al Caso

Haciendo clic en este ícono se accede a la conversación asociada al caso que está consultando. El título de la conversación es la descripción del caso y los participantes son los usuarios suscriptos.

Adjuntos

Haciendo clic en este ícono se accede una ventana donde se visualiza el detalle de la información que fue adjuntada al caso: nombre del archivo, usuario, fecha, hora y el origen del adjunto (desde el formulario, chat o caso).

Consulta Gráfica

Esta opción permite visualizar el caso de forma gráfica, con animaciones que muestran cómo fue pasando por las diferentes actividades que componen el proceso al que pertenece y los tiempos de duración de cada tarea.

Panel Izquierdo

En el panel izquierdo se muestra la consulta delformulario principaldel caso.

Panel Derecho

En el panel derecho se muestra la información del caso, indicando el número que lo identifica.

Estado
Estado del caso y color que identifica su estado.

- Activo
- Finalizado
- Cancelado
- Suspendido

Proceso
Nombre del proceso al que pertenece el caso.

Inicio
Fecha de inicio del caso. Si el caso tiene fecha de inicio en el mismo día, entonces se visualiza la hora de inicio (incluye horas y minutos). De lo contrario se muestra la fecha y hora de inicio.

Duración del caso
Se informa en días, horas, minutos y segundos.

Iniciador
Usuario que inició el caso.

Prioridad
Indica la prioridad que tiene la tarea mediante íconos de diferentes colores.

- Urgente
- Alta
- Media
- Baja

Formularios
Formularios creados durante la ejecución del caso.

Actividades en Ejecución

Visibles solo si el caso no está finalizado. Son las actividades del caso que se encuentran en ejecución al momento de la consulta.

Inicio
Fecha de inicio de la actividad. Si la actividad tiene fecha de inicio en el mismo día, entonces se visualiza la hora de inicio (incluye horas y minutos). Delo contrario se muestra la fecha y hora de inicio.

Responsable
Responsable de la ejecución de la actividad. No se informa responsable si el caso está finalizado. Puede ser un usuario, un rol o una unidad organizacional y se representa de diferentes formas.

- La tarea es responsabilidad directa del usuario, se visualiza la imagen del usuario y su nombre completo.
- La tarea es responsabilidad de la unidad organizacional a la cual pertenece el usuario, se visualiza el icono de la unidad organizacional y el nombre de la misma.
- La tarea es responsabilidad de un rol que tiene asignado el usuario, se visualiza el ícono del rol y la descripción.
- La tarea es responsabilidad de otro usuario pero ha sido delegada a quien consulta, se visualiza la imagen y el nombre completo del usuario que consulta.

Vencimiento
Indica la fecha en que vence la tarea en ejecución, o la hora de vencimiento cuando la misma vence en el día actual.
Mediante la propiedad Duración de la Actividad se modela el origen de la fecha de vencimiento. La misma se calcula según la duración de la tarea, o se toma de un campo perteneciente a un formulario asociado al proceso.
La fecha se muestra en diferentes colores, logrando una semaforización de las tareas vencidas, las que vencen en el corto plazo o las que vencen con un plazo más largo. Siendo un dato de definición opcional en el modelado de procesos, la fecha puede no estar informada.

Botón "Consultar"
Es visible cuando la actividad puede ser ejecutada por el usuario que consulta. Permite acceder a la ventana de ejecución de la tarea.

Casos Finalizados

Si el caso está finalizado solamente se muestran algunas propiedades.

Ultima Actividad
Actividad con la que finalizó el caso.

Finalización
Fecha de finalización de la última actividad del caso. Se visualiza de la misma manera que fecha de inicio de la actividad.

Cierre del Panel Derecho

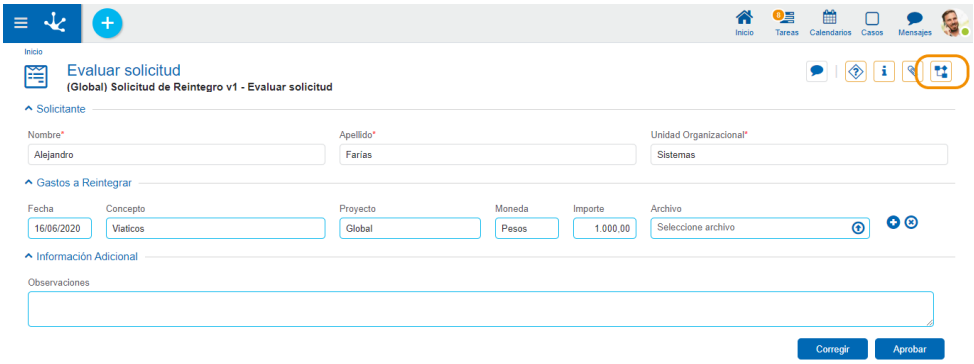
Cierra el panel derecho de información del caso.

Nueva Ventana

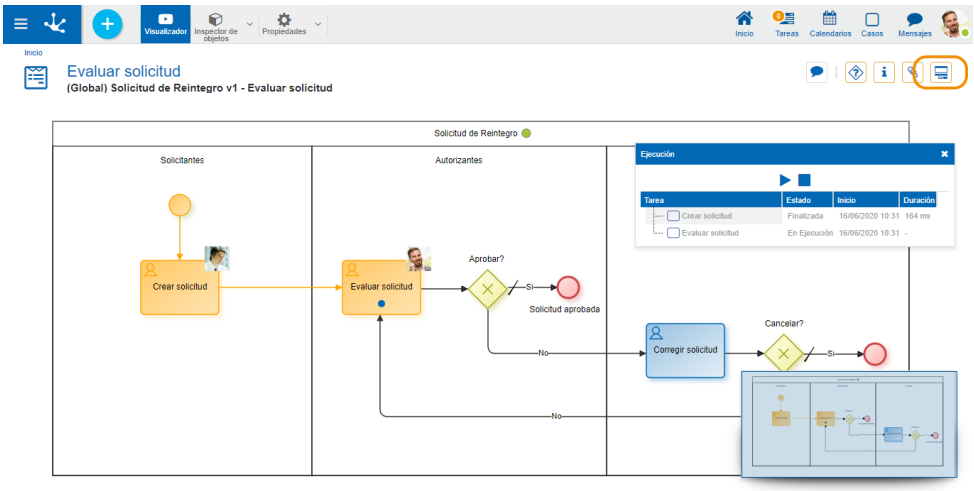
Para visualizar la información del caso en una nueva ventana del navegador, se debe hacer clic en el ícono que se encuentra en el ángulo inferior derecho del panel lateral.

3.4.9.3. Consulta Gráfica

El usuario puede ingresar a esta funcionalidad desde el ícono correspondiente a la definición gráfica dentro de la consulta puntual del caso.



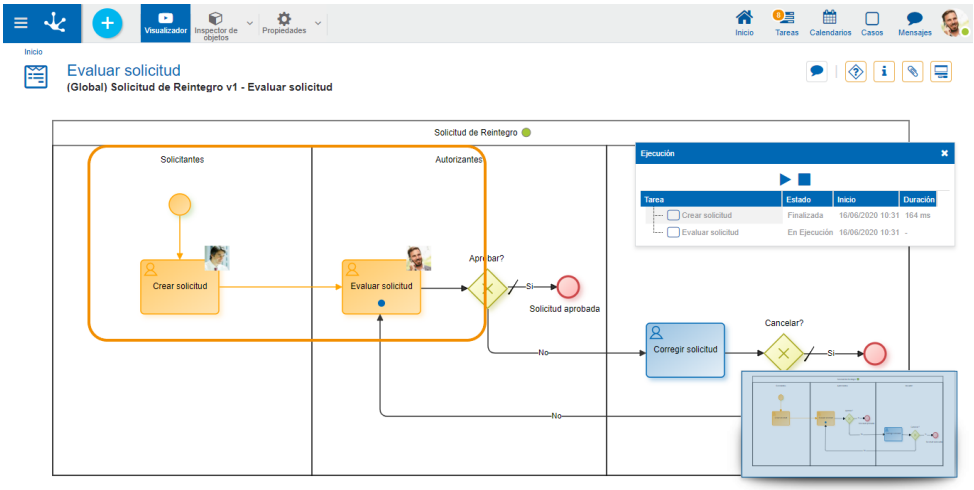
Se abre la ventana del visualizador del proceso, compuesta por el diagrama del proceso y un conjunto de herramientas. Para volver a la consulta puntual del caso, se debe presionar el ícono



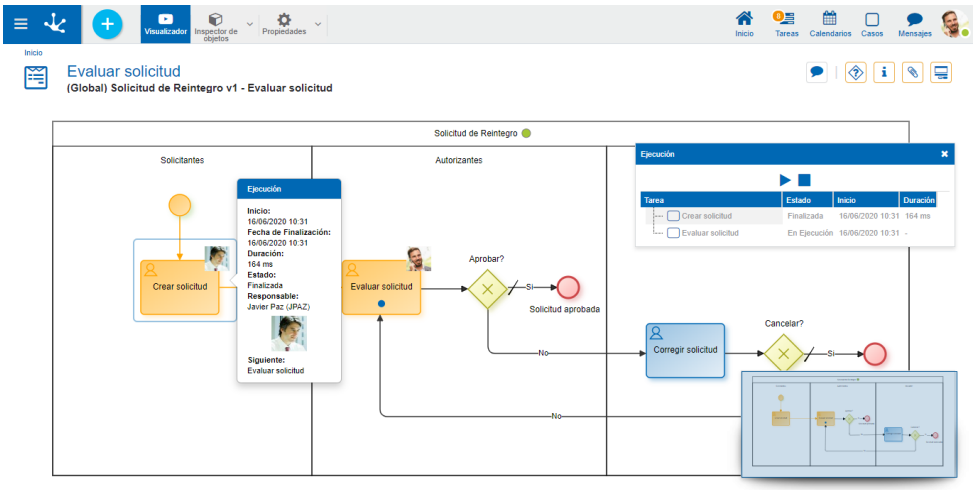
Ejecución de Actividades

Dentro del gráfico del proceso, se visualizan las actividades ejecutadas del caso en color naranja. Además, se puede observar en la parte superior derecha de cada actividad, la imagen del responsable de la ejecución.

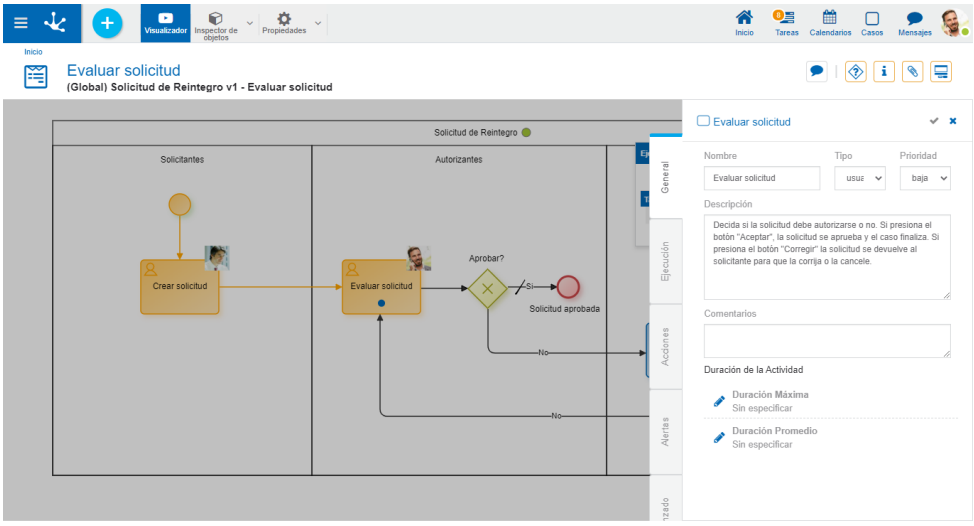
Si la actividad contiene en su interior un punto azul, significa que dicha actividad se encuentra en ejecución.



Al pasar el mouse sobre una actividad ejecutada, se visualiza un panel con información de la ejecución de la actividad, actividades automáticas ejecutadas e información de los eventos capturados. También se visualiza la próxima actividad o evento a ejecutar.

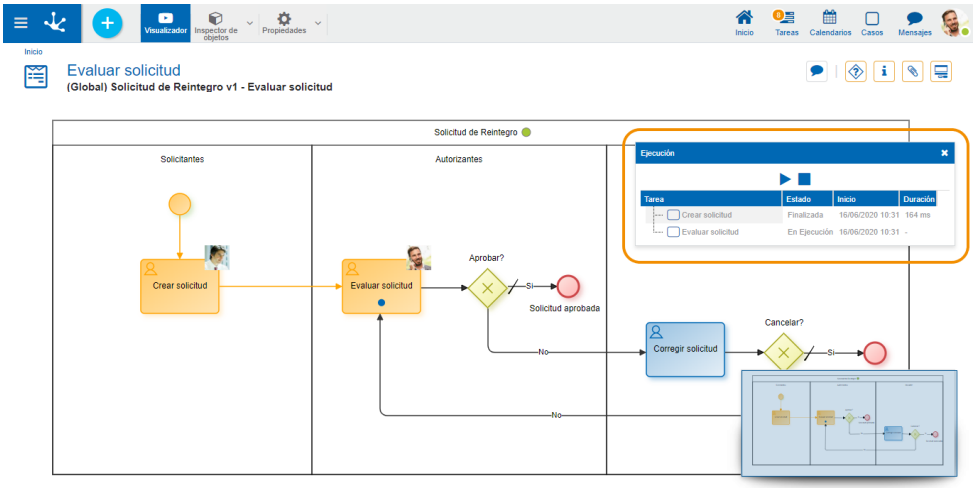


Se puede seleccionar cualquiera de los elementos del diagrama del proceso para abrir su panel de propiedades, donde no pueden ser modificados sus valores, como en el modelado del proceso.



Visualizador

Al presionar el ícono Visualizador se abre el panel de ejecución, donde se visualiza la lista de actividades ejecutadas, con su información resumida.



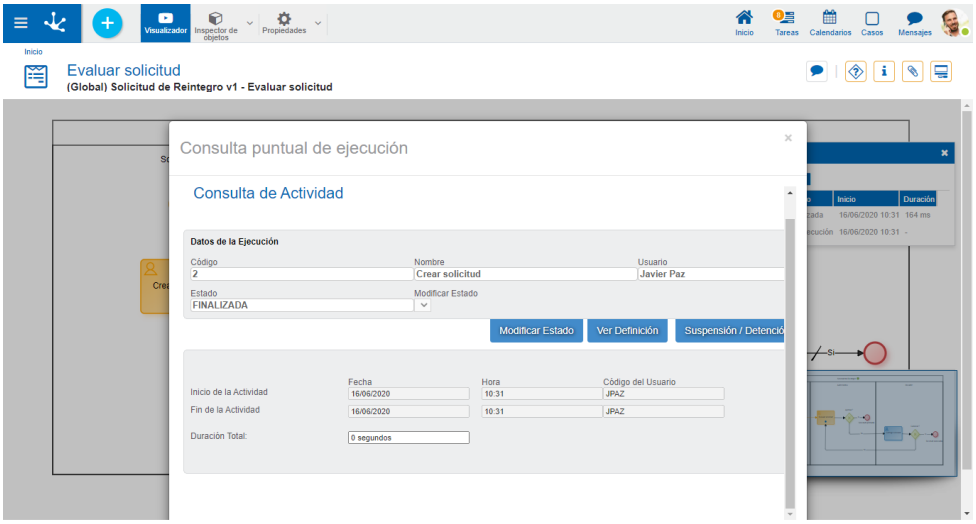
Este panel permite simular la ejecución del caso mediante tres íconos:

▶ Comienza la simulación de la ejecución. Mientras se realiza la simulación se visualiza el ícono correspondiente a la pausa.

⏸ Pausa la simulación. Una vez detenida la misma, se visualiza el ícono correspondiente a la ejecución.

■ Finaliza la simulación de la ejecución.

Al seleccionar una actividad de la lista en el visualizador, se abre una ventana con la información detallada de la misma.



3.4.9.4. Adjuntar Información

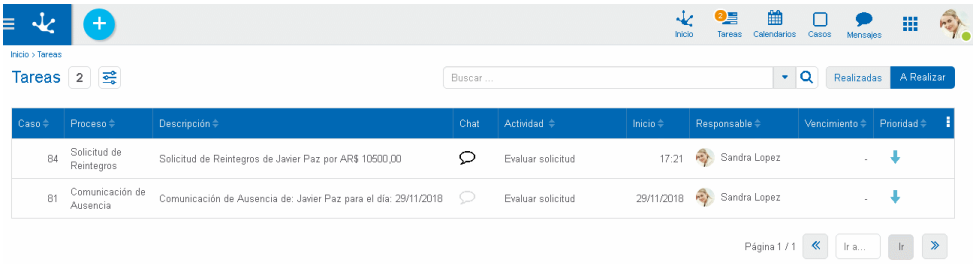
Un caso puede tener información adjunta en forma de archivos, imágenes, enlaces a Google Drive y Google Docs, la misma se pueda adjuntar de diferentes formas.

- Desde el formulario, cuando en el proceso de modelado se incorporan campos tipo Archivo, Imagen o Google Doc. El caso debe encontrarse en una actividad que posibilita actualizar información y el usuario debe tener permiso para hacerlo.

- Desde el chat asociado al caso. Todos los usuarios intervinientes pueden adjuntar información en todo momento. Esta información queda disponible para consulta aún cuando el caso se ha cerrado.

- Desde la consulta del caso, el usuario responsable de ejecutar la actividad en que se encuentra el caso puede adjuntar información almacenada localmente en su equipo o en Google Drive.

Para acceder a la ventana de Adjuntos, hacer clic en el ícono y para retornar a la ventana del caso hacer clic en el ícono .



Desde la ventana de adjuntos se visualiza toda la información adjuntada al caso y se indica el modo en que se adjuntó. Esta posibilidad le otorga al usuario facilidad y rapidez para la consulta de dicha información.

Indica que el archivo fué adjuntado desde el formulario.

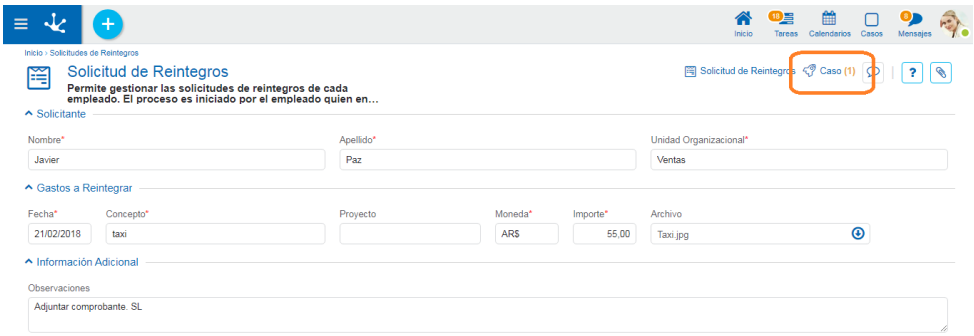
Indica que el archivo fué adjuntado desde el chat del caso.

Indica que el archivo fué adjuntado directamente del caso.

También se visualiza el nombre del usuario que adjuntó la información, fecha y hora de la acción y nombre del archivo o enlace.
Pasando el mouse sobre cada adjunto se habilita la posibilidad de descargar el archivo al equipo local del usuario.
La información adjuntada solo la puede eliminar el usuario que la adjuntó.

3.4.9.5. Casos Vinculados a un Formulario

Los casos vinculados a un formulario son los casos donde el formulario es o fue utilizado. Si el formulario tiene marcada la propiedad Mostrar Casos Vinculados, desde las instancias del formulario se visualiza la opción para poder consultar sus casos vinculados y la cantidad de los mismos.



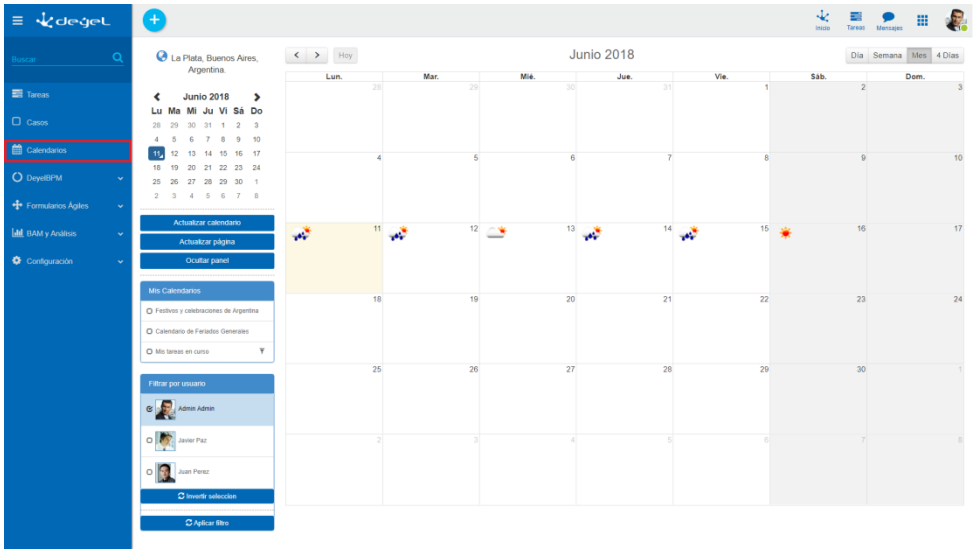
Al hacer clic sobre el ícono correspondiente al caso o casos vinculados, se accede a la ventana de consulta del caso cuando solo hay un caso vinculado, o a la grilla de casos vinculados, cuando existe más de uno.

3.4.10. Calendarios Gráficos

Fase 3: Portal > Calendario y herramienta colaborativa

Para utilizar la funcionalidad de calendarios gráficos existen distintas alternativas.

- Hacer clic sobre la opción " Calendarios" del menú principal.
- En la búsqueda rápida del menú ingresar " Calendarios" y hacer clic en la opción listada.



En la porción superior central de la ventana se indica la fecha o periodo que se está considerando para la visualización de eventos dentro del **panel de eventos**.
Este panel de eventos se encuentra debajo y es donde se visualizan los indicadores del clima y los eventos de los calendarios.

Hacia la derecha se presentan una serie de botones que permiten controlar la forma de visualización de los eventos:

"Día" - Visualiza los eventos organizándolos por día. Se visualizan dentro de una agenda diaria.
"Semana" - Visualiza los eventos de la semana.
"Mes" - Es la visualización por defecto. Presenta los eventos del mes.
"4 Días" - Permite ver al calendario como una agenda con los eventos de los próximos 4 días.

Hacia la izquierda se presentan botones que permiten establecer la fecha o periodo que se considera para la visualización de eventos.

Permite navegar la vista hacia atrás.
Ejemplo: Si la vista activa es por mes, retrocede un mes. Si la vista activa es por día, retrocede un día.

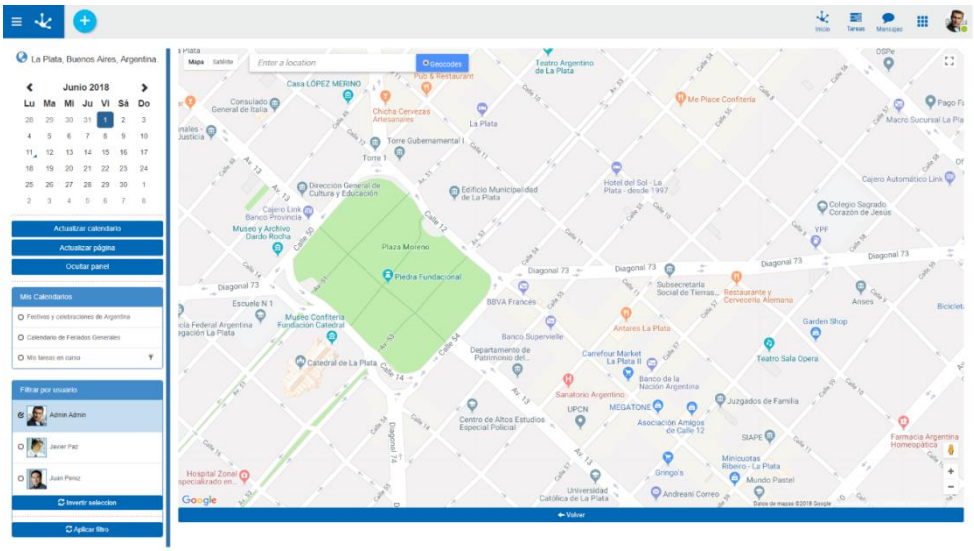
Permite navegar la vista hacia adelante.
Ejemplo: Si la vista activa es por mes, avanza un mes. Si la vista activa es por día, avanza un día.

Acceso rápido al día de hoy.

Ejemplo: Si la vista activa es por mes, visualiza el mes en curso. Si la vista activa es por día, muestra la fecha actual.

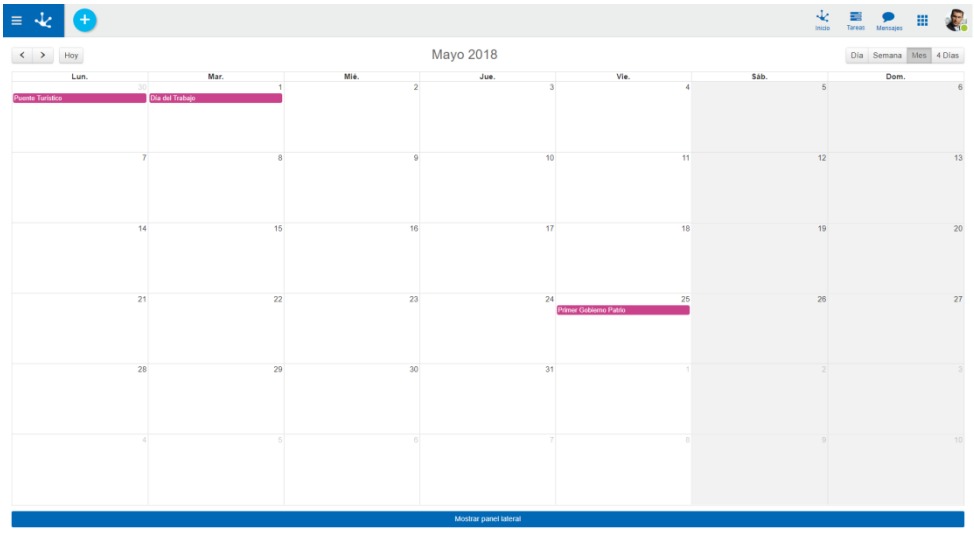
Además de estos mecanismos de navegación descritos, se puede usar el selector de fechas o **"Date picker",** que permite establecer una fecha específica.

Ubicación Geográfica
En la porción superior izquierda de la pantalla se indica el lugar que **Deyel** está considerando para recuperar información del clima.
Esta ubicación puede ser cambiada abriendo el panel de configuración de ubicación:



La búsqueda de la ubicación debe hacerse mediante el cuadro de texto superior. Una vez modificada la ubicación actual, los iconos del clima se actualizarán. El botón "Volver" permite retornar a visualizar el calendario.

“Actualizar Calendario” - Actualiza el contenido del panel de eventos. Evita tener que recargar la página para obtener los últimos cambios.
“Actualizar página” - Actualiza la página completa. Esto implica que se inicializan los filtros, debiendo establecerlos nuevamente.
“Ocultar panel” - Oculta el panel lateral, para que el calendario ocupe el ancho total de la pantalla. Con el botón **‘Mostrar panel lateral’** se volverá a visualizar el panel ocultado.



Mis Calendarios.

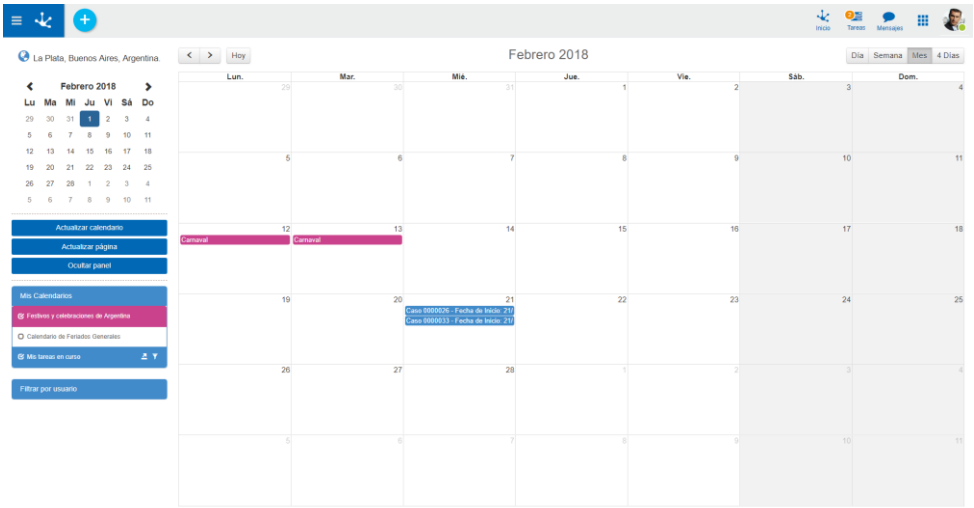
Este panel contiene la lista de los calendarios gráficos que el usuario puede visualizar.

Deyel ofrece algunos calendarios predefinidos, que pueden ser de utilidad al usuario.

- Calendario de Feriados en el país de origen de la instalación.
- Calendario de Feriados Generales. Muestra los días Feriados y Fechas Particulares definidos en la organización.
- Calendario de Mis Tareas

Estos calendarios gráficos se definen y configuran en la tabla auxiliar “Configuración de calendarios gráficos – T246”, referirse a la definición de la tabla y al tópico “Configuración” para más información.

Al hacer clic sobre un calendario gráfico, los eventos que corresponden a este calendario son visualizados en el panel de eventos.
Pueden seleccionarse varios calendarios simultáneamente y cada uno presenta sus eventos con un color específico.

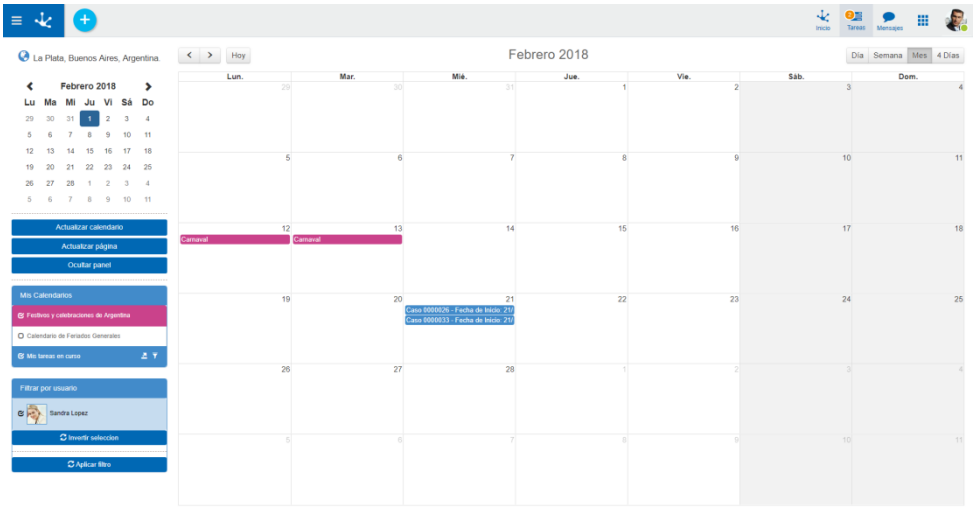


Cada calendario ofrece la posibilidad de filtrar los eventos. Al presionar este control se despliega el panel de filtrado, que presenta una serie de campos por los cuales filtrar los eventos a visualizar.

Permite integrar el calendario con otras aplicaciones, como por ejemplo Google Calendar. Esta funcionalidad se encuentra detallada en el tópico “Integración con otras aplicaciones.”

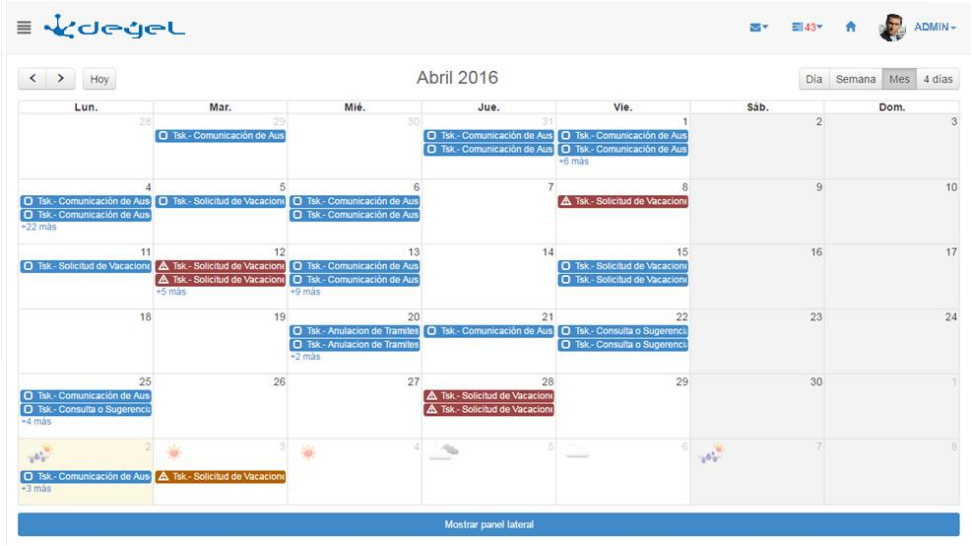
Panel de Selección de Usuarios

Si el usuario es responsable de una unidad organizacional , aparecerá el listado de usuarios registrados en dicha unidad. Este mecanismo permite que un coordinador visualice no sólo sus propios eventos sino también los de sus colaboradores. Una vez seleccionados los usuarios deseados, presione el botón **‘Aplicar filtro’** para actualizar el contenido del panel de eventos.

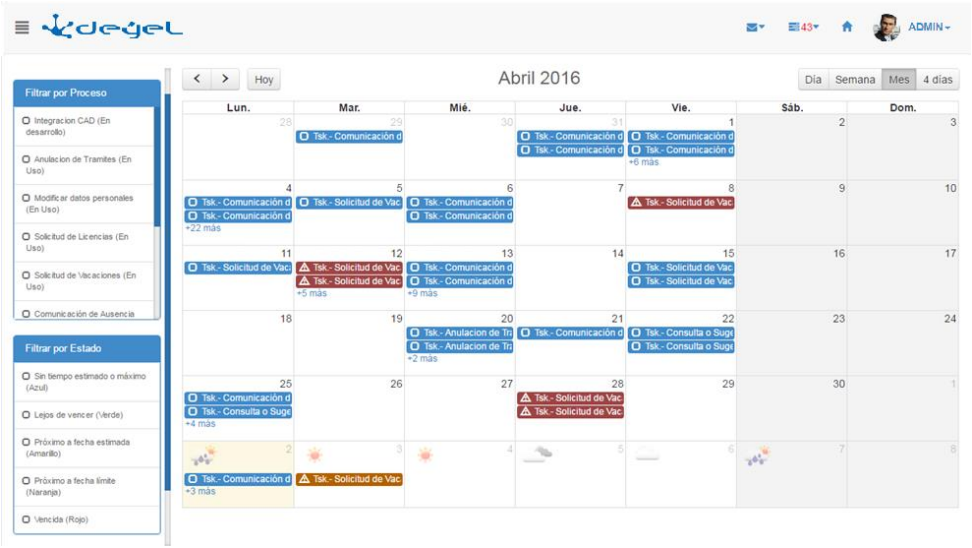


3.4.10.1. Calendario "Mis Tareas"

Este es uno de los calendarios predefinidos de **Deyel**.
Cada evento de este calendario representa a una tarea del usuario que está en ejecución, con lo cual este calendario es una manera alternativa de visualización de la lista de tareas del usuario.
Los eventos se visualizan con diferentes colores, dependiendo del vencimiento de la tarea que representan. Se utiliza el mismo esquema de semaforización que en la lista de tareas pendientes.



El calendario de "Mis Tareas" ofrece filtrar los eventos por los siguientes criterios:



Filtrar por Proceso - Permite ver las tareas pertenecientes a los procesos seleccionados.
Filtrar por Estado - Permite ver las tareas de acuerdo a los tiempos de ejecución de la tarea. Se considera la definición de tiempos promedios y tiempos máximos de ejecución de la actividad y se clasifica la tarea.

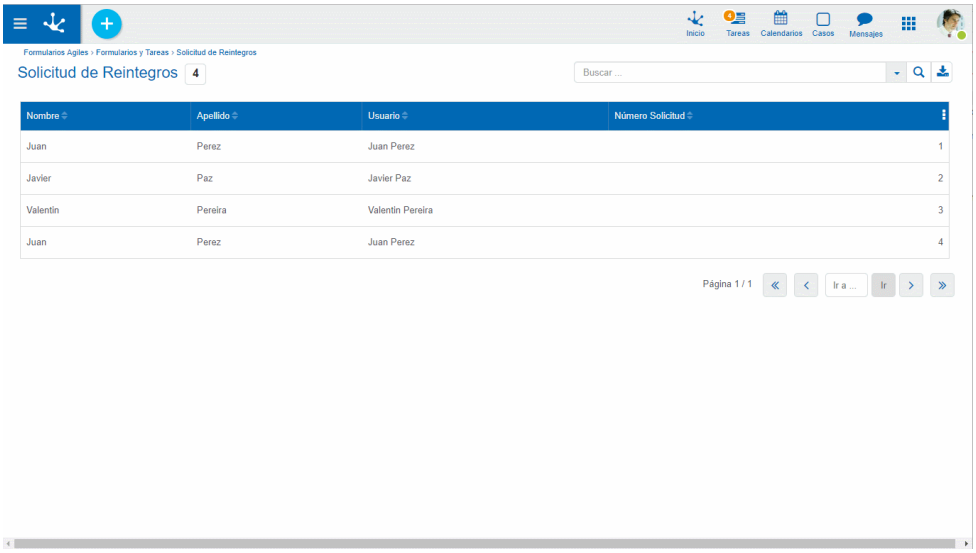
Haciendo clic sobre uno de los eventos se puede acceder a la consulta del caso correspondiente y a la ejecución de la tarea.

3.4.11. Grilla de Resultados y Búsqueda

La grilla de resultados permite visualizar la lista completa de instancias del formulario, paginando los resultados y permitiendo aplicar filtros de búsqueda para reducir la grilla.

Cada fila representa una instancia de formulario y cada columna representa datos propios de la instancia. Los datos que se visualizan en las columnas son aquellos que fueron ingresados en campos de formularios, modelados con la propiedad Incluido en Grilla marcada.

Las columnas pueden mostrarse u ocultarse para simplificar la lectura de la grilla. Para ocultar una columna basta con presionar el ícono , el cual se visualiza al pasar el mouse sobre la cabecera de la columna. También se pueden ocultar o mostrar columnas utilizando las marcas que se visualizan al presionar el ícono .

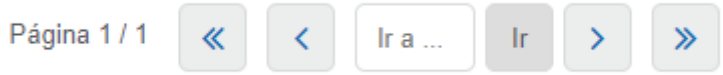


Ordenamiento

El ícono se visualiza en cada columna de la grilla y permite ordenar las líneas en base a dicha columna, de manera ascendente o descendente. El ordenamiento puede realizarse solamente por una única columna.

Paginado

Si la cantidad de líneas supera la cantidad máxima a mostrar en la ventana, se pueden utilizar los botones que se encuentran en la parte inferior derecha de la grilla para moverse hacia la página anterior y siguiente, al inicio o fin, o bien seleccionar una página específica.



« » Permiten ir al inicio o al final de la lista.

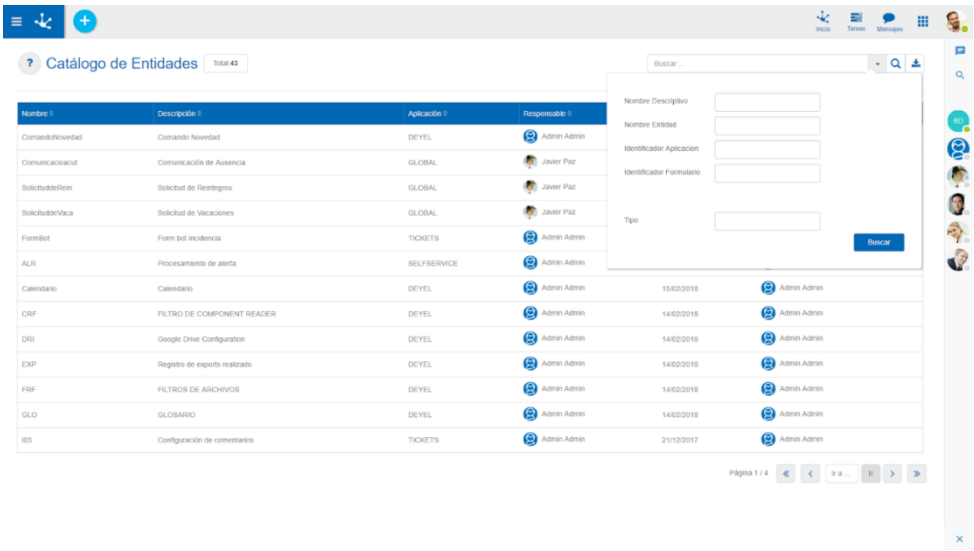
< > Permiten ir a la página anterior o siguiente.

Ir a ... Permite acceder a la página indicada.

Barra Búsqueda y Filtros

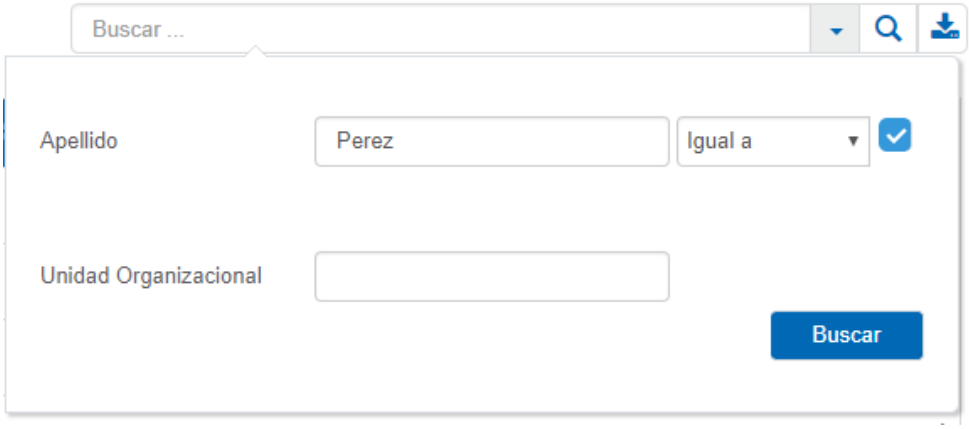
Al ingresar un texto en la barra de búsqueda y presionar el ícono o la tecla “Enter”, se filtran en la grilla todos los objetos que contienen el valor ingresado en algunas de sus columnas.

El ícono permite desplegar el panel con filtros que pueden combinarse para definir el criterio de selección de las líneas. Los filtros que se visualizan son aquellos que hayan sido seleccionados como filtros de búsqueda al momento del modelado del formulario. Una vez realizada la selección de filtros se presiona el botón “Buscar” o bien el ícono para reducir la grilla visualizada.



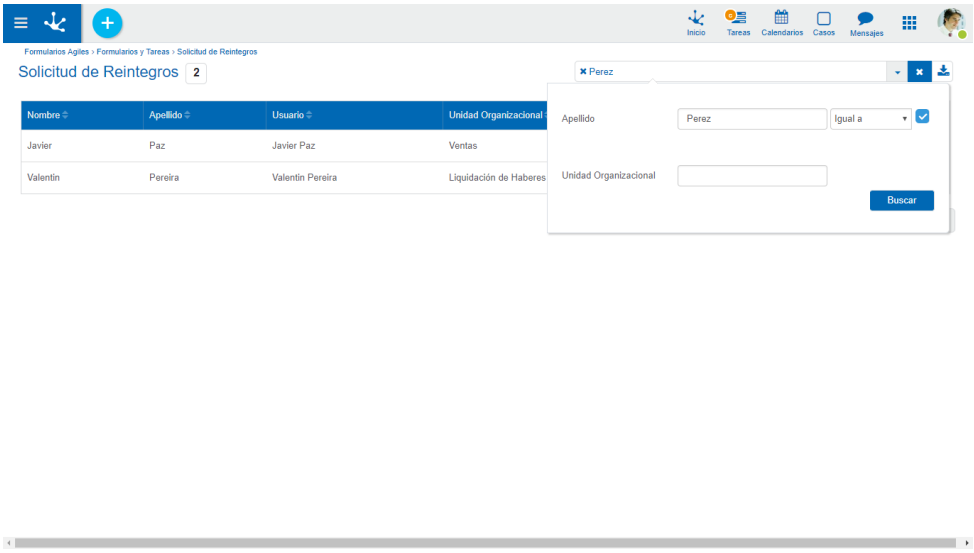
Elementos del Criterio de Selección

- Nombre del campo de filtro
- Valor de búsqueda
- Comparación (Igual, Contiene, Inicia con, Incluido, Mayor, Mayor o igual, Menor, Menor o igual)
- Marca para negar la condición



Ejemplo

Se desean listar las solicitudes de reintegro que no contienen el valor "Perez" en el campo Apellido, mediante el uso de la marca para negar la condición de cada campo.
Presionando el ícono o el botón "Buscar" se obtiene el resultado de la consulta de instancias con el filtro ingresado.



El buscador permite filtrar resultados en base a la negación de una condición, mediante el uso de la marca para negar la condición de cada campo.

Descarga de Datos

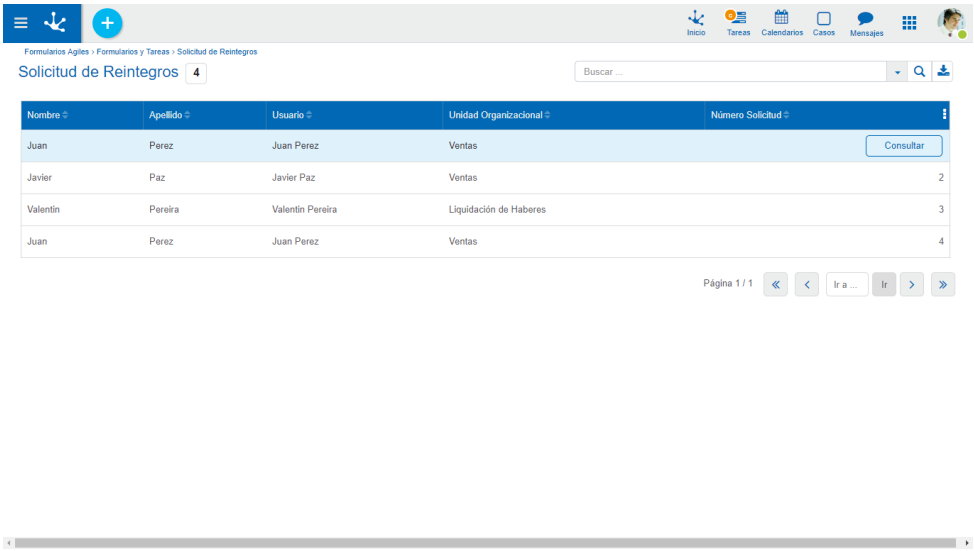
El ícono permite descargar a un archivo Excel los datos de los formularios que se visualizan en la grilla de resultados, ya sea en su totalidad o los que resulten de aplicar un filtro. En caso que el usuario posea permisos de seguridad para realizar la funcionalidad "Exportar", además puede realizar la descarga a un archivo XML, que se comprime con extensión .ZIP.

Operaciones

Al desplazar el mouse sobre cada una de las filas de la grilla de resultados, se visualizan los botones correspondientes a las operaciones que se pueden realizar sobre el formulario, dependiendo de los permisos de seguridad que el usuario tenga definidos.

- Consultar
- Modificar
- Eliminar

La excepción es la operación Crear, que puede realizarse desde el ícono correspondiente al menú contextual.

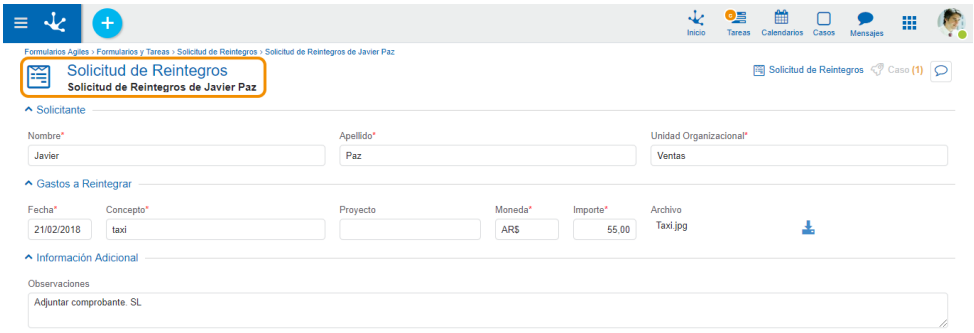


3.4.12. Instancias de Formularios

Fase 2: Modelado de Formularios > Visualización de los datos del formulario

Cada instancia de un formulario se compone de secciones. La presentación visual depende de cómo se haya modelado el formulario, puede verse en forma centrada o clásica, con o sin color de fondo.

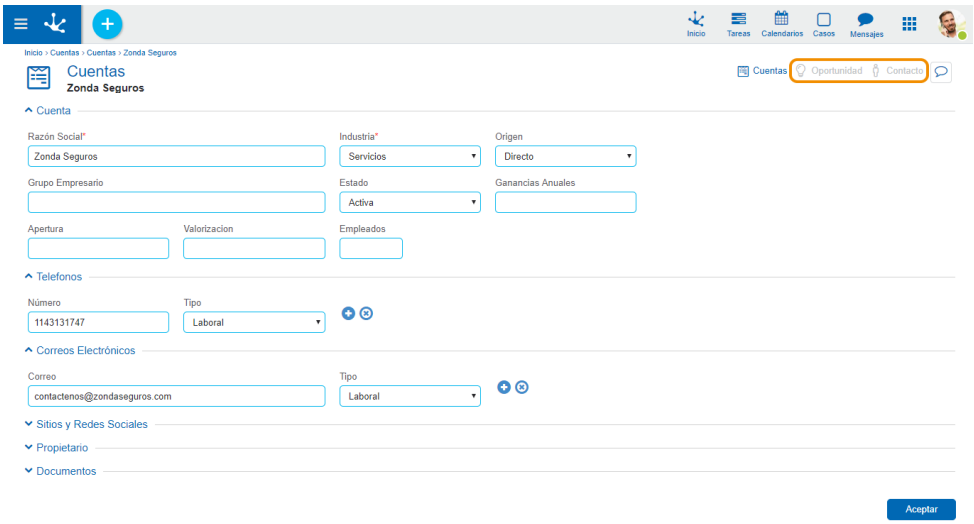
Encabezado del Formulario



Los elementos que se visualizan en el encabezado del formulario son los definidos en el modelador, como propiedades generales del mismo.

- Nombre descriptivo del Formulario
- Icono asociado: Definido como propiedad del formulario.
- Descripción

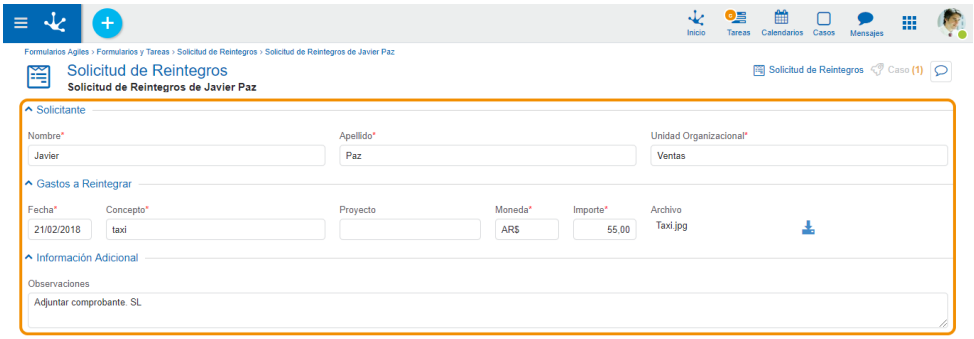
Area de Relaciones



El área de relaciones permite visualizar el conjunto de entidades que referencian a la instancia con la que se está trabajando.

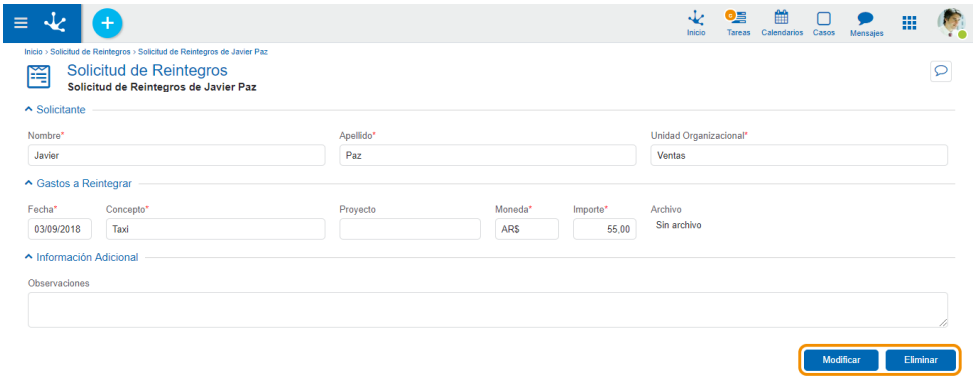
Permite al usuario poder navegar entre entidades vinculadas de forma sencilla. Cada relación se representa como una pestaña distinta, cuyo nombre es el nombre de la relación. Al hacer clic sobre cada pestaña se visualizan las instancias relacionadas.

Area de Campos



En el área de campos se muestra el conjunto de campos del formulario con sus contenedores. Dichos campos pueden estar visibles, no visibles, editables y no editables, dependiendo de las propiedades definidas para el campo en el momento de su modelado o de la definición de la actividad del proceso asociado al formulario.

Botones de Operaciones



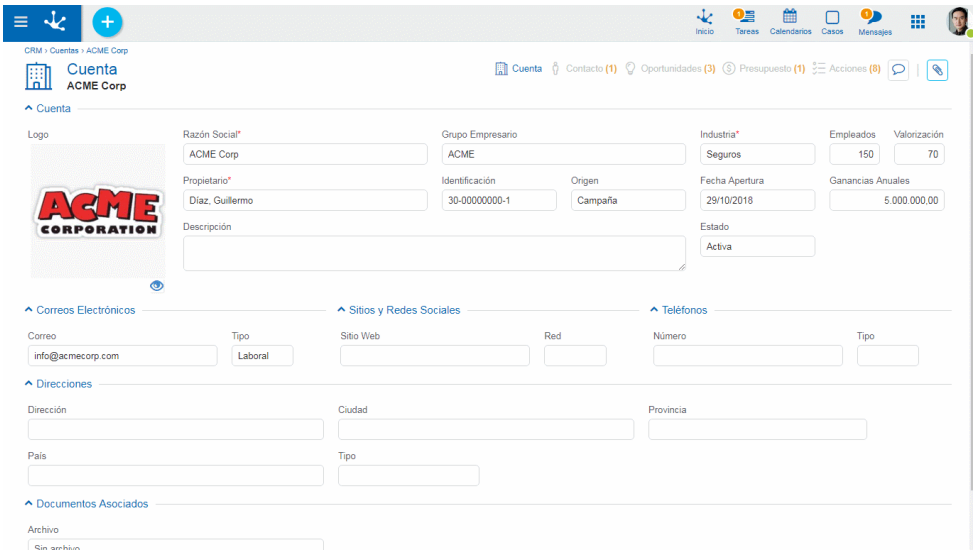
En esta sección se visualizan los botones para realizar acciones sobre el formulario dependiendo de la operación que se esté realizando sobre la instancia del formulario.

Para cada operación sobre la instancia del formulario, los botones que se visualizan varían, de acuerdo a los permisos que el usuario tenga definidos en su configuración de permisos de seguridad.

3.4.12.1. Entidades Relacionadas

Cuando se utiliza un formulario, las relaciones que han sido modeladas se visualizan en la sección superior derecha con sus nombres y la cantidad de instancias que se han asociado para cada relación. Para acceder a las mismas, se debe hacer clic sobre su nombre.

Si la relación se da a una única instancia se muestra el nombre singular modelado, si es a más de una se muestra el nombre plural modelado. Si la relación tiene una sola instancia de formulario, se accede directamente al mismo, en cambio, si tiene más de una se visualiza la lista de instancias.



Ver Relaciones

Este ícono se visualiza cuando la cantidad de entidades relacionadas ocupan un espacio mayor al disponible en la línea superior. Al presionarlo, se despliega una lista con las entidades relacionadas.

Uso del Menú Contextual en Entidades Relacionadas

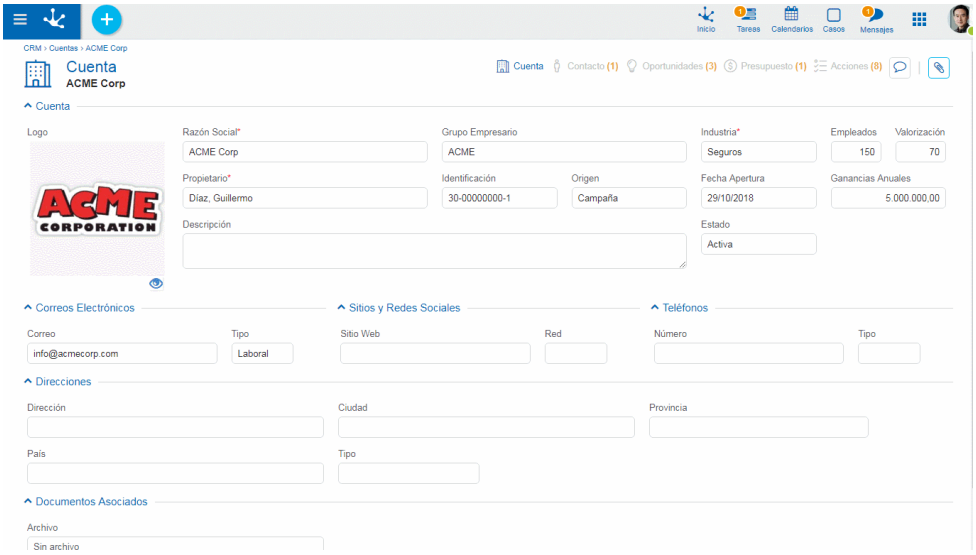
Menú Contextual

El menú contextual adopta las características generales del portal pero adaptado a las entidades relacionadas.

Menú de Iconos

El menú de íconos se habilita al pasar el mouse sobre el ícono y permite realizar la creación de entidades para relacionarlas. Se visualizan los íconos de las entidades relacionadas a las que el usuario les puede crear una nueva instancia.

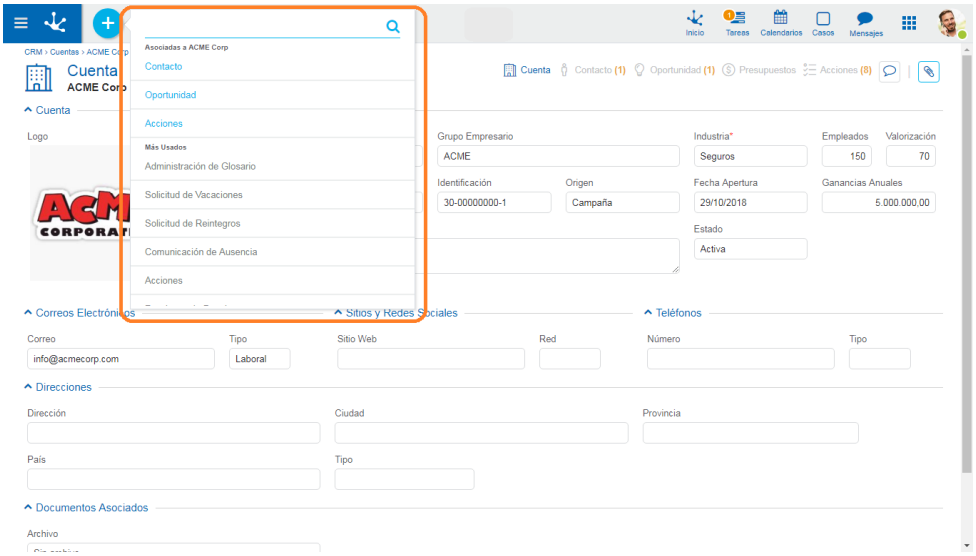
Los valores predeterminados en la entidad creada, toman lo configurado en el panel de correspondencia de valores, si los tuviesen.



Menú Desplegado

Presionando el ícono y se abre un panel con las siguientes secciones:

- Lista de objetos relacionados con la entidad (color celeste).
- Lista de objetos más usados (color gris).



3.4.12.2. Operaciones

Los botones de las operaciones de las instancias de formularios se habilitan al pasar el cursor sobre la línea de la instancia seleccionada en la en la grilla de resultados, a excepción de la operación de creación, que se inicia desde el ícono correspondiente al menú contextual.

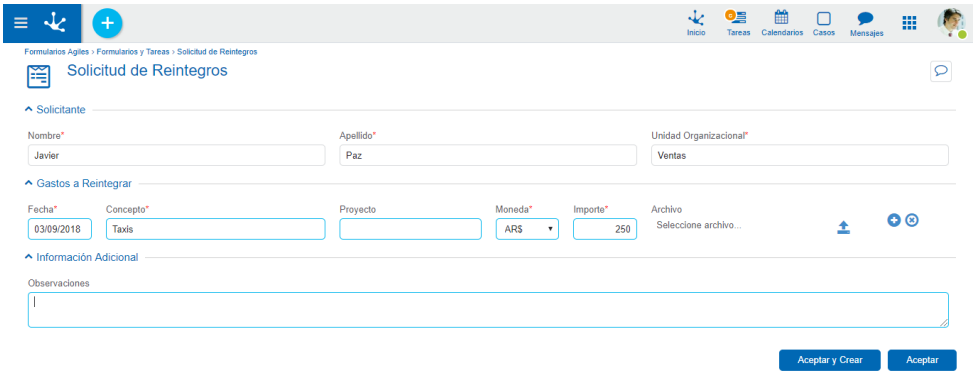
Para realizar operaciones sobre las instancias de un formulario, el mismo no debe tener definidos procesos relacionados en el modelado o en caso de tenerlos, no debe estar seleccionada la propiedad Variable de Proceso.

Crear

El usuario puede crear una nueva instancia de un formulario mediante el ingreso de los valores en los campos, al presionar cualquiera de los botones disponibles se crea la instancia y el usuario recibe un mensaje indicando que los datos fueron guardados. . Al momento de crear una instancia se ejecutan las validaciones definidas para el formulario y cada uno de sus campos.

Botones

- Aceptar y Crear: se confirma la creación de una instancia del formulario y se abre un nuevo panel para crear otra.
- Aceptar: se confirma la creación de una nueva instancia del formulario.

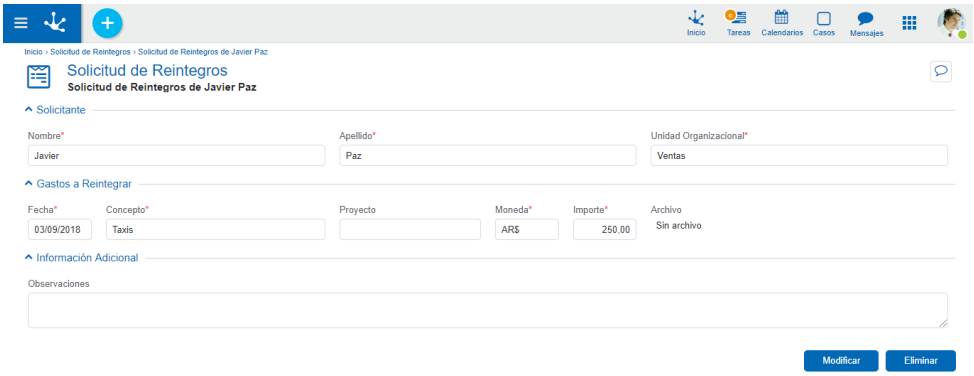


Consultar

Permite visualizar los datos correspondientes a la instancia seleccionada que fueron previamente creados.
Abre la instancia del formulario seleccionada. Los campos se visualizan como no editables y dependiendo los permisos de seguridad del usuario, se habilitan los botones disponibles para operar con la instancia que es está consultando.

Botones

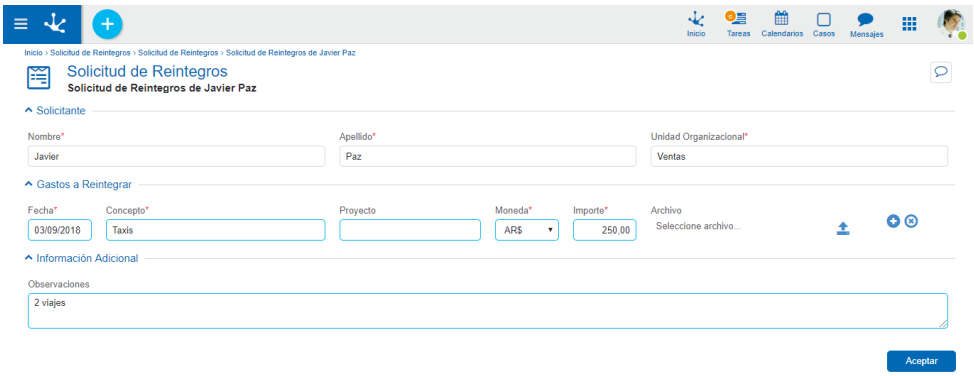
- Modificar
- Eliminar



Modificar

Abre la instancia del formulario seleccionada con aquellos campos que pueden modificarse, como editables. La modificación se realiza al presionar el botón “Aceptar” y el usuario recibe un mensaje indicando que los datos fueron guardados.

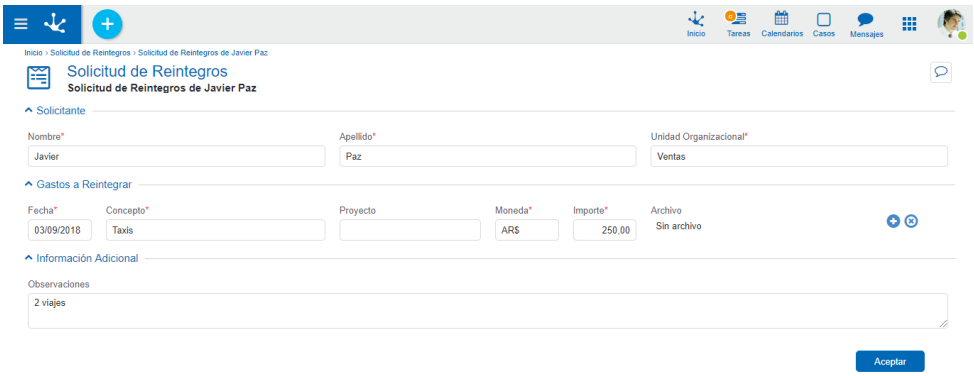
Al momento de modificar una instancia se ejecutan las validaciones definidas para el formulario y cada uno de sus campos.



Eliminar

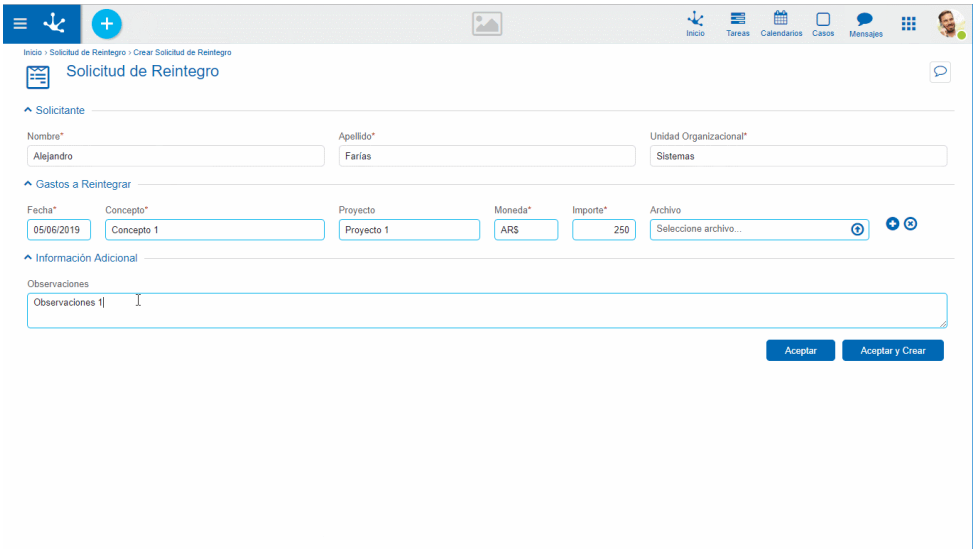
Abre la instancia del formulario seleccionada y sus campos se visualizan como no editables. La eliminación se realiza al presionar el botón “Aceptar” y el usuario recibe un mensaje indicando que los datos fueron eliminados.

> *Una instancia del formulario no puede ser eliminada si se encontrara relacionada a un caso.*



Versionar

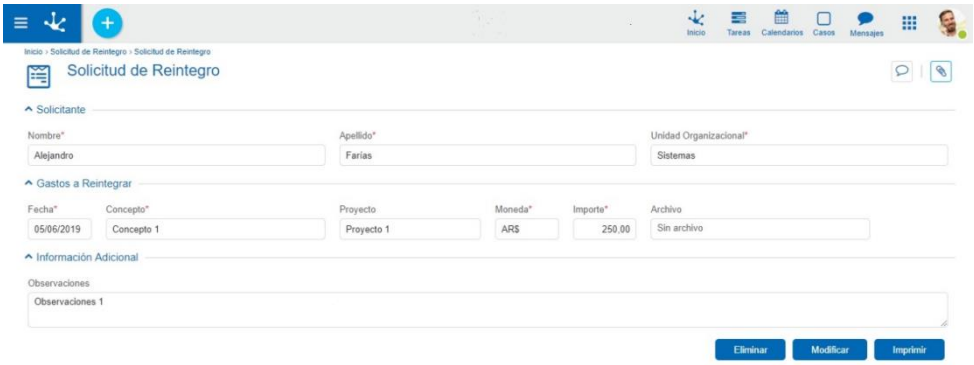
Cuando un formulario utiliza la propiedad versionado se pueden generar copias históricas del mismo denominadas versiones, permitiéndose asi, hacer un seguimiento de los cambios que se van realizando sobre el formulario. Cada versión es una copia de la instancia del formulario, que almacena el valor de todos los campos al momento de crearse la misma.
Para generar una versión, desde la consulta del formulario, presionar el boton **“Versionar”** . De esta manera se accede a la modificacion del formulario, previa generación de una copia del mismo.
Para acceder a las versiones asociadas a una instancia se debe hacer click en el boton **“Versiones”.**



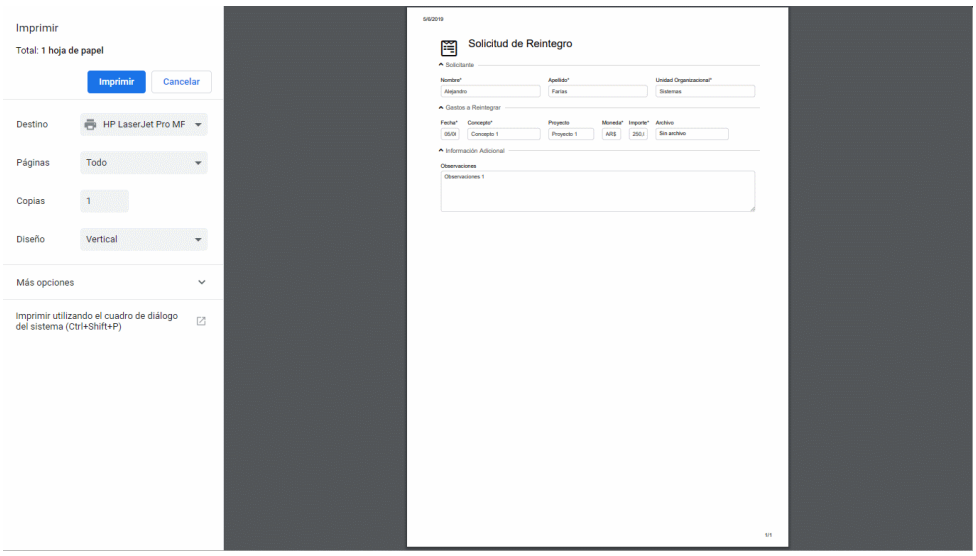
3.4.12.3. Impresión

Para imprimir la instancia de un formulario, el mismo tiene que haber sido modelado con la propiedad Imprimible habilitada.

Cuando se consultan las instancias de un formulario, si esta propiedad está habilitada, a la izquierda al pie del formulario, se muestra el botón “Imprimir”.



Presionando este botón se accede directamente a la impresión del formulario.



La impresión de un formulario tiene las mismas propiedades que un documento externo a **Deyel**, pudiendo cancelar la impresión con el botón “Cancelar”.

- Total de hojas
- Destino

- Páginas a imprimir
  - Todas
  - Rango de hojas
- Copias
- Diseño
- Color
- Tamaño del papel
- Páginas por hoja
- Márgenes
- Escala

Estas propiedades se ajustan en función del diseño del formulario.

> *Es recomendable seleccionar Diseño Horizontal y Escala al 80%.*

Para extender el comportamiento predeterminado de la impresión, ver la función modifyPrint().

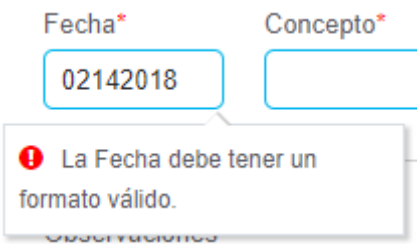
3.4.12.4. Validaciones

Las validaciones permiten el control de formato de los datos, obligatoriedad de campos y lógica de negocio al trabajar con instancias del formulario, de modo que no se ingresen datos incorrectos o que no sean válidos para el negocio.

Validaciones de Formato de Datos

Esta validación se aplica implícitamente a todos los campos de un formulario, y mediante la misma se controla que el formato de los datos ingresados por los usuarios es el correcto para cada campo.

Ejemplo: si una fecha no cumple con el formato DD/MM/AA, se muestra un mensaje de alerta.

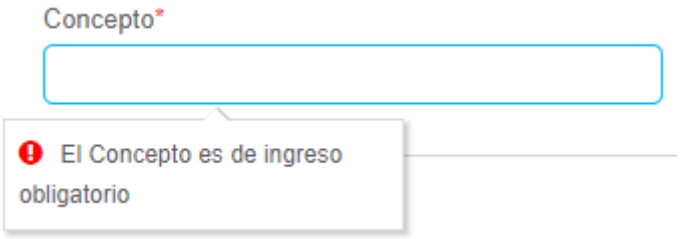


> Los formatos de fecha son ajustables en las propiedades del sistema.

Validaciones de Obligatoriedad

Esta validación controla que se ingresen datos en aquellos campos que son de carácter obligatorio. No se puede crear una instancia del formulario si no se completaron los campos que son de ingreso obligatorio.

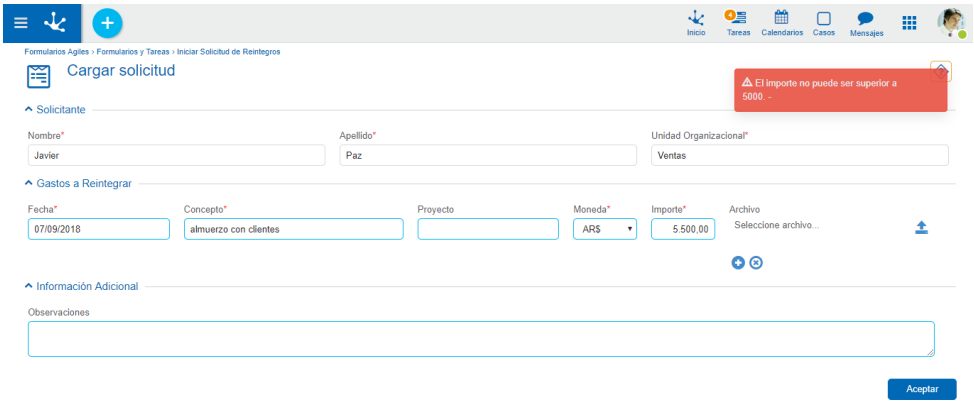
Un campo de ingreso obligatorio está indicado con un asterisco en la etiqueta.



Si no se completa un campo obligatorio, al crear una instancia del formulario aparece una alerta.

Validaciones Lógicas

Las validaciones lógicas controlan que los campos ingresados cumplan con condiciones establecidas al momento de modelar el formulario por medio de la propiedad Validación de cada campo.



3.4.13. Elementos de Formularios

Un formulario se compone de un conjunto campos y elementos que permiten manipular la información que contiene. Sus propiedades se definen al momento de modelar en formulario y se agrupan de la siguiente manera:

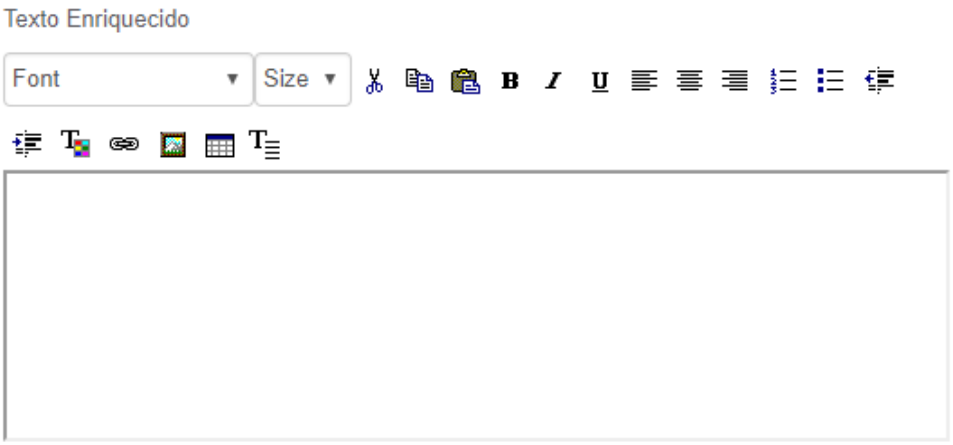
- Controles de Campo
- Control de Relación
- Contenedores

3.4.13.1. Campos

A continuación se detallan los distintos controles que pueden presentarse en el área de campos al utilizar el formulario, según las definiciones realizadas para los distintos tipos de campos al momento de modelar el formulario.

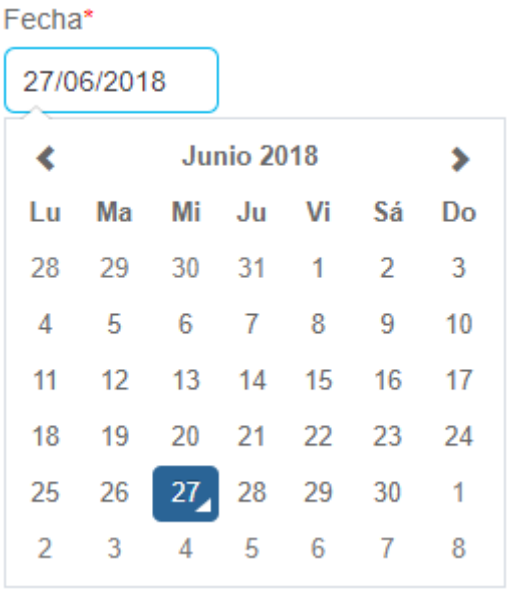
Texto Enriquecido

El control de texto enriquecido presenta un área para ingreso y modificación de texto en la que se permite modificar su formato. Se puede modificar el tamaño de la fuente, así como también aplicar subrayado, intensificado y colores, entre otras opciones. La definición de un control de tipo texto enriquecido se realiza al modelar el formulario.



Selector de Fecha

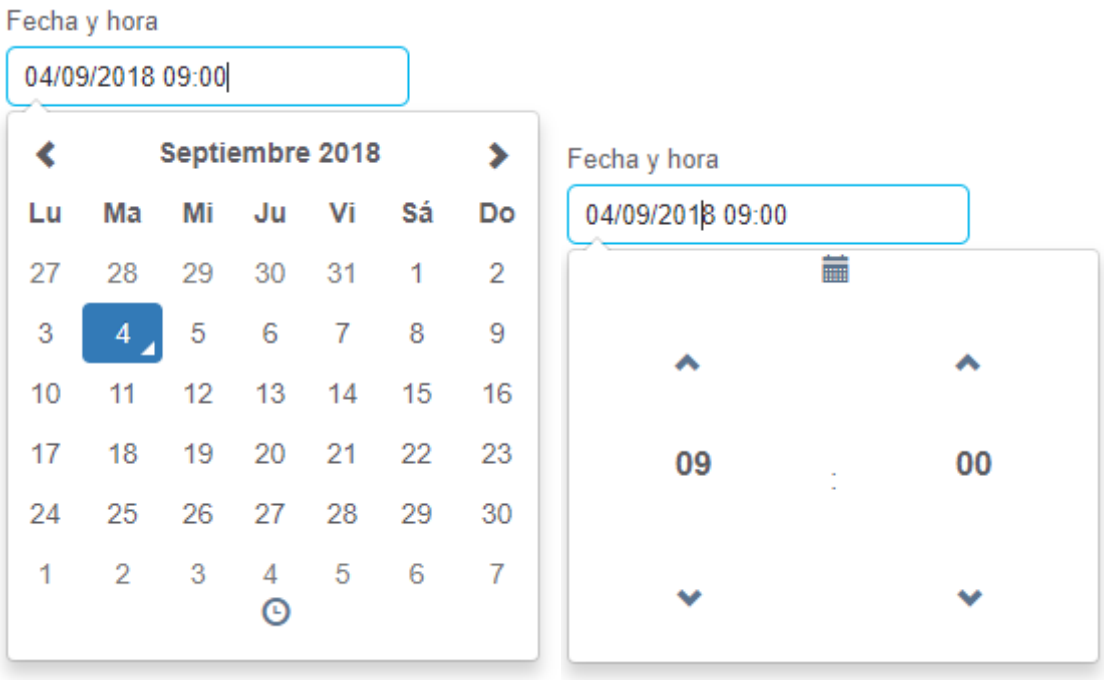
Un selector de fecha se hace visible al hacer clic sobre un control de tipo fecha. Permite seleccionar el día, mes y año, completando fácilmente el campo.



Selector de Fecha y Hora

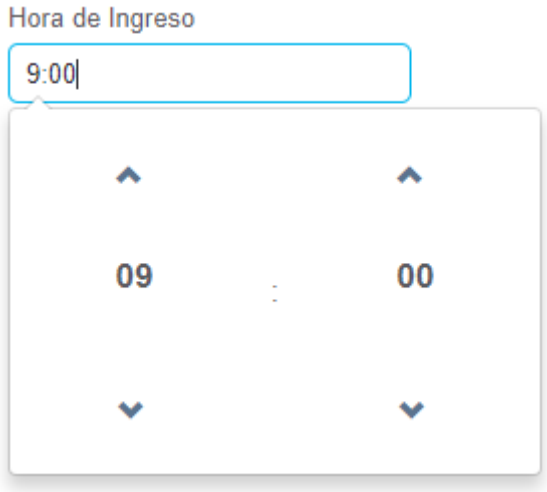
Un selector de fecha y hora se hace visible al hacer clic sobre un control de tipo fecha. Permite seleccionar el día, mes y año, completando fácilmente el campo.

Presionando el icono se accede a modificar la hora y minutos. Para volver a la fecha se debe presionar el botón .



Selector de Hora

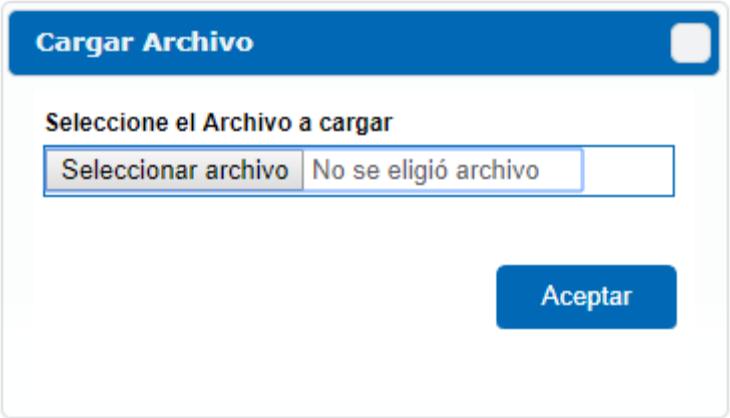
Un selector de hora se hace visible al hacer clic sobre un control de tipo hora. Permite seleccionar la hora y los minutos.



Imágenes

El ingreso de una imagen en un control de tipo imagen se realiza desde un selector que se encuentra en la parte inferior derecha del campo imagen, dentro del área de campos.

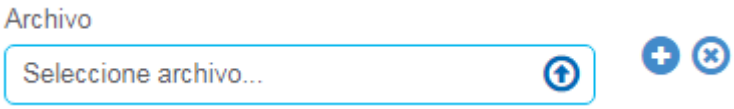
Al presionar el icono se abre la siguiente ventana:



Presionando el botón "Seleccionar archivo" se abre una ventana que permite seleccionar un archivo desde el ambiente donde el usuario esté trabajando. Al presionar el botón "Aceptar" se incorpora la imagen a la instancia del formulario.

Archivos

Un control de tipo archivo permite adjuntar uno o varios archivos desde el equipo del usuario y/o almacenados en Google Drive.



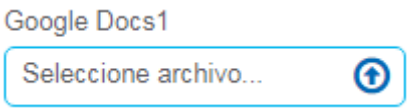
Abre la ventana para selecciona archivos.
Genera una nueva ocurrencia para adjuntar archivos.
Elimina una ocurrencia de archivos adjuntados.

Una vez adjuntados los archivos - cualquiera sea su origen y de acuerdo a los permisos del formulario y de la actividad en que encuentra la instancia del formulario dentro de un caso, pueden administrarse desde la barra de selección.

Reemplazar el archivo adjuntado por otro.
Descargar el archivo adjuntado en el equipo local.
Eliminar el archivo adjuntado de la instancia del formulario.

Google Docs

Un control del tipo Google Doc permite adjuntar uno o varios archivos desde el equipo del usuario y/o almacenados en Google Drive y alojarlos en Google para que puedan ser accedidos en la web por otros usuarios.



Abre la ventana para selecciona archivos.

Una vez adjuntados los archivos - cualquiera sea su origen y de acuerdo a los permisos del formulario y de la actividad en que encuentra la instancia del formulario dentro de un caso, pueden administrarse desde la barra de selección.

Reemplazar el archivo adjuntado por otro.
Descargar el archivo adjuntado en el equipo local.
Editar el archivo en el entorno colaborativo de Google.
Eliminar el archivo adjuntado de la instancia del formulario.

La administración de los permisos de consulta y modificación de documentos Google es externa a **Deyel**.

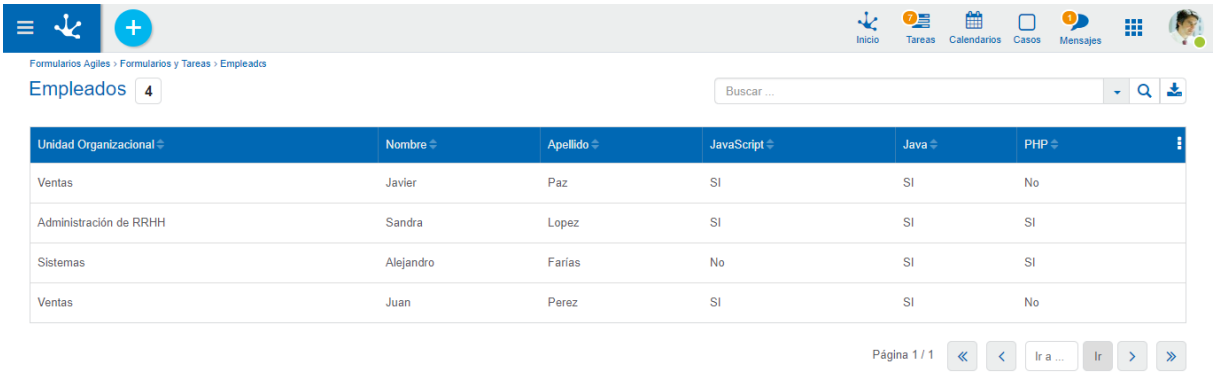
CheckBox

Un control checkbox recibe datos de tipo lógico, pudiendo tomar valores “SI/NO”. Al hacer clic sobre el control, el campo queda marcado indicando que su contenido es “SI”, caso contrario su contenido es “NO”.

Lenguajes de Programación

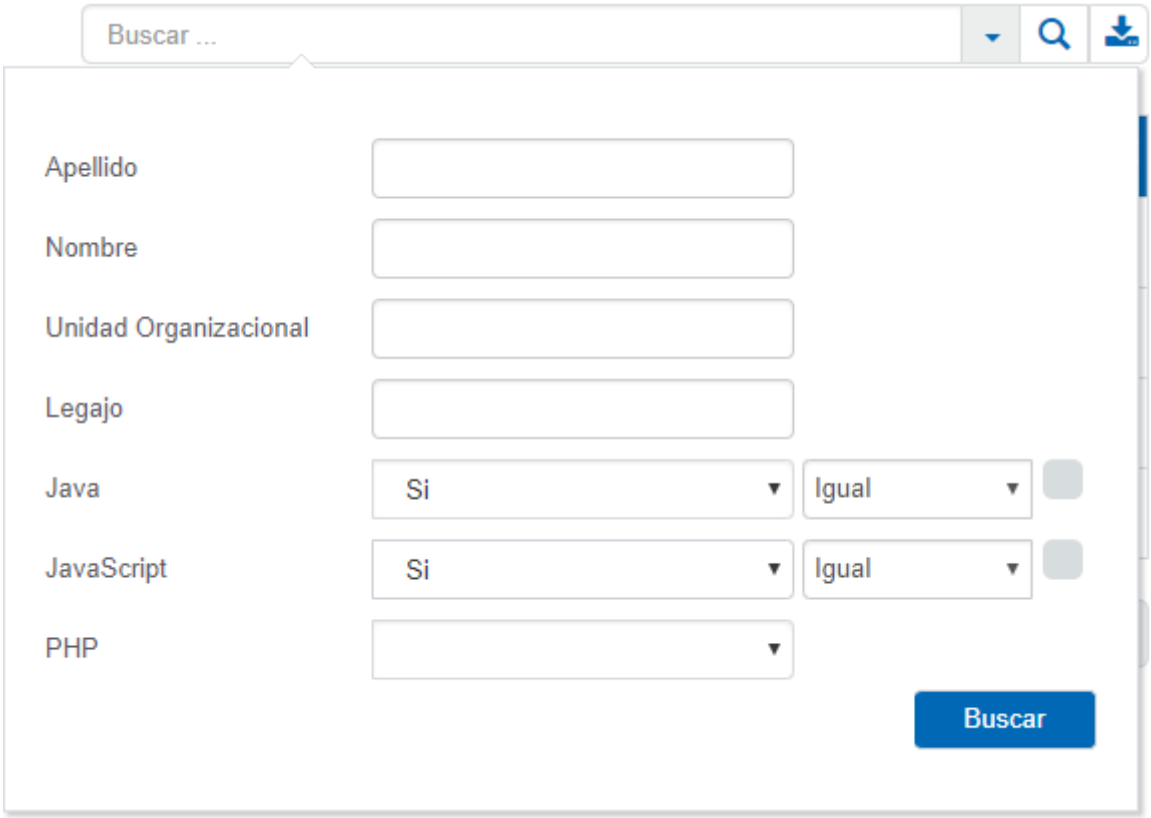
Java ☑ JavaScript ☑ PHP ☐

Si el campo es incluido en la grilla de resultados, el contenido del mismo se representa con valores "SI/NO" en la columna correspondiente.



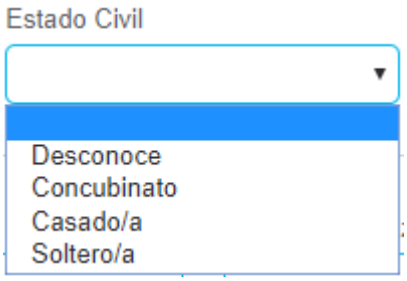
Unidad Organizacional	Nombre	Apellido	JavaScript	Java	PHP
Ventas	Javier	Paz	SI	SI	No
Administración de RRHH	Sandra	Lopez	SI	SI	SI
Sistemas	Alejandro	Farías	No	SI	SI
Ventas	Juan	Perez	SI	SI	No

Si se utiliza como filtro de búsqueda se pueden utilizar los valores "SI/NO" para reducir la lista resultante.



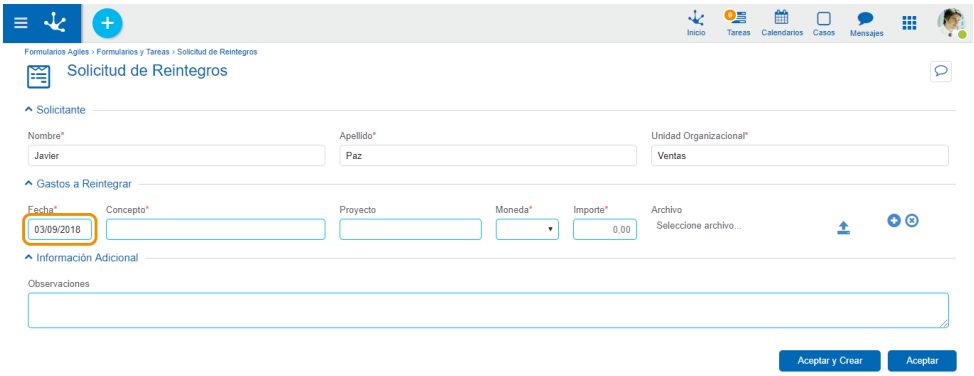
Lista de Valores

Los campos representados por este control proponen la lista de valores definida al modelar el campo,



3.4.13.1.1. Valores Predeterminados

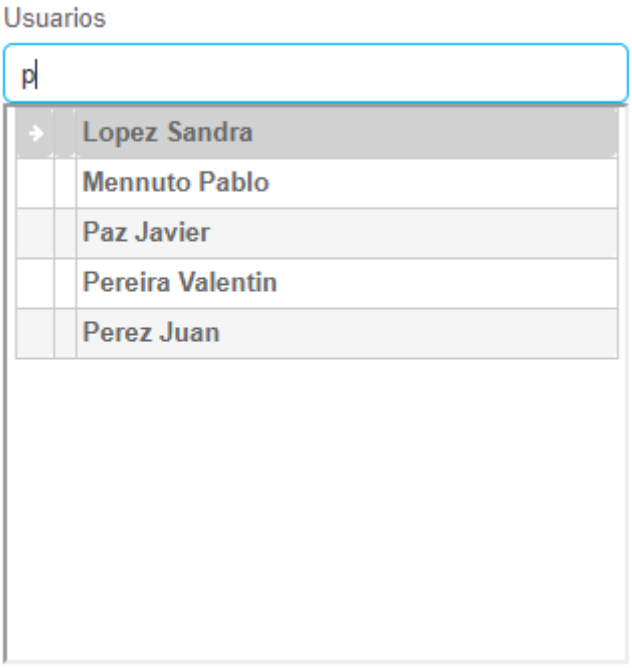
En el modelado de un formulario se pueden definir valores predeterminados para sus campos. Al crear una nueva instancia, este tipo de campos se visualizan con dicho valor contenido automáticamente.



3.4.13.1.2. Campos Autocompletables

Los campos configurados como autocompletables proponen al momento de ingresar los datos del formulario, una lista de valores que coinciden con lo que el usuario va escribiendo. De esta manera los valores propuestos varían según las coincidencias entre los datos ingresados y los datos del sistema.

En el ejemplo, al escribir la letra "p", la lista de valores de usuarios se acota a aquellos que contienen dicha letra en alguna posición.



3.4.13.1.3. Campos Relacionados

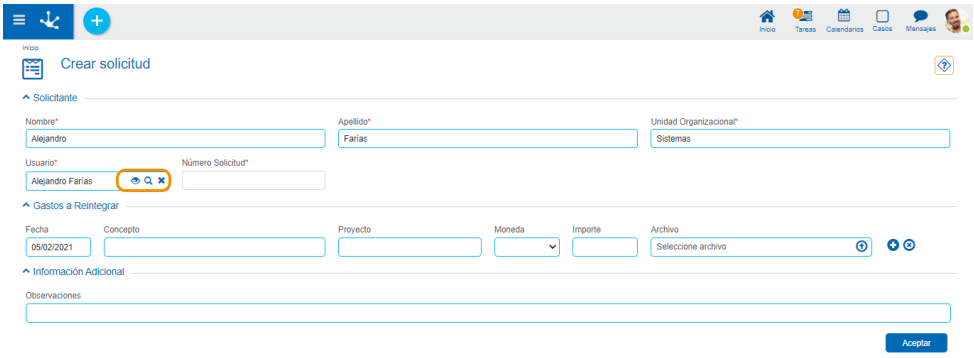
Se describen ejemplos de uso de los campos con definiciones en la pestaña "Relación" del panel de propiedades.

Selección Simple

Los campos con una relación a listas de valores definida al modelar el formulario, se visualizan como un conjunto de valores, donde el usuario puede realizar la selección de uno de ellos.

Relación con Entidades

En los campos definidos en el modelador de formularios que están relacionados con una entidad mediante un atributo clave, se visualizan iconos que permiten realizar operaciones.
:



Limpiar Campo

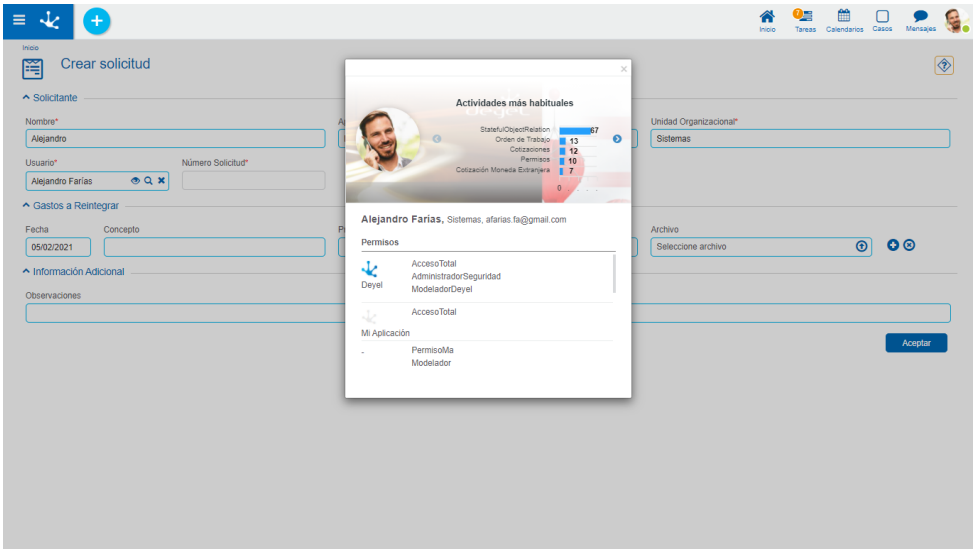
Permite borrar el dato ya ingresado en el campo.

Seleccionar

Permite buscar y seleccionar instancias de entidades con las que el campo está relacionado. Al hacer clic en este botón se abre una nueva ventana que muestra la grilla de resultados de la entidad relacionada.

Consultar

Permite ver los datos de la entidad relacionada con el campo, abriendo una nueva ventana con el formulario correspondiente a la entidad relacionada.
En el ejemplo se muestran los datos del usuario, que es la entidad relacionada con el campo.



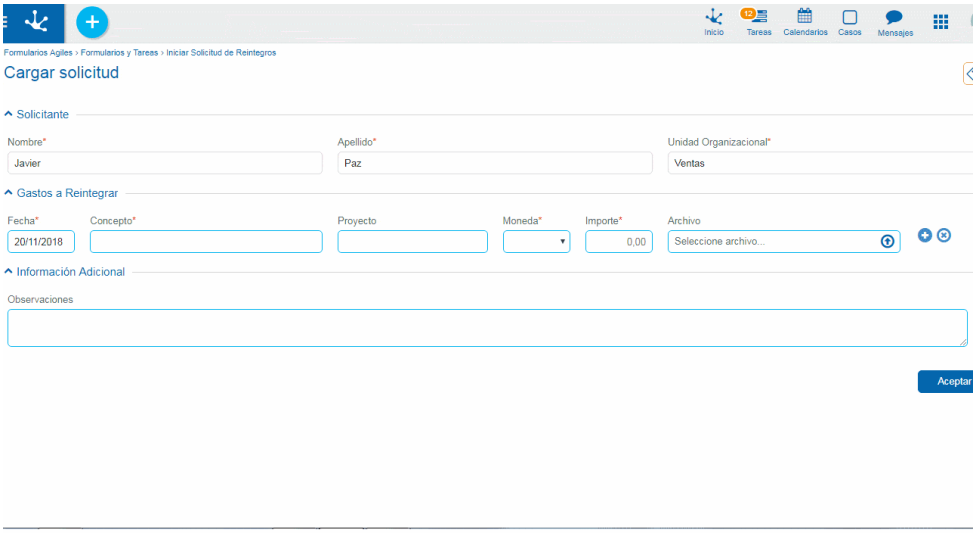
3.4.13.1.4. Archivos Adjuntos

Adjuntar Archivos desde el Equipo del Usuario.

Abre la ventana del equipo del usuario donde se encuentra el archivo a adjuntar y se selecciona con doble clic.
Se pueden adjuntar múltiples archivos con la modalidad “Arrastrar y soltar”, presionando botón izquierdo del mouse sobre el archivo y sin soltar, se arrastra hasta el centro del área que presenta el texto “Arrastra los archivos aquí”.

Adjuntar el Enlace a un Archivo Almacenado en Google Drive.

Accede al drive de Google perteneciente al usuario y se selecciona el archivo a adjuntar. Se pueden adjuntar múltiples archivos seleccionándolos y presionando el botón insertar.



Permite cambiar el archivo o enlace adjunto. Si se trata de un campo de múltiples ocurrencias, permite agregar archivos o enlaces.

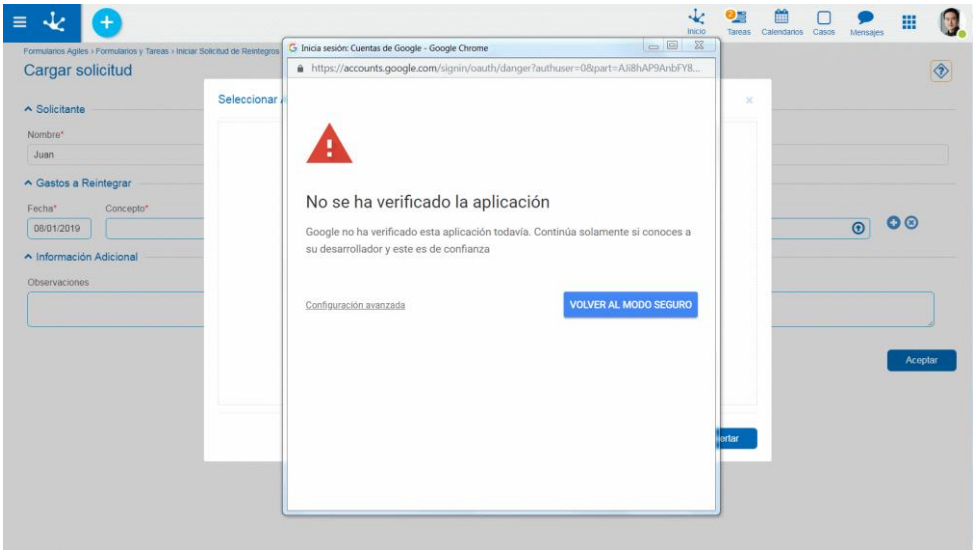
Retorna a la ventana de la instancia del formulario sin tomar las últimas modificaciones realizadas en el campo archivo.

Retorna a la ventana de la instancia del formulario confirmando la información registrada en el campo archivo.

El usuario tiene que contar con los permisos de acceso a la información alojada en este servicio. La administración de los permisos de Google Drive es externa a **Deyel**.
Si no tiene iniciada la sesión, al momento de consultar o adjuntar un enlace se presenta la ventana para ingresar el usuario y contraseña de Google Drive.

Primer acceso a Google Drive desde Deyel

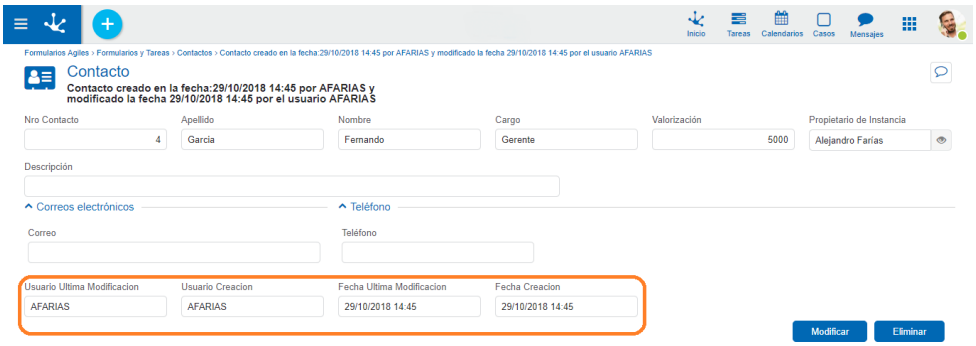
Cuando el usuario se conecta desde **Deyel** a Google Drive por primera vez, se abre una ventana adicional para autorizar a **Deyel** a acceder en forma segura a sus datos.



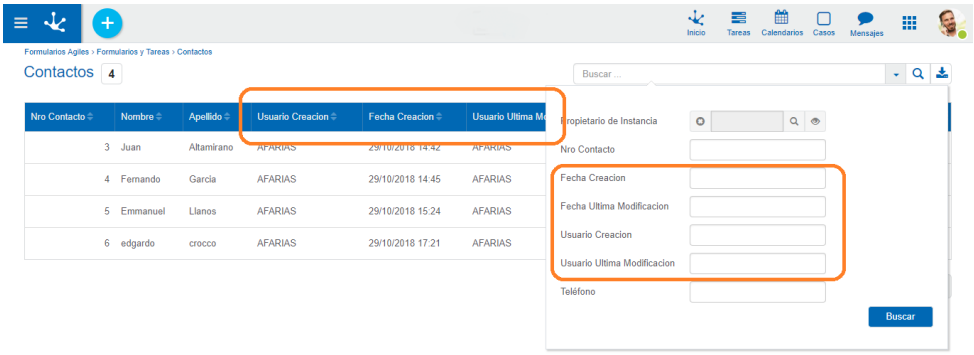
- Presionar enlace configuración avanzada
- Hacer clic en Ir a **Deyel**
- Presionar botón Permitir

3.4.13.1.5. Campos de Auditoría

Un formulario puede tener visibles algunos o todos sus campos de auditoria visibles, o bien no mostrarlos pero no pueden ser editados por el usuario ya que se completan en forma automática.

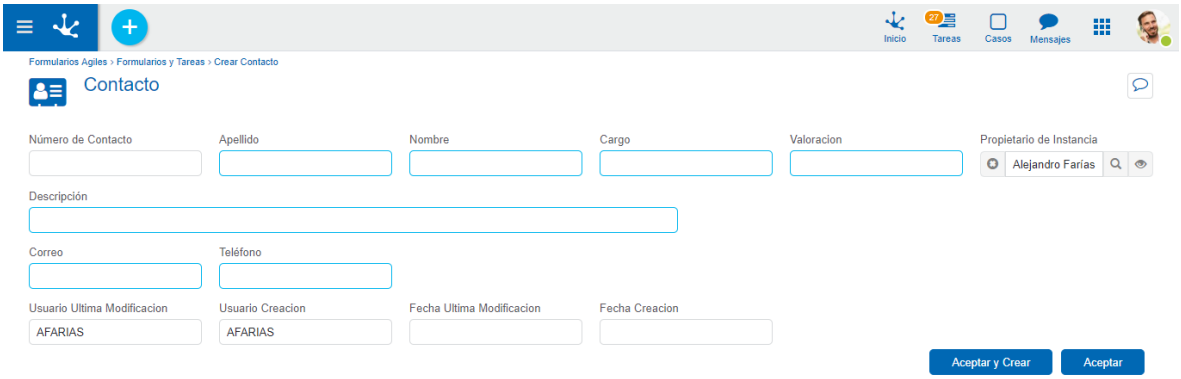


Los campos de auditoría pueden estar presentes en la grilla de resultados y ser parte de los filtros de búsqueda independientemente a que estén visibles en el formulario.



3.4.13.1.6. Campo Propietario de Instancia

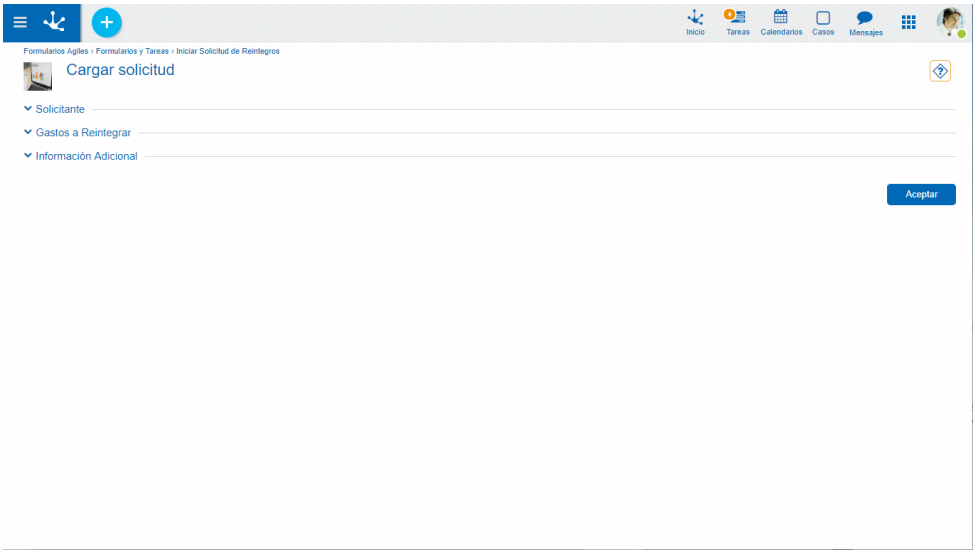
Un campo propietario de instancia puede representarse con el control de relación vinculado a la entidad Usuario y en ese caso se visualiza como cualquier campo relacionado a una entidad definido en el formulario.



En caso que esté relacionado a una tabla o una regla, no se visualiza el control de relación, en ese caso se visualiza la correspondiente lista con los valores que la tabla o regla asociada retornen.

3.4.13.2. Contenedores

Los contenedores agrupan un conjunto de elementos dentro de la instancia del formulario de acuerdo a un criterio común. Al momento de modelar se pueden definir diferentes tipos de contenedores, cada uno con características particulares.



Contenedor Simple

Son los contenedores simples que tienen la funcionalidad de permitir la agrupación de elementos y hacen modelable la visibilidad de secciones del formulario, según reglas del negocio.

Contenedor Gráficos

Este tipo de contenedor tal como el contenedor descripto anteriormente, tiene la funcionalidad de permitir la agrupación de elementos y hace modelable la visibilidad de secciones del formulario, según reglas del negocio. Además posee una cabecera y puede plegar o desplegar su contenido haciendo clic en la misma.

Contenedores de Múltiples Ocurrencias

Este tipo de contenedor permite generar múltiples ocurrencias de los campos que se encuentran dentro del mismo. Puede ser configurado como visible o no visible según reglas del negocio.

3.5. Agile Forms

Fase 1: Introducción a Deyel > Modelador Agil - Introducción

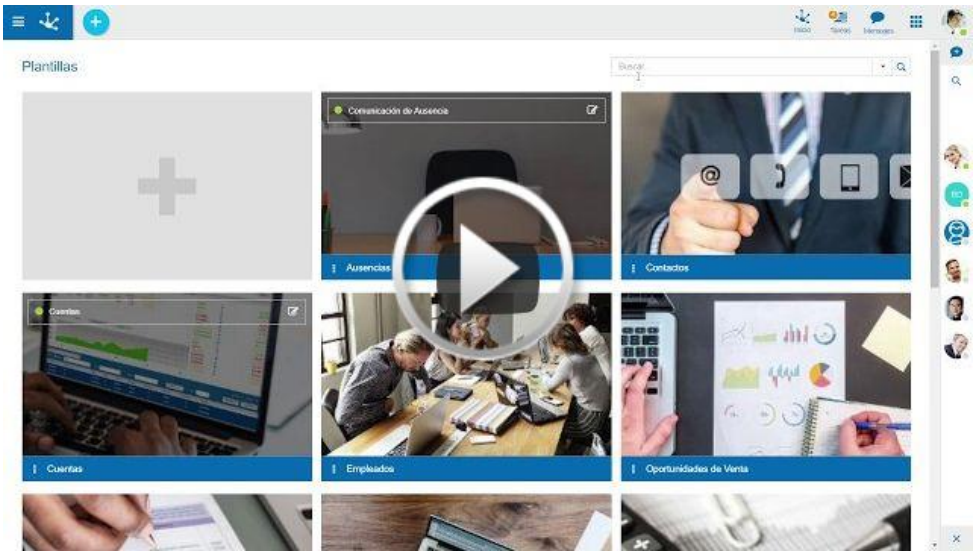
Formularios Agiles es una plataforma para crear y mantener aplicaciones basadas en formularios y procesos de negocio, de forma inmediata.

Los usuarios de negocio de una empresa pueden definir sus formularios web a partir de plantillas de una galería existente o realizando su diseño completo, adecuarlos a las necesidades del negocio, agregarles un proceso, así como también definir los permisos de acceso y su visibilidad, finalmente se publican para su uso.

De esta forma se pueden automatizar y optimizar fácilmente una gran cantidad de procesos de negocio que actualmente están basados en documentos en papel y tareas manuales. Ejemplos de este tipo

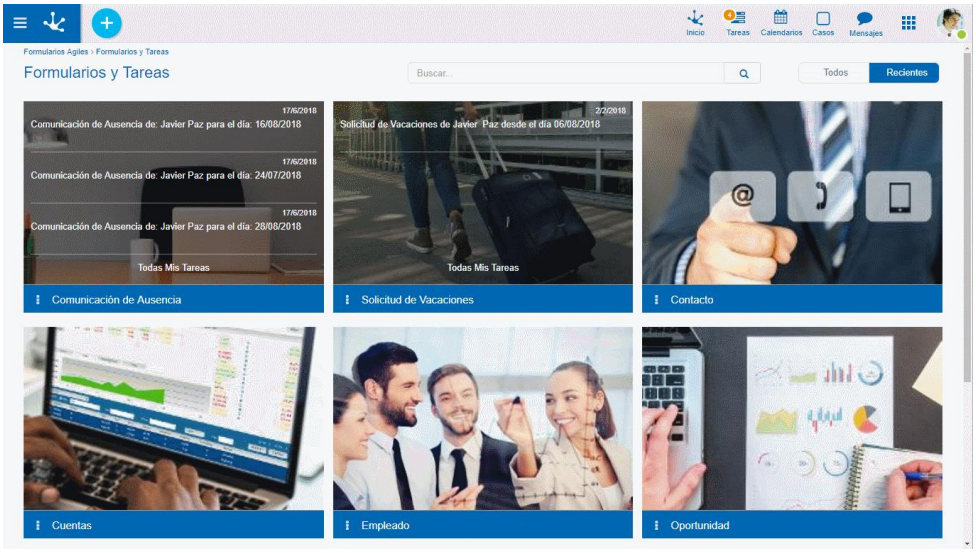
de aplicaciones van desde la autorización de facturas y comprobantes de gastos hasta solicitudes de diversos tipos, formularios de consultas, encuestas y registros, entre otros.

La utilización del formulario permite a los usuarios que participan del proceso, completar información, aprobar o rechazar desde su lista de tareas o usar la red social empresarial para comentar acerca de cada instancia.



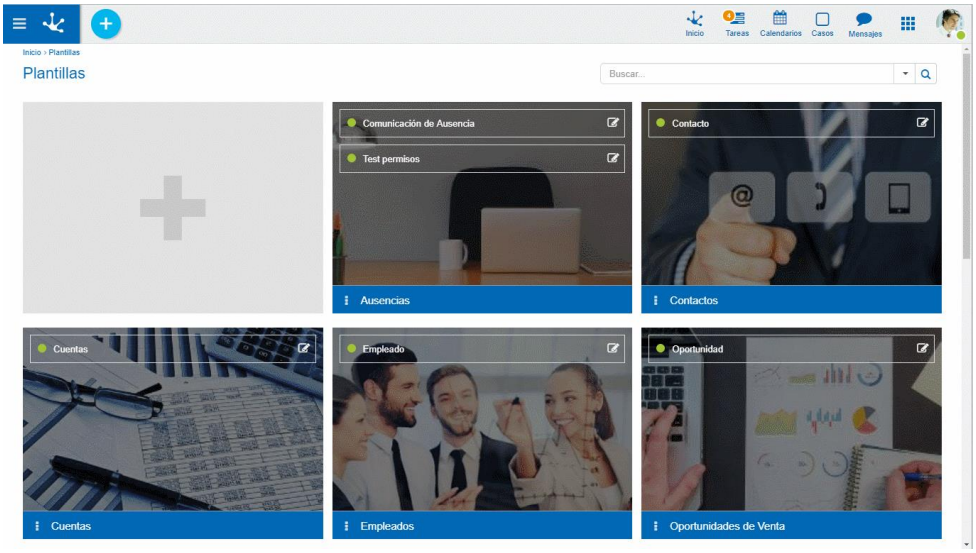
Formularios y Tareas

Los usuarios de negocio que participan del proceso pueden iniciar diferentes formularios y tareas que fueron previamente definidos como formulario o como proceso.



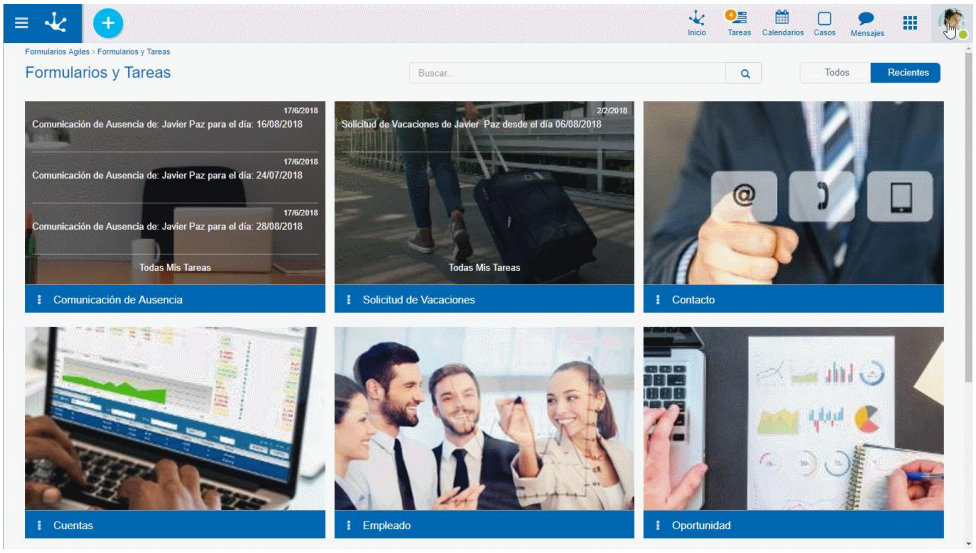
Plantillas

Los usuarios de negocio pueden definir sus propias plantillas o sus formularios, éstos últimos a partir de un diseño preexistente o en forma completa.



Configuración

Los usuarios administradores pueden realizar la configuración de usuarios, permisos, roles y unidades organizacionales.



3.5.1. Conociendo Formularios Agiles

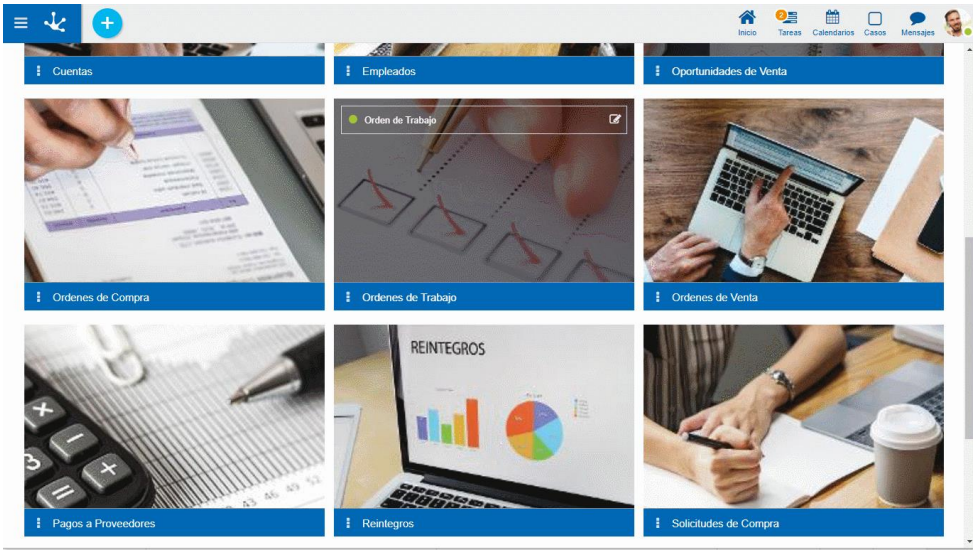
Fase 1: Introducción > Modelador Agil - Parte 1

Fase 1: Introducción > Modelador Agil - Parte 2

Se describen a continuación los pasos para crear rápidamente aplicaciones ágiles.

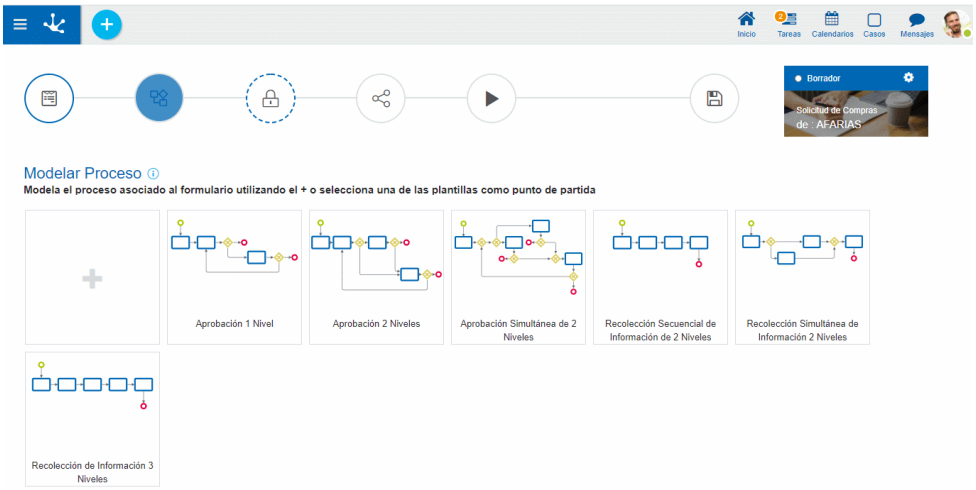
Paso 1

Seleccionar una de las plantillas existentes en la galería para crear un formulario, la cual contiene un conjunto de campos y secciones predefinidos. El formulario puede adecuarse a la necesidad del usuario, modificando sus elementos gráficos, incorporando relaciones y reglas de negocio.



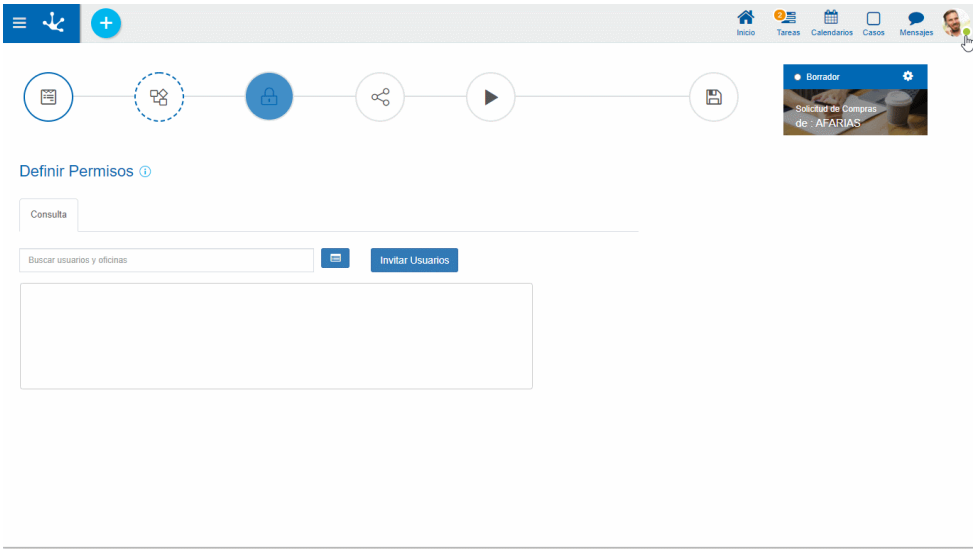
Paso 2

Asociar un proceso al formulario, vinculándolo con uno de los procesos predefinidos, donde se pueden definir participantes y crear condiciones que permitan determinar el flujo de los datos y la visibilidad de los mismos.



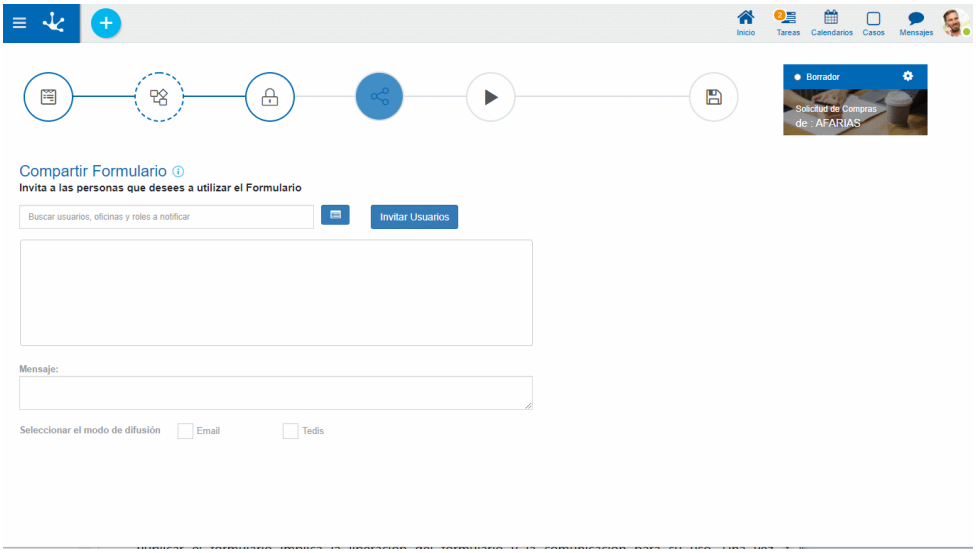
Paso 3

Permitir la utilización del formulario a los usuarios o grupos dentro de la organización para consulta, modificación y anulación, o bien invitar usuarios externos.



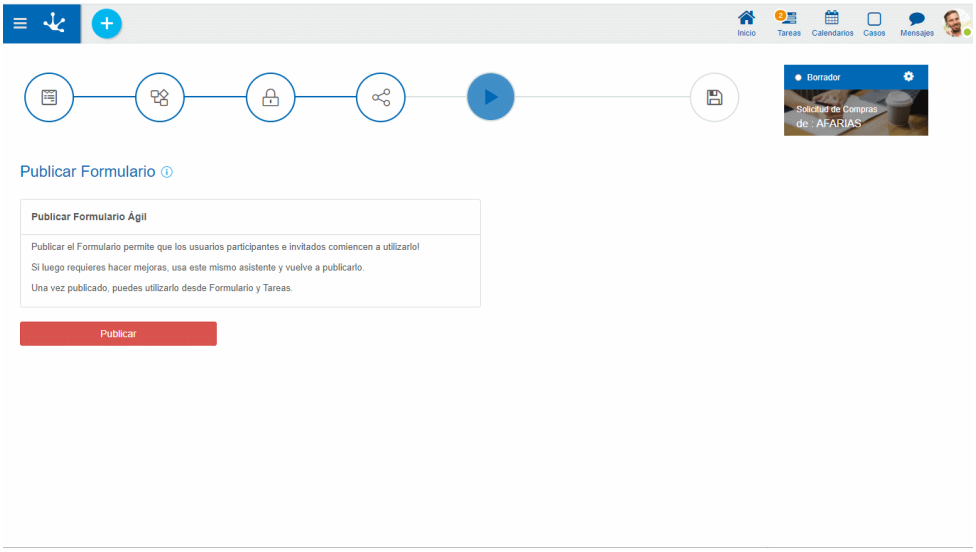
Paso 4

Compartir el formulario difundiendo la funcionalidad del mismo a través de email o de la red social empresarial.



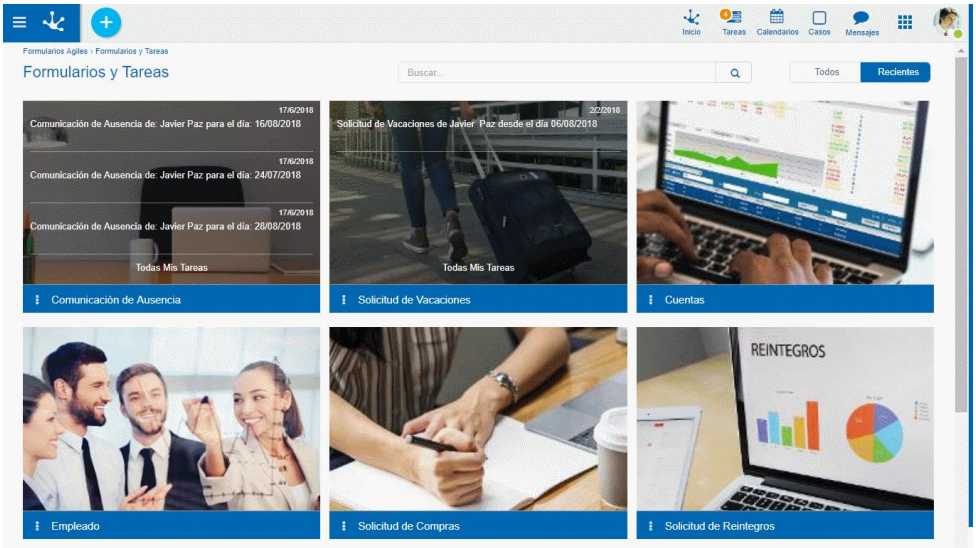
Paso 5

Publicar el formulario implica la liberación del formulario y la comunicación para su uso. Una vez publicado, el formulario puede ser utilizado por los usuarios participantes de acuerdo a los permisos otorgados tanto a nivel del formulario como del proceso si tuviera uno asociado.



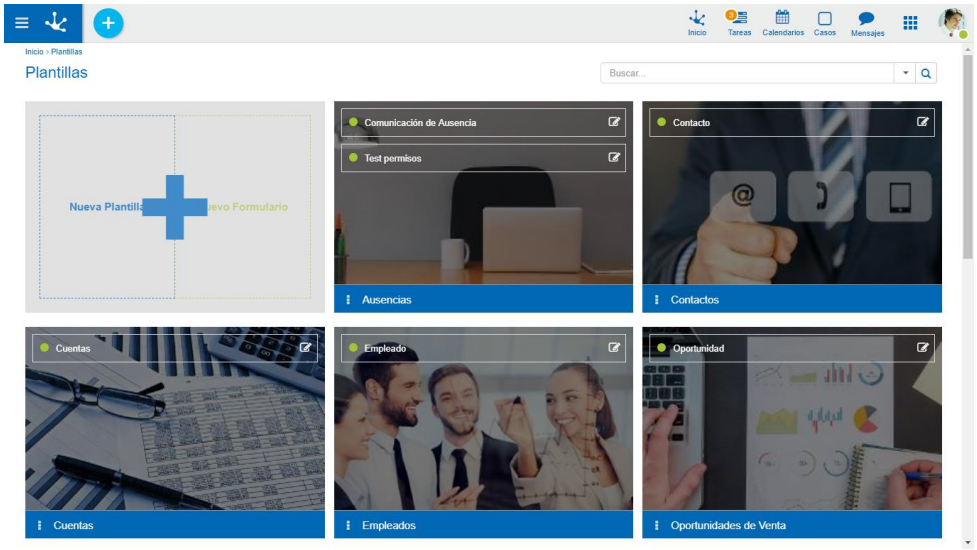
Ejecución de Aplicaciones Agiles

Una vez publicado, el formulario puede ser utilizado por los usuarios participantes de acuerdo a los permisos otorgados tanto a nivel del formulario como del proceso, si tuviera uno asociado.



3.5.2. Plantillas

Para comenzar a utilizar la galería de plantillas se debe abrir el menú de la izquierda, seleccionar la opción “Formularios Agiles”.



El primer elemento permite iniciar diseños nuevos, mientras que los siguientes corresponden a plantillas y formularios previamente definidos. Los elementos identificados con una barra color azul corresponden a plantillas, mientras que las identificados con una barra verde corresponden a formularios sin plantilla.

Cada elemento de la galería permite realizar diferentes acciones dependiendo de lo que representa, se describen a continuación las diferentes posibilidades.

Nueva Plantilla

Desde el signo del primer elemento, se puede crear una plantilla a partir de un diseño propio, que luego será utilizada para definir formularios ágiles basados en la misma.

Nuevo Formulario

Desde el signo del primer elemento, también se puede crear un formularios ágil a partir de un diseño propio, es decir sin utilizar una plantilla preexistente. Dicho diseño se realiza mediante el modelado de formularios de **Deyel**.

Plantilla

Desde un elemento identificado con una barra azul se puede crear un formulario basado en una plantilla preexistente.

Secciones del Elemento

Líneas de Formularios

Si la plantilla tiene formularios ágiles definidos basados en ella, los mismos se visualizan en diferentes líneas dentro de cada elemento. Cada formulario se identifica con la propiedad Nombre Descriptivo definida en las propiedades del formulario, en el momento de su creación.

En cada linea pueden visualizarse los siguientes íconos:

- Permite editar el formulario ágil.
- Este ícono solamente se visualiza si el formulario no se encuentra publicado y permite eliminar el formulario ágil.
- Identificador de Estado: El estado del formulario indica la etapa de desarrollo en la que se encuentra el mismo y se identifica por un color.
- Si una plantilla tiene definido una cantidad de formularios mayor a la que puede visualizarse dentro de un elemento, se puede avanzar y retroceder en la lista hacia derecha e izquierda mediante el uso de este ícono.

Menú

Permite visualizar las acciones disponibles para la plantilla.

- Nuevo: Permite crear un nuevo formulario ágil basado en la plantilla.
- Editar: Permite modificar la plantilla utilizando las facilidades de modelado.
- Eliminar: La plantilla puede eliminarse solamente si fueron eliminados todos los formularios basados en la misma, esto se hace desde el cuadro de identificación de cada uno.

Formulario

Desde un elemento identificado con una barra verde se puede editar un formulario creado a partir de un diseño propio, es decir que no está basado en una plantilla.

Secciones del Elemento

Identificador de Estado

El estado del formulario indica la etapa de desarrollo en la que se encuentra el mismo y se identifica por un color.

Menú

Permite visualizar las acciones disponibles para el formulario.

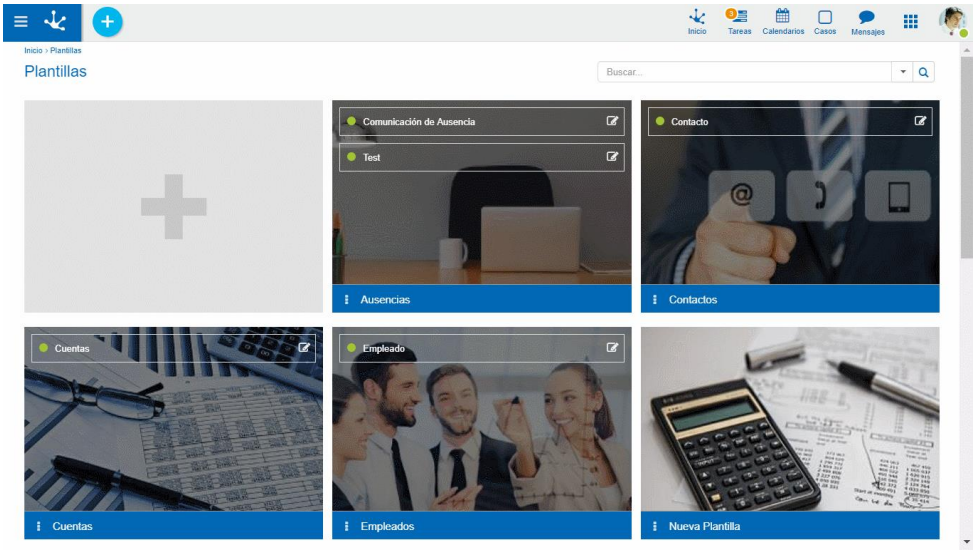
- Editar: Permite modificar el formulario utilizando las facilidades de modelado
- Eliminar: El formulario puede eliminarse solamente si fue eliminada su publicación, esto se hace desde el cuadro de identificación.

Filtros de Búsqueda

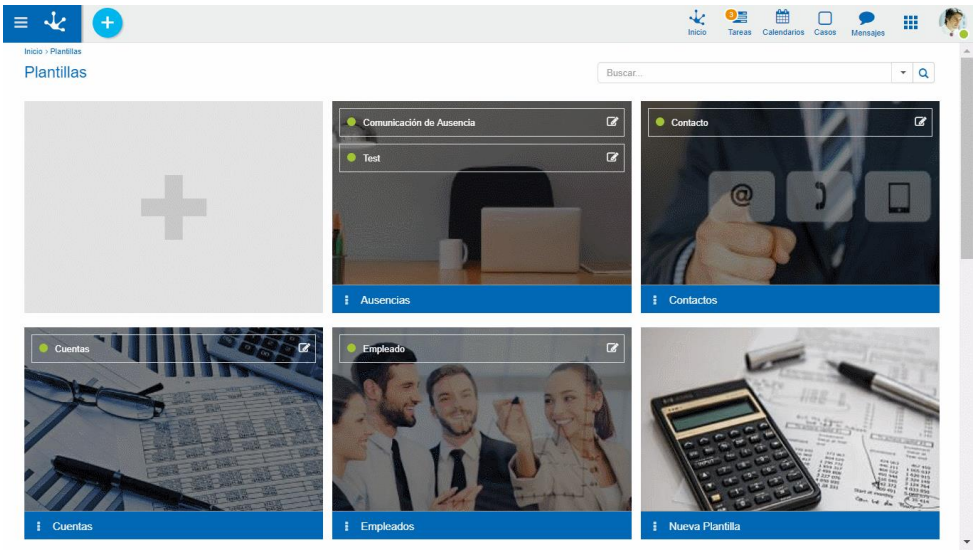
La cantidad de elementos de la galería puede reducirse utilizando un filtro de búsqueda, el cual se aplica al nombre del formulario. A medida que el usuario ingresa caracteres en el campo de búsqueda, la cantidad de elementos en la grilla se reduce a aquellos formularios cuyos nombres contienen los caracteres ingresados.

También pueden agregarse filtros por tipo de elemento, donde puede optarse por plantillas o formularios, y en caso de tratarse de estos últimos también se puede sumar un filtro por estado.

- **Filtro por Tipo**

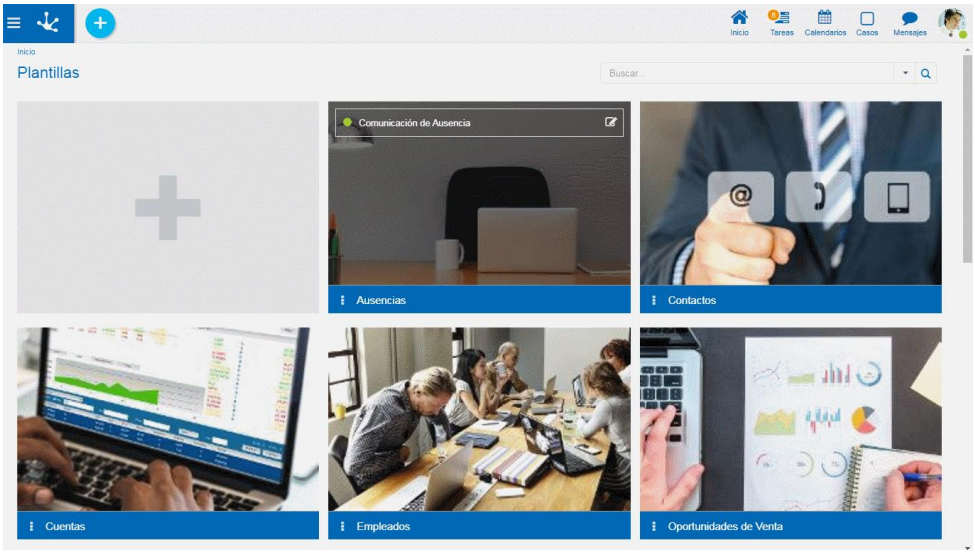


- **FIltro por Tipo y Estado**



3.5.2.1. Crear Plantilla

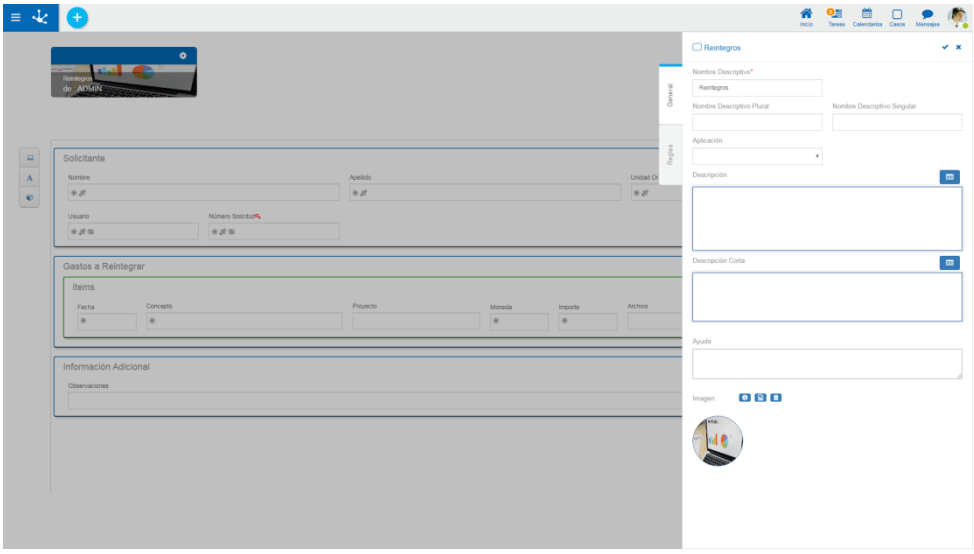
El usuario modelador puede diseñar su propia plantilla para ser utilizada para la creación de nuevos formularios ágiles. El diseño se inicia desde el área de edición vacía, pudiendo incluir campos, contenedores y diferentes elementos gráficos.



La creación se inicia partiendo de la opción "Nueva Plantilla" en la galería de plantillas, lo que lleva a un área de modelado vacía para comenzar un diseño propio. Se pueden definir diferentes elementos gráficos, relaciones y reglas de negocio.

Identificación de la Plantilla

Sobre área de diseño se visualiza un cuadro con la identificación de la plantilla y la imagen que se haya definido para representarla. Las propiedades de la plantilla pueden modificarse en el panel que se abre al seleccionar el ícono ⚙



> *Un asterisco "*" en la etiqueta indica que la propiedad es obligatoria.*

Cuadro de Identificación de la Plantilla

Nombre de la Plantilla
Corresponde con la propiedad Nombre Descriptivo que se indica al crear la plantilla.

Propietario
Es el responsable de la plantilla, quien hizo la definición de la misma.

Imagen
Corresponde con la propiedad Imagen que se indica al crear la plantilla.

Propiedades – Pestaña General

Se accede haciendo clic en el ícono que se encuentra en el cuadro de identificación de la plantilla.

Nombre Descriptivo
Es el nombre que se visualiza en la grilla de plantillas. Soporta multi-idioma.

Nombre Descriptivo Plural
El texto ingresado en esta propiedad es utilizado como título en la grilla de resultados del formulario, mientras que si no se completa, se utiliza la propiedad Nombre. Soporta multi-idioma.

Nombre Descriptivo Singular
El texto ingresado en esta propiedad es utilizado en la lista de entidades relacionadas, mientras que si no se completa, se utiliza la propiedad Nombre. Soporta multi-idioma.

Descripción
Texto que define al modelo describiendo su funcionalidad y opcionalmente su contenido. Soporta multi-idioma y puede incluir valores de variables definidas en el área de diseño, que se incorporan mediante el uso del asistente de variables. Los formularios definidos a partir de la plantilla heredan este texto.

Descripción Corta
Versión resumida de la propiedad Descripción. Soporta multi-idioma y puede incluir valores de variables del mismo formulario, que se incorporan mediante el uso del asistente de variables. Puede ser utilizada a modo descriptivo al relacionar entidades. Los formularios definidos a partir de la plantilla heredan este texto.

Ayuda
Texto informativo que se visualiza al presionar el ícono ? cuando se utiliza el formulario generado a partir de esta plantilla, solamente cuando se utiliza fuera del workflow. Soporta multi-idioma.

Imagen
Imagen que se visualiza en la grilla de plantillas y en la barra superior al crear modelos. Se recomienda que esté relacionada con la funcionalidad o información que el modelo representa. Los formularios definidos a partir de la plantilla heredan esta imagen.
Informa sobre la utilidad de la imagen.
Permite elegir una imagen desde una ventana de selección de archivos.
Permite eliminar la imagen actual asociada a la plantilla.

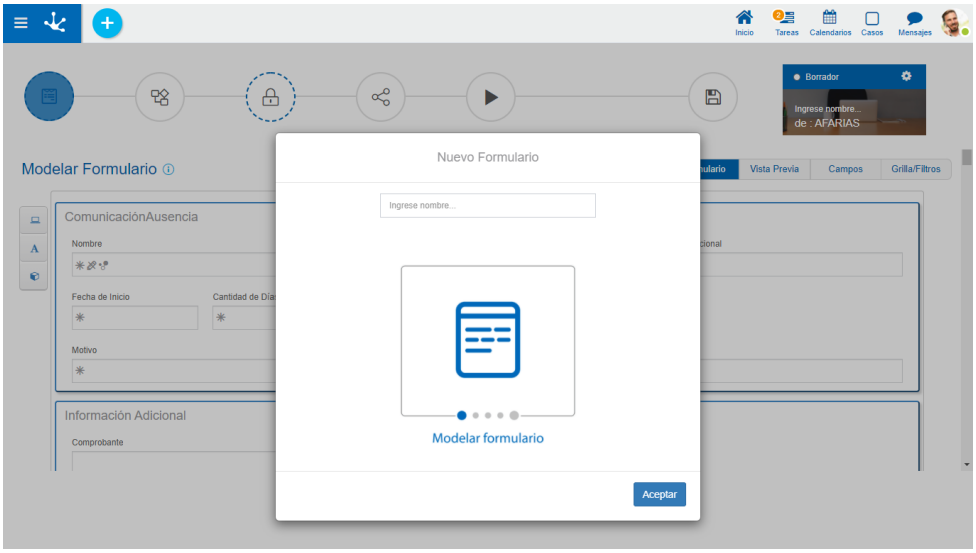
Propiedades – Pestaña Reglas

La última pestaña del panel lateral corresponde a las reglas de validación asociadas a la plantilla.

3.5.2.2. Crear Formulario

La creación de un formulario ágil puede realizarse a partir de un diseño propio o bien a partir de una plantilla preexistente en la galería.

En ambos casos se inicia el Asistente de Formularios Agiles, donde se define el nombre del formulario y se completan los pasos que permiten llegar a su publicación.



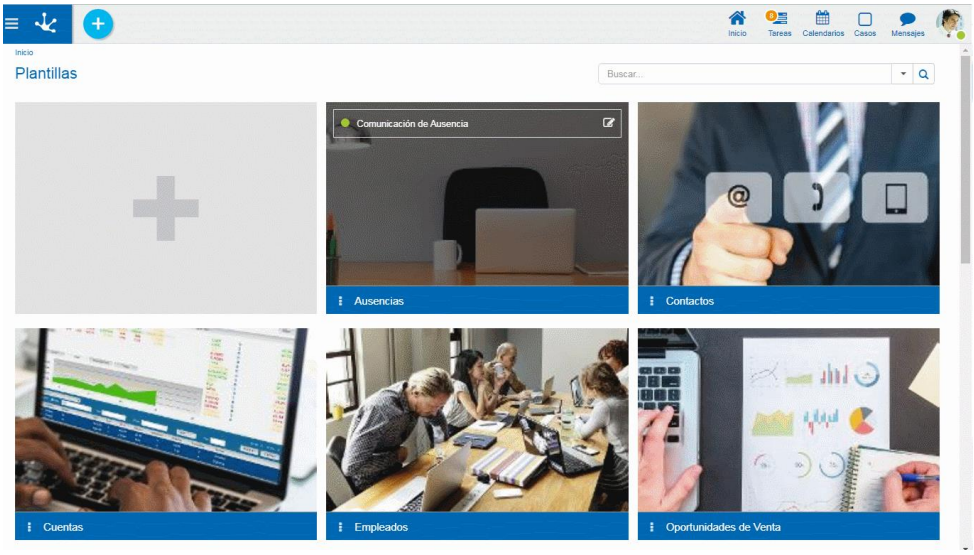
La creación de un formulario ágil incluye la creación de la entidad correspondiente, así como también la generación de permisos para usuarios y unidades organizacionales.

Crear Formulario Agil sin Plantilla

Crear Formulario Agil con Plantilla

3.5.2.2.1. Crear Formulario sin Plantilla

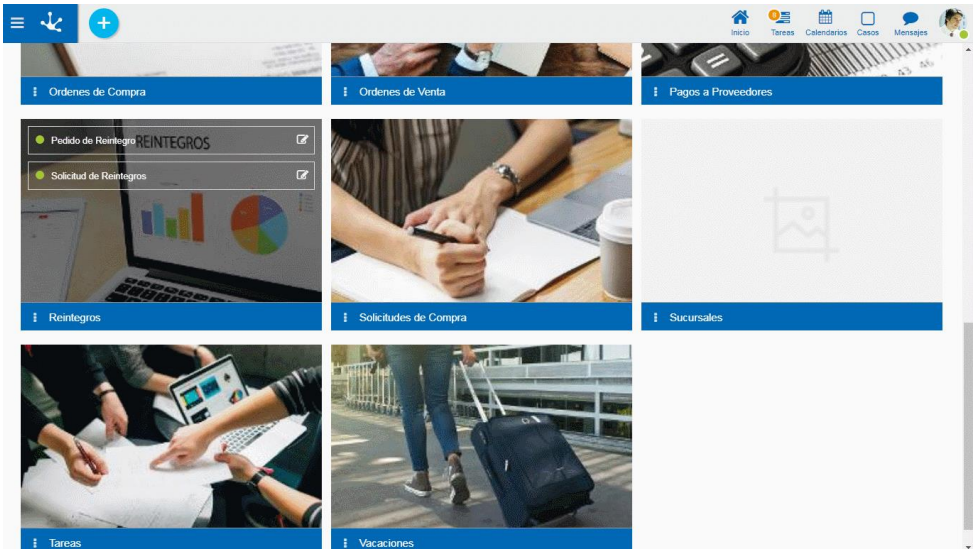
El usuario modelador puede diseñar su formulario desde el área de edición vacía, es decir, sin utilizar una plantilla como base, pudiendo incluir campos simples, iterativos, contenedores y diferentes elementos gráficos.



La creación se inicia partiendo de la opción "Nuevo Formulario" en la galería de plantillas, lo que lleva al Asistente de Formularios Agiles, donde se presenta un área de modelado vacía para comenzar un diseño propio. Se pueden definir diferentes elementos gráficos, relaciones y reglas de negocio.

3.5.2.2.2. Crear Formulario con Plantilla

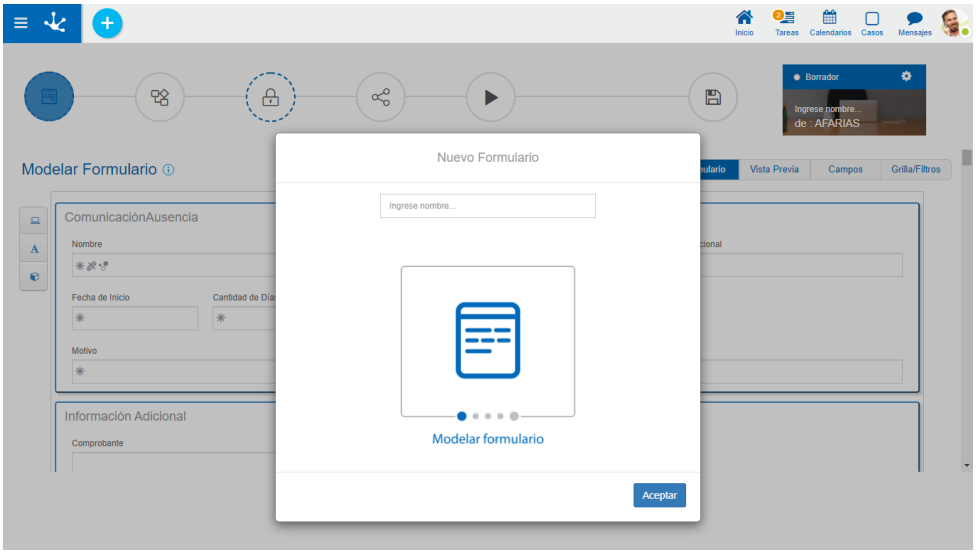
El usuario modelador puede crear su formulario tomando como base una plantilla de la galería.



Se selecciona la plantilla y se presiona la opción de menú "Nuevo" en el ícono para crear un formulario ágil, de esta forma se ingresa al Asistente de Formularios, donde se presenta un área de modelado con el diseño predefinido. El formulario puede adecuarse modificando los elementos gráficos propuestos, incorporando relaciones y reglas de negocio.

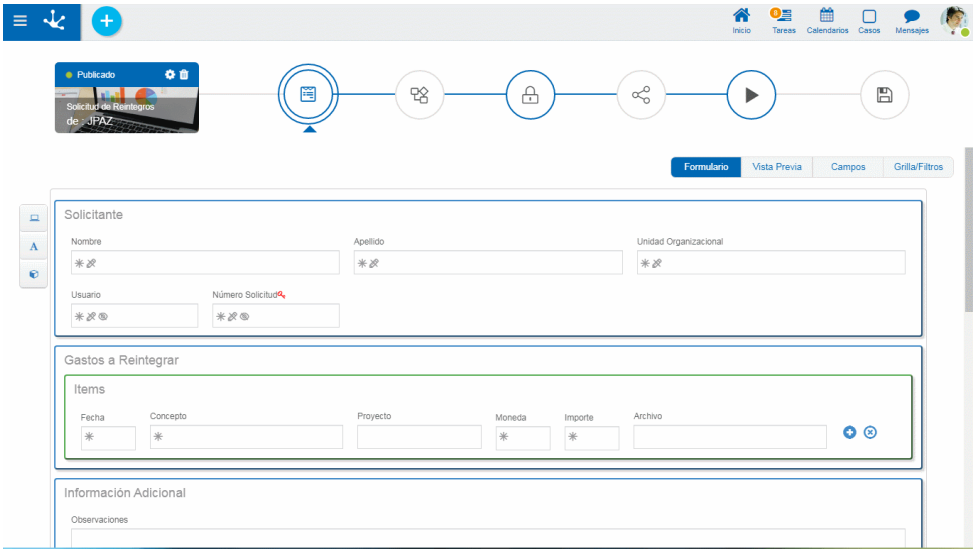
3.5.2.2.3. Asistente de Formularios

Para crear un formulario ágil en primer lugar se debe definir su nombre y luego completar los pasos necesarios para su definición con la ayuda del asistente.



Identificación del Formulario Agil

En la parte superior izquierda se visualiza un cuadro con la identificación del formulario y la imagen que se haya definido para representarlo. Las propiedades del formulario pueden modificarse en el panel que se abre al seleccionar el ícono



Cuadro de Identificación

Estado

El estado del formulario indica la etapa de desarrollo en la que se encuentra el mismo y se define automáticamente a medida que el usuario va trabajando sobre el formulario.

- Borrador: cuando empieza a definirse o fue guardado pero aún no fue publicado.
- Modificado: cuando se modifica luego de haber sido publicado.

- Publicado: se liberó y está disponible para ser utilizado.

Propiedades
Al presionar este ícono se abre el panel de propiedades del formulario.

Eliminar
Al presionar este ícono se abre un panel para que el usuario seleccione la información que desea eliminar.
- Eliminar Datos
- Eliminar Datos y Publicación
- Eliminar como Formulario Agil

Nombre
Corresponde a la propiedad Nombre Descriptivoque se define en las propiedades del formulario, en el momento de su creación.

Propietario
Es el responsable del formulario, quien hizo la definición y es el encargado de publicar el formulario.

Imagen
Corresponde a la propiedad Imagen que se define en las propiedades del formulario, en el momento de su creación.

Pasos para la Generación de un Formulario Agil

La creación de un formulario consta de 5 pasos representados en la barra superior del asistente por medio de círculos, donde el color y el tipo de borde definen la obligatoriedad de cada acción.

- Línea azul continua: acción obligatoria.
- Línea azul cortada: acción obligatoria condicional, es decir, puede ser obligatoria dependiendo de algún paso previo.
- Línea gris continua: acción opcional.

	Modelar Formulario Agil Se utiliza para crear y modificar el diseño del formulario partiendo del inicio o de un modelo existente. Este paso es obligatorio.
	Modelar Proceso (opcional) Se selecciona un proceso de la galería de procesos predefinidos a los que el formulario puede asociarse para su configuración. Este paso no es obligatorio.
	Definir Permisos (opcional) Define permisos para los usuarios que utilizan el formulario. Este paso es obligatorio si no se modela un proceso (Paso 2).
	Compartir (opcional) Permite difundir la funcionalidad que brinda el nuevo formulario. Este paso no es obligatorio.

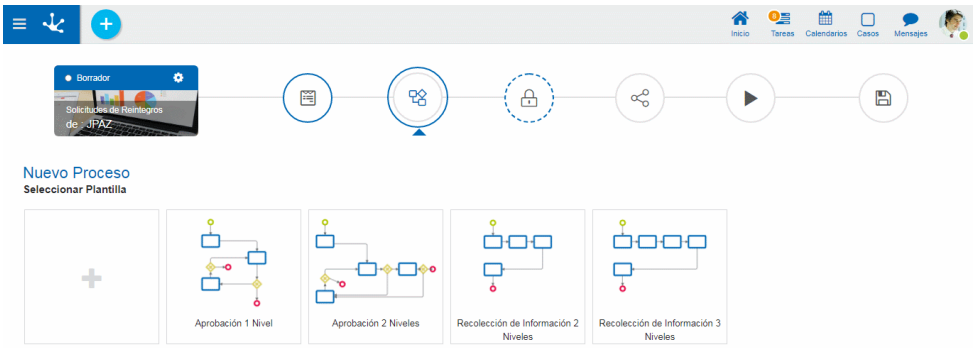
	Publicar Implica la liberación del formulario para su uso. Este paso no es obligatorio.

Existe un paso adicional que corresponde a la acción de guardar

	Guardar Guarda la configuración y el diseño del formulario, sin que esto implique la puesta en uso del mismo.

3.5.2.2.3.1. Modelar Proceso Agil

Se puede asociar un proceso al formulario ágil, vinculándolo con un workflow predefinido, en el que se definen los participantes y las condiciones que permitan determinar el flujo de los datos y la visibilidad de los mismos.



La opción Modelar Proceso, permite seleccionar una plantilla de la galería de procesos predefinidos. Este paso del asistente de formularios es opcional, ya que no es obligatorio que un formulario tenga un proceso asociado.

Desde el diagrama del proceso seleccionado, se configuran las propiedades de la pestaña General del proceso, las propiedades de los participantes y las de los elementos gráficos, haciendo doble click sobre los mismos o accediendo al menú contextual de cada figura. En el caso de las actividades, están disponibles las propiedades que se encuentran en la pestaña General y Ejecución, exceptuando la opción Editar en Modelador. En el caso de los eventos, algunos tipos están restringidos.

El ícono indica que es obligatorio configurar las propiedades del elemento gráfico. Una vez que las mismas fueron configuradas, el ícono desaparece.

La opción Seleccionar otra Plantilla, en la barra de herramientas superior, permite reiniciar el modelado de proceso. Al elegir esta acción se pierde la configuración realizada .

3.5.2.2.3.2. Modelar Permisos Agiles

Para cada una de las acciones que pueden realizarse sobre el formulario ágil, se pueden seleccionar los usuarios y unidades organizacionales que tengan autorización para ejecutarlas.

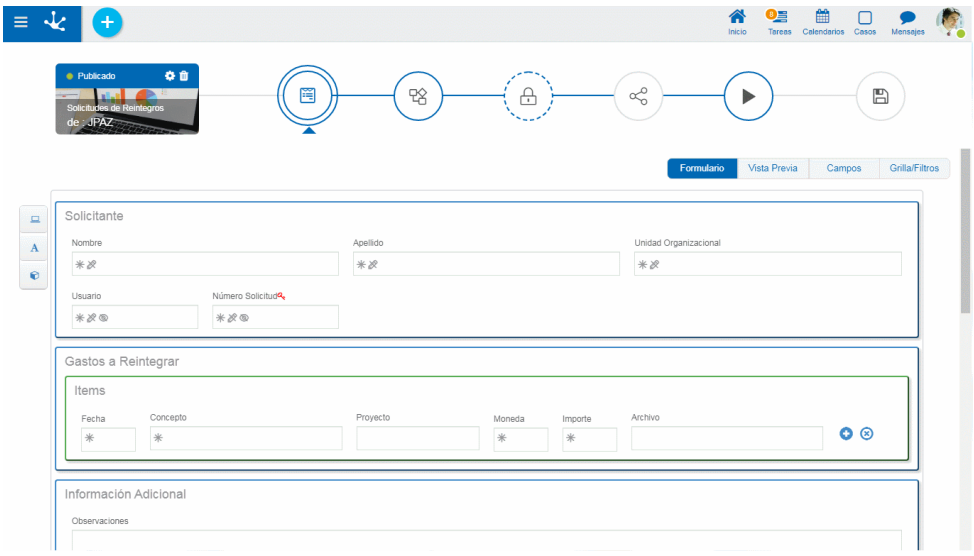
En caso de haberse modelado un proceso asociado al formulario, este paso es opcional y solamente se pueden definir permisos de consulta para los usuarios, mientras que en caso de no tener un proceso asociado al formulario, es obligatorio completar este paso del asistente de formularios.

En cada una de las pestañas se definen los permisos de los usuarios que utilizan el formulario para realizar siguientes acciones:

- **Consulta**
  Con el permiso de consulta el usuario puede buscar y visualizar los datos del formulario cargados por él o por otros usuarios.

- **Creación y Modificación**
  Con el permiso de creación y modificación el usuario puede crear y modificar los datos de los formularios, incluyendo implícitamente el permiso de consulta.

- **Eliminación**
  Con el permiso de eliminación el usuario puede eliminar los datos de los formularios, incluyendo implícitamente el permiso de consulta.



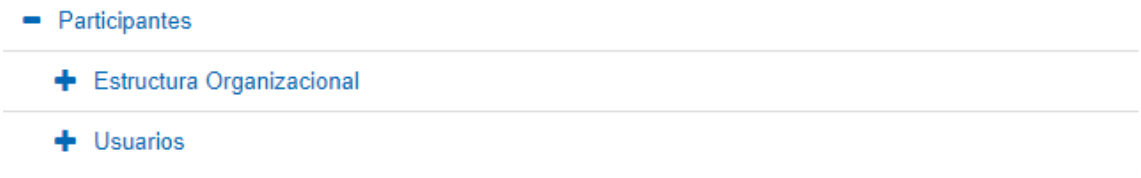
Propiedades

Barra de búsqueda

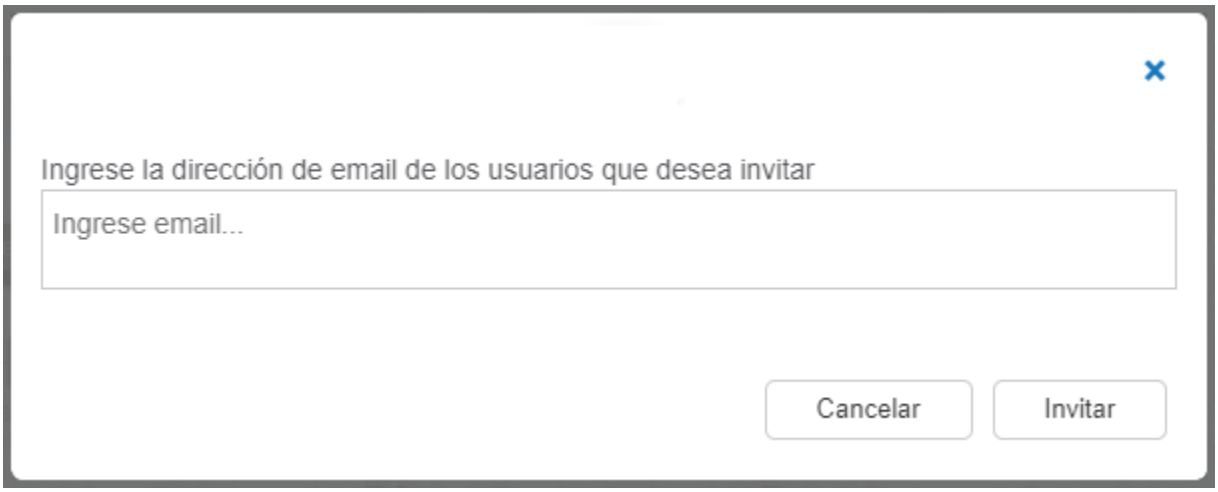
Se pueden buscar los usuarios y unidades organizacionales por nombre, utilizando la barra de búsqueda con la facilidad de autocompletar.

Asistente

Mediante el uso del asistente se pueden seleccionar los usuarios y unidades organizacionales desde un árbol jerárquico.



Invitar Usuarios
El botón permite definir como responsable a usuarios invitados por email, para lo que se debe ingresar la dirección o lista de direcciones de email. Si se invita a más de un usuario, automáticamente se inicia la generación de un nuevo rol.
Una vez presionado el botón "Invitar", se le concede el permiso al usuario, agregándolo en el Area de Usuarios y Oficinas.

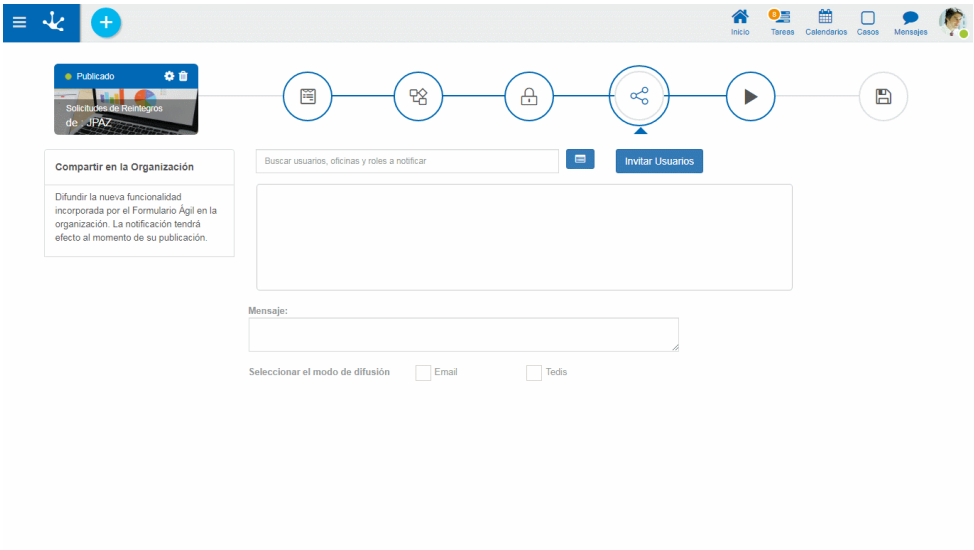


Area de Usuarios y Oficinas
Representa el conjunto de usuarios y unidades organizacionales para los distintos permisos.
Recordar que los responsables que presenten permisos de creación o eliminación, implícitamente podrán consultar.

3.5.2.2.3.3. Compartir Formulario Agil

Es posible definir el envío automático de un mensaje, el cual es enviado al momento de la publicación del formulario, pudiendo seleccionar como destinatarios varios tipos distintos de participantes.

Este paso del asistente de formularios puede no completarse ya que no es obligatoria la difusión de un mensaje.



Propiedades

Barra de búsqueda

Se pueden buscar los usuarios, unidades organizacionales y roles por nombre, utilizando la barra de búsqueda con la facilidad de autocompletar. En el caso de que se seleccione una unidad o un rol, el mensaje será enviado a sus integrantes.

Asistente

Mediante el uso del asistente de participantes se pueden seleccionar los usuarios, unidades organizacionales y roles desde un árbol jerárquico.

Invitar Usuarios

El botón permite definir como participante a usuarios invitados por email, para lo que se debe ingresar la dirección o lista de direcciones de email.
Una vez presionado el botón "Invitar", se agrega el usuario en el Area de Destinatarios de Notificación.

Area de Destinatarios de Notificación

El área de destinatarios de notificación indica los usuarios, unidades o roles que serán notificados.

Mensaje

Permite definir el texto que se utiliza en el mensaje que se utiliza para la difusión.

Modo de Difusión

Marcas que permiten seleccionar el medio para notificar la publicación del nuevo formulario. Puede realizarse la difusión por medio de un email o por medio de chats.

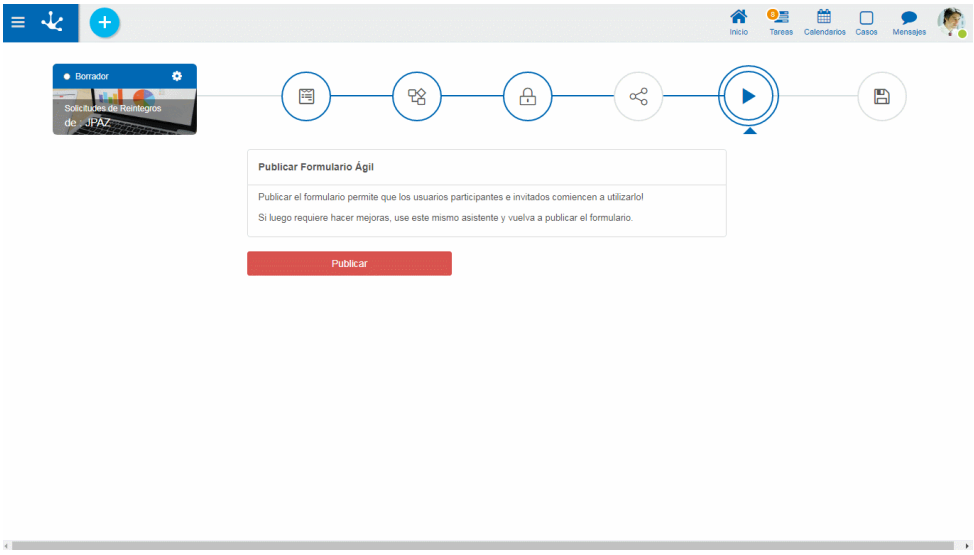
3.5.2.2.3.4. Publicar Formulario Agil

Publicar un formulario ágil permite que los usuarios participantes e invitados comiencen a utilizarlo. Una vez publicado, este puede ser utilizado por los usuarios de acuerdo a los permisos otorgados tanto a nivel del formulario como del proceso si tuviera uno asociado.

En caso de haberse definido un proceso asociado al formulario ágil en el segundo paso del asistente, la publicación implica que el workflow estará disponible para su uso.

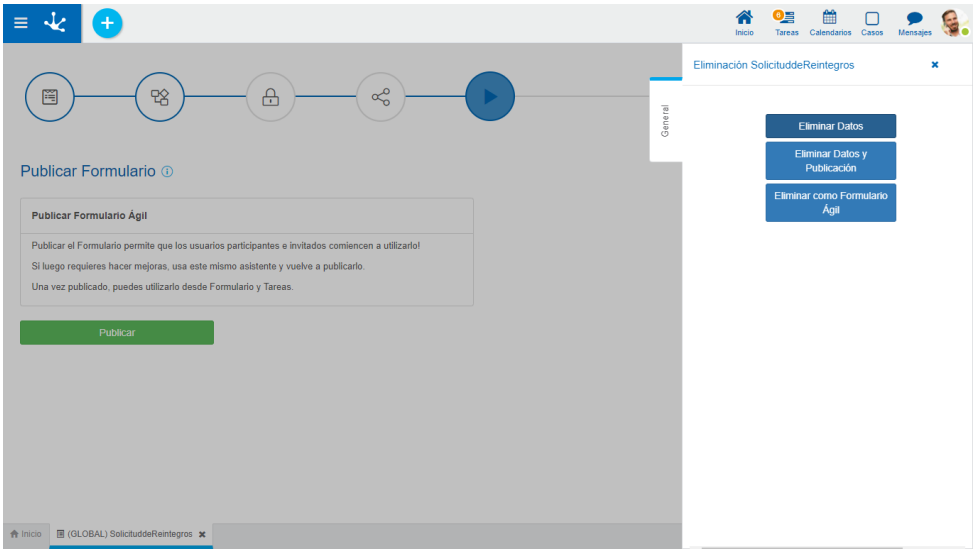
Si se requiere aplicar mejoras posteriores a la publicación, se debe volver a publicar con este mismo asistente.

El botón para publicar el formulario pasa de color rojo a verde luego una vez realizada la publicación.



3.5.2.2.3.5. Eliminar Formulario Agil

En el panel de eliminación se visualizan las tres modalidades de eliminación de un formulario ágil.



Opciones

Eliminar Datos

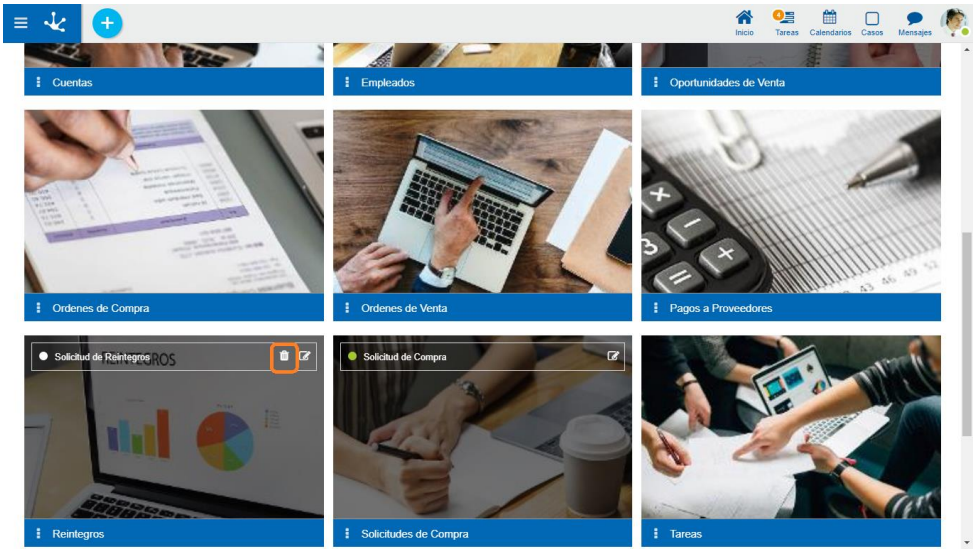
Permite eliminar las instancias del formulario.

Eliminar Datos y Publicación

Permite eliminar las instancias del formulario y volver atrás la publicación del mismo, es decir que su estado pasa de "Publicado" a "Borrador".

Una vez eliminados los datos del formulario ágil y su publicación, en la galería de plantillas se habilita el ícono en la línea correspondiente al formulario en el elemento del formulario ágil. Dicho ícono

permite eliminar el formulario que fue creado a partir de la plantilla. Si se tratara de un formulario que no tiene asociada una plantilla, entonces el mismo se borra desde el menú del elemento ⋮.



Eliminar como Formulario Agil

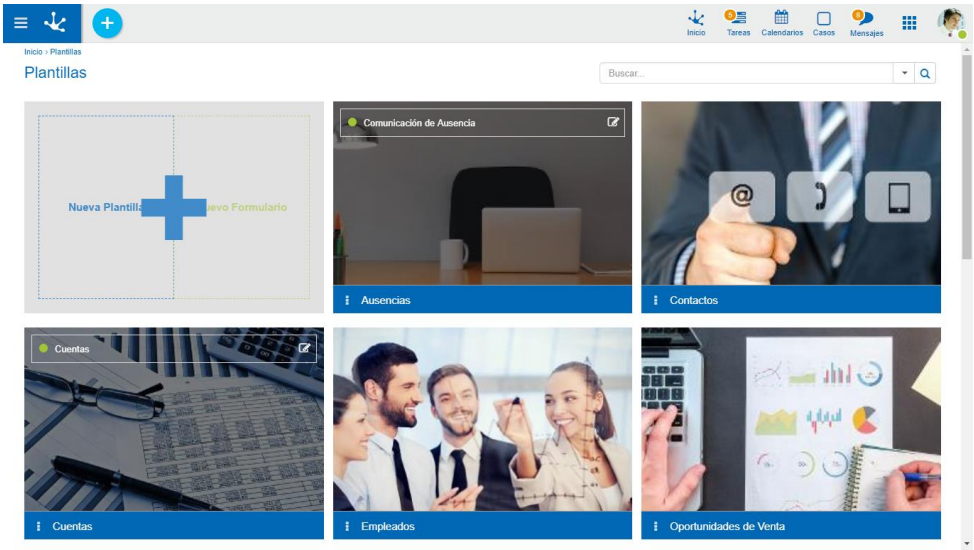
Permite convertir el formulario ágil en un formulario de **Deyel**, es decir que puede utilizarse desde el modelador de formularios. Si el formulario ágil incluye un proceso modelado, entonces este último también se convierte en un objeto de **Deyel**, que puede utilizarse desde el modelador de procesos.

Una vez eliminado el formulario ágil, deja de visualizarse en la galería de plantillas en el modelador ágil y en el modelador Deyel deja de verse como un objeto de tipo "Formulario Agil" para pasar a visualizarse como un objeto de tipo "Formulario". Si el formulario ágil incluyera un proceso, entonces también se crea un objeto de tipo "Proceso".

A partir de esta conversión los objetos resultantes pueden ser utilizados por los modeladores de TI y ya no por los usuarios de negocio.

3.5.3. Modelado de Formularios Agiles

El modelador de formularios es una herramienta que permite diseñar gráficamente formularios, así como también definir las características de sus campos y comportamiento en los procesos asociados.



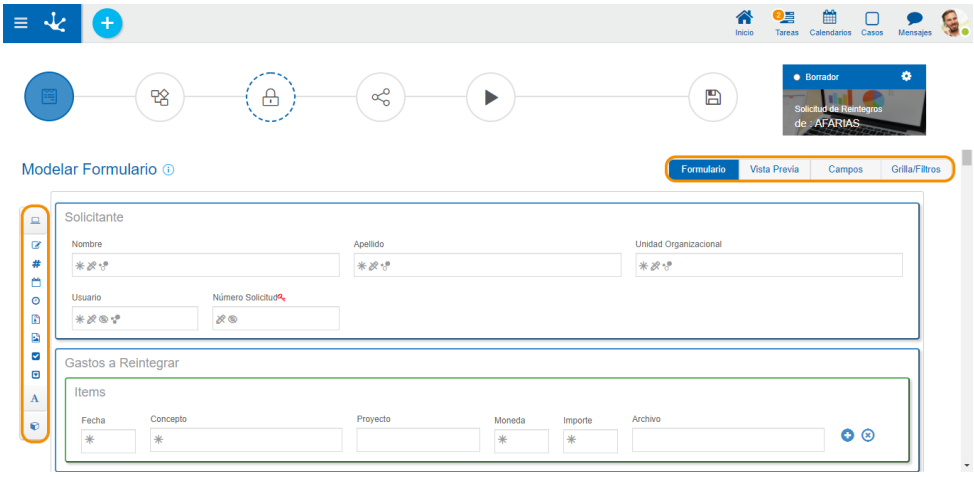
3.5.3.1. Facilidades de Modelado

Nuevo Formulario

El usuario modelador puede definir un nuevo formulario, que luego de ser publicado puede ser utilizado en el portal.

El área del modelado de formulario cuenta con diferentes secciones:

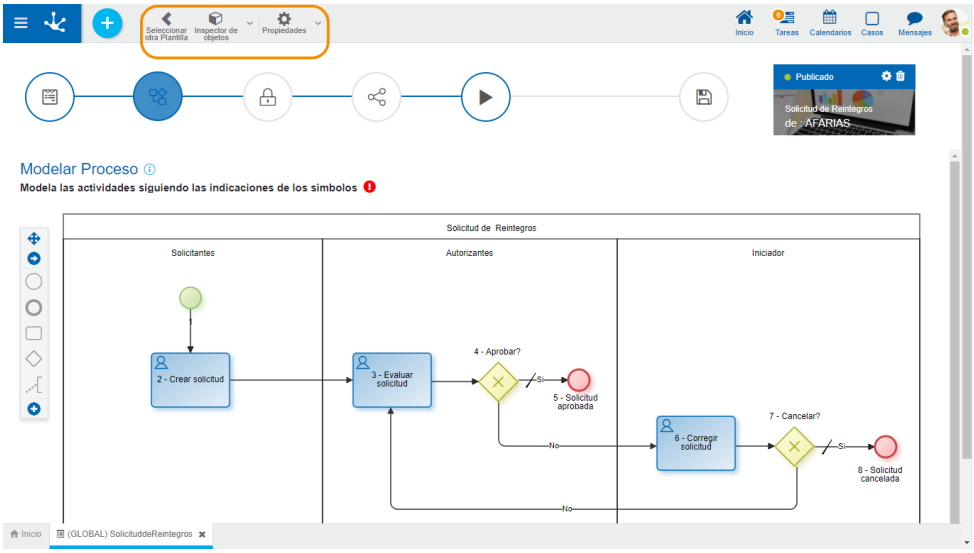
- Opciones de Diseño
- Barra de Herramientas Lateral Izquierda



3.5.3.1.1. Barra de Herramientas Superior

This toolbar contains icons and submenus from which you can perform operations on the object. Depending on its state, some options may be disabled.

The bar is displayed on the toolbar of **Deyel**.



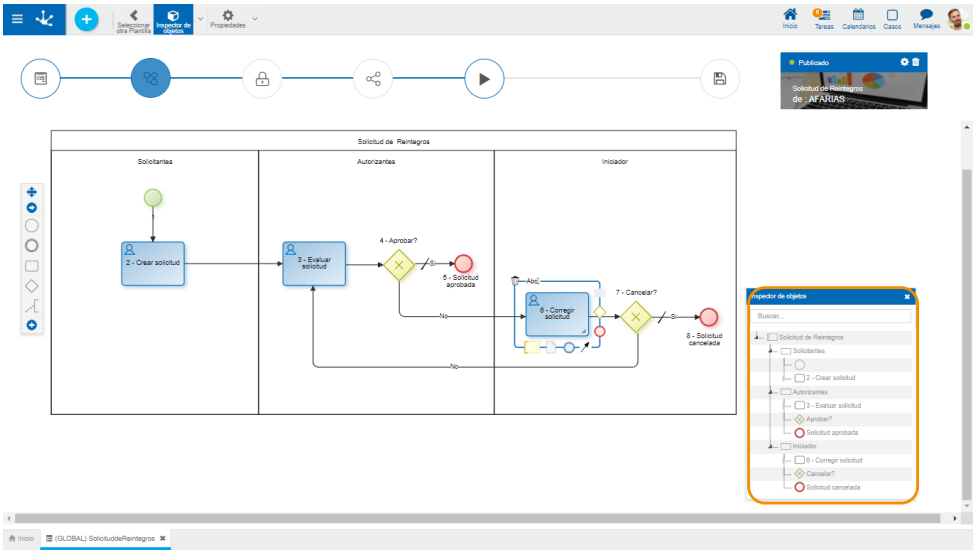
Seleccionar otra Plantilla

Permite seleccionar una plantilla de proceso diferente o generar una nueva, para el formulario ágil que se está definiendo.

Inspector de Objetos

Al presionar este ícono se visualiza/oculta un panel donde se muestran todos los elementos del proceso con estructura de árbol.

Esta visión facilita la tarea de análisis y desarrollo, especialmente en diagramas complejos.



En la parte superior del inspector se encuentra un campo de búsqueda que permite buscar elementos en base a su nombre, código de identificación, comportamiento referido a la ejecución de una actividad, o acciones automáticas definidas. Una vez ingresado el texto en el campo de búsqueda, el inspector muestra en forma resaltada los elementos del proceso que coincidan con el texto ingresado.

Desde el inspector de objetos se puede abrir el panel de propiedades de cada elemento al hacer doble clic sobre el nombre del mismo.

A la derecha del ícono se visualiza el ícono para abrir el segundo submenú.

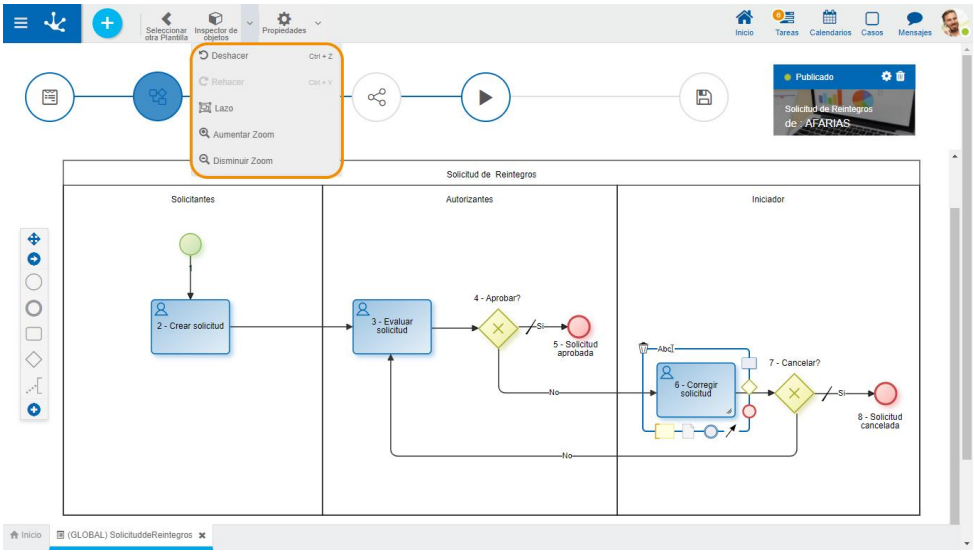
Propiedades

Abre el panel de propiedades del formulario.

A la derecha del ícono se visualiza el ícono para abrir el segundo submenú.

3.5.3.1.1.1. Primer Submenú

Este submenú puede abrirse presionando el ícono que se encuentra inmediatamente a la derecha del ícono .



Deshacer

Vuelve atrás la última modificación realizada al diagrama. El mismo comportamiento se obtiene con la combinación de teclas ctrl+z.

Rehacer

Rehace la acción sobre el diagrama que fue eliminada con la opción "Deshacer". El mismo comportamiento se obtiene con la combinación de teclas ctrl+y.

Lazo

Permite seleccionar los elementos gráficos que quedan enlazados al presionar el botón derecho del mouse sobre un punto del diagrama y arrastrar hacia otro extremo sin soltar. Las figuras enlazadas quedan marcadas temporalmente con un borde y pueden ser desplazadas utilizando el cursor o eliminadas en conjunto.

Aumentar Zoom

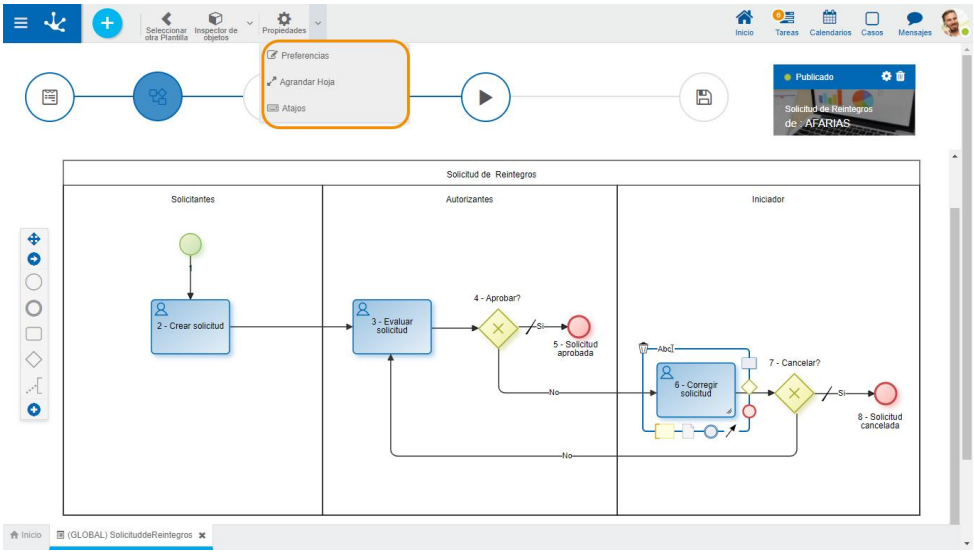
Aumenta el tamaño de visualización del diagrama.

Disminuir Zoom

Disminuye el tamaño de visualización del diagrama.

3.5.3.1.1.2. Segundo Submenú

Este submenú puede abrirse presionando el ícono que se encuentra inmediatamente a la derecha del ícono .



Preferencias

Define las preferencias de visualización para el modelado de procesos.

- Visualizar Identificadores
  Indica si se muestran en el diagrama los identificadores internos de los símbolos. El valor predeterminado es “No”.

- Orientación
  Indica la orientación en que se visualiza el diagrama. El valor predeterminado es “Vertical”.

- Tamaño de Grilla
  Definir la cantidad de pixeles utilizada cuando se mueven los símbolos en el diagrama, cuando se arrastran con el mouse. El valor predeterminado es “1”, aunque puede seleccionarse “10” o “20” para utilizar un movimiento más pronunciado cuando se desplazan los símbolos con el mouse.

- Vista en Miniatura
  Indica si se visualiza la vista en miniatura en el ángulo inferior derecho. Esta vista consiste en un pequeño panel donde se visualiza el proceso completo.

Agrandar Hoja

Incrementa el área de modelado, lo que hace que se visualicen barras de scroll y que en la vista en miniatura se vea sombreada la zona del diagrama que se encuentra visible en la ventana. A medida que se desplazan las barras de scroll, la zona sombreada se desplaza en la vista en miniatura.

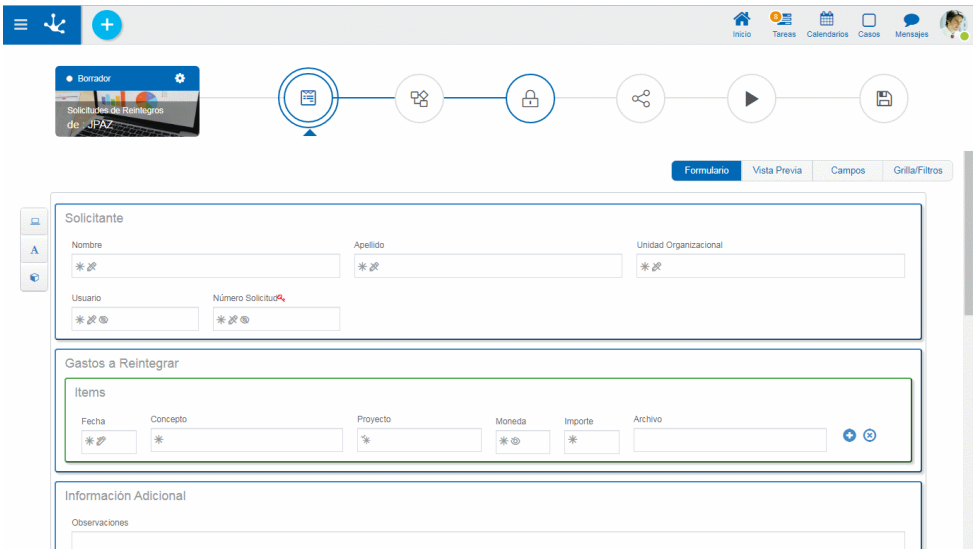
Atajos

Abre un panel con todos los atajos de teclado disponibles para utilizar en el modelado de procesos.

3.5.3.1.2. Area de Modelado Gráfico

El área de modelado gráfico es el espacio donde se arrastran los diferentes elementos desde la Barra de Herramientas Lateral Izquierda, permitiendo modelar la disposición gráfica de los elementos en el formulario.

El área de modelado se divide en filas y cada elemento del formulario ocupa al menos una celda dentro de la fila.



Operaciones sobre los elementos

Al arrastrar un elemento desde la barra de herramientas izquierda al área de modelado gráfico, el mismo se inserta en el lugar elegido. Al pasar el mouse sobre el elemento, se visualiza un conjunto de íconos que permiten realizar distintas operaciones. Las operaciones disponibles dependen del tipo de elemento con el que se esté trabajando.

Permite eliminar un elemento. Al hacer clic sobre este ícono se muestra un mensaje de confirmación que, al ser presionado, elimina el elemento modelado.
En caso que el formulario se encuentre publicado y ya cuente con datos cargados, este ícono se encuentra deshabilitado.

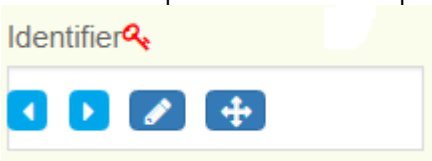
Permite modificar el ancho gráfico del elemento. El ícono derecho agranda el ancho en una unidad y el izquierdo lo disminuye en la misma medida.

Permite ingresar a la sección de propiedades del elemento. Para mayor información sobre la configuración de los distintos tipos de elementos, ver Propiedades de Elementos.

Permite mover el campo a cualquier otro sector del área del modelador. Se puede arrastrar el campo manteniendo el clic sobre este símbolo y soltándolo sobre una fila, o sobre los separadores de celdas, así el campo se desplaza a la zona indicada.

Permite definir el campo como clave única del formulario. Una vez definido un campo clave en el formulario, el botón llave no se visualiza sobre el resto de los campos.

Al definir el campo como clave se representa con una llave a la derecha de la etiqueta.

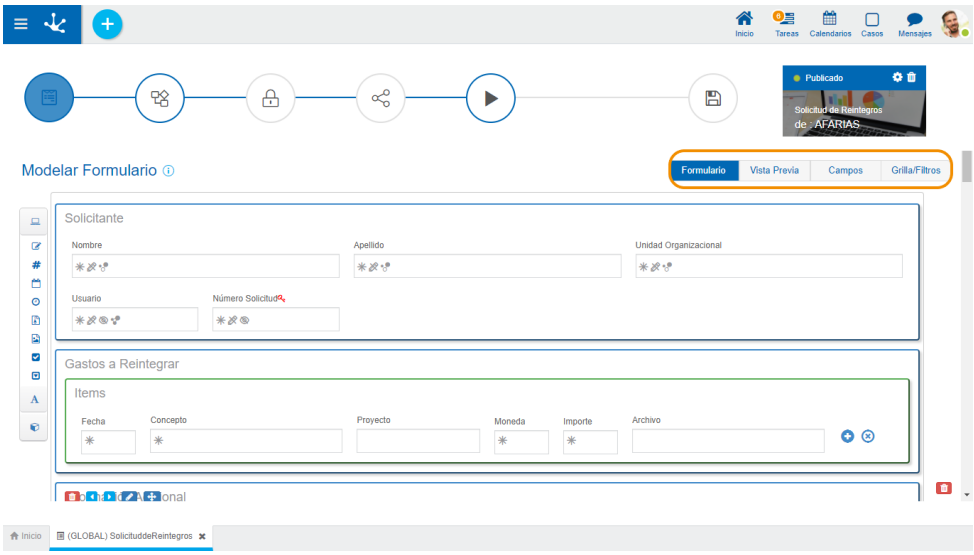


Visualización de íconos de reglas

En cada campo del formulario se visualizan los íconos correspondientes a las reglas establecidas en ellos, de acuerdo al criterio de visualización de reglas en el modelado

3.5.3.1.3. Opciones de Diseño

Al crear o modificar un formulario, permite definir el conjunto de campos, el tipo de información que se almacena en cada uno, la distribución de los mismos dentro del formulario y con qué características se presentan al usuario. De esta forma se puede trabajar sobre el modelo de datos y su visualización al mismo tiempo.

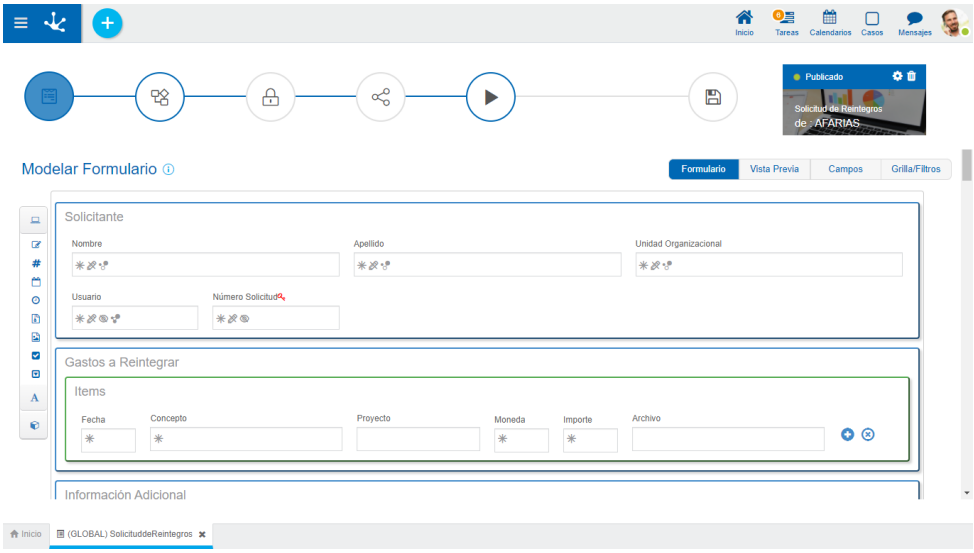


- Formulario
- Vista Previa
- Campos
- Grilla y Filtros

3.5.3.1.3.1. Formulario

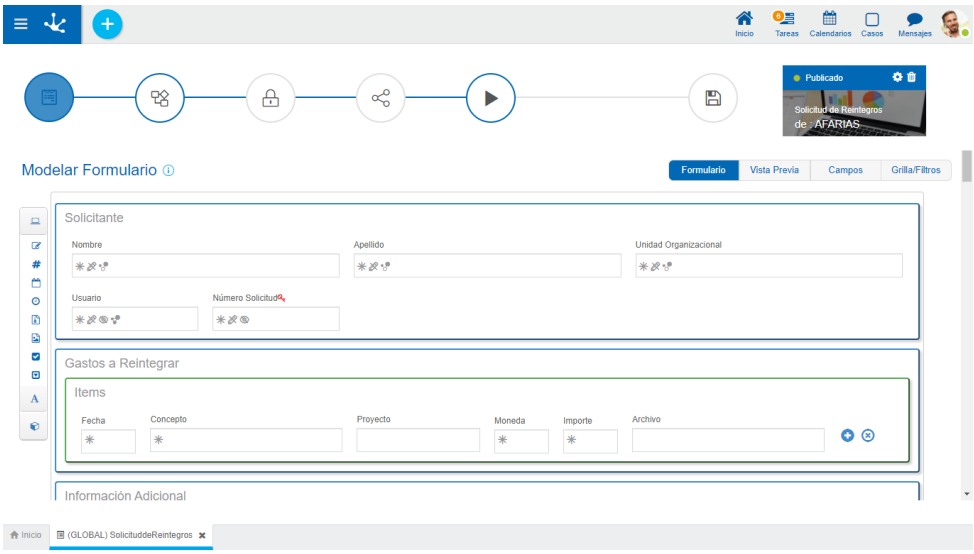
Se presenta el área de modelado gráfico cuando se inicia el modelador, donde pueden definirse campos, elementos gráficos y contenedores.

Estos elementos están disponibles en la barra de herramientas lateral, donde se explica su estructura y funcionamiento en detalle.



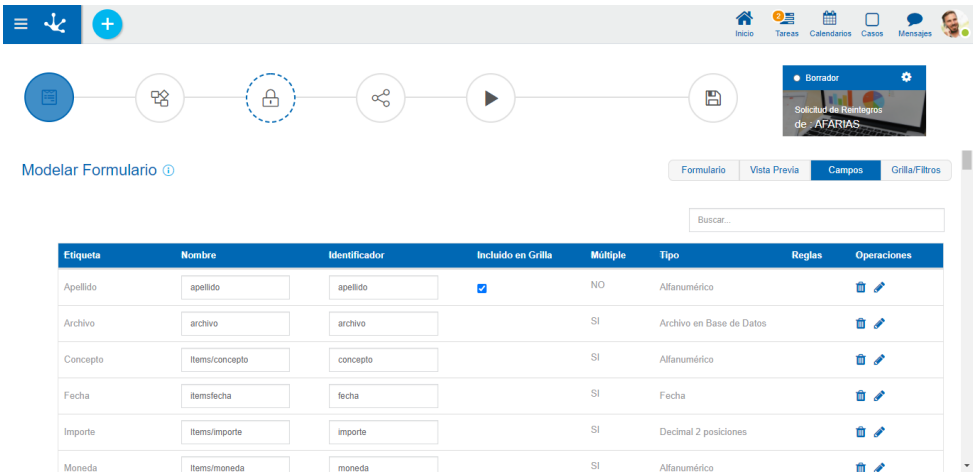
3.5.3.1.3.2. Vista Previa

Permite tener una visión del formulario modelado, mostrando cómo se vería en el portal de usuarios. Por ejemplo, se ocultan botones propios de la edición de la pestaña "Formulario" y se visualizan los contenedores (acordeones y pestañas), se ocultan los contenedores que tengan activa la propiedad de ocultar, etc.



3.5.3.1.3.3. Campos

Permite ver la lista de todos los campos del formulario en una tabla paginada, con capacidades de búsqueda rápida, filtros, orden, visualizando las propiedades de cada uno y permitiendo realizar operaciones.



Columnas

Etiqueta
Es el nombre que se asigna al campo para ser reconocido por el usuario.

Nombre
Es el nombre generado automáticamente a partir de la propiedad Etiqueta, por el que el campo es reconocido por el sistema.

Edición de Nombre
Permite la edición de la propiedad Nombre para que el usuario pueda modificarla. Este campo se encuentra habilitado solo cuando el formulario no presenta datos.

Incluido en Grilla
Indica si el atributo es parte de la grilla de resultados.

Múltiple
Indica si el campo tiene múltiples ocurrencias.

Tipo
Indica el tipo de dato del campo.

Clave
Indica si el campo es único y sirve como identificador en el formulario.

Reglas

En caso que el campo tenga alguna regla asociada de obligatoriedad, de visibilidad, de edición o de validación, se muestra un ícono indicando el tipo de regla.

Operaciones

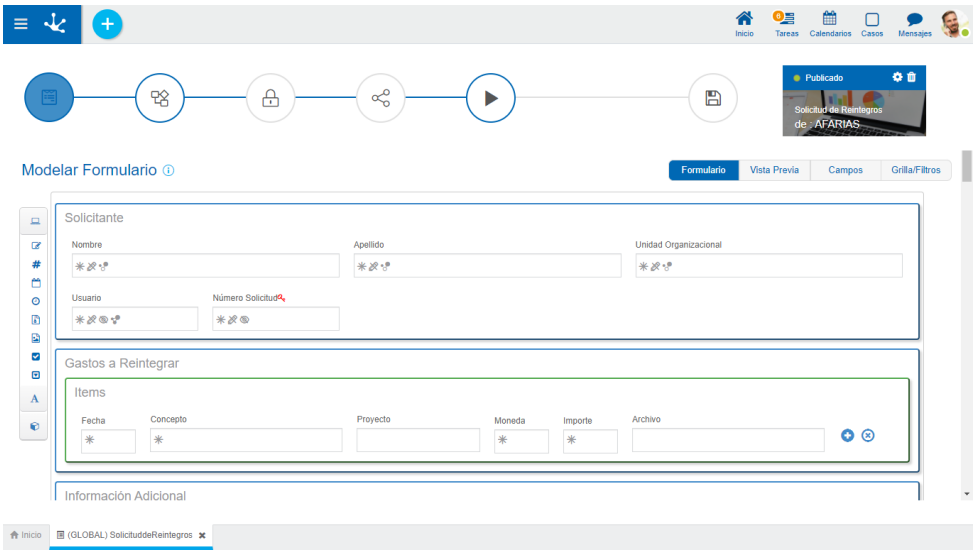
Permite eliminar el campo únicamente en formularios sin datos y requiere confirmación por parte del usuario.

La edición de atributos abre una nueva ventana que permite editar las propiedades del campo.

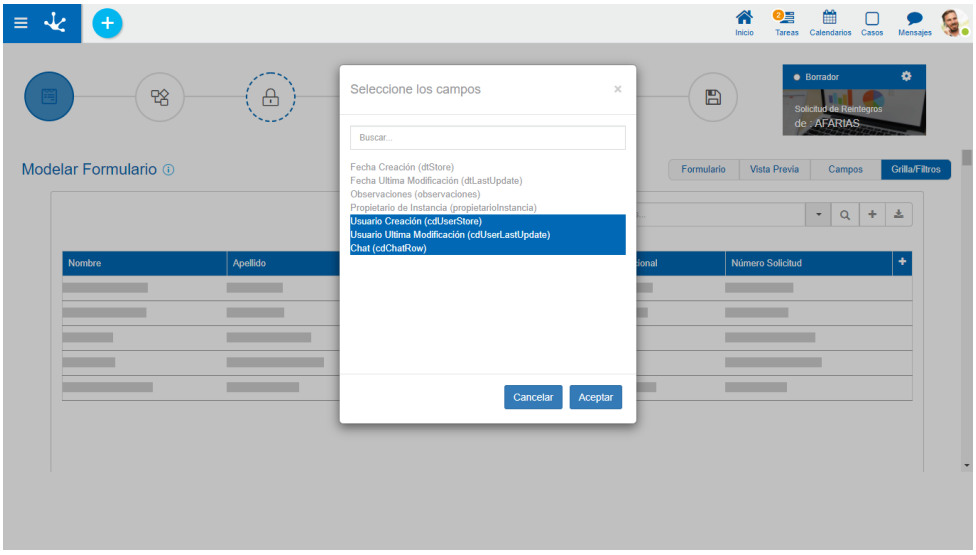
3.5.3.1.3.4. Grilla y Filtros

Grilla

Permite modelar las columnas que se visualizan en los listados de instancias. Las columnas predeterminadas de la grilla corresponden a aquellos campos que se han configurado con la propiedad Incluido en Grilla marcada. Pueden agregarse nuevos campos presionando el icono y su orden es modelable arrastrando las columnas para intercambiar sus posiciones.



Los formularios con procesos asociados pueden utilizar información de la ejecución de los mismos al modelar la grilla de resultados.



Propiedades Relacionadas a la Ejecución del Proceso

Actividad del caso

Es la actividad del proceso en la que se encuentra el caso.

Responsable del Caso

Hace referencia al usuario, rol u oficina que tiene pendiente el caso en su lista de tareas.

Estado del Caso

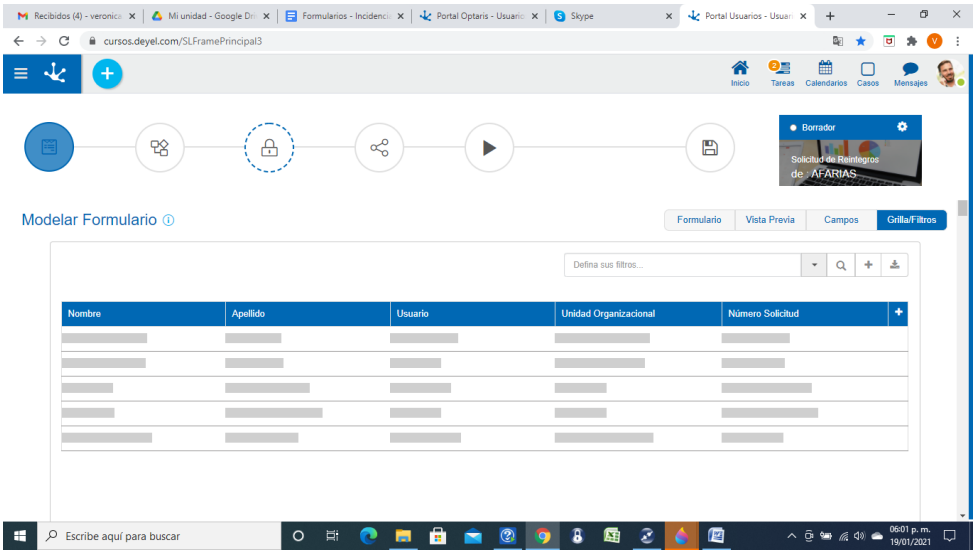
Los posibles estados se visualizan con colores.

- Activo: Verde
- Finalizado: Azul
- Cancelado: Rojo
- Suspendido: Gris

Filtros

La definición de filtros permite enriquecer las variantes de búsqueda que se realizan sobre las instancias de formularios. Cada filtro corresponde a un campo del formulario e incorpora un criterio más por el cual un usuario puede buscar instancias.

Presionando el ícono se accede a la lista de campos disponibles para incluír en el filtro.



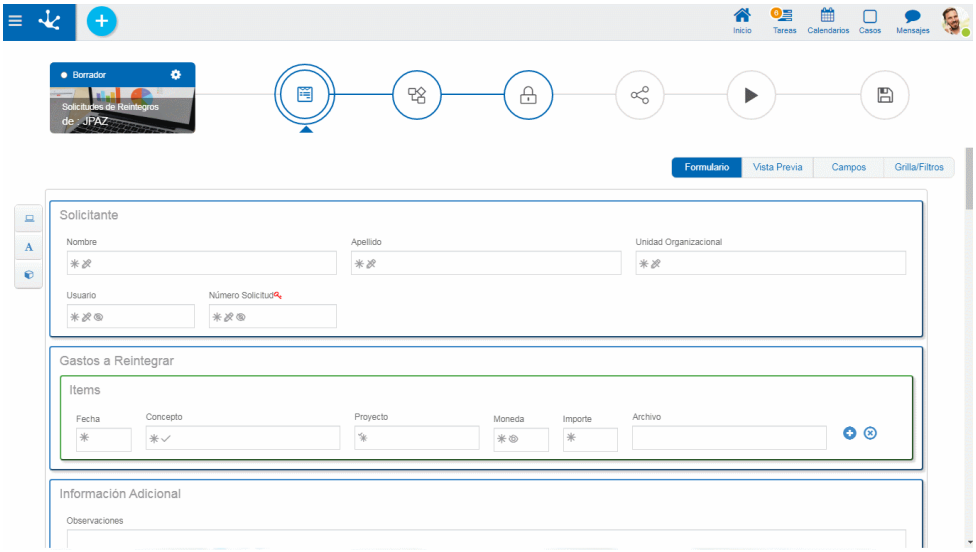
3.5.3.1.4. Barra de Herramientas Lateral

Es la barra de herramientas que contiene los íconos correspondientes a los diferentes tipos de elementos que pueden definirse en un formulario.

Para su uso, es necesario hacer clic sobre la sección correspondiente de la barra y seleccionar el tipo de elemento a agregar, arrastrándolo hacia el lugar del formulario que se desee.

Separadores

Son cuadros de color celeste que se visualizan a la derecha e izquierda, o arriba y abajo de los elementos definidos en el área de modelado, al arrastrar con el mouse desde la barra de la izquierda un campo, elemento gráfico o contenedor. Indican las posiciones dentro del área de modelado donde se puede soltar un campo, elemento gráfico o contenedor elegido para incluir en el formulario.



Elementos de la Barra

Creación de Campos

Al presionar este icono se despliega un subconjunto de elementos que permiten modelar campos representados por controles dentro del formulario. Cada campo puede tener asociado un tipo de dato, según el control que lo represente.

Texto

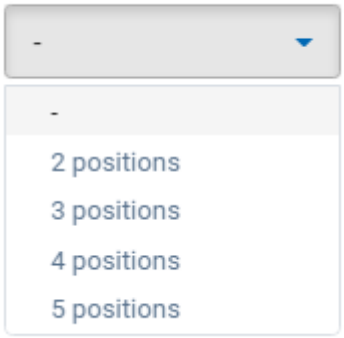
- Alfanumérico (Tipo predeterminado)
  Los valores se guardan respetando mayúsculas y minúsculas ingresadas.
- Alfanumérico Mayúscula
  Los valores se guardan en mayúscula.
- Alfanumérico Extenso
  Funciona como un alfanumérico predeterminado, con la salvedad que permite almacenar textos extensos (usualmente hasta 4GB).
- Texto Enriquecido
  Tiene las características del tipo alfanumérico. Presenta un editor extendido que permite formatear el texto aplicando distintos estilos, colores, tamaños, etc. Ver detalle del uso del control.

Número

- Entero
  Puede contener valores enteros entre -2147483648 y 2147483647.
- Decimal

Si el tipo de dato es decimal, se habilita la selección de cantidad de posiciones decimales (2 a 5).

Amount of Decimals



Fecha

Date

- Fecha
  Permite visualizar una fecha o ingresarla a partir de un selector. Ver detalle deluso del control.
- Fecha y Hora
  Permite visualizar una fecha y una hora o ingresarlas a partir de selectores. Ver detalle deluso del control.

Hora

Time

- Hora
  Permite visualizar una hora o ingresarla a partir de un selector. Ver detalle del uso del control.

Archivo

File

- Archivo en Base de Datos
  Permite utilizar como adjunto del formulario archivos del usuario. Ver detalle del uso del control.

Imagen

Image

- Imagen en Base de Datos
  Permite utilizar como adjunto del formulario, archivos del usuario de tipo imagen. Las ex-

tensiones permitidas son jpg, png y gif.
Ver detalle del uso del control.

Google Docs

Archivo Cloud

Permite modelar un campo que sirve para adjuntar archivos que serán alojados en Google Docs. Los archivos adjuntados se suben a la web relacionados a la cuenta Google configurada para la aplicación.

El formulario sirve como punto de acceso a los archivos en Docs, pero también puede ser accedido desde la web por todo usuario que tenga acceso a la cuenta google asociada.

CheckBox

CheckBox

Permite modelar campos representados por un control checkbox con valores del campo de tipo lógico, donde los valores posibles son “SI/NO”. Ver detalle del uso del control.

Lista de Valores

Value List

Example List

Permite modelar campos representados por un control de tipo lista de valores, cuyos valores posibles están predeterminados en base a una lista preexistente o bien se definen en el momento que se modela el campo, en la pestaña "Relación" de las propiedades del mismo. Ver detalle del uso del control.

Creación de Elementos Gráficos.

Al presionar este icono se despliega un subconjunto de elementos gráficos.

Título

Se utiliza para incluir un texto con un formato destacado.

My Title

Etiqueta

Se utiliza para incluir un texto libre y poder ubicarlo en cualquier sección.

Label

Separador de Línea

Permite dividir secciones del formulario.

Espacio en Blanco

Se utiliza para incluir líneas en blanco.

Botón Múltiples Ocurrencias
Permite agregar y eliminar iterativos. Puede agregarse en cualquier sector del formulario, aunque solo tiene sentido que se encuentre dentro de contenedores iterativos.

Creación de Contenedores

Los contenedores son elementos gráficos que permiten agrupar otros elementos bajo ciertos criterios.

Para más detalles sobre su configuración y atributos, ver[Propiedades de Contenedores](#).
Al presionar sobre este icono se despliega un subconjunto de elementos contenedores.

Filas
Son el agrupamiento general en el área de modelado gráfico. Dentro de una fila se puede poner una cantidad ilimitada de elementos, aunque se recomienda que no se exceda el ancho de la ventana ya que la fila se expande aumentando el ancho con respecto a las demás.

Las filas pueden ser desplazadas hacia arriba o hacia abajo moviendo todo su contenido. Para mover una fila se debe mantener presionado el botón izquierdo del mouse con el cursor, haciendo foco sobre la fila que se desea mover y arrastrarla hacia arriba o abajo de otra. Las filas solo pueden ser ordenadas dentro del contenedor donde se encuentran, ya sea el área general o un contenedor creado.

Para eliminar una fila se debe posicionar el mouse sobre la misma, donde se visualiza a la derecha la opción de eliminar .

Como mecanismo de prevención de errores, se muestra un aviso de confirmación y si se acepta se elimina la fila y todo su contenido, incluyendo campos.

Si el formulario ya contiene datos, no se permite eliminar filas que contengan campos.

Contenedor Simple
Se visualiza un rectángulo a modo de caja, el cual puede contener cualquier elemento del formulario (campos, elementos gráficos e inclusive otros contenedores).

Para su distinción de los otros contenedores, su contorno es de color naranja.

Contenedor Gráfico
La diferencia con un contenedor simple es que todo lo que se encuentre dentro del contenedor gráfico se visualiza en modo de ejecución como dentro de un acordeón.
Se puede ver en la vista previa dicho contenedor transformado en acordeón.

Para su distinción de los otros contenedores, su contorno es de color azul.

Contenedor de Múltiples Ocurrencias
Puede contener cualquier elemento del formulario (campos, elementos gráficos, e inclusive otros contenedores).

A diferencia de los demás contenedores, todos los campos que se incluyan dentro se definen como un conjunto de campos de múltiples ocurrencias. Un contenedor de múltiples ocurrencias sirve para agrupar secciones del formulario, al que se le agregan operaciones (ocultar, relacionar reglas, etc).

Su principal funcionalidad es que todo el contenido es definido como múltiple, es decir, se pueden ir generando copias y eliminarlas desde los íconos a la derecha del elemento. Este contenedor se crea por defecto con estos botones, pero se puede eliminar en caso de que no se requieran.

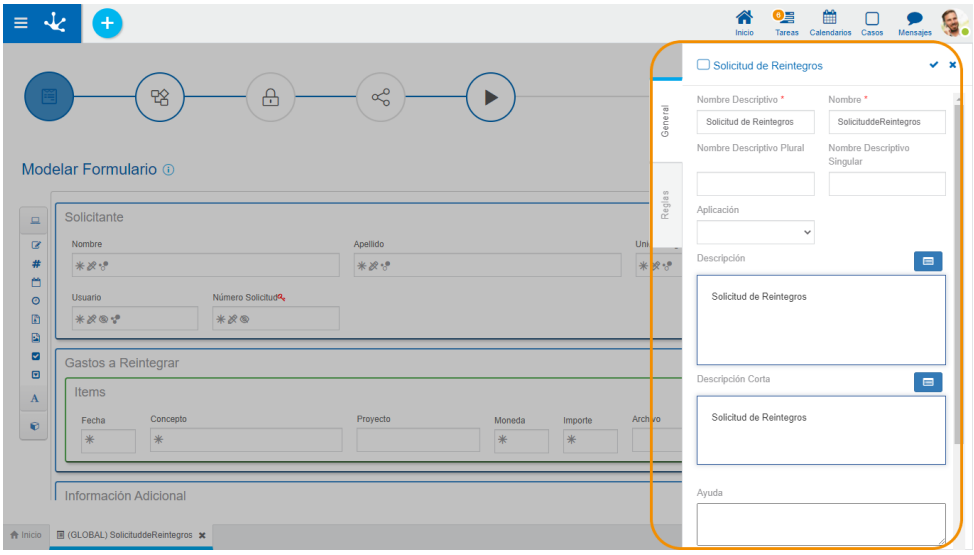
A diferencia de los otros tipos de contenedores, éste no puede contener otros contenedores iterativos dentro, ni tampoco contenedores básicos con la propiedad Grupo de Campos marcada. Ver Propiedades de Contenedores.

Para diferenciarlo de otros contenedores, su contorno es de color verde.

3.5.3.2. Propiedades de Formularios

Las propiedades de los formularios pueden ingresarse tanto en el momento de su creación, como en la modificación de uno existente.

El ingreso al panel de propiedades del formulario se realiza utilizando el ícono que se encuentra en la imagen que representa al formulario en el ángulo superior izquierdo.

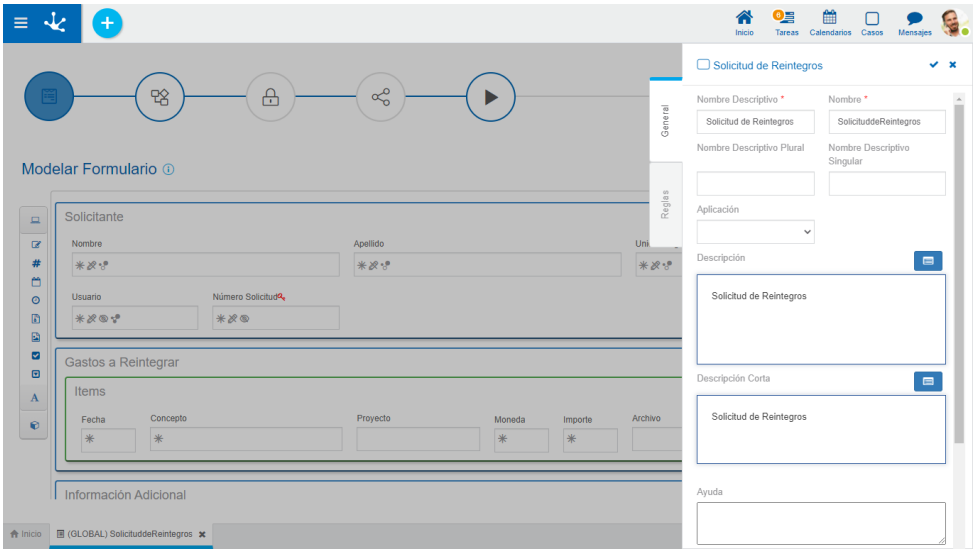


Pestañas

- General
- Reglas

3.5.3.2.1. General

El panel de propiedades se visualiza en la parte derecha del asistente de formularios, donde la primera pestaña corresponde a información general.



> *Un asterisco "*" en la etiqueta indica que la propiedad es obligatoria.*

Propiedades

Nombre Descriptivo
Es el nombre que utiliza el usuario para referenciar el formulario, visualizarlo en la galería de formularios y en el árbol de objetos. Soporta multi-idioma.

Nombre
Se usa internamente para referenciar el formulario dentro de reglas o como parámetro. Soporta multi-idioma.

Nombre Descriptivo Plural
El texto ingresado en esta propiedad es utilizado como título en la grilla de resultados del formulario, mientras que si no se completa, se utiliza la propiedad Nombre Descriptivo.

Nombre Descriptivo Singular
El texto ingresado en esta propiedad es utilizado en la lista de entidades relacionadas, mientras que si no se completa, se utiliza la propiedad Nombre Descriptivo.

Descripción
Texto que define al formulario describiendo su funcionalidad y opcionalmente su contenido. Soporta multi-idioma, y puede incluir valores de variables del mismo formulario que se incorporan mediante el uso del asistente de variables.

Descripción Corta
Versión resumida de la propiedad Descripción. Soporta multi-idioma y puede incluir valores de variables del mismo formulario, que se incorporan mediante el uso del asistente de variables. Puede ser utilizada a modo descriptivo al relacionar entidades.

Ayuda
Texto informativo que se visualiza en el portal, al presionar el ícono cuando se utiliza el formulario y el mismo no tiene proceso asociado. Soporta multi-idioma.

Imagen
Imagen que se visualiza en las grillas de Formularios y Tareas, Plantillas y en la barra superior al crear formularios. Se recomienda que esté relacionada con la funcionalidad o información que el formulario representa.
Informa sobre la utilidad de la imagen.
Permite elegir una imagen desde una ventana de selección de archivos.
Permite eliminar la imagen actual asociada al formulario.

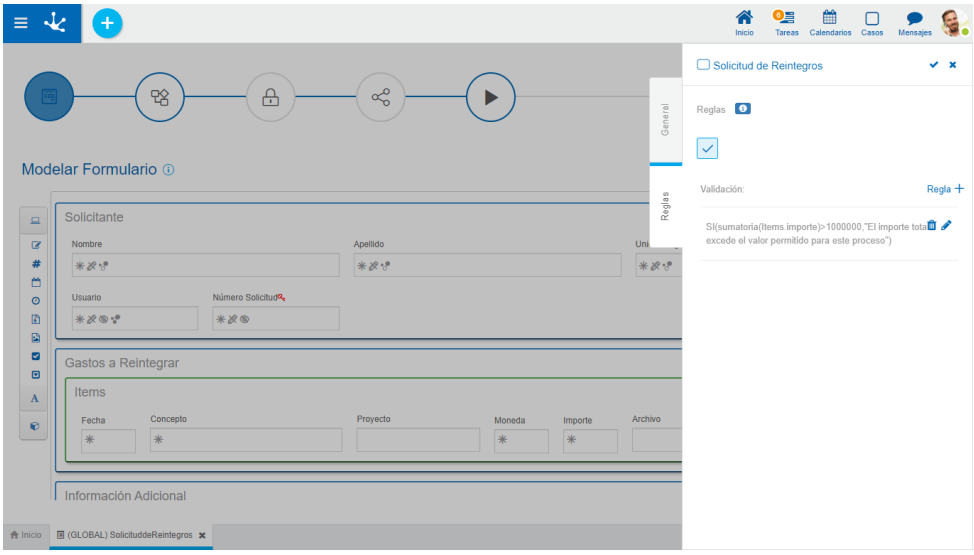
Actions

The icon is used to confirm the modifications made in the properties panel.

The icon is used to close the properties panel, if it was not previously saved, changes are discarded.

3.5.3.2.2. Reglas

La última pestaña del panel lateral corresponde a las reglas de validación asociadas al formulario ágil.



Rules

It is possible to define validation rules associated with the form.

Validation

Rule + Opens an edit area where you can define the condition that determines if the form is correct or not. It is possible to define more than one rule. If rules are defined, the icon is displayed with light blue borders.

Shows syntax examples for writing the rules.

Operations

- Saves the new or modified rule
- Cancels the operation
- Edits the existing rule
- Deletes the rule

Validations with Advanced Rules

If a validation with an advanced rule needs to be incorporated, an existing rule must be selected from the drop-down list and press to add the rule to the grid. It can be selected using the options "Create", "Modify" and "Delete" when the validation is executed, either when creating a form instance, modifying it or deleting it. Through this icon you can unlink the rule from the grid.

3.5.3.3. Propiedades de Elementos

El ícono se hace visible al pasar el cursor sobre cada elemento del Area de Modelado Gráfico del formulario, permitiendo ingresar al panel lateral de propiedades que define a cada uno de los diferentes tipos de los elementos.

Tipos

- Gráficos
- Campos
- Contenedores

Panel Lateral

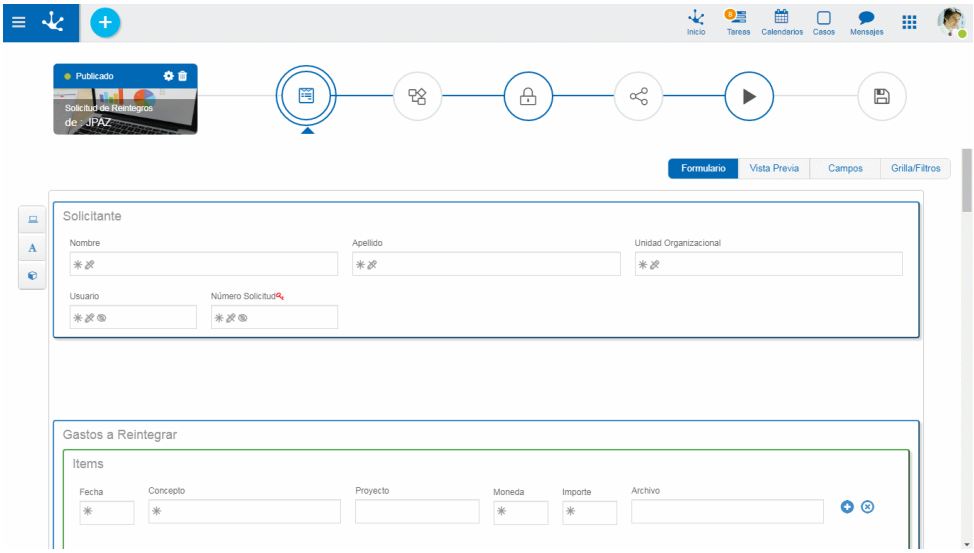
En el panel lateral se pueden visualizar y modificar las propiedades correspondientes a cada tipo de elemento. En la parte superior derecha del mismo se pueden seleccionar íconos para realizar las siguientes acciones:

- Aceptar: Confirma las modificaciones realizadas a las propiedades del elemento y cierra el panel.
- Cerrar: Cierra el panel de propiedades sin guardar las modificaciones que se hayan realizado.

3.5.3.3.1. Propiedades de Elementos Gráficos

En esta sección se explican las propiedades de los distintos tipos de elementos gráficos presentes en el modelador de formularios.
De todos los elementos de este tipo, solo es posible editar los elementos "Título" y "Etiqueta", ya que el resto no tiene disponible el ícono .

Presionando el ícono sobre un elemento gráfico se abre el panel de propiedades.



> *Un asterisco "*" en la etiqueta indica que la propiedad es obligatoria.*

Propiedades

Texto
Texto del título o la etiqueta.

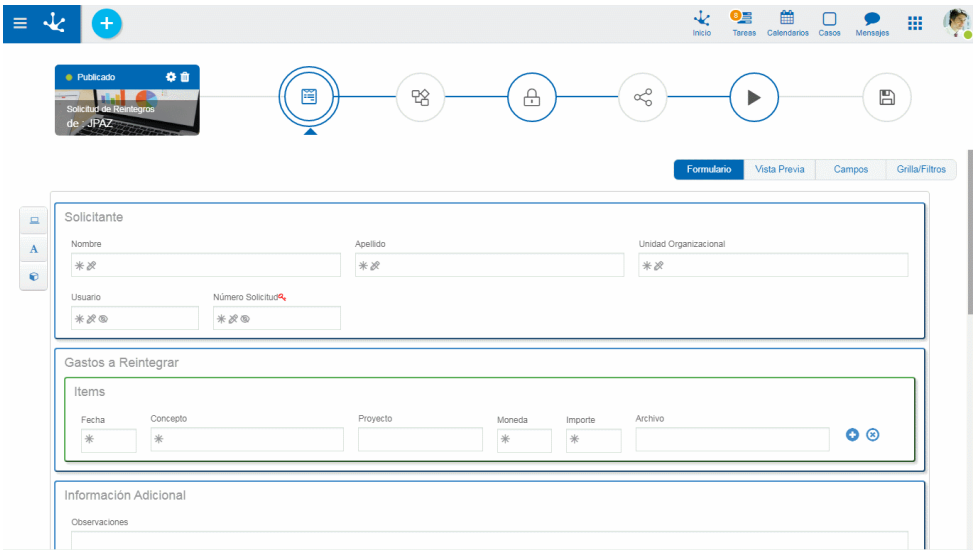
Ancho

Se utiliza para modificar el ancho que utiliza el campo en la fila, pudiendo seleccionar valores entre 1 y 12, siendo 12 el ancho total de la fila.

3.5.3.3.2. Propiedades de Campos

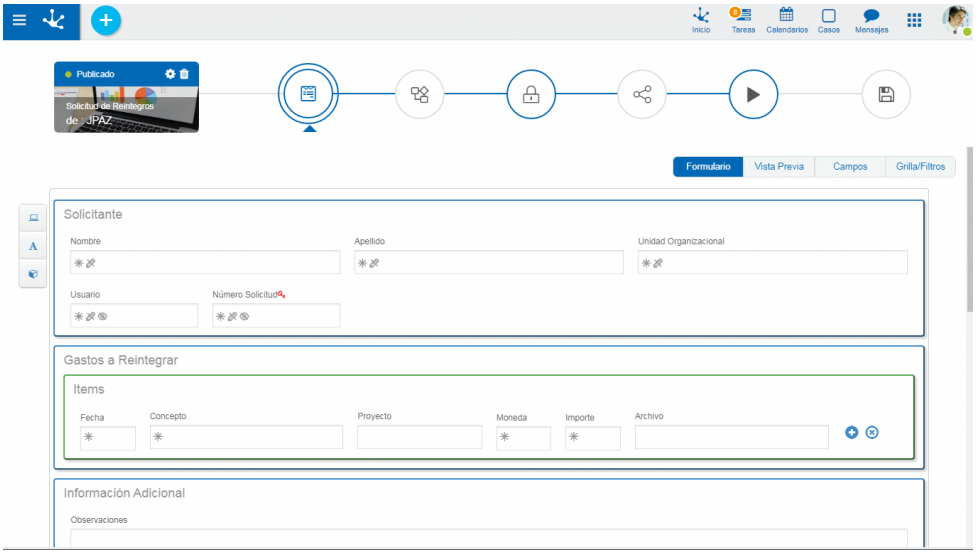
Presionando el ícono sobre el campo se abre el panel vertical de la derecha, que contiene las siguientes pestañas:

- General
- Avanzado
- Relación



3.5.3.3.2.1. General

El panel de propiedades se visualiza a la derecha del modelador de formularios, donde la primera pestaña corresponde a información general.



Un asterisco "" en la etiqueta indica que la propiedad es obligatoria.*

Propiedades

Etiqueta
Permite ingresar el texto que se visualiza sobre el campo. Actúa conjuntamente con el prefijo para referenciar al campo en los mensajes de validaciones y admite espacios en blanco.

Prefijo
Este indicador de prefijo se utiliza para la conformación de mensajes de error durante la carga de información. El mismo es de ingreso obligatorio. Permite seleccionar los valores: "El", "La", "Los", "Las".

Nombre
Es el nombre que se asigna para referenciar a un campo en el modelado, permitiendo identificar unívocamente el campo dentro del formulario. Se utiliza en los asistentes de reglas para referirse al campo dentro de las condiciones. Se genera automáticamente a partir de la propiedad Etiqueta, puede ser modificado por el usuario y no admite espacios ni caracteres especiales.

Identificador
Es el nombre que se asigna para referenciar a un campo en el código de programación, se utiliza para referirse al campo dentro del código Java en la pestaña "Código de Ejecución" de las reglas avanzadas y en el código JavaScript en la pestaña "Edición Avanzada" del modelador de formularios. Permite identificar unívocamente el campo dentro del formulario modelado. Puede ser modificado por el usuario, mientras no se hayan cargado datos al formulario y no admite espacios ni caracteres especiales.

Descripción
Texto que define al campo y opcionalmente su contenido. Soporta multi-idioma.

Reglas
Se pueden definir reglas de comportamiento, validación y cálculo, asociadas a un campo.

Obligatorio

Indica si el campo debe informarse en forma obligatoria al crear una nueva instancia del formu-

lario o al modificar una ya existente.

Obligatorio No obligatorio (predeterminado)

Rule + Abre un área de edición donde se puede definir una regla para determinar la condición de obligatoriedad, mediante el uso de un asistente<<al asisatente>> (ctrl + espacio). Si se define una regla, el ícono se visualiza con bordes color celeste.

Visible

Indica si el campo es visible. Si no se marca esta propiedad, el campo no se visualiza en las instancias del formulario.

Visible (predeterminado) No visible

Rule + Abre un área de edición donde se puede definir una regla para determinar la condición de visibilidad, mediante el uso de un asistente (ctrl + espacio). Si se define una regla, el ícono se visualiza con bordes color celeste.

Editable

Indica si el campo es editable. Si no se marca esta propiedad, el usuario no puede ingresar o modificar valores en el campo.

Editable (predeterminado) No editable

Rule + Abre un área de edición donde se puede definir una regla para determinar la condición de editabilidad, mediante el uso de un asistente (ctrl + espacio). Si se define una regla, el ícono se visualiza con bordes color celeste.

Válido

Rule + Abre un área de edición donde se puede definir la condición que determina si el valor del campo es correcto. Es posible definir más de una regla. Si se definen reglas, el ícono se visualiza con bordes color celeste.

Cálculo

Rule + Abre un área de edición donde se puede definir la expresión que debe ejecutarse para calcular el valor del campo. Si se define una regla, el ícono se visualiza con bordes color celeste.

Visualización en el Modelador

Obligatorio / Obligatorio con Regla

No visible / Visible con Regla

No editable / Editable con Regla

Valida con Regla

Calcula con Regla

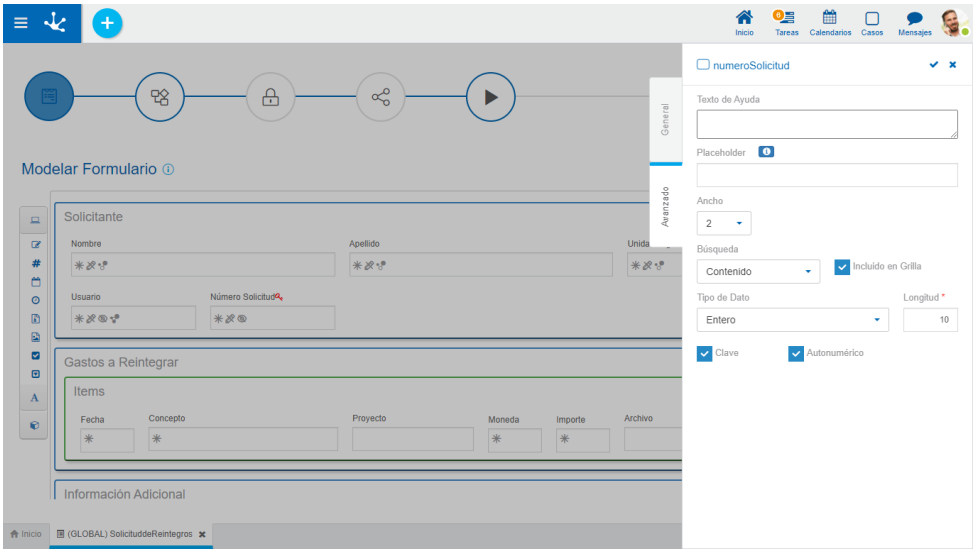
Muestra ejemplos de sintaxis para escribir las reglas.

Operaciones

- Guarda la regla nueva o modificada
- Cancela la operación
- Edita la regla existente
- Elimina la regla

3.5.3.3.2.2. Avanzado

La segunda pestaña del panel lateral corresponde a propiedades adicionales para definir el comportamiento y funcionalidad de un campo.



> *Un asterisco "*" en la etiqueta indica que la propiedad es obligatoria.*

Propiedades

Texto de ayuda

Su función es orientar al usuario sobre el contenido a cargar en el campo. El texto ingresado como ayuda se muestra cuando el usuario posiciona el cursor sobre el campo.

Placeholder

Su función es orientar al usuario sobre el contenido a cargar en el campo. A diferencia de la propiedad Texto de Ayuda, el texto ingresado como placeholder se visualiza dentro del campo.

Ancho

Se utiliza para modificar el ancho que utiliza el campo en la fila, pudiendo seleccionar valores entre 1 y 12, siendo 12 el ancho total de la fila.

Multilínea
Esta propiedad determina si el campo se visualiza en modo de caja de texto al utilizar el formulario, permitiendo editar el texto en un área de mayor tamaño. Esta propiedad solamente está visible si el campo es de tipo texto.

Valor Predeterminado
Permite asignar el valor que el campo tiene en forma predeterminada al momento del primer ingreso de datos, agilizando la carga de valores. Los valores predeterminados se definen a partir de las funciones disponibles para el tipo de campo. Cada función retorna un valor al momento de utilizar el formulario y pueden combinarse resultados de más de una función con texto.

Búsqueda
Permite utilizar el campo como filtro en las búsquedas de instancias del formulario.

Incluido en Grilla
Determina si el campo se visualiza en la grilla de resultados donde se muestran las instancias del formulario.

Tipo de Dato
Determina el formato del dato que puede ingresarse en el campo. El conjunto de datos soportado varía, dependiendo del tipo de campo que se haya creado a partir de los elementos de la barra de herramientas lateral izquierda.

Longitud
Determina la longitud máxima del valor dependiendo del tipo de dato. Este atributo sólo está visible si el campo es de tipo texto o numérico.

Clave
Cada formulario requiere que al menos un campo sea definido como clave, la misma permite identificar unívocamente una instancia del formulario, es decir dicho valor no puede repetirse entre instancias del mismo formulario.

Clave Alternativa
Determina que el valor del campo es único entre instancias del mismo formulario pero no es utilizado como clave.

Autonumérico
Determina que el valor del campo es generado automáticamente al momento de crear la instancia del formulario. Con esta propiedad marcada, un número único es asignado a la instancia del formulario en forma automática. El campo no puede definirse con la propiedad Editable marcada,

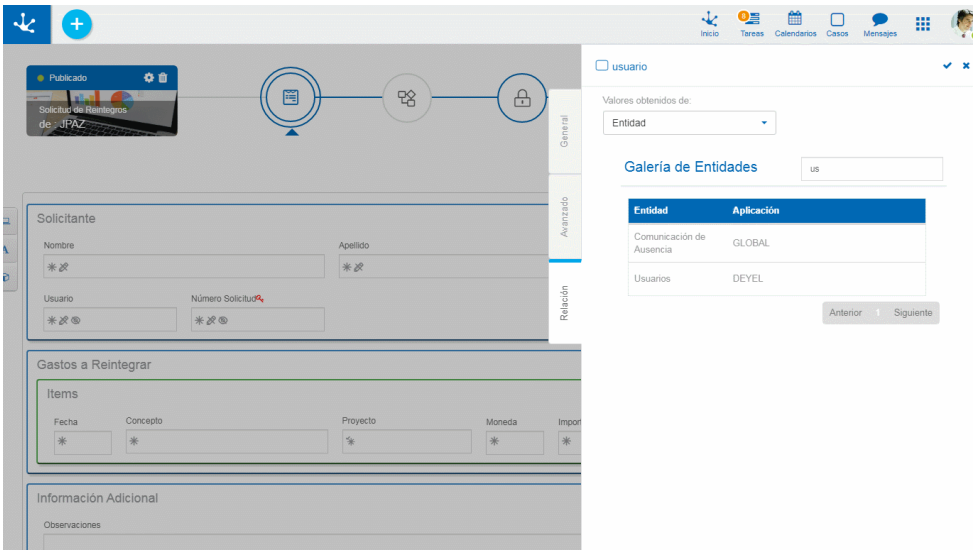
3.5.3.3.2.3. Relación
Una relación permite obtener valores de diferentes orígenes para el campo que se está modelando. Se describen a continuación los diferentes tipos de orígenes.

- Entidades

- Lista de valores

Entidades

A través del modelador de formularios se modelan las entidades del negocio. Es posible además modelar relaciones entre entidades para representar lo interacción de las mismas.
Para definir una relación con una entidad, primero se debe seleccionar la entidad con la que se se quiere establecer una relación, para luego completar un conjunto de propiedades.



> *Un asterisco "*" en la etiqueta indica que la propiedad es obligatoria.*

Propiedades

Nombre Descriptivo de Relación
Nombre representativo de la relación, orientado al usuario con perfil de ejecución.

Nombre de Relación
Nombre de la relación entre las dos entidades, es un atributo orientado al modelado. No admite espacios y debe ser único por Formulario modelado.

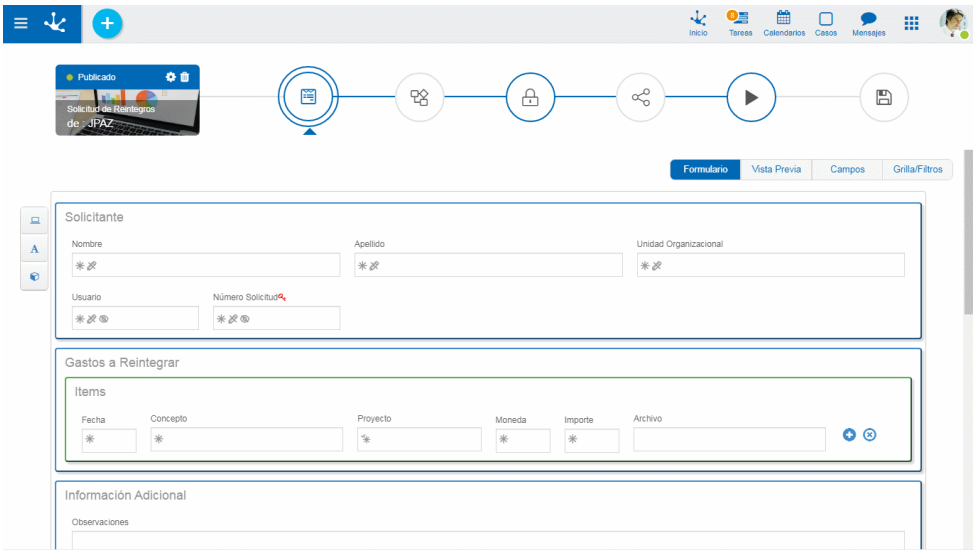
Atributo Clave
Representa el atributo de valor único de la entidad con la que se quiere establecer una relación. La entidad que se está relacionando puede tener más de un atributo cuyo valor sea único, uno es su clave y en caso de poseerla su clave alternativa.

Permisos
Los permisos de la relación definen que funcionalidades se habilitan en el control de la relación.

Consulta
Al marcarse esta propiedad se le permite al usuario visualizar la instancia de entidad con la que se establece relación en el uso del formulario.

Búsqueda
Al marcarse esta propiedad se le permite al usuario acceder a la grilla de resultados para buscar la instancia de entidad con la que se quiere establecer relación en el uso del formulario.

Atributo Relacionado
Esta marca indica que el valor del campo se recupera a partir del valor de un atributo de otra entidad relacionada al crear o actualizar una instancia de formulario. Se debe seleccionar una de las relaciones definidas en el formulario y luego indicar a qué atributo de la entidad relacionada se vincula.



> *Un asterisco "*" en la etiqueta indica que la propiedad es obligatoria.*

Al marcar la propiedad Atributo relacionado, se habilitan las siguientes propiedades:

Nombre de la Relación
Se visualiza la lista de relaciones definidas en el formulario, permitiendo seleccionar una de ellas.

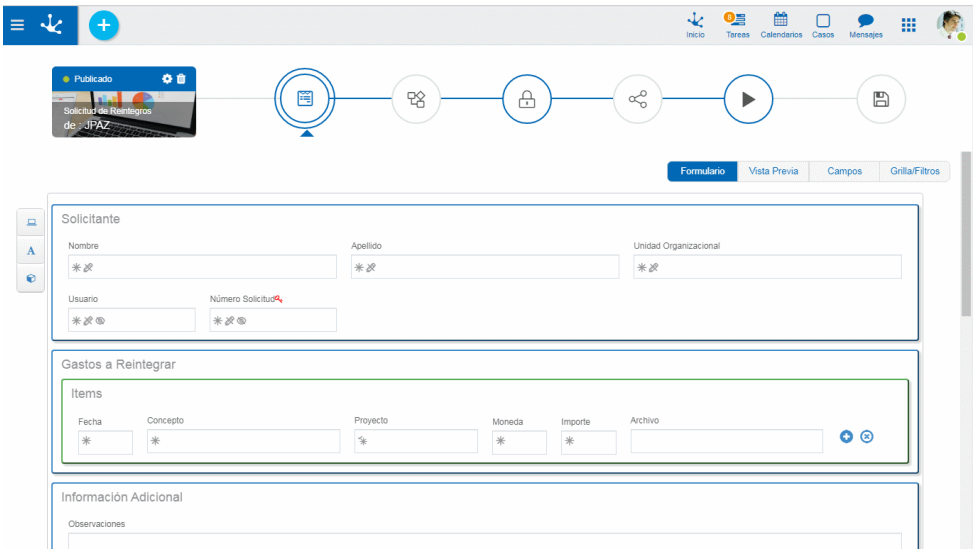
Atributo de la Entidad
Se visualiza la lista de atributos de la entidad seleccionada compatible con el tipo del campo.

Referencia
Esta propiedad sin marcar indica que una vez cargado el valor recuperado, no vuelve a modificarse automáticamente aunque el valor del atributo recuperado se modifique. Al estar marcada, el valor del campo siempre está sujeto al valor actual del atributo recuperado, es decir se visualiza automáticamente el valor actualizado.

Lista de Valores

Allows to define a set of values grouped under some criteria so that the application user knows the possible values a field can take.

New lists can be created from the modeler, then edit and relate them to form fields.



> *Un asterisco "*" en la etiqueta indica que la propiedad es obligatoria.*

To create a new list of values, click on the icon .
A panel opens to enter the name of the list, the application to which it belongs is selected, if alphabetical sorting is indicated, descending order can be selected. Pressing the "Accept" button creates the list of values, subsequently allowing the entry of new values.

The list of values can be used in any other field of the same or another form.

Properties

Select a list
Allows to select a particular list from the set of already existing lists, being able to filter such set by entering text into the search field that is above the list.

Allows empty
If this property is checked, the empty value is included as the first option in the list. Otherwise the first value in the list is displayed in the field.

Autocomplete
If this property is checked, predictive text functionality is presented to users. Based on the characters that the user types, a subset of values is proposed, which coincide with all or part of the text entered. This facility is called autocomplete.

Other Elements

Open in modeler
Allows using the list of values modeler for the definition, instead of the field properties panel.

Allows to filter values from the list based on the characters entered. If a list is very long it helps users to easily visualize the desired values.

Enables an option that allows to add the internal code to the list values.

Operations on Values

- Allows adding each entered value to the list of values.
- Double click: Allows to modify a value in the list.
- Move: Allows to change the position of a value within the list by dragging the value with the mouse.

Hovering the cursor over each of the values entered, a set of icons is displayed and this allows to perform different operations.

- Allows to delete a value from the list of values. Once deleted, it is displayed in gray and crossed out.
- Allows to restore a previously deleted value.

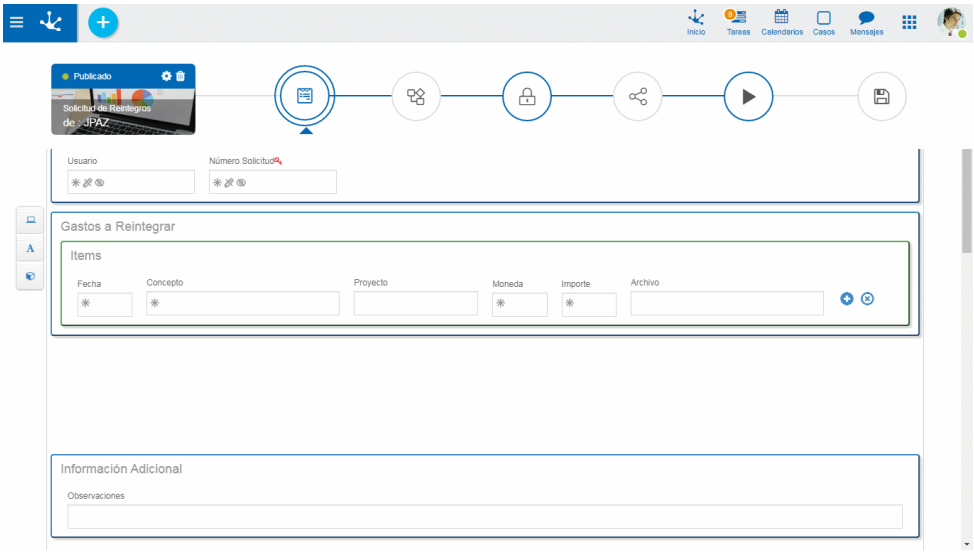
Display the Selected Line

- Hides the icons that are displayed.
- Shows hidden icons.

3.5.3.3.3. Propiedades de Contenedores

Las propiedades de los contenedores varían según el tipo.

- Contenedor Simple
- Contenedor Gráfico
- Contenedor Múltiples Ocurrencias



Un asterisco "" en la etiqueta indica que la propiedad es obligatoria.*

Propiedades

Nombre Descriptivo

Permite identificar al contenedor con un nombre orientado al usuario final. Se admiten espacios y caracteres especiales.

Nombre

Permite identificar al contenedor con un nombre orientado al usuario modelador. No admite espacios ni caracteres especiales.

Ancho

Se utiliza para modificar el ancho del contenedor, pudiendo seleccionar valores entre 1 y 12, siendo 12 el ancho total.

Desplegado

Indica si el contenedor se visualiza desplegado, mostrando su contenido, o cerrado al utilizar la instancia del formulario. Se utiliza solamente para contenedores de tipo gráfico.

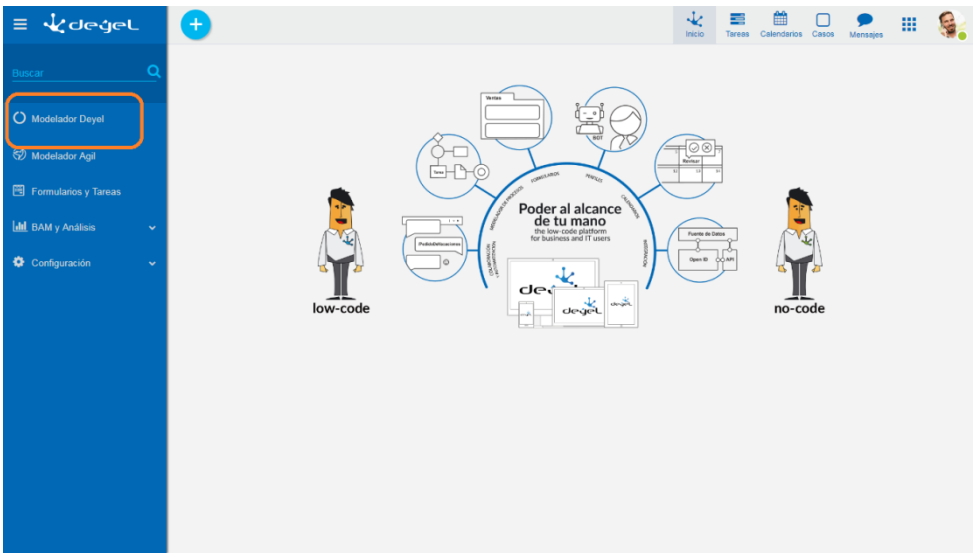
Visible

Esta propiedad indica si el contenedor y su contenido son visibles al utilizar la instancia del formulario. Permite la definición de reglas de visibilidad asociadas al contenedor al igual que las definidas para los campos.

3.6. Modeler

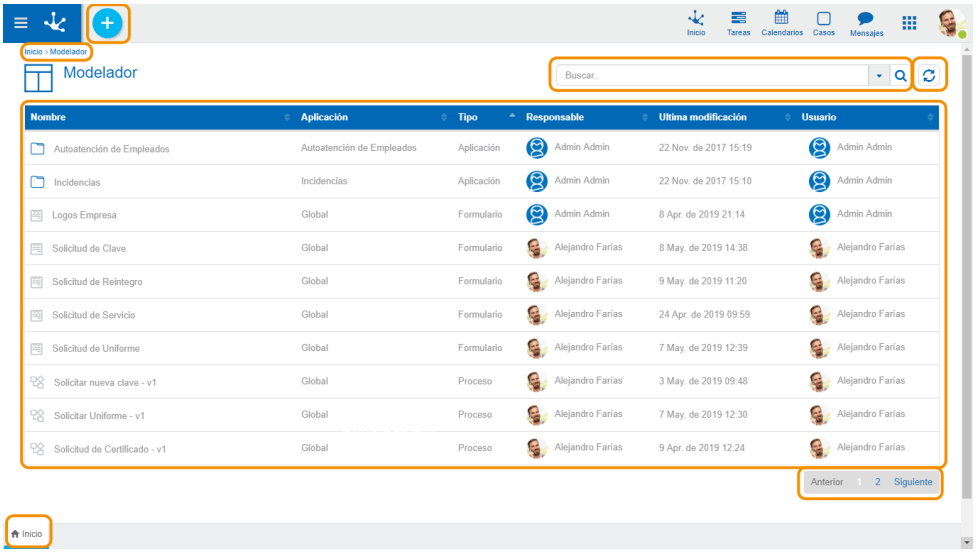
The menu option "Deyel Modeler" allows IT users to design and build low-code applications. Using a single interface, different **Deyel**objects can be created and maintained. For each of them, specific modeling tools are used in an intuitive and easy-to-use environment.

For each type of object, its modeler allows to use a set of development functions, such as graphic modeling areas, toolbars, properties panels, and design wizards, among others.



Sections

- Portal Top Toolbar
  - Context menu with the modeler's own functionality.
- Breadcrumb
- Objects Grid
- Search Bar
- Update Grid
- Ordering
- Pagination
- Start



3.6.1. Characteristics

Phase 2: Forms Modeling > Deyel Modeler

Modeler Context Menu

The context menu adopts the general characteristics of the portal, but adapted to the modeler's own functionalities.

Icons Menu

This menu is enabled by hovering over the icon and allows to create different types of objects and import them.

New Process

New Form

New Rule

New Adapter

New Application

New Dashboard

New Indicator

New Value List

Import

Displayed Menu

This menu is displayed by pressing the icon , unfolding a vertical panel with the following sections:

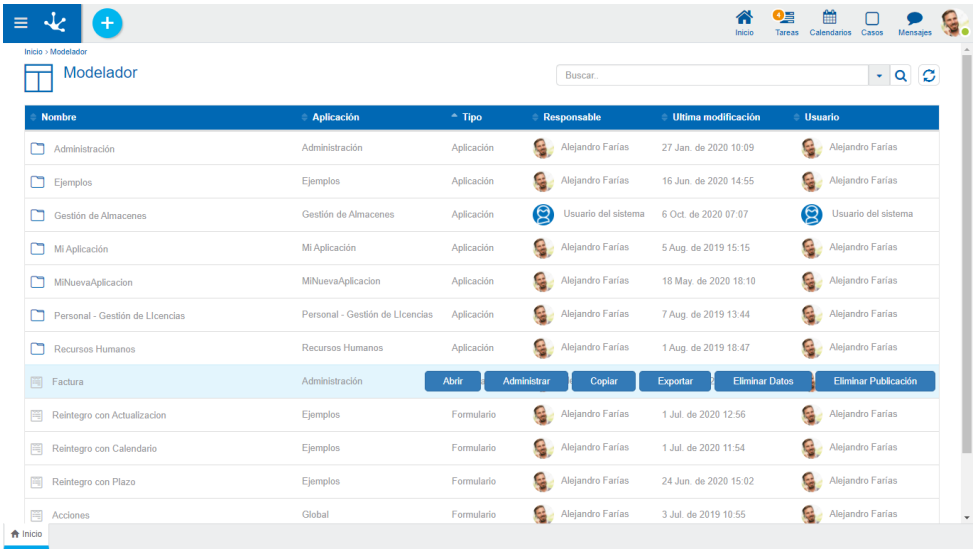
- Options associated with the modeler, in light blue.
  - Process
  - Form

- Rule
- Adapter
- Application
- Dashboard
- Indicator
- Value List
- Report
- Import

- Most used, in gray.
  - Forms and processes most used by the user.

Objects Grid

The objects grid allows to display all objects defined in **Deyel** and perform operations with each one of them.



Columns

Name
Name used in the interface to reference the object, corresponds to the Name property of each object.

Application
Application the object belongs to, corresponds to the Application property of each object.

Type
Identifies the type of object defined in **Deyel**.

Responsible Agent
The user that created the object.

Last Modification
Date of last modification made to the object.

User
Last user that modified the object.

Operations

The operation buttons are enabled when hovering over the row selected in the grid and they depend on the type of object selected.

Adapter

- Open
  Opens the selected adapter properties panel in a new tab, for modification.

- Delete
  Deletes the selected adapter, after confirmation.

Application

- Open
  Opens a reduced view of the objects grid, only with those that belong to the selected application.

- Modify
  Opens the selected application properties panel, for modification.

- Delete
  Deletes the selected application, if it is not associated with other objects of **Deyel**.

- Export
  Opens a panel for the user to select and confirm the application export.

Form

- Open
  Opens the form modeler in a new tab.

- Administrate
  Opens a panel with the results grid corresponding to the selected form.

- Copy
  Copies the selected form into a new modeler tab. Its properties must be modified to save it as a new form.

- Delete Data
  Deletes the existing instances of the selected form, after confirmation.

- Delete Data and Publication
  Removes the existing instances of the selected form and its publication after confirmation, that is, it cannot be used from the portal.

- Export
  Opens a panel for the user to select and confirm the form export.

Agile Form

- Open
  Opens the agile forms modeler in a new tab.

Permissions

- Open
  Opens a panel to show the permission properties.

- Modify
  Opens a panel with the permission properties for modification.

- Delete
  Allows to delete the permission selected if not in use, otherwise the user receives a message.

- Export
  Opens a panel for the user to select and confirm the permission export.

Processes

- Open
  Opens the process modeler in a new tab.

- Tests Plan
  Opens a panel with the grid corresponding to the tests plan defined for the process.

- Delete
  Deletes the selected process, after confirmation.

- Export
  Opens a panel for the user to select and confirm the process export.

Rule

- Open
  Opens the rules modeler in a new tab.

- Delete
  Deletes the selected rule, after confirmation.

- Export
  Opens a panel for the user to select and confirm the rule export.

Roles

- Open
  Opens a panel to show the role properties.

- Modify
  Opens a panel with role properties for modification.

- Delete
  Allows to delete the role selected if not in use, otherwise the user receives a message.

- Export
  Opens a panel for the user to select and confirm the role export.

Dashboard

- Open
  Opens thedashboard modeler in a new tab.

- Delete
  Deletes the selected dashboard.

- Export
  Opens a panel for the user to select and confirm the dashboard export.

Indicator

- Open
  Opens the indicators modeler in a new tab.

- Delete
  Deletes the selected widget, after confirmation.

- Export
  Opens a panel for the user to select and confirm the indicator export.

Value List

- Open
  Opens the value list modeler in a new tab.

- Delete
  Deletes the selected value list, after confirmation.

- Export

Opens a panel for the user to select and confirm the value list export .

Search Bar and Filters

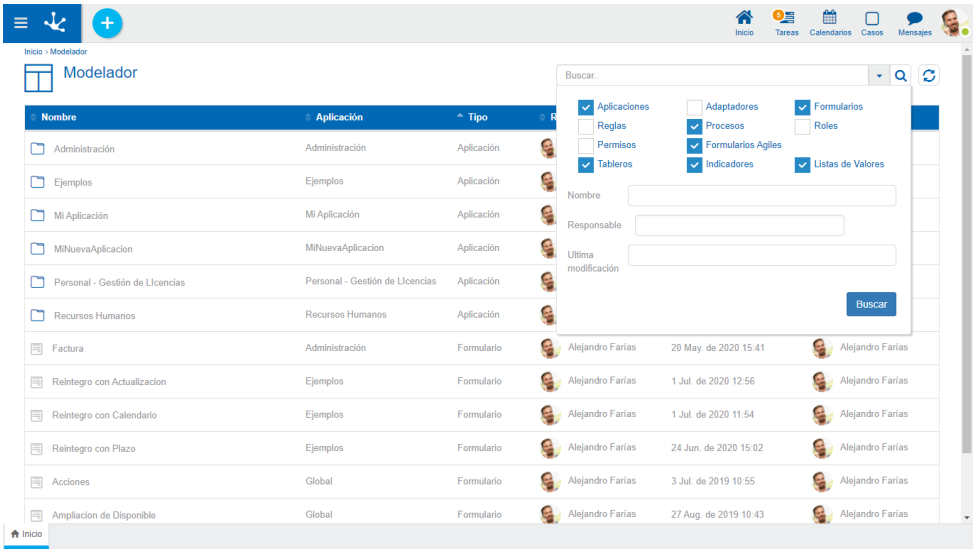
The search bar allows to see a reduced grid of objects, based on the definition of filters.

When entering a text in the bar and pressing the icon or the "Enter" key, all the objects that contain the text entered in the Name column are displayed in the grid.

Filters

The icon allows to display a panel with filters, they can be combined to define the selection criteria of the grid rows.

Filters can be combined by checking object types and values in the grid columns. Once the filter selection was made, press the "Search" button to apply them.



Filters for Object Types

- Applications
- Adapters
- Forms
- Agile Forms
- Rules
- Processes
- Roles
- Permissions
- Dashboard
- Indicator
- Value List
- Reports

Filters for Column values

- Name
- Responsible Agent
- Last Update

Update Grid

The icon allows to update the grid of objects to reflect recent changes.

Sorting

The icon is displayed in each column of the grid and allows to sort rows, ascending or descending, based on that column. Sorting can be done on a single column.

Paging

If the number of rows exceeds the maximum number to display on a page, the buttons at the bottom right of the grid can be used to move to the previous and next page, or to select a specific page.

Start

The tab with the icon corresponds to the modeler's grid. When opening other objects in different tabs, the home tab always allows to return to the modeler's grid.

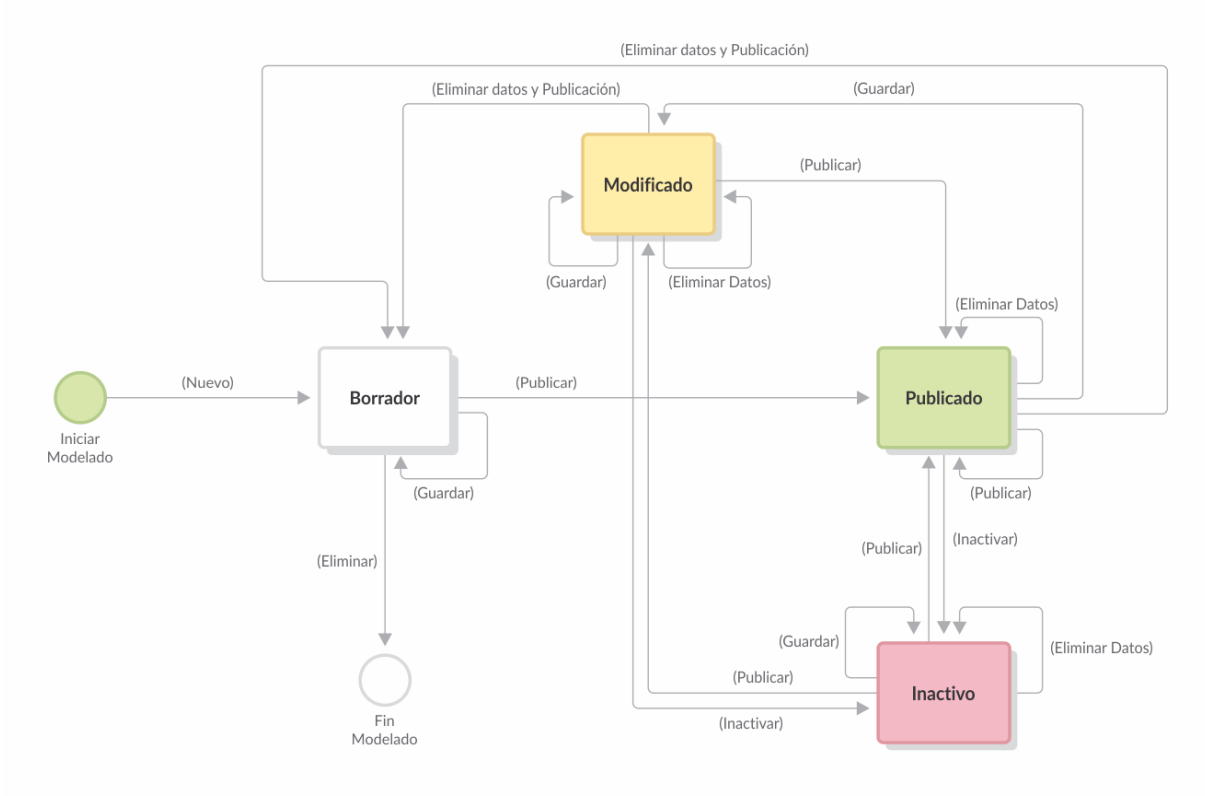
3.6.2. Object States

Sates indicate the stage of development in which **Deyel** objects are and if they are operational.

The following **Deyel** objects defined in applications have states:

- Form
- Agile Form
- Process
- Rule
- Dashboard
- Indicator
- Value List

The possible states and the events that produce transitions among them are described below, ensuring the modelers of predictable behavior for each type of object.



Possible States

Draft
State corresponding to the beginning of the modeling or when it was saved but not published.

Published
State of the object once published, that is, it is available for use.

Modified
If an object is modified when it is in the "Published" state, it becomes "Modified".

Inactive
State corresponding to a published object for which only instances or cases can be shown.
As of processes, forms and agile forms, instances or cases of objects that are in this state cannot be created, delete or modified. On the other hand, rules can move to inactive state, if they are not being used in another object, that is, since they are inactive, they cannot be used in the modeling of another object.

> *Available in future versions ..*

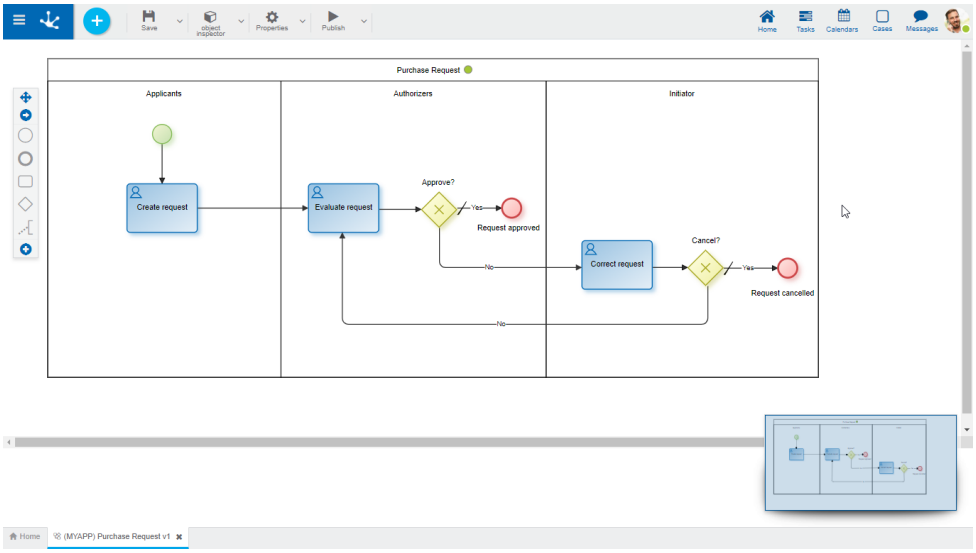
3.6.3. Processes Modeling

Processes Modeling

The process modeler from **Deyel** is a tool which allows IT modelers to design business processes in a collaborative work environment, using symbols from **BPMN 2.0** (Business Process Model and Notation)

By using this intuitive tool, the different business units can visualize the processes that are carried out within an organization and easily understand them.

When creating a new process or opening one already defined, it is allowed the access to a design space where the business process can be defined. Once it is finished, the process can be published and is available for use within the organization.



The general characteristics of the processes modeling environment and the main elements that compose it, are described in the topics:

- Modeling Facilities
- Graphic Elements

3.6.3.1. Modeling Facilities

Process Modeling> Introduction to process modeler

Process Modeling> Modeling a process from a template

New Process

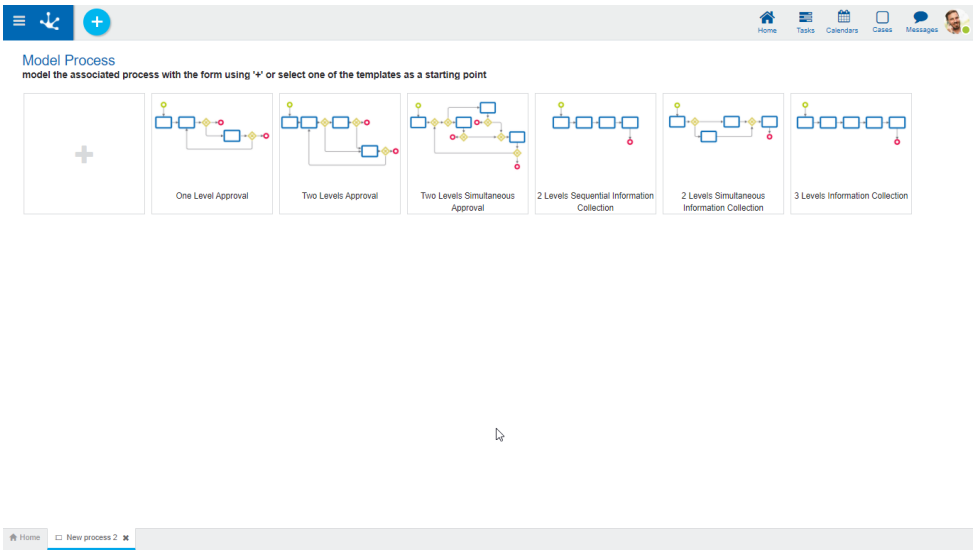
The modeler user from IT can design new procesess which, after being defined and published, can be used in the portal of **Deyel**.

In the same work space and by using concepts from **BPMN 2.0** (Business Process Model and Notation), it can be simply modelated in an intuitive environment:

- The process diagram with its activities, interconnections, events, time, alerts and actors involved, in a way that allows interaction with the business user and that they easily understand the process scope.

- Business rules which define the behaviour of each of the diagram elements, as well as the validations, automatic actions for sending messages and integration with other applications.

Steps for Creating a New Process



Step 1: Open the Assistant with Templates

A new process can be created from the icon corresponding to the context menu.

- From the icon
- From the drop down menu with option “Process”

In both cases, an assistant is opened which allows to define the new process by selecting one of the predefined templates to take as a basis, or starting from an empty design area, represented by the first element on the grid.

Step 2: Enter the Templates Properties

The properties are grouped in containers.

Basic Information

Presents information common to all processes.

Name of the Process

The modelator should enter the name of the process they are creating.

Owner

User, organizational unit or role that is modelating the process, it is the responsible of the process maintenance. The predetermined owner is the logged in user.

Open panel of profile query.
Remove the selected participant.
Select an existing participant. This icon is enabled when removing a participant.

Initiator

The participant who can initiate the process, it can be a user, an organizational unit, a role or an agent. For a participant to start a process, it must be published. The predetermined initiator is the agent All users.
It is possible to invite an external user as initiator, informing their email. If more than one user is invited, the generation of a new role starts automatically.

Open a window "New Participant" to define a new role or invite an external user.
Open panel of profile query.
Remove the selected participant.
Select an existing participant. This icon is enabled when removing a participant.

Participants

This container is only visualized if the new process is defined from a predefined template. The participants in the container depend on the type of template that has been selected. If an authorization template is selected, authorizing users will be displayed, while if a data collection template is selected, participating areas will be displayed. The predetermined value for the participants is the logged in user.

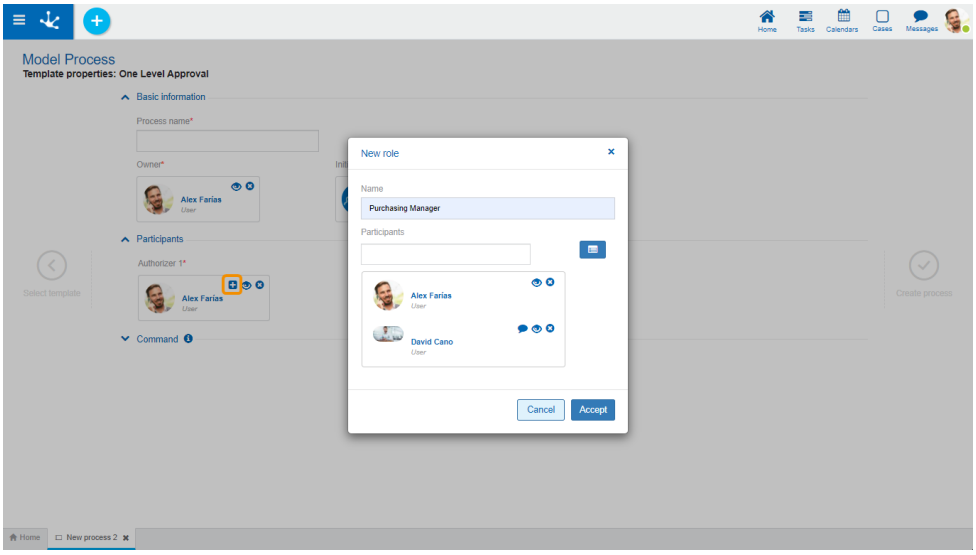
As many participants as there are actors in the selected template are defined. For each of the participants, a particular user can be selected, as well as an organizational unit, a role or an agent.

Open a window to define a new role or invite an external user.
Open panel of profile query.
Remove the selected participant.
Select an existing participant. This icon is enabled when removing a participant.

New Participant

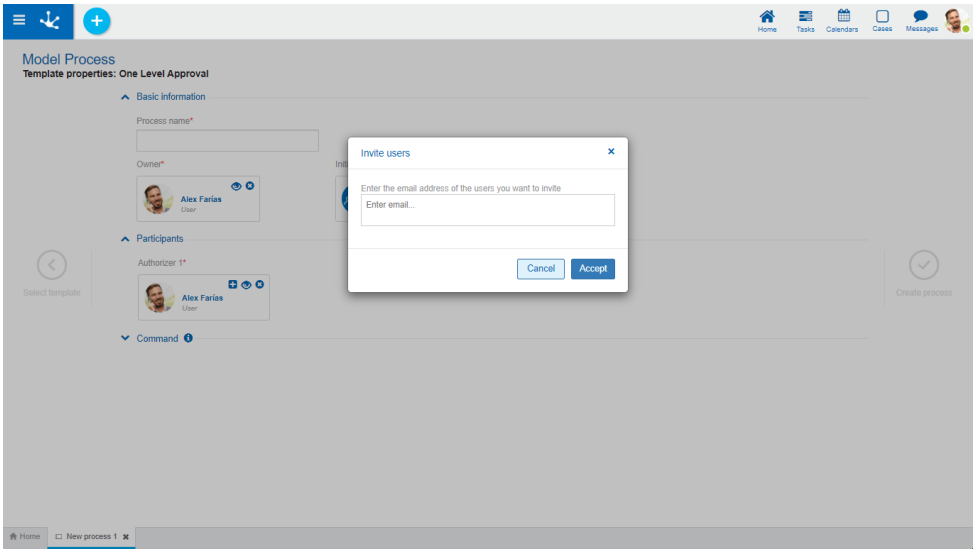
New Role

Allows to enter the name of the new role and the users that make it up.



Invite User

Allows to enter the mail of the external user who is invited. If more than one user is invited, the generation of a new role starts automatically.



Commands

A process can be used as a command, in which case the following properties must be completed.

Start from Chat

Indicates if a user can initiate th process from a conversation.

Command Name

Allows to enter the command name with which the process can be initiated from a conversation.

Define New Assistant

Indicates if a new chatbot is defined or if an existing one is selected.

Assistant Bot Name
If you mark the property Define New Assistant, the name of it must be indicated.

Select Assistant Bot
If you do not mark the property Define New Assistant, you must select which chatbot has the command.

Step 3: Close Properties Panel

Select Template
You can return to the templates assistant to select a new one.

Create Process
You move on to the process workspace and depending on the option selected in the assistant, the graphic modeling area is visualized with a different content:

- With the process diagram corresponding to the selected template.
- With the minimal figures so that a process can be saved, if none of the predefined templates has been selected.

Workspace Sections

- Top Toolbar
- Graphic Modeling Area
- Side Toolbar

3.6.3.1.1. Predefined Templates

Deyel provides a collection of standard process templates applicable to any organization. The modeler user of **Deyel** can select the template that best suits their needs to start modeling a new process immediately.

Besides, modifications deemed necessary to the process generated from the selected template can be added, as well as the definition of participants of each activity and the conditions that determine the data flow and its visibility.

According to the characteristics of the process they model, templates can be classified into groups.

Approval Process Templates

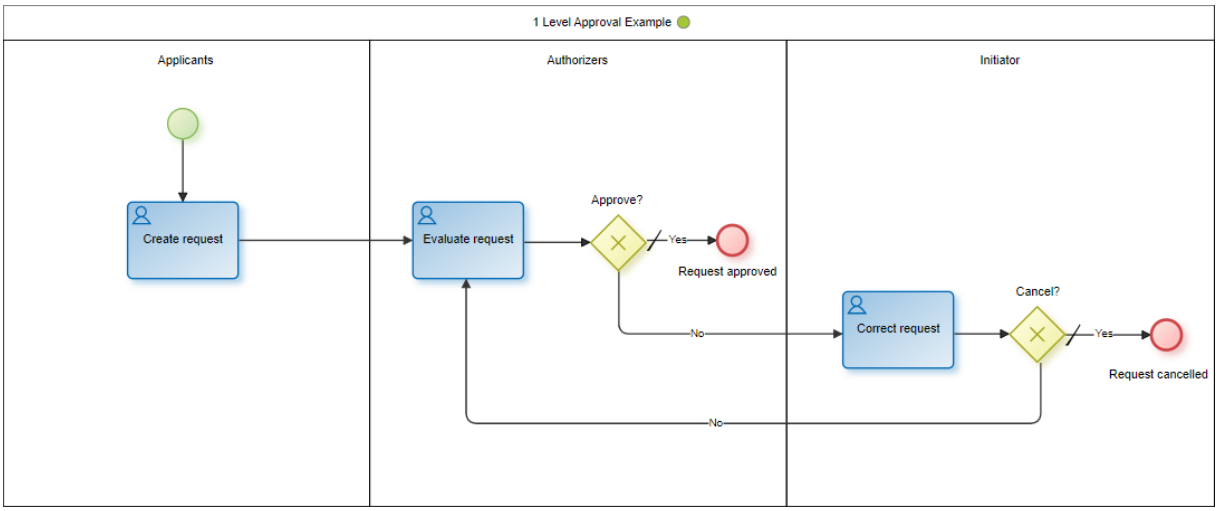
They model conditional branching situations, such as requests that are approved or sent for correction by the requestor.
These templates are suitable for defining vacation request processes, expense reimbursement requests or absence from work notice, among others.

When executing the process, business users display the buttons corresponding to the available actions in each activity, such as “Approve”, “Correct”, “Deny” or “Cancel”.

1 Level Approval

This template is used to implement processes that consider a single approval activity.

It allows users or organization areas, called “Applicants”, to enter a request and send it for evaluation to other users or areas called “Authorizers”. If the form is approved, the process ends, otherwise the form is sent back to the person who started the process to correct it and send it again for re-evaluation, or to cancel it outright.

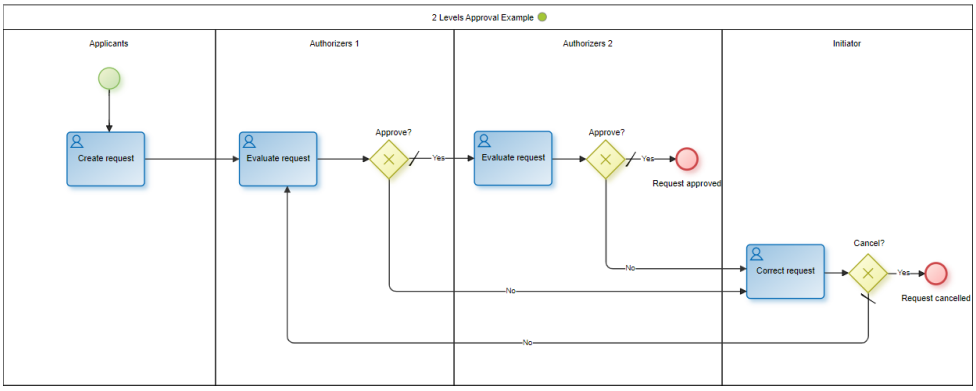


2 Levels Approval

This template is used to define processes with 2 sequential approval activities.

Adds an approval level to the template explained in "1 Level Approval ", that is, after the application is approved by the "Authorizers 1" level, it goes to the "Authorizers 2" level for evaluation. If the form is not approved in any of the authorization levels, it is sent back to the person who started the process to correct it and send it again for re-evaluation, or to cancel it outright.

These templates can be used to define authorization processes in organizations with different responsible persons for each approval level.

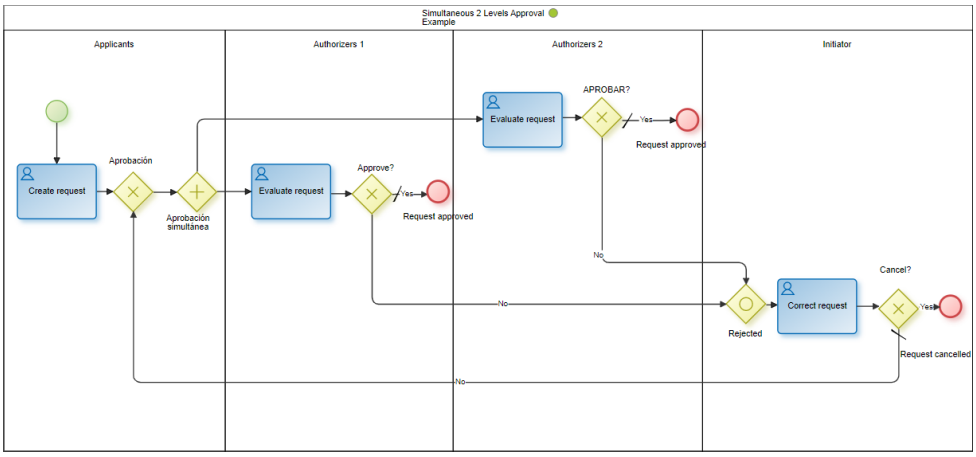


Simultaneous 2 Levels Approval

This template is used to define processes with 2 parallel approval activities.

It has two levels of approval as well as the template explained in “2 Levels Approval”. Once the application is created it must be evaluated by the "Authorizers 1" level and the "Authorizers 2" level for approval, regardless of the order in which it is authorized. The case is finished only after having both approvals. If the form is not approved in any of the authorization levels, it is sent back to the person who started the process to correct it and send it again for re-evaluation, or to cancel it outright.

These templates can be used to define authorization processes in organizations with different responsible persons for each approval level and where the approval sequence is not relevant .



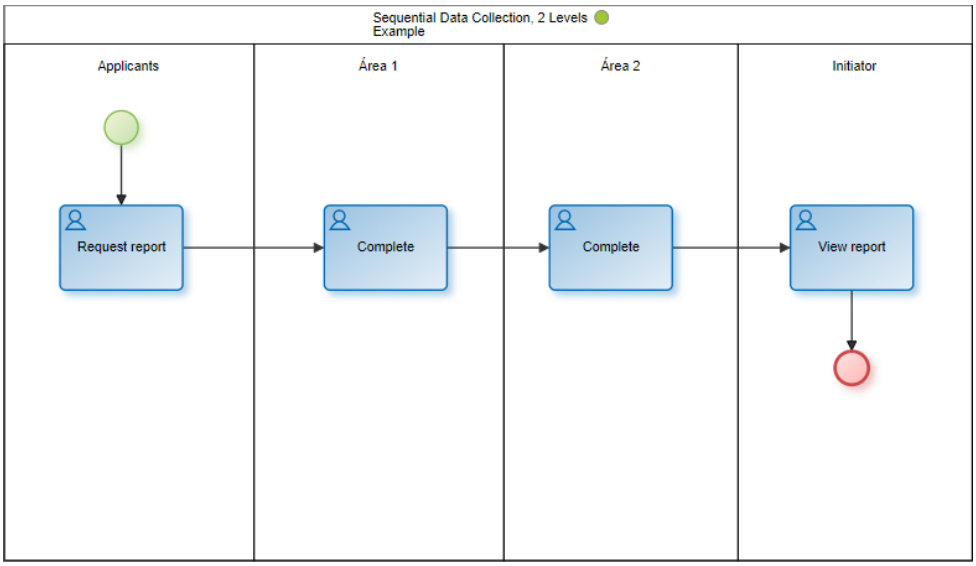
Data Collection Process Templates

They model processes where different business or office users, or the same but at different times, update the information of a case.

These templates can be used to define processes for hiring requirements, password generation, quote requirements, content publication, among others.

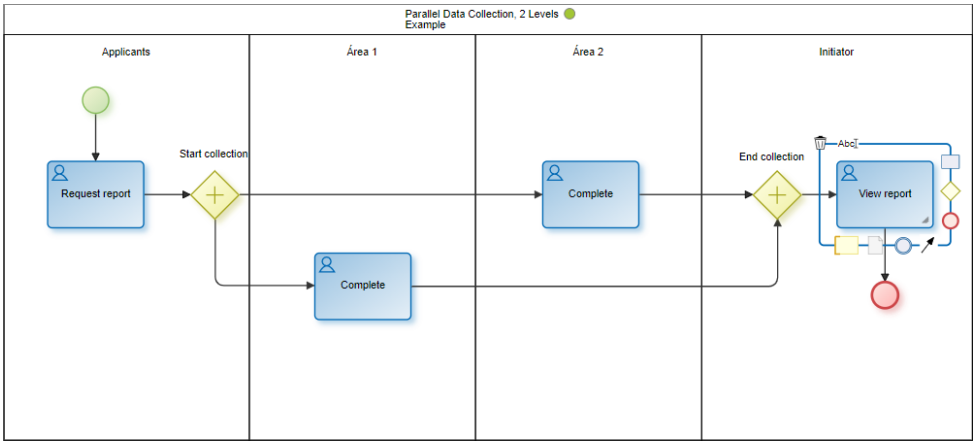
Sequential Data Collection, 2 Levels

This template is used to define processes that collect data in 2 different stages. It can involve more than one business or office user and the request is ultimately forwarded to whoever started the process.



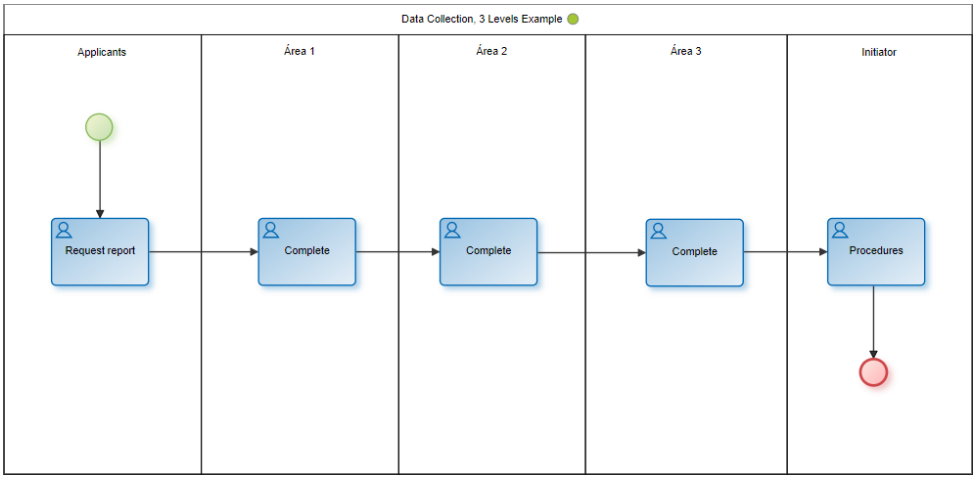
Parallel Data Collection, 2 Levels

This template is used to define processes that collect data in 2 different areas, regardless of the order in which they are performed. It can involve more than one business or office user and the request is ultimately forwarded to whoever started the process.



Data Collection, 3 Levels

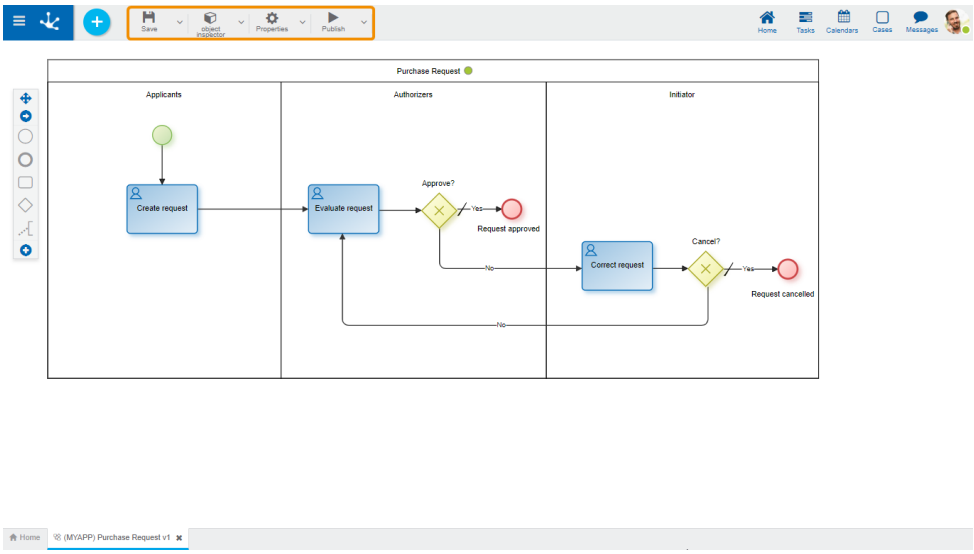
This template is used to define processes that collect data in 3 different stages. It can involve more than one business or office user and the request is ultimately forwarded to whoever started the process.



3.6.3.1.2. Top Toolbar

This toolbar contains icons and submenus from which you can perform operations on the object. Depending on its state, some options may be disabled.

The bar is displayed on the toolbar of **Deyel**.



Save

This icon allows to save the object in the repository of **Deyel**, leaving its state as "Draft" or "Modified". If certain conditions are met, the modeler user receives a message indicating that the operation was performed correctly, otherwise they receive an explanatory message.

Main Conditions

- The object´s application must exist.
- The name must be unique in the application.

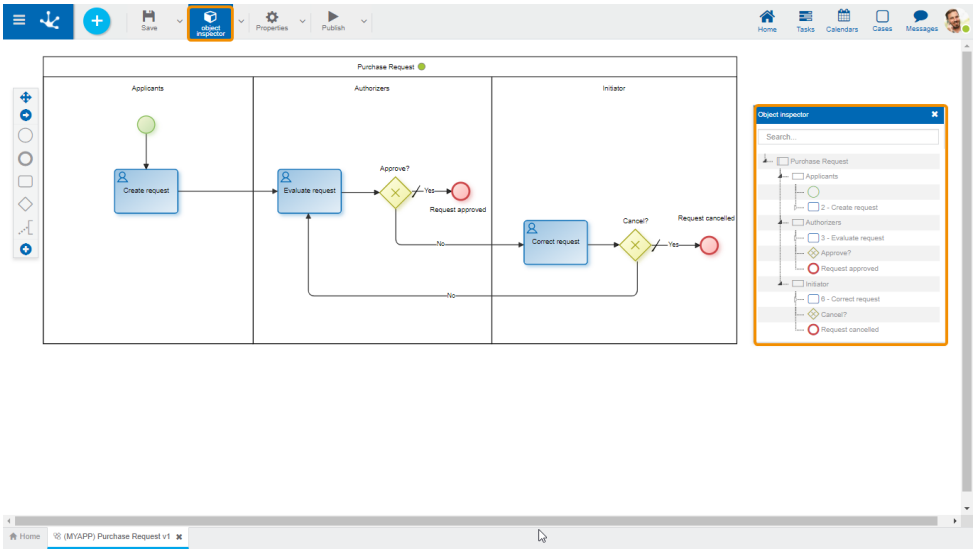
- The object must not be locked by other user.

To the right of the icon it is displayed the option of opening the submenu.

Object Inspector

When pressing this icon, a panel is displayed in which all the elements of the process with tree structure are shown.

This vision facilitates the analysis and development tasks, especially in complex diagrams.

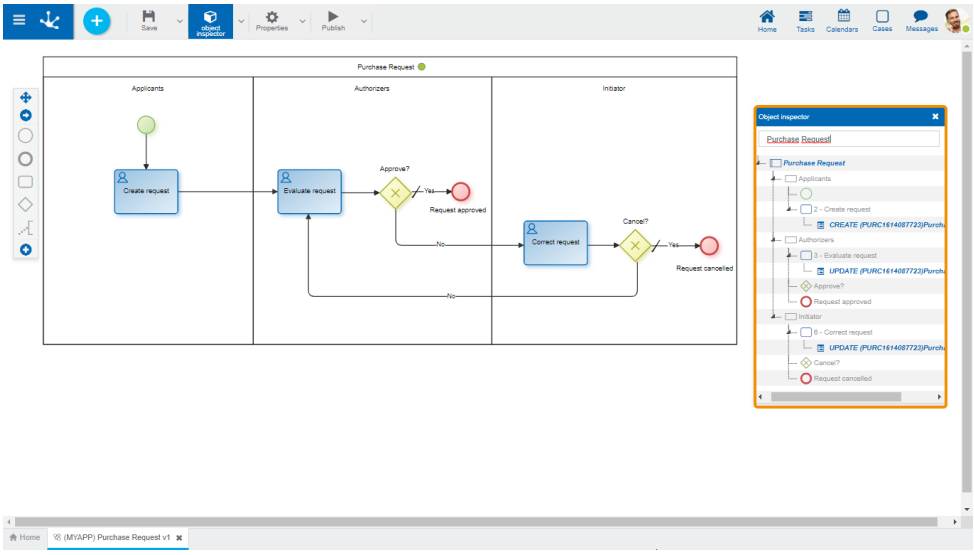


A search field is found on the top of the inspector which allows to search for elements by their names, identification codes, behaviour referred to the execution of an activity, or automatic actions already defined. Once the text has been introduced in the search field, the inspector shows the elements highlighted of the process that coincide with the inserted text.

From the object inspector, the properties panel of each element can be opened by double clicking on the name of said element.

To the right of the icon it is displayed the option of opening the submenu.

Usage Example
Identify in a process all of the activities which have the form named Expenses attached to them. When introducing such text in the search area, the inspector shows the references to the form in all of the activities that use it.



Properties

Open panel of process properties.

To the right of the icon it is displayed the option of opening the submenu.

Publish

By means of this icon the object goes to "Published" state, after verifying a set of additional conditions to those of the "Save" operation. The modeling user receives the corresponding message, indicating the result of the operation.

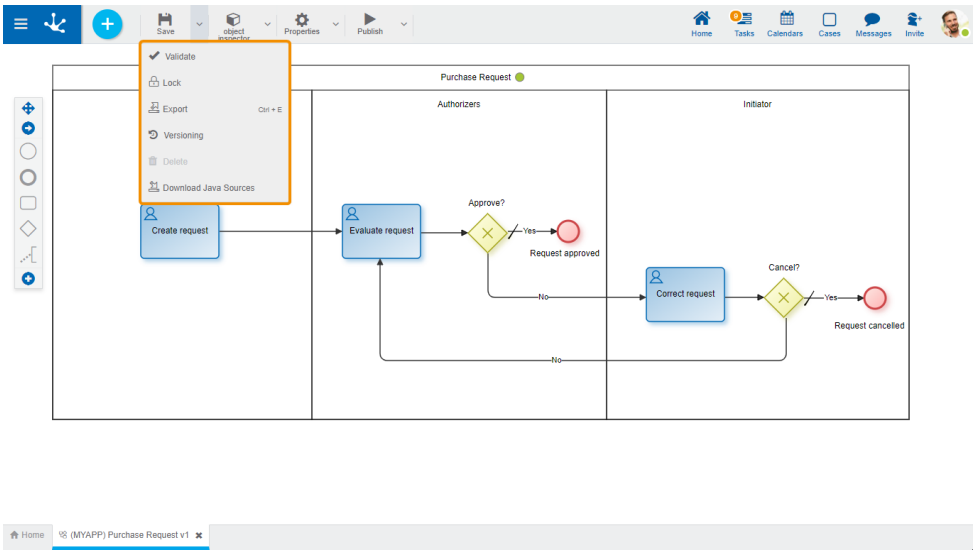
Main Conditions

- The process must contain at least one lane with an activity and a participant.
- Send task properties must contain valid values for its type.
- Maximum duration must be modeled to define alerts.
- If there are initiated cases, activities must not be deleted.
- Activities cannot have more than one output.
- Outputs of the inclusive and exclusive gateways must have condition or button.
- In tasks, subprocesses and processes the name is required.
- In tasks as “User” or “Not defined”, “Execution” tab must be defined.
- For subprocesses, a subprocess should be selected.
- For processes, the modeling participant is required.
- If an executing agent from a given activity is chosen in the lane, an activity must be defined for the participant.
- If an agent is chosen based on the form field in the lane, a form field must be defined for the participant.

To the right of the icon it is displayed the option of opening the submenu.

3.6.3.1.2.1. Save Submenu

This submenu is opened by pressing the icon immediately to the right of the icon corresponding to "Save".



Validate

This icon validates if the form is ready to be published, that is, validations performed at the moment of publicationare applied and the their result is reported.

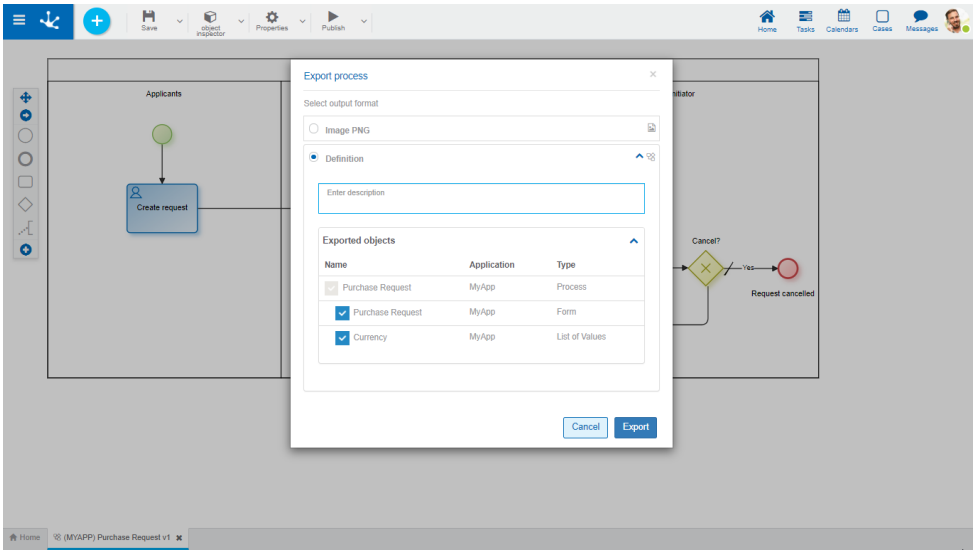
/ Lock/Unlock

A process diagram can only be modified by users, organizational units or roles defined as owner in the process properties.

- Allows blocking a process diagram to ensure that no one can modify it until the person who is using it unlocks it, that is, releases it.
- Allows to unlock a process so that another user, organizational unit or role, defined as owner can modify it.

Export

This icon opens a window for the user to select and confirm the export of the object and the related objects included in it.



Export Formats

PNG Image

A file with a png extension is generated, with the image of the process selected area.

Definition

It is the default export format for all objects.

Description

In this property a text explaining the reason for the operation can be entered.
This text can be modified upon import and is displayed in the description column of the export record.

Exported Objects

Expanding the container, name, application and type of the objects related to the export process are displayed. Objects not meant to be exported are unchecked.

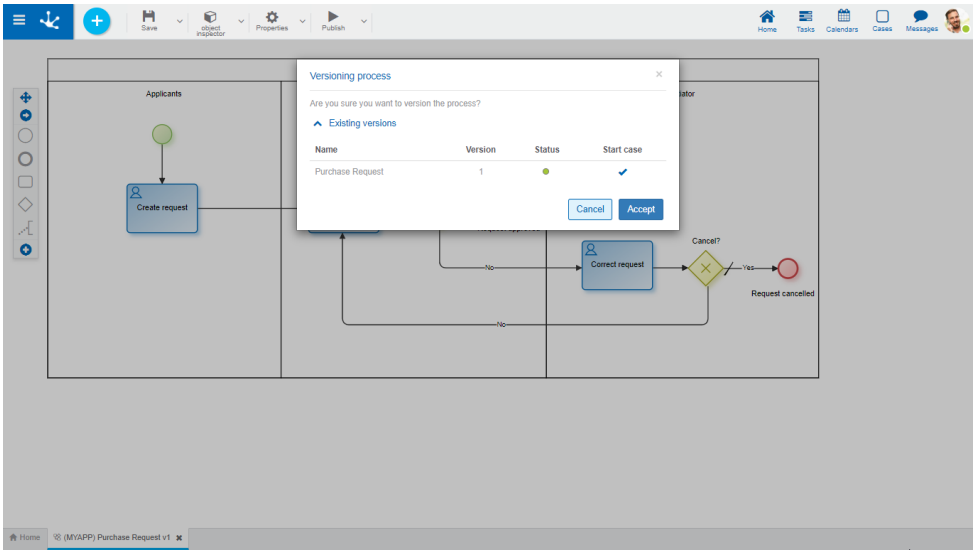
- Forms related to the process. If there are value lists associated with fields, their definitions are also included.
- Advanced rules used in automatic actions, in fields, from the "Relation" tab of the field properties or included in embedded rules.

> *Forms representing related entities are not included.*

Press the "Cancel" button to undo export or the "Export" button to finish.

Version

Allows the user to version the process, display the previous versions and show them. By hovering over each previous version, an "Open" button is available to open that version.



Existing Versions

Name
Name of the process being modeled.

Version
Number of each existing version.

States
State of each existing version.

Starts Case
Indicates the version to start cases. Only cases of a single version of the same process can start; being this the latest version that is in the "Published" or "Modified" state.
Press the "Cancel" button to cancel this versioning or press the "Accept" button to generate a new version from the previous one, that is, to generate a process whose version number immediately follows the current version.

Delete

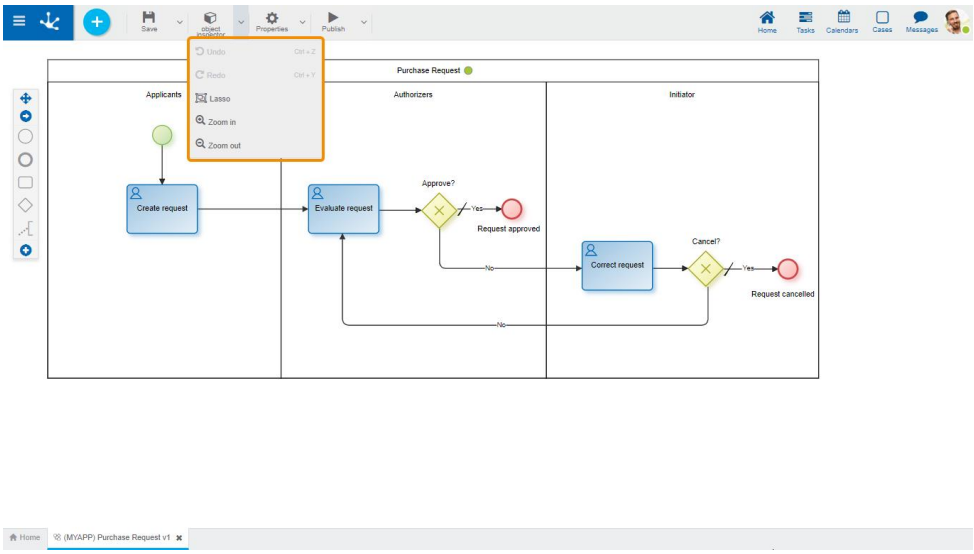
Allows to delete the object only if it is in "Draft"state, closing the tab in which it is located and deleting it from the modeler's grid.

Download Java Fonts

This icon allows to download the Java files that represent the object's model and service, so that it can be used in advanced rules.
Pressing the icon displays a message to confirm file download.

3.6.3.1.2.2. Object Inspector Submenu

This submenu is opened by pressing the icon immediately to the right of the icon corresponding to "Object Inspector".



Undo

Returns the last modification made to the diagram. The same behavior is obtained with the keyboard combination ctrl + z.

Redo

Redoes the action on the diagram that was eliminated with the "Undo" option. The same behavior is obtained with the keyboard combination ctrl + z.

Link

Allows to select the graphic elements that are linked by pressing the right mouse button on a point in the diagram and dragging to another end without releasing. Linked figures are temporarily marked with a border and can be moved using the cursor or deleted altogether.

Increase Zoom

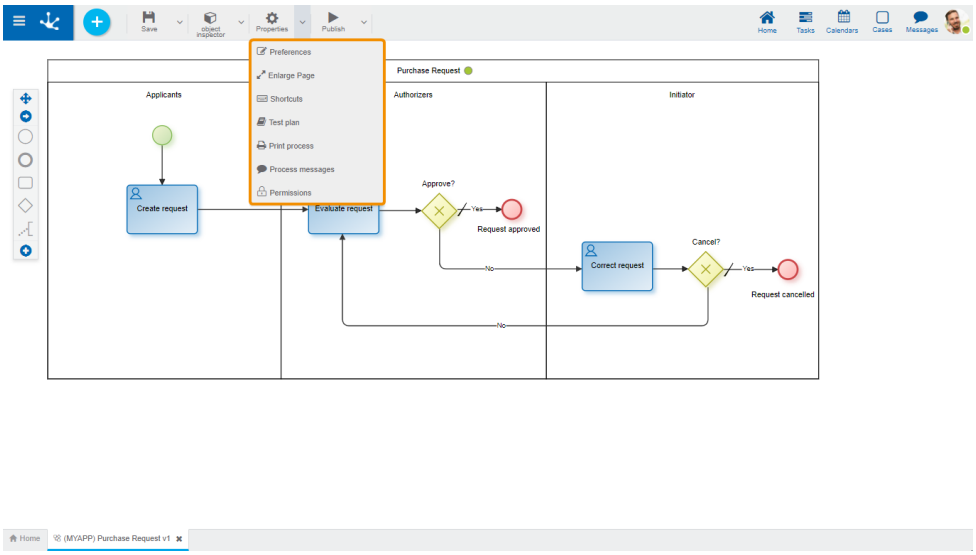
Increases the display size of the diagram.

Decrease Zoom

Decreases the display size of the diagram.

3.6.3.1.2.3. Properties Submenu

This submenu can be opened by pressing the icon immediately to the right of the icon corresponding to "Properties".



Preferences

Defines the display preferences for process modeling.

- Show symbol identifiers
  Indicates whether the internal identifiers of symbols are shown in the diagram. The default value is "No".

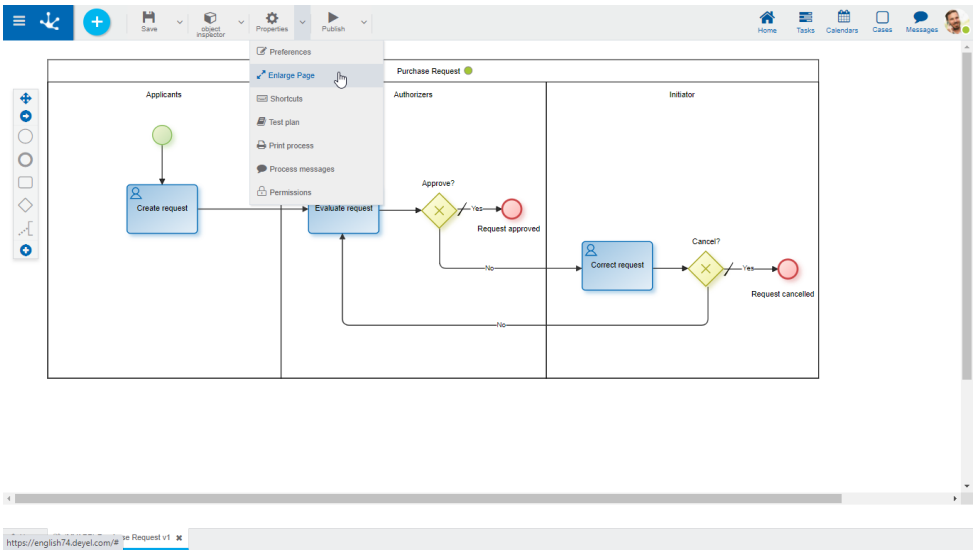
- Grid size
  Define the number of pixels used when moving symbols in the diagram, when dragging them with the mouse. The default value is “1”, although “10” or “20” can be selected for larger movements of the symbols with the mouse.

- Display Minimap
  Indicates whether the thumbnail view is displayed in the lower right corner. This view consists of a small panel where the complete process is displayed.

- Display validations on shapes
  Indicates whether the exclamation mark is displayed on shapes with validation errors.

Enlarge Sheet

Increases the modeling area, which makes scroll bars to be displayed and the area of the diagram that is visible in the window is shaded in the thumbnail view. As scroll bars move, the shaded area scrolls in the thumbnail view.



Shortcuts

Opens a panel with all available keyboard shortcuts to use in process modeling.

Test Plan

In a new tab, it opens a panel with the grid corresponding to the test plans defined for the process. Operation buttons are available for each line of the grid. This option is enabled only when the process was saved.

Print Processes

Allows to print the process design. Opens a panel with the printing functionality for the process in a new tab. This option is enabled only when the process was saved.

Process Messages

Opens the case modelers chat window.

Permissions

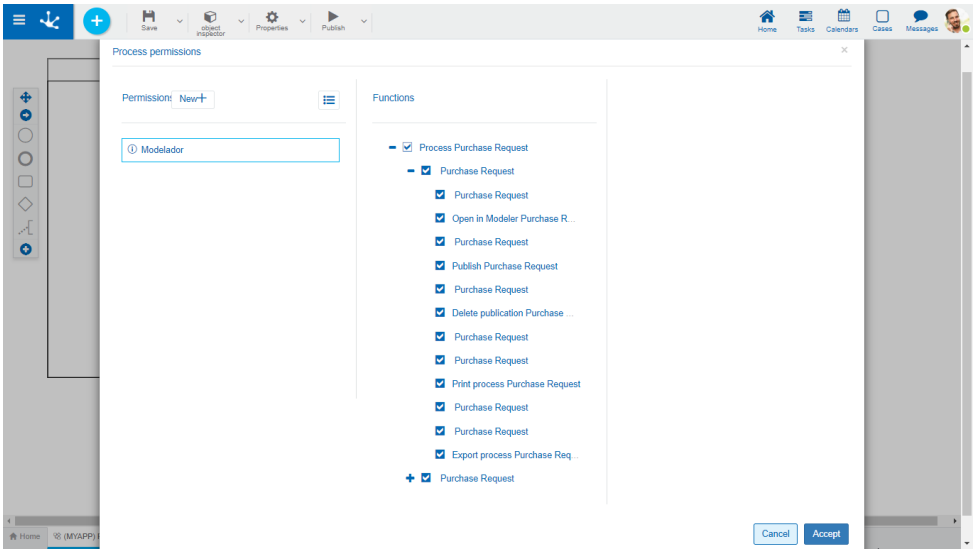
Allows to assign the security functions for use and modeling of the object to the existing permissions or to new permissions that the user creates, without having to go to the permission settings option from the menu.

Sections

- Permissions: Permissions to which object functions are assigned.
- Security Functions: Represents the total set of security functions, modeling and use of the object. Those that are marked are the ones included in the selected permission.

By default, all security functions for a new object are assigned to the permission Modeler.

Users who are assigned the permissions have access to the functions included in it.



New+ Opens a panel to create a new permission and once created, the security functions included in it must be selected .

Search permissions... Allows to select a permission from a list and enables the input of characters to filter the values in the list.

Opens the wizard to select a permission and once chosen, the necessary object security functions must be checked.

To unrelate a permission from the object, hover the cursor over the permission and press the icon ✖. If there are functions selected for that permission, they must be unchecked in order to delete it.

Modeling Security Functions

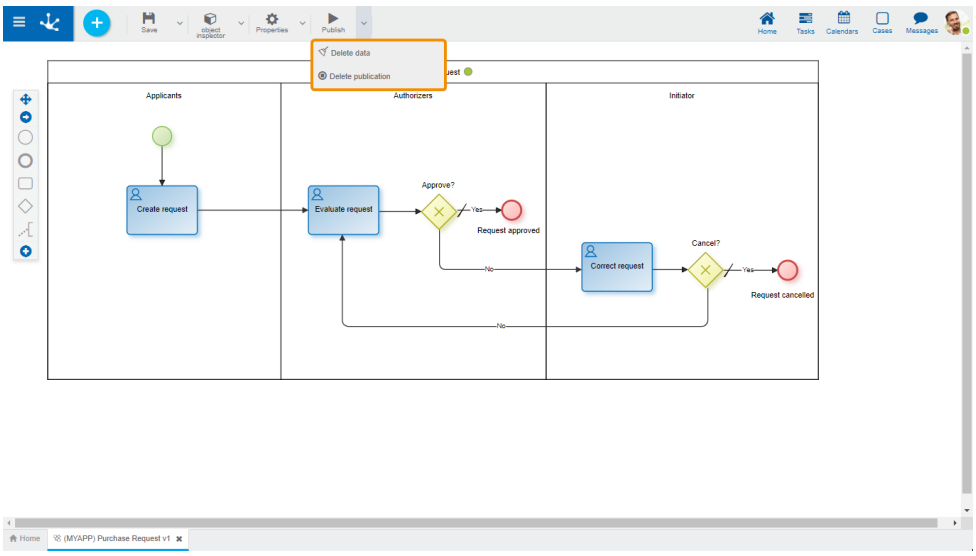
- Visibility in the modeler: Allows to display the object in the Deyel modeler.
- Open in modeler: Allows to show the object from the Deyel modeler.
- Save changes: Enables the operation of saving modifications made to the object.
- Publish: Enables the operation of publishing the object leaving its state as "Published".
- Deletes cases: Enables the delete data operation.
- Delete publication: Enables the operation of deleting the object publication leaving its state as "Draft".
- Delete draft definition: Enables the operation of deleting the object.
- Block process: Enables the Lock/Unlock operation, only the user who locks it can modify it.
- Print Process: Enables the process print operation.
- Scenarios: Enables the operation of creating different testing scenarios.
- Test plan: Enables the operation of creating a test plan and adding scenarios to it.
- Export: Enables the operation to export the object from the save submenu of the top toolbar.

Security Functions of Use

- Start by agent: Allows to start a process if "Initiator Restricted by Function" was defined in the first lane as a participant.

3.6.3.1.2.4. Publish Submenu

This submenu can be opened by pressing the icon immediately to the right of the icon corresponding to "Publish".



Delete Data

All cases are deleted from the process. If each case had an associated form instance, it would also be deleted.

Delete Publication

Allows to delete the use process, returning it to the "Draft"state, in addition to deleting the process cases and the associated form instances, if any.

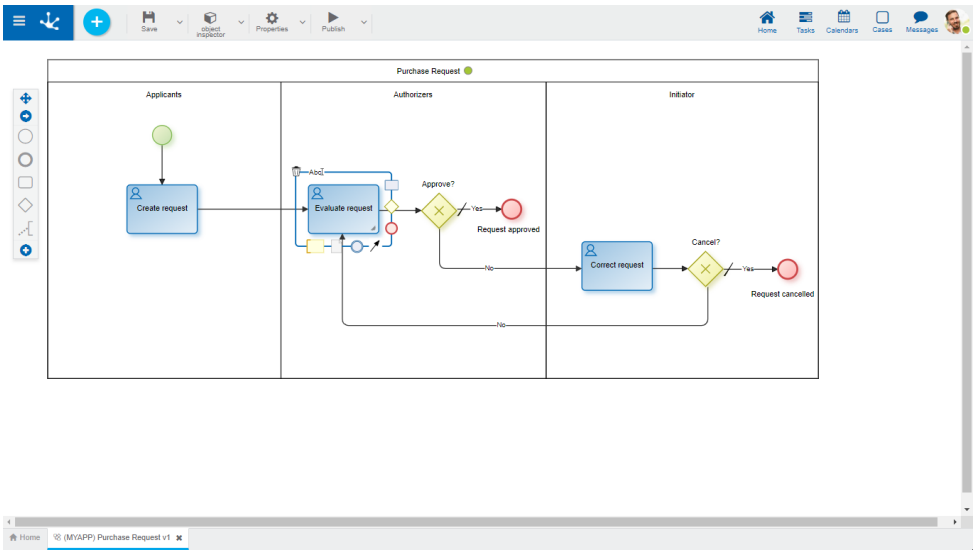
3.6.3.1.3. Graphic Modeling Area

Processes Modeling > Graphic Modeling Area

Graphic modeling area is where processes are designed by using graphic elements from **BPMN 2.0** (Business Process Model and Notation)

Pie Menu

Circular menu which is displayed by making click on each modeled graphic element in the design area, showing the different symbols with which the selected object can connect to.



To incorporate a new connected symbol, pie menu of the graphic element selected must be opened, drag and drop the symbol to be connected on the modeling area.

Common Options

Allows to delete the symbol

Allows to edit the name of the symbol for its modification

Context Menu

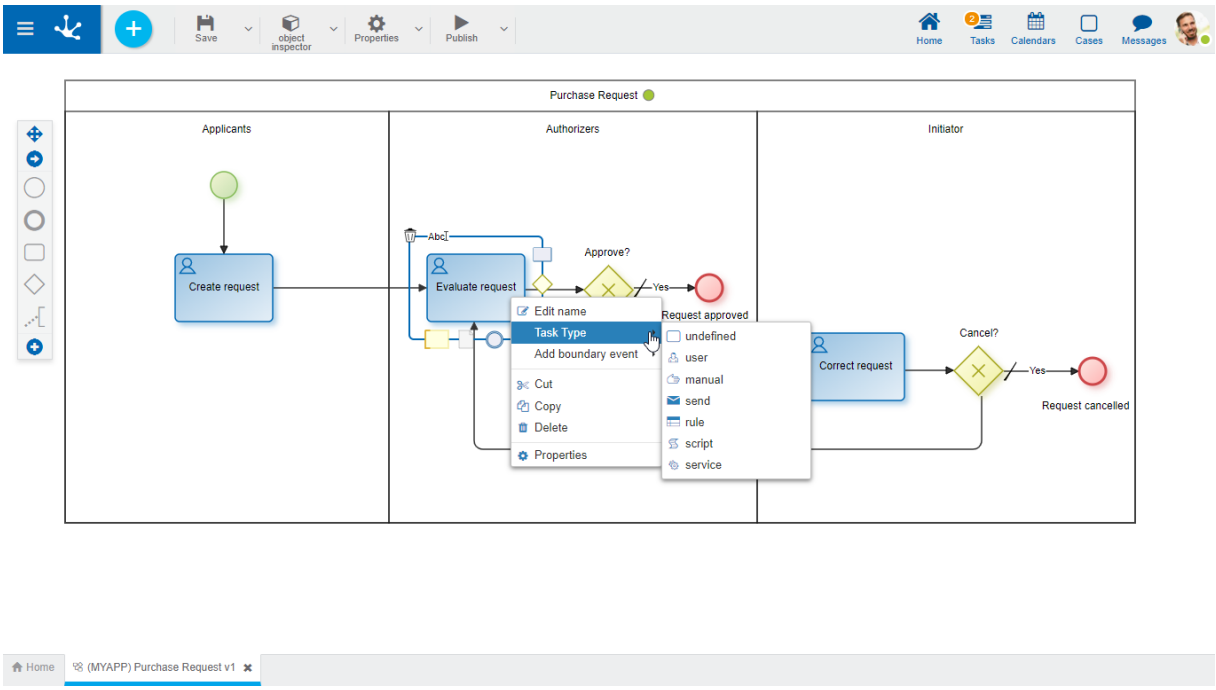
On clicking on the selected graphic element with the right mouse button, it opens a vertical menu with its options grouped as own of the object and common to all objects, besides from an option to open the properties.

Options

- Own of the Object
  - Edit name
    - Available types according to the object
    - Other own actions of each object

- Common to All Objects
  - Cut
  - Copy
  - Delete

- Properties
  Opens the properties panel of the graphic element, it is the same panel that is displayed by double clicking on the symbol.



Thumbnail View

Complete image of the process which is visualized on the lower right corner of the graphic modeling area, the active work area is reflected in such image. It is visualized whenever you activate the option Thumbnail View of the option Preferences in the top toolbar.

Scrolling in the Modeling Area

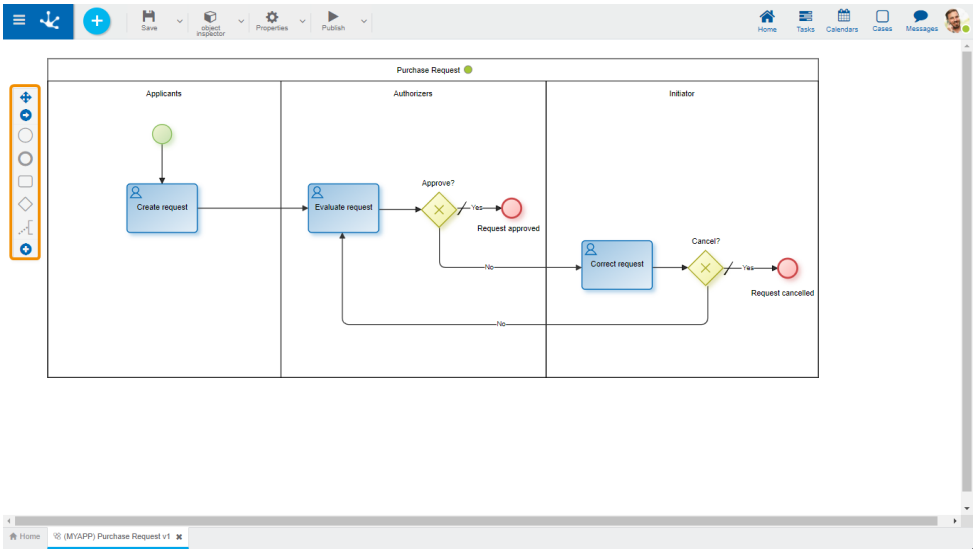
The vertical and horizontal scroll bars are used to slide the graphic modeling area, in order to allow the complete edition of the process. They are automatically activated when the diagram exceeds the size of the work window.

The modeling area can be enlarged by using the option Enlarge Page on the top toolbar.

3.6.3.1.4. Side Toolbar

Process Modeling > Graphic modeling area

It is the toolbar that contains the graphic elements corresponding to the different types of symbols that can be defined in the process diagram, based on **BPMN2.0** (Business Process Model and Notation).



Bar Elements

Scroll Bar

Allows to move the icon bar to any position in the graphic modeling area.

/ Show/Hide Names

Either displays a panel to the right of the bar with the names of icons, or closes it.

Start Event

A start event Indicates the beginning of a process, so they have no input flow but may have a trigger event.

Types

- Standard
- File
- Command
- Rule
- Email
- Signal
- Timer

End Event

An end event Indicates the end of a process, so it has no output flows.

Types

- Standard
- Signal
- Terminal

Task

A task represents a work unit that is carried out as part of a process execution.

Gateway

An gateway represents a branch point. Gateways can be of different types.

- Exclusive without marker
- Exclusive with marker
- Inclusive
- Parallel

Annotation

An annotation allows the process modeler to add additional data, intended for the diagram reader.

/ Show/Hide more Shapes

Allows to display the sidebar in full or reduced form.

Subprocess

A subprocess is an activity that refers to another independently defined process.

Intermediate Event

An intermediate event indicates where events may occur between the beginning and the end of a process. They affect the process flow, but they will not start nor end a process directly.

Types

- Capture link
- Catch signal
- Throw link
- Throw signal
- Timer

Data Object

A data object represents information that flows through the process such as documents or emails.

Group

A group allows to group symbols visually.

3.6.3.2. Graphic Elements

The graphical elements of **Deyel** process modeler are based on **BPMN 2.0** (Business Process Model and Notation).

BPMN 2.0 is a standardized graphical notation that allows to model business processes in a workflow format. It provides an easy and understandable notation for all those involved in the business, whether they are business analysts (who define the processes), technical developers (responsible for implementing them) or business managers (who monitor and manage them).

BPMN 2.0 is intended to be the common language to reduce the communication gap that frequently occurs between the business process design and its implementation.

To model processes, elements must be dragged from the left palette to the selected position in the graphic modeling area and the properties of each element must be entered.

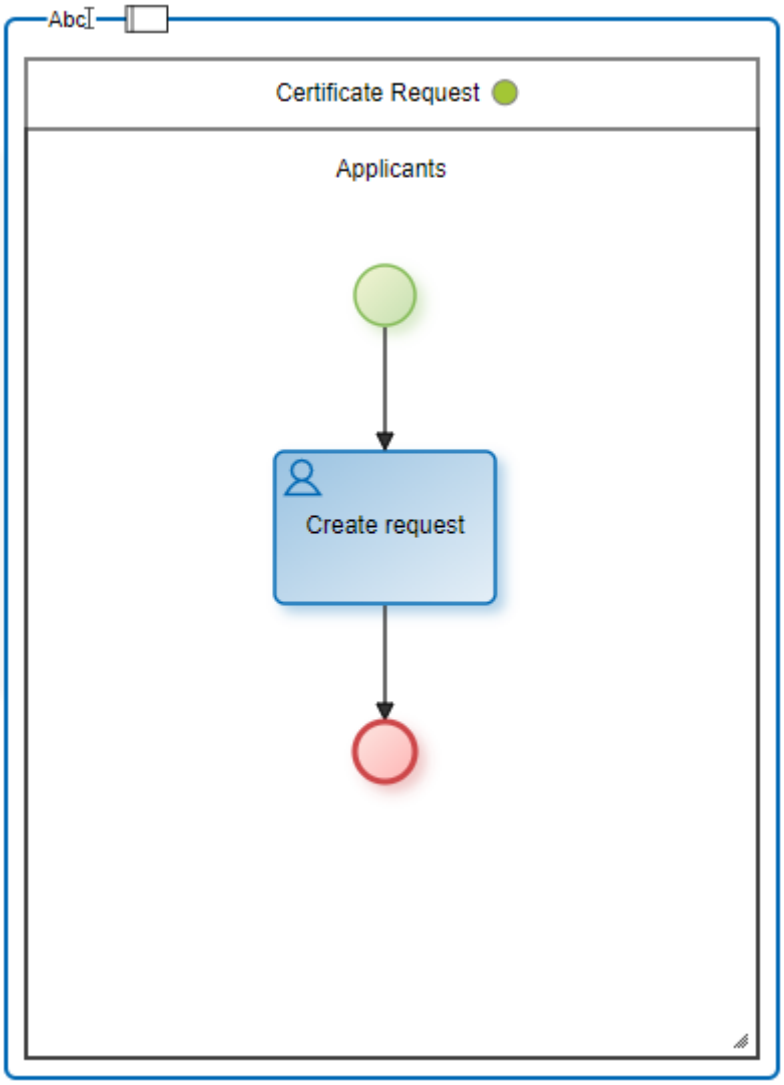
- Pool
- Lane
- Activity
- Flow
- Event
- Gateway
- Artifact

3.6.3.2.1. Pool

The pool is a graphical representation of a business process (Business Process).

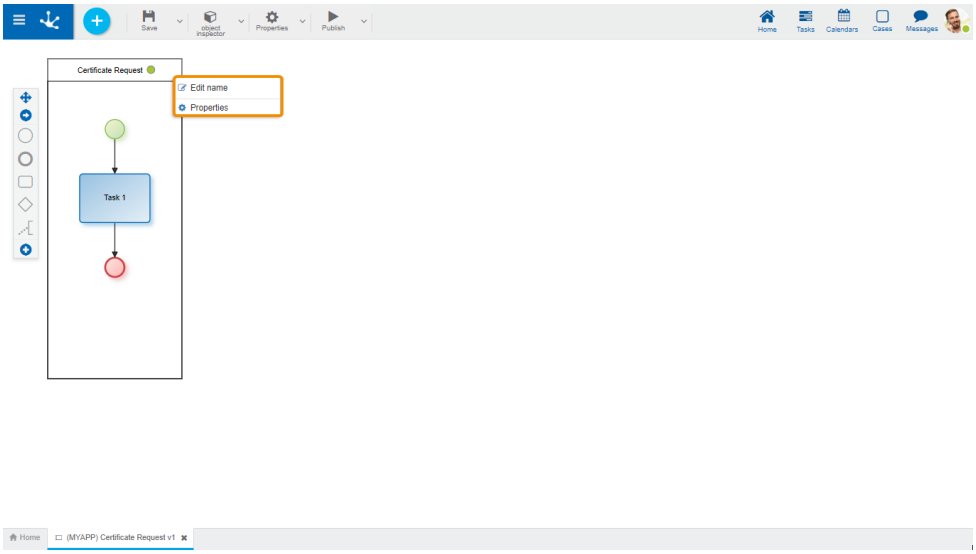
"It is a set of one or more activities whose performance allows meeting a business objective, normally within an organizational structure context, which defines functional roles and relationships among them." [Workflow Management Coalition]

Pie Chart Menu



Abc	Opens an area for editing the name.
	Adds a lane. Clicking on this icon adds a lane to the right inside the pool. Every pool must have at least one lane. When you create a process without a template, a lane is automatically generated.

Context Menu



- Edit name: Allows to edit the name. Same functionality as the icon Abc
- Properties: Opens the properties panel to show and/or modify.

3.6.3.2.2. Lane

Lanes allow organizing the activities of a process. Each lane represents one participant.

"The lane is a partition that is used to organize and categorize activities within a pool. It is generally used to represent internal roles (for example managers), systems (e.g. business application) or internal departments (e.g. finance)." [Workflow Management Coalition]

Types of Participants

User
The activities included in the lane are executed by a specific user or by their delegated users, if any.

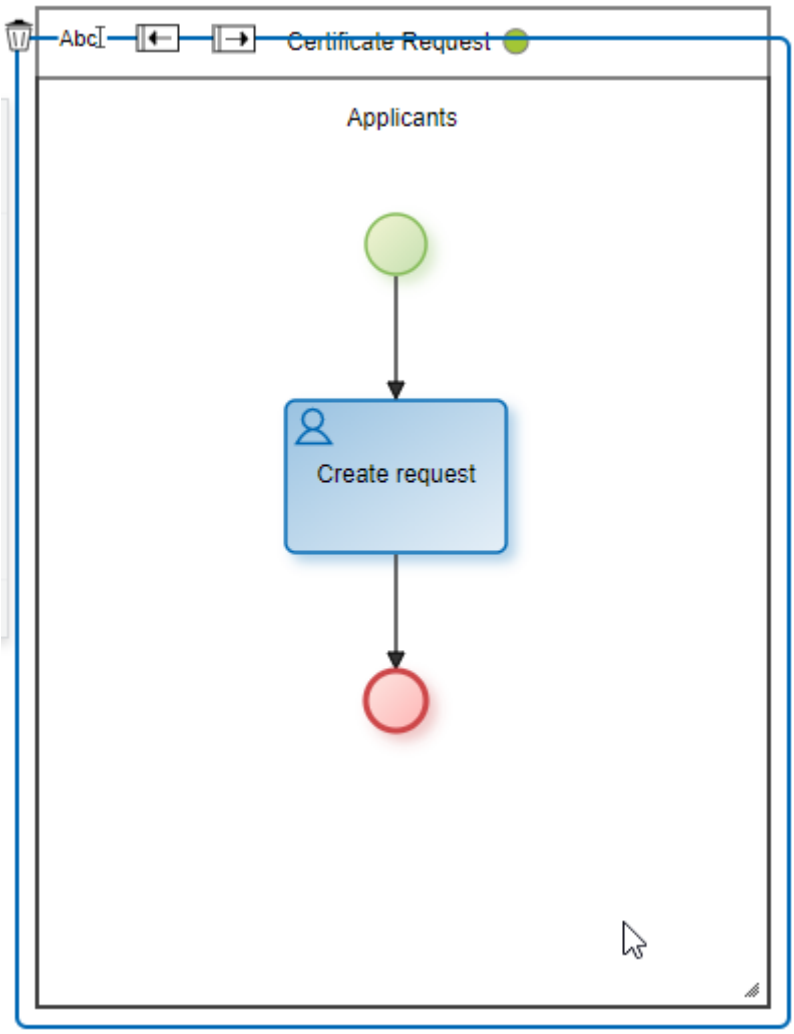
Organizational Unit
The activities included in the lane are executed by every user that belongs to the organizational unit informed.

Role
The activities included in the lane are executed by the actors defined within the role.

Agent
The activities included in the lane are executed by a participant who is dynamically identified when the activity is executed.
A predefined agent can be assigned as lane manager or else, define a new agent.

Lanes without Participants
If at the time of modeling a process, the persons responsible for each activity are not known, the lane may be left without an assigned participant as long as the process state is "Draft" or "Modified".

Pie Chart Menu



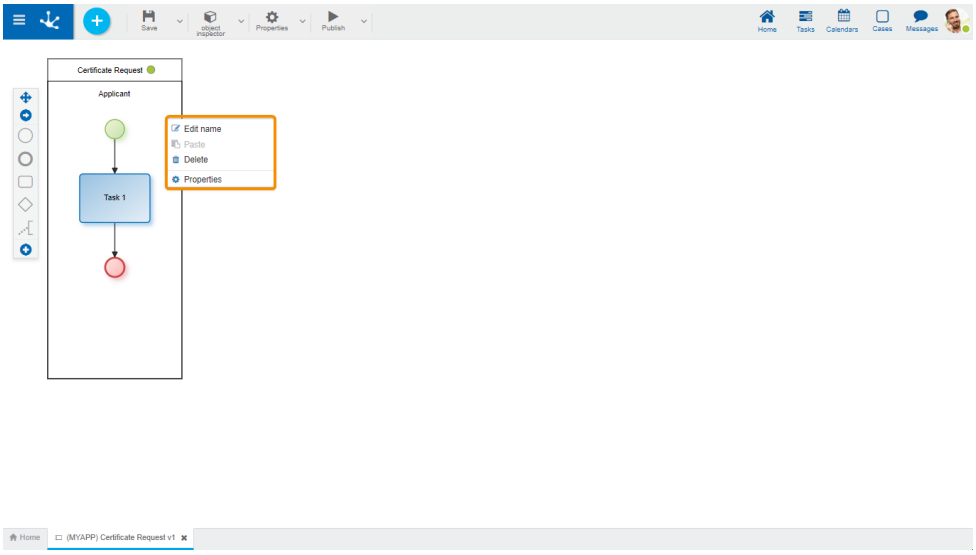
Abc Opens an area for editing the name.

Deletes the lane.

Add a lane to the left.

Add a lane to the right.

Context Menu



- Edit name: Allows to edit the name. Same functionality as the icon Abc.
- Paste: Allows to paste a previously copied graphic element into the lane.
- Delete: Allows to delete the lane. Same functionality as the icon .
- Properties: Opens the properties panel to show and/or modify.

Lane extension
When more space is required to incorporate elements into the lane, either horizontally or vertically, click on the icon and drag the cursor until the required lane size is achieved.

3.6.3.2.3. Activity

An activity can be understood as the representation of a work unit that is carried out as part of a process execution.

"An activity is the description of a task that is a logical step within a process." [Workflow Management Coalition]

The graphic representation of this element is a rectangular image with rounded corners that contains a label with the name of the task inside it, identifying the types of activities with different icons inside.

Types

- Tasks
- Subprocess

3.6.3.2.3.1. Task

A standard activity or task is an indivisible work unit performed as part of the execution of a process.

Types

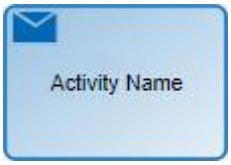
The **BPMN 2.0** notation (Business Process Model and Notation) allows to typify standard activities or tasks in such a way that the graphic element used indicates the task type it represents.
The task type definition is made in different ways:

- From the task context menu, selecting the option "Task type".
- From the task property panel.

Undefined Task
Task without type.

User Task
Task that needs human intervention to be performed. When an activity modeled as a user task must be executed, it is informed to the participants of the process in My Tasks, in the top toolbar It is recommended to use this notation to model all the activities in which the user has to create, modify or show a form.

Manual Task
Represents a task to be performed by the user without any application support. Only moves on to the next activity.
It is recommended to use this notation to model activities such as making a telephone call to the customer, paying a visit or installing audio equipment, among others.



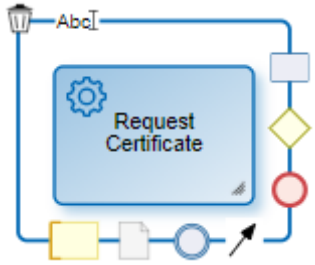
Send Task
It is used to send a message to an external participant regarding the process. When the message has been sent, the task ends.
This task type is used to model automatic mailing tasks. They do not define user participation.

Business Rule Task
Represents a call to an advanced rule, being able to exchange parameters between the task and the rule. They do not define user participation.

Script Task
This task type only performs automatic actions. They do not define user participation.

Service Task
Represents the execution of a service, automatically and without user intervention, which is carried out through the adapters. This notation is recommended to model the use of a web service, an automated software component or an integration rule. They do not define user participation.

Pie Chart Menu



Abc Opens an area to edit the name.

Deletes the task.

Adds an activity.

Adds a gateway.

Adds an end event.

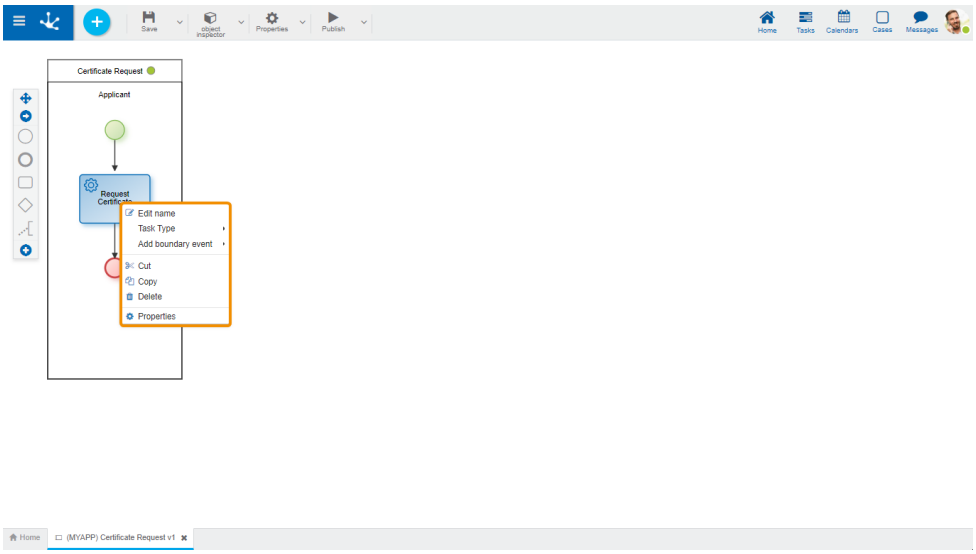
Adds a flow.

Adds an intermediate event.

Adds a data object.

Adds a comment.

Context Menu



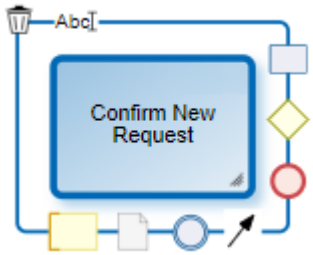
- Edit name: Edits the name inside the rectangle that represents the activity. Same functionality as the icon Abc.
- Task type: Defines the type of task.
- Add border event: Incorporates an border event to the task.

- Cut: Deletes the selected task. It can be pasted in another process location.
- Copy: Copies the selected task. It can be pasted in another process location.
- Delete: Deletes the selected task.
- Properties: Opens the property panel to show and/or modify.

3.6.3.2.3.2. Subprocess

It is an abstract activity that represents an independently defined process. It is used to hide levels of detail in processes and allows to reuse modeled processes. For example, if different processes have a common part, it can be defined only once and be reused in each process.

Pie Chart Menu



Abc Opens an area for editing the name.

Deletes the subprocess.

Adds an activity.

Adds a gateway.

Add an end event.

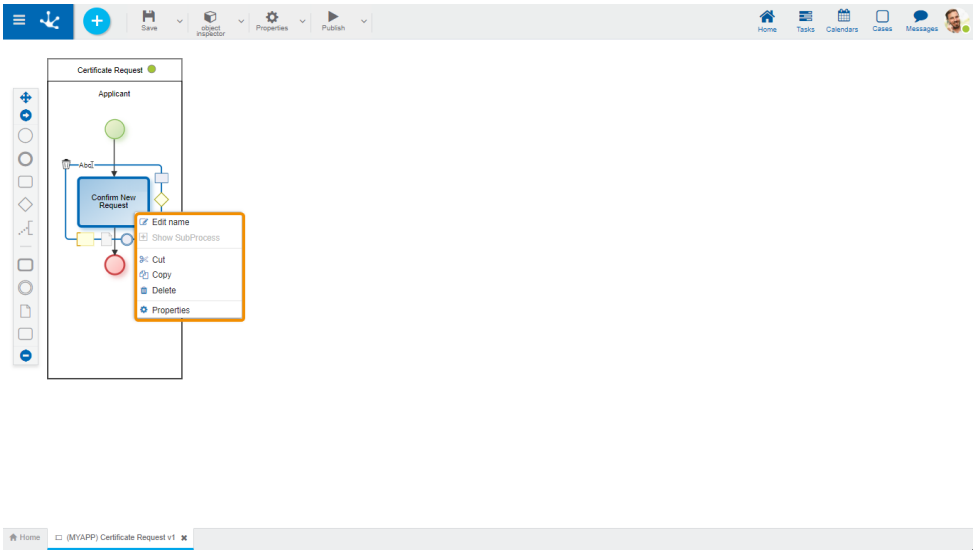
Adds a flow.

Adds an intermediate event.

Adds a data object.

Adds a comment.

Context Menu



- Edit name: Edits the name inside the rectangle that represents the activity. Same functionality as the icon Abc.
- See subprocess: Opens the subprocess in a new modeler tab.
- Cut: Deletes the selected subprocess. It can be pasted in another location.
- Copy: Copies the selected subprocess. It can be pasted in another location.
- Delete: Deletes the selected subprocess.
- Properties: Opens the properties panel to show and/or modify.

3.6.3.2.4. Flow

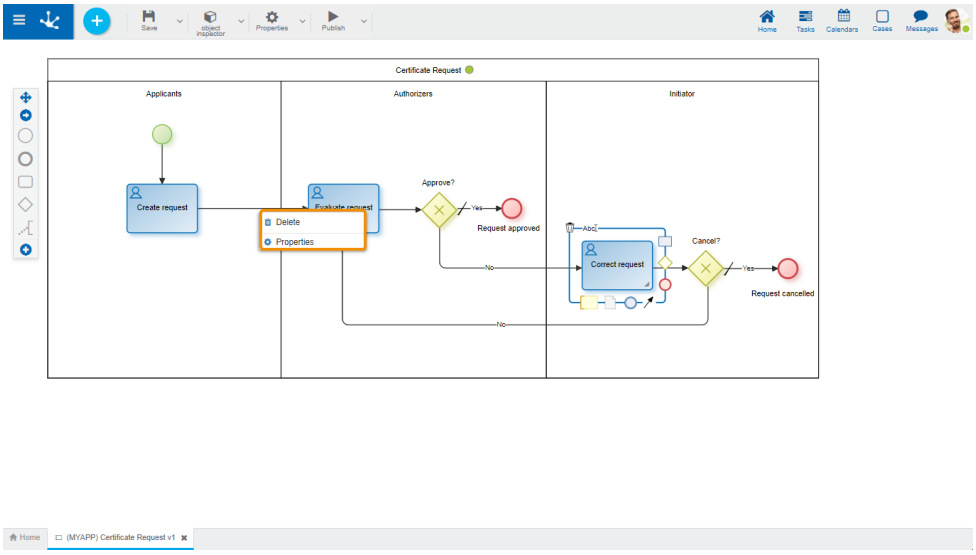
A flow represents the connection between an activity and its subsequent activities. Flows are used to establish the sequence in which defined activities are executed.

"Sequence flows are used to show the order in which activities are carried out within a process." [Workflow Management Coalition]

Normal flow
A flow can determine a required path, that is, the process has no option not to execute it nor an optional path, in which case, the subsequent activity may or may not be executed.

Default Flow
The exclusive and inclusive gateways must have a defined predetermined flow so that the process can execute its path in case none of the conditions for the output flows of the aforementioned gateways are met.

Context Menu



- Delete: Allows to eliminate the flow.
- Properties: Opens the properties panel to show and/or modify.

3.6.3.2.5. Event

An event is an fact or incident that occurs during the execution of a business process. In general, they require an action and allow a reaction.
Represents situations such as start of activities, end of activities, document change of state, message arrivals, among others.

"An event is something that happens during the course of a process. They generally have a cause and an impact." [Workflow Management Coalition]

Start events
They indicate the way in which a process is started. They have no input flow.

Standard
It does not have a trigger event, therefore it indicates that the process does not start automatically.

Command
Indicates that the process starts by executing a command from a chat window of the instant messaging of **Deyel**.
These events are used to model processes that are executed from a chat window, by means of actionable messages if data entry or decision making is required.

Email
Indicates that the process starts when an email arrives to a certain email account, which may come from another participant, another process or other application. They are detected by scheduled tasks called EMAIL-type Event Generators.

Timer
Indicates that the process starts after a certain time cycle or on a specific date.

File
Indicates that the process starts when a new file is detected in a certain folder.

Rule
Indicates that the process starts from the execution of a certain business rule.

Signal
Indicates that the process starts each time it catches the indicated signal. The current process may not be the only receiver of the transmitted signal.

End events
Indicate the completion of the process. They have no outflow.

Standard
Indicates the completion of the process without adding any behavior.

Signal
Indicates the end of a process and sends a signal to other processes that are waiting to catch it.

Terminal
Indicates that all process activities are immediately finished when the process comes to an end. When any of the paths reaches this event, the entire process is finished.

Intermediate events
Indicate possible event occurrences that take place between the beginning and the end of a process.

Throw Link and Catch Link
These events facilitate the use of diagrams. They represent the connection of two sections of the same process distantly located in the process diagram.

Throw Signal
Used to send a signal that is received by signal catching events, whether they are initial, intermediate or final indistinctly, of the current process or of another process.

Catch Signal
Indicates that the process is waiting for the reception of a signal sent by the same process or another, in order to advance.

Timer
Acts as a mechanism that waits until a certain date or during a specified cycle.

Border event
They are intermediate events that are directly associated with an activity.
These events can be triggered only when the activity to which they are associated is being executed.
The output flows of border events are called exception flows, as they are executed only if the border event is activated.
They are used in particular for managing exceptions and waiting periods with certain deadlines.

- Interrupting events
  When the border event is activated, the execution of the current activity is interrupted and the process continues through the output flow of the activated border event.
  They are represented by a solid line.
- Non-interrupting events
  When the border event is activated, the process starts a path to treat the exception, in parallel with the normal process.
  They are represented by a dashed line.

Signal
This event is waiting for the reception of a signal that can be emitted by this or another process.

Timer
This event is activated at a specified moment or at certain time intervals.

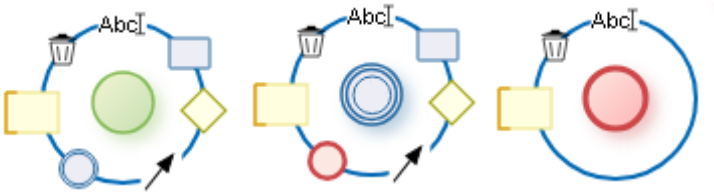
Rule
These type of events represent the execution of a business rule. They are detected by scheduled tasks called "Component-type Event Generators".

Email
These type of events represent the arrival of an email to a certain email account. They are detected by scheduled tasks called EMAIL-type Event Generators.

File
These type of events represent the appearance of a new file in a certain folder. They are detected by scheduled tasks called "FILE-type Event Generators".

Pie Chart Menu

The options presented in the pie chart menu vary depending on whether it is a start, intermediate or end event.



Abc Opens an area to edit the name.

Deletes the event.

Adds an activity.

Adds a gateway.

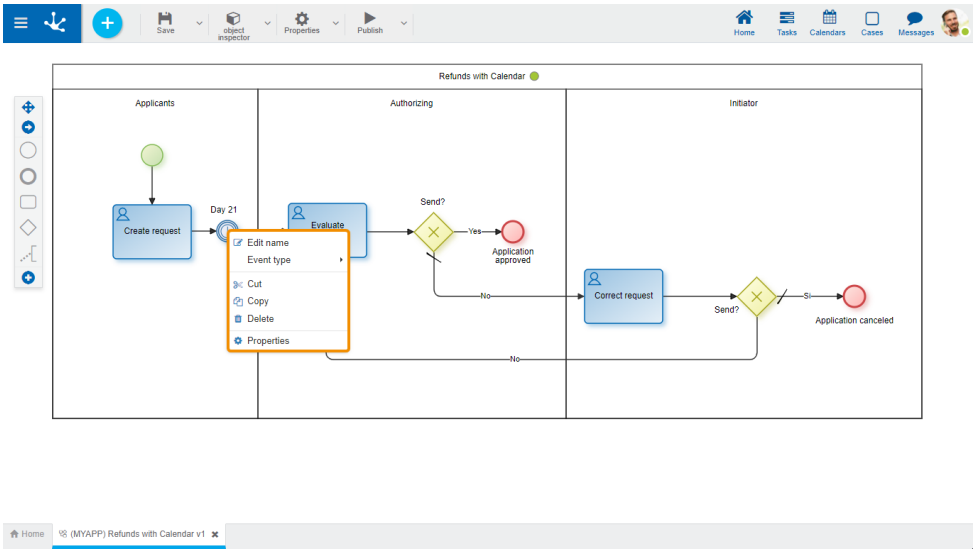
Adds an end event.

Adds a flow.

Adds an intermediate event.

Adds a comment.

Context Menu



- Edit name: Allows to edit the event name. Same functionality as the icon Abc.
- Event type: Allows to define the type of event. Its options vary depending on whether it is a start, intermediate or end event.
- Cut: Deletes the selected event. It can be pasted in another process location.
- Copy: Copies the selected event. It can be pasted in another process location.
- Delete: Deletes the selected event.
- Properties: Opens the properties panel to show and/or modify.

3.6.3.2.6. Gateway

A gateway is incorporated into the model to determine whether the data flow routes within a process branch or merge depending on the definition of conditions.

"Gateways are used to control how sequence flows interact as they converge and diverge within a process." [Workflow Management Coalition]

Types

Exclusive Gateway

An exclusive gateway is where only one of the exit paths continues with the task sequence. Gateways have two representations, although both have the same behavior. By convention, the same representation must be used for the entire definition of a process.

- Exclusive gateway without marker.
- Exclusive gateway with marker.

The possible exit paths of the gateway are set by selecting one of the following options:

- **Definition of conditions**
  It is necessary to define conditions for gateway output flows, which are evaluated sequentially. When a condition is met, the process continues along this path, without evaluating the remaining conditions.
- **Button definition**
  It is necessary to define buttons for gateway output flows. When users press a button, the process continues along that path.

The exclusive gateway must have an output flow defined as default so that the process can continue when no conditions are met.

Parallel Gateway

It is used to create parallel paths and synchronize activities. All exit paths are activated without evaluating the conditions.
Parallel gateways can be defined in pairs, one as a divergence element to activate several parallel paths simultaneously and the other as a convergence element to synchronize previously activated paths. In the latter case, when the gateway is used for synchronization, the output is activated when the input paths are completed, that is, all activities preceding such point have ended.
The output flows of a parallel gateway do not require definition of conditions, since all exit paths must be completed.

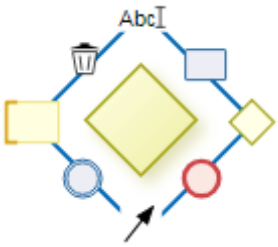
Inclusive Gateway

It is used to activate one or more paths and synchronize activities. All output flow conditions are evaluated.
Inclusive gateways can be defined in pairs, one as a divergence element to generate parallel activities and the other as a convergence element to synchronize previously activated paths. For divergence, at least one exit path must be activated, while for convergence, all activated paths must be completed.

- **Definition of conditions**
  It is necessary to define conditions for inclusive gateway output flows. All conditions are evaluated and the process takes all the paths where conditions are met.
- **Button definition**
  It is necessary to define buttons for gateway output flows. When users press a button, the process continues along that path.The same button can be defined for more than one flow.

An inclusive gateway should have an output flow defined as default, in order to prevent execution errors, when no flow condition is met.

Pie Chart Menu



Abc Opens an area for editing the name.

Deletes the gateway.

Adds an activity.

Adds a gateway.

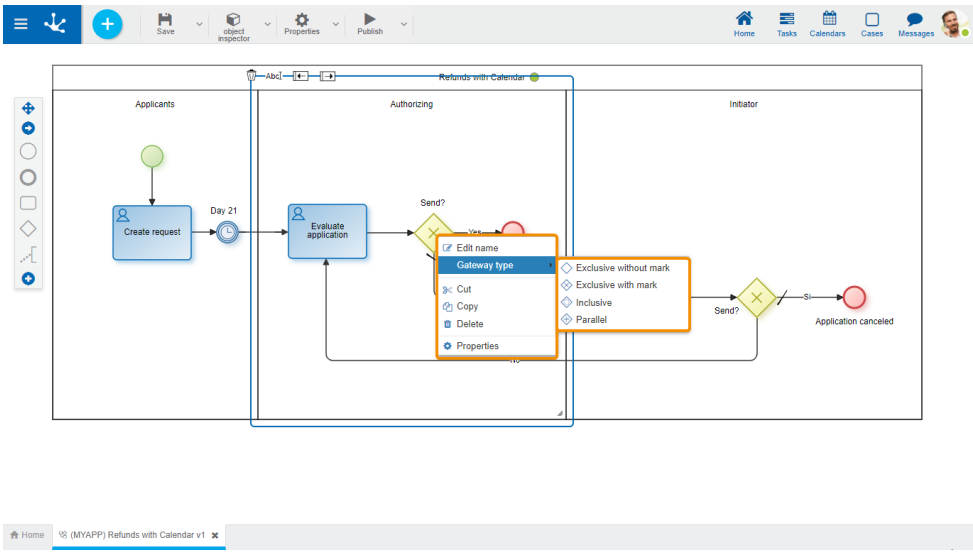
Adds an end event.

Adds a flow.

Adds an intermediate event.

Adds a comment.

Context Menu



- Edit name: Opens an area for editing the name. Same functionality as the icon Abc.
- Gateway Type: Defines the type of gateway.
- Cut: Deletes the selected gateway. It can be pasted in another process location.
- Copy: Copies the selected gateway. It can be pasted in another process location.
- Delete: Deletes the selected gateway.
- Properties: Opens the property panel to show and/or modify.

3.6.3.2.7. Artifact

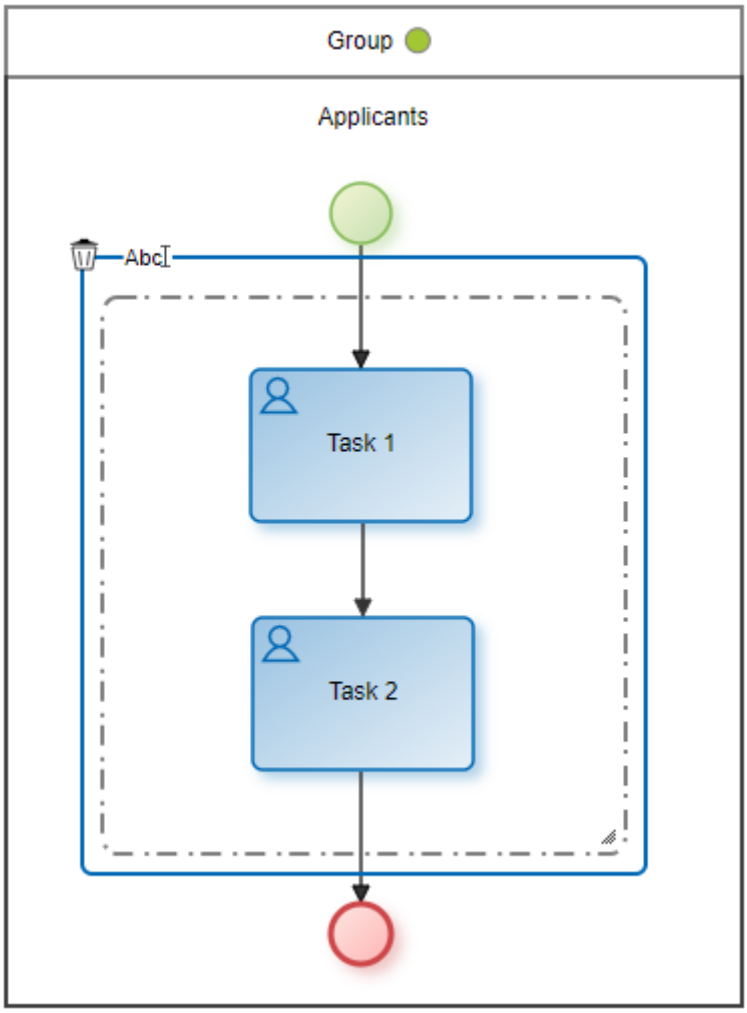
They are shapes that improve the expressiveness of diagrams, representing data related to the process but with no influence on performance.

"Artifacts are used to provide additional information about the process."[Workflow Management Coalition]

Group

Visually groups shapes in the diagram.

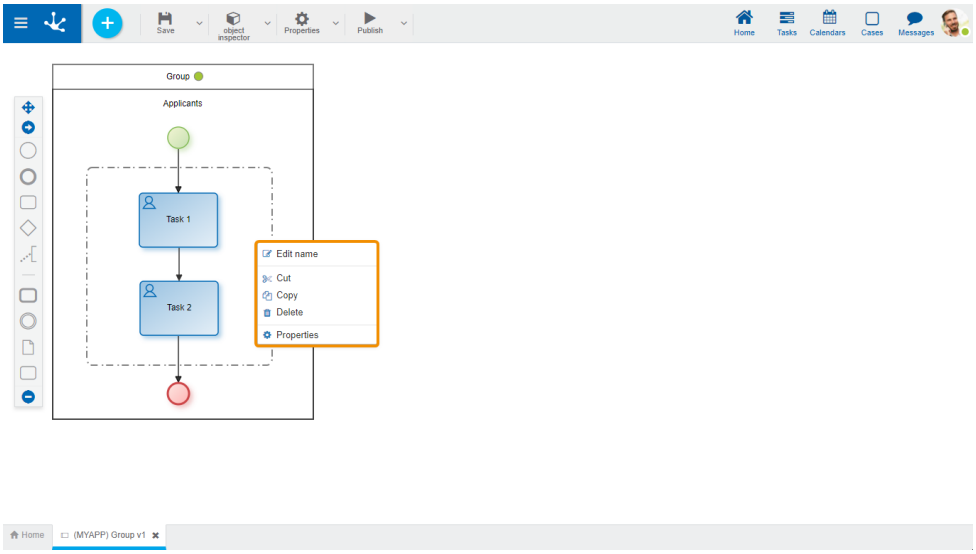
Pie Chart Menu



Abc Opens an area for editing the name.

Deletes the group.

Context Menu

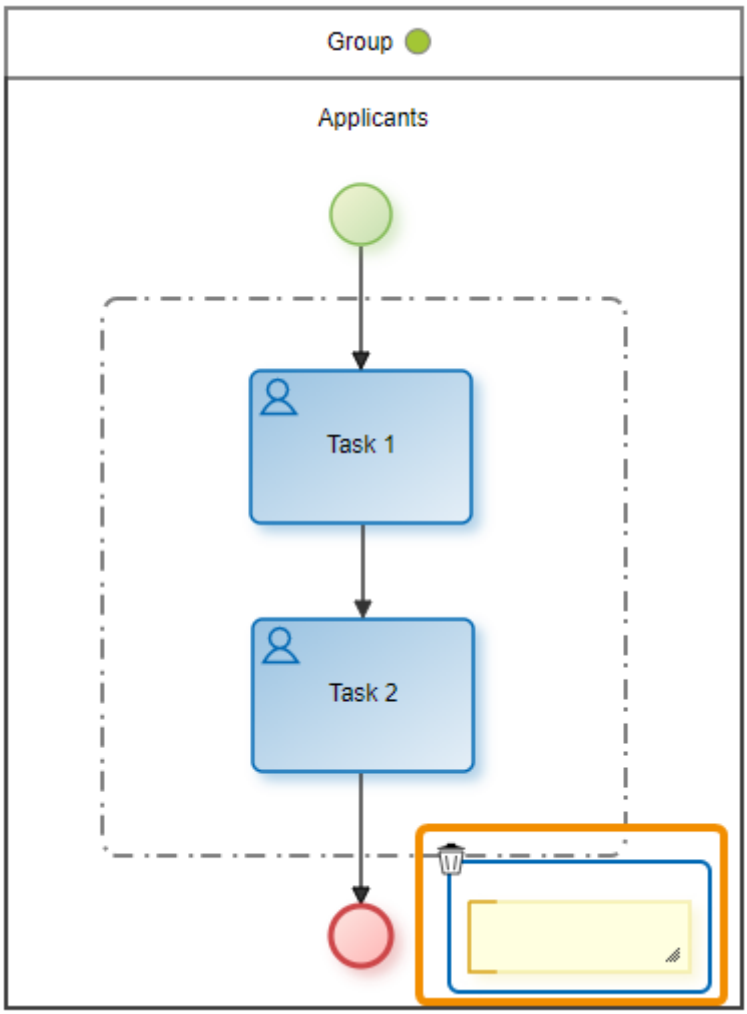


- Edit name: Edits the group name. Same functionality as the icon Abc.
- Cut: Deletes the selected group keeping the shapes it groups. It can be pasted in another process location.
- Copy: Copies the selected group. It can be pasted in another process location. It does not copy the elements it groups.
- Delete: Deletes the selected group keeping the shapes it groups. Same functionality as the icon .
- Properties: Opens the properties panel to show and/or modify.

Annotation

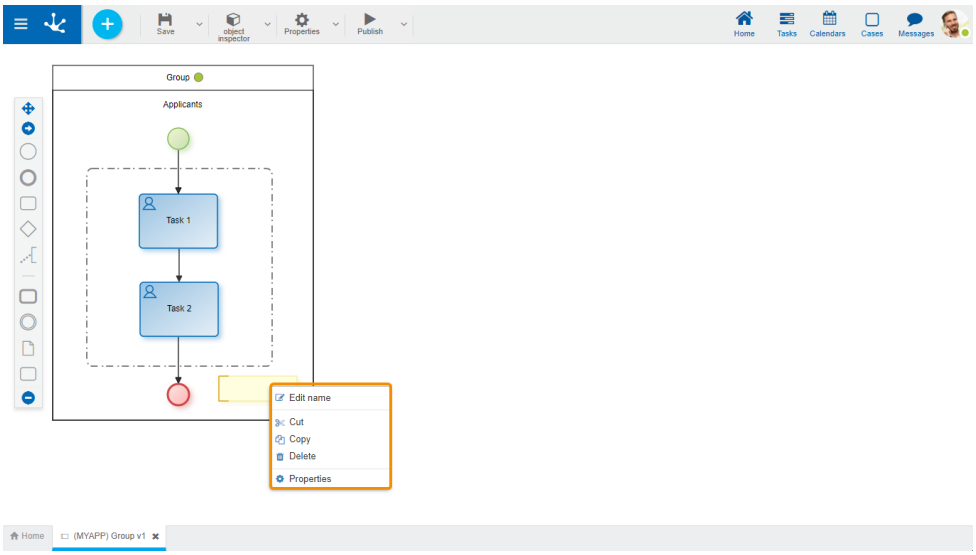
Area where the modeler can write explanatory texts.

Pie Chart Menu



Deletes the selected annotation.

Context Menu

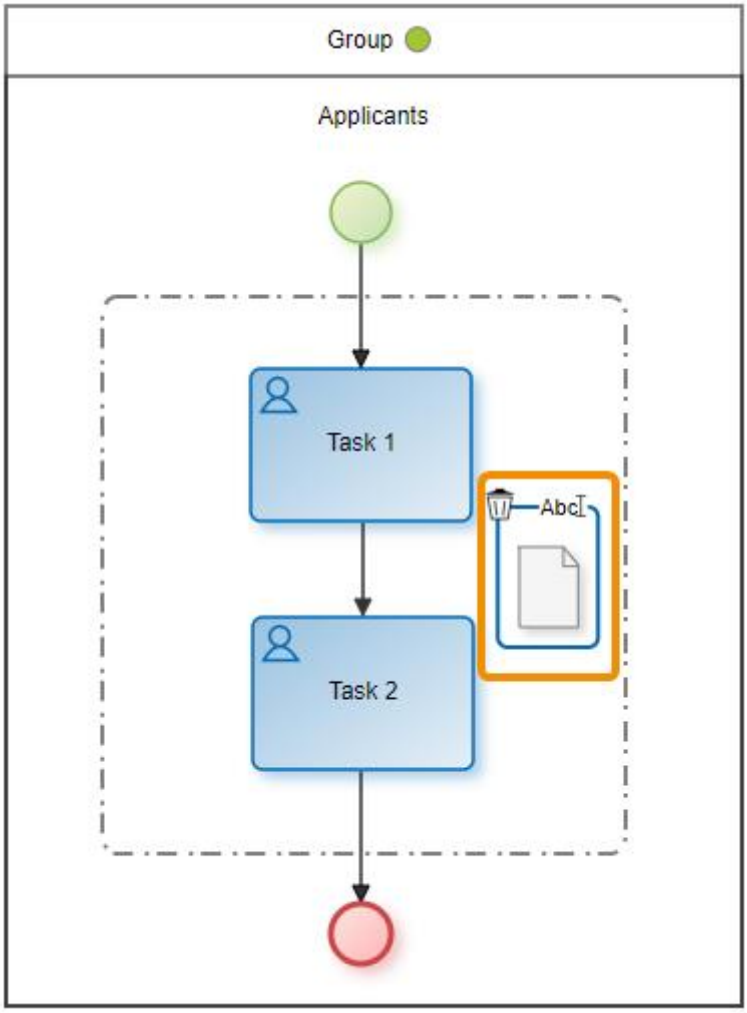


- Edit name: Allows to edit the annotation text.
- Cut: Deletes the selected annotation. It can be pasted in another process location.
- Copy: Copies the selected annotation. It can be pasted in another process location.
- Delete: Deletes the selected annotation. Same functionality as the icon 🗑.
- Properties: Opens the properties panel to show and/or modify.

Data Object

Represents the information that accompanies the process circuit or is generated during it, such as documents or notifications.

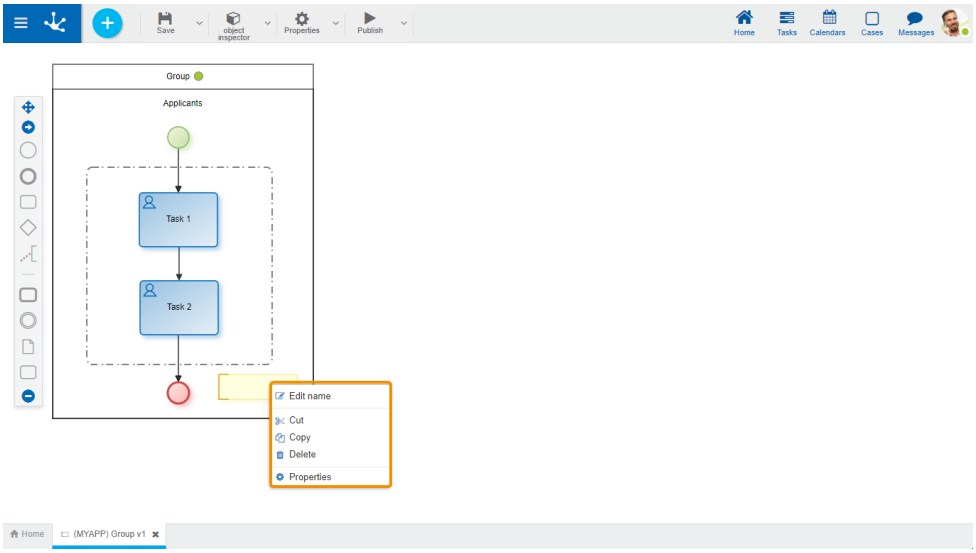
Pie Chart Menu



Abc Opens an area for editing the name.

🗑 Deletes the selected data object.

Context Menu

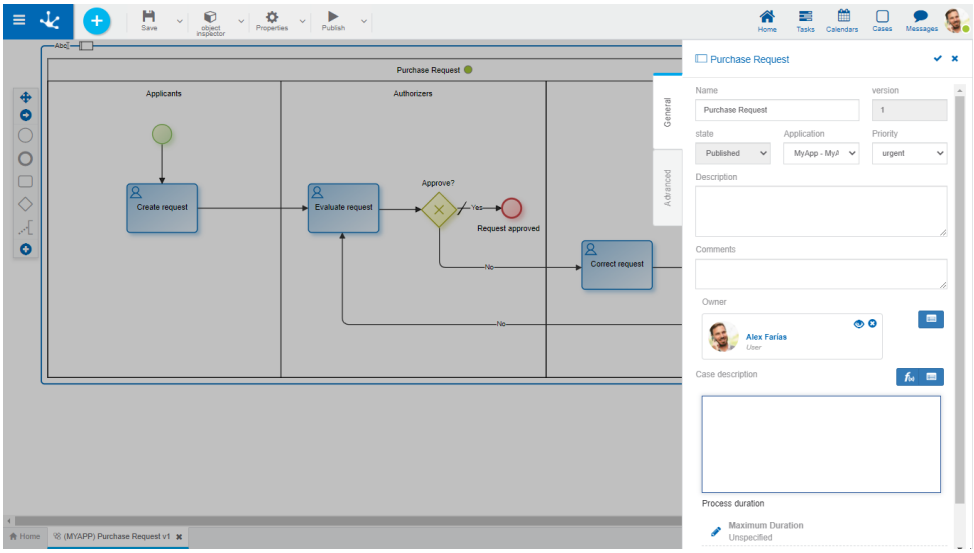


- Edit name: Edits the name of the data object. Same functionality as the icon Abc.
- Cut: Deletes the selected data object. It can be pasted in another process location.
- Copy: Copies the selected data object. It can be pasted in another process location.
-Delete: Deletes the selected data object. Same functionality as the icon .
- Properties: Opens the properties panel to show and/or modify.

3.6.3.3. Process Properties

There are different ways to enter a process properties panel:

- Press the icon from the top toolbar.
- Double click on the graphic element corresponding to pool.
- Access the context menu of the pool graphic element.

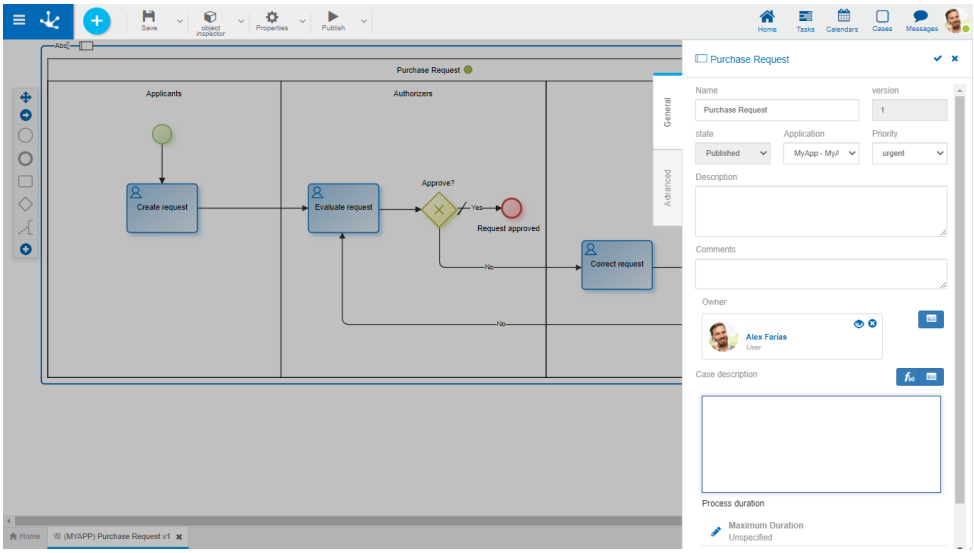


Tabs

- General
- Advanced

3.6.3.3.1. General

The properties panel is displayed on the right side of the process modeler, where the first tab corresponds to general information.



Properties

Name

It is the name used at the modeling level to refer to the process, for example, in the modeler's grid and in the object inspector. Supports multi-language.

Version

Process version number. Along with the Identifier property, they form the unique identification of the process.
This attribute allows to incorporate modifications in the process design without impacting executions that are still active (unfinished cases).
It is a non-modifiable attribute. The modeler user can version the process from the top toolbar.

State

Indicates the current state of the process. It is automatically updated when start, modify, save, publish and inactivate operations are executed.
The state of the process determines the level of validations that are carried out on it and the possibility of being used.

Application

Allows to define the application to which the process belongs. If no application is reported, the process is assigned to the GLOBAL application.

Priority
Indicates the priority assigned to the process activities, when they do not have an expressly defined priority. Possible values are URGENT, HIGH, MEDIUM, LOW.

Description
Text that defines the process describing its functionality. Supports multi-language.

Comments
Observations on the process.

Owner
Indicates the person who has control and vision of the process from start to finish. The owner can be a user, an organizational unit, a role, or an agent.
It is, by default, the logged in user.

Opens the profile show property.

Deletes the informed owner

Allows to select an existing owner

You can also use the autocomplete function. As characters are entered into this field, the first five matches dynamically appear to select one of them.

Case Description
It is a description that represents each one of the process executions.

Allows to select functions from a list to incorporate dynamic information into the case description.

Allows incorporating attributes of the entities that are used in the execution of the process to the case description. For example, in the "Refund Request" process, the request date can be incorporated, which is a form variable used in such process.

The description entered is displayed in:

- The last tasks, by pressing the Tasks icon on the top toolbar.
- The task execution panel, below the activity name.
- The Description column in the tasks grid.
- The last tasks, on the element that represents the form in the grid of "Forms and Tasks".

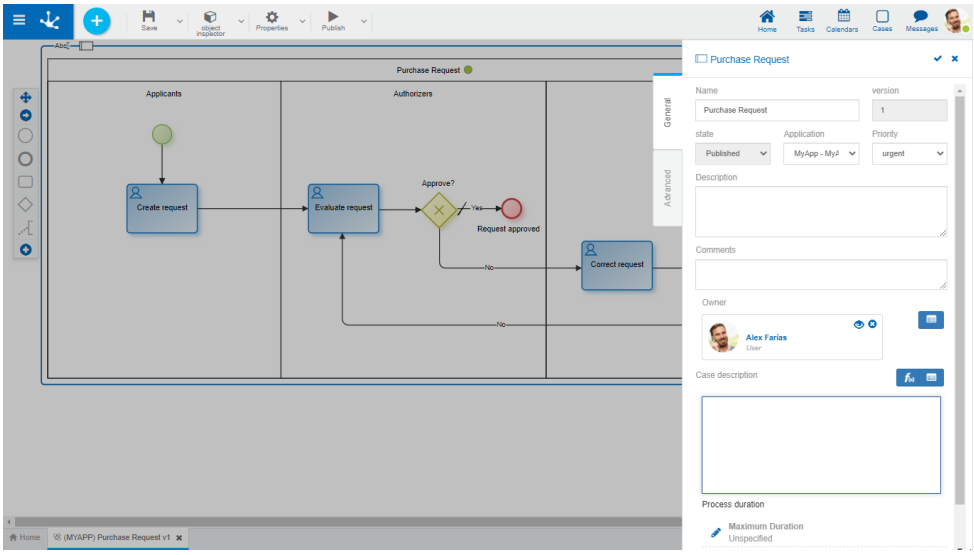
Process Duration
Allows to set the Maximum duration and Average duration to carry out the process.

Allows to edit the maximum and average duration.

- By duration
  A deadline in days, hours, minutes and seconds is reported.
- By due date
  A date-type variable is selected, from a form associated with the process, using the icon
  In the process execution, the value of such variable is used as due date. If the value of the variable used to define the due date is modified, process duration is automatically updated.

Saves changes made to duration.

Cancels changes made to duration.



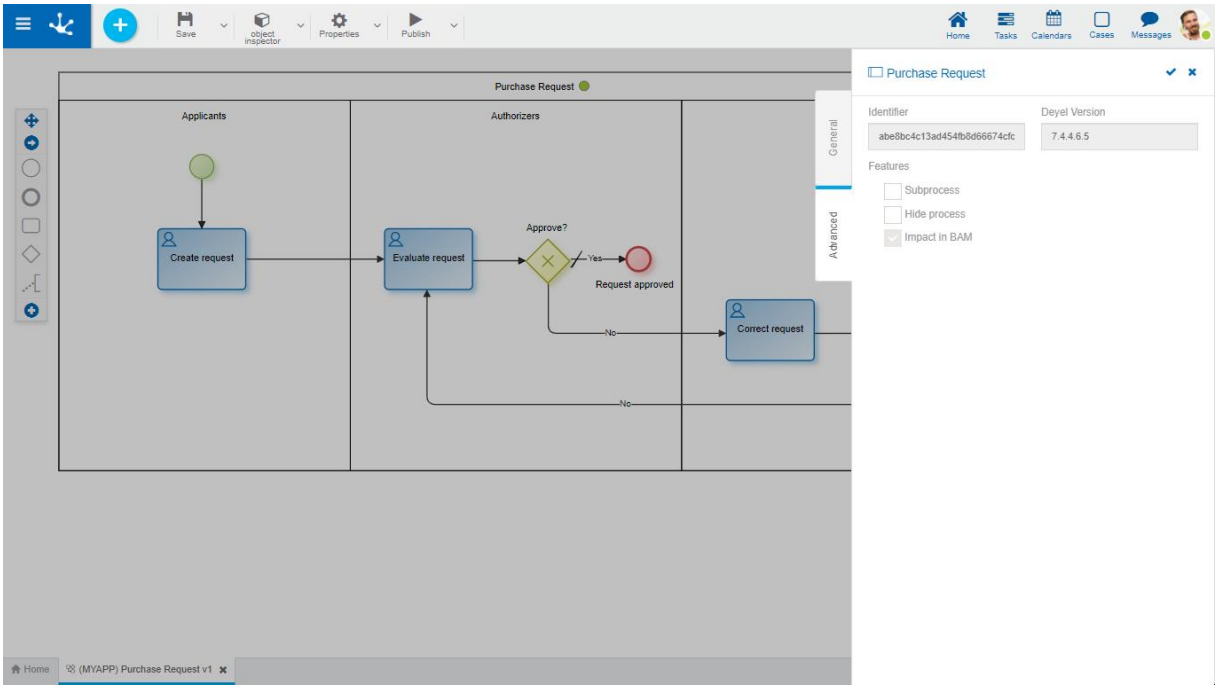
Actions

The icon ✔ is used to confirm the modifications made in the properties panel.

The icon ✖ is used to close the properties panel, if it was not previously saved, changes are discarded.

3.6.3.3.2. Advanced

The second tab of the side panel corresponds to advanced properties of the process.



Properties

Identifier
Alphanumeric code that is automatically assigned to the process. Users cannot modify it. Along with the Version property, they form the unique identification of the process.

Deyel Version
Indicates a version by **Deyel** with which the process was last saved or published.

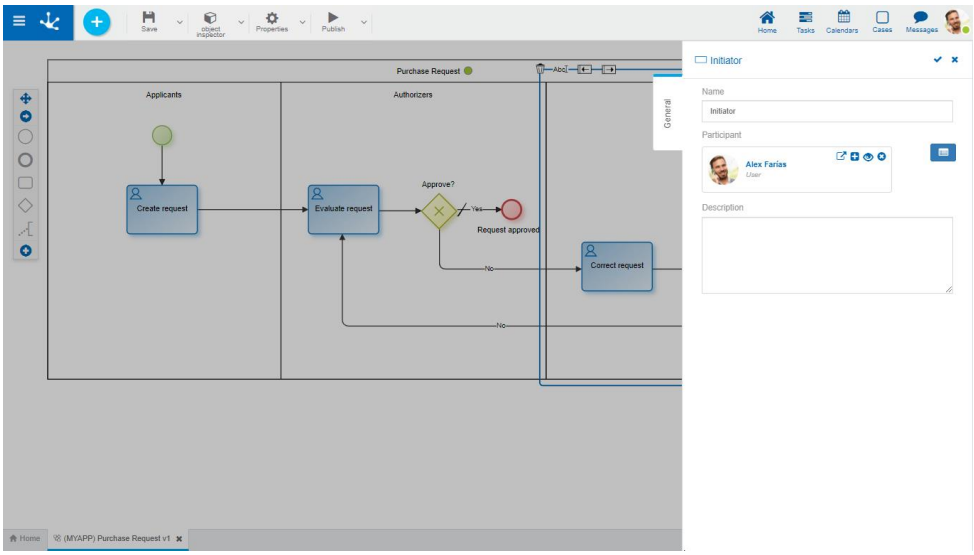
Characteristics
One of the following options can be indicated:

- Subprocess: Indicates that a process is defined to be used exclusively as a subprocess of others. It can only be started when it is invoked from another process in a Subprocess activity. It cannot be started by a user manually from the functionality "Start Cases".
- Hide process:Indicates if a process remains visible in the Context Menu and from Forms and Tasks so that users can start it.
- Impacts on BAM: Indicates if process data are taken into account in BAM reports.

3.6.3.4. Lane Properties

There are different ways to enter the properties panel of a lane:

- Double click on the graphic element corresponding to the lane.
- Access the context menu of the lane graphic element.



General Tab

Properties

Name
Lane name. It must clearly represent those who perform the tasks within the lane.

Participant

It is the participant in charge of executing the activities included in the lane.

Allows to select an existing participant, which can be a user, an organizational unit, a role, or an agent. You can also use the autocomplete formula. As characters are entered into this field, the first five matches dynamically appear to select one of them.

Allows to display a window with complete information on the participant, if he is defined as a user, organizational unit or role.

Opens a window to define a new role or invite an external user.

Opens the profile show panel only if it is a user.

Deletes the informed participant.

External users can be invited to participate by providing their email. If more than one user is invited, the generation of a new role starts automatically.

Description

Conceptual description of who executes the tasks within the lane.

Actions

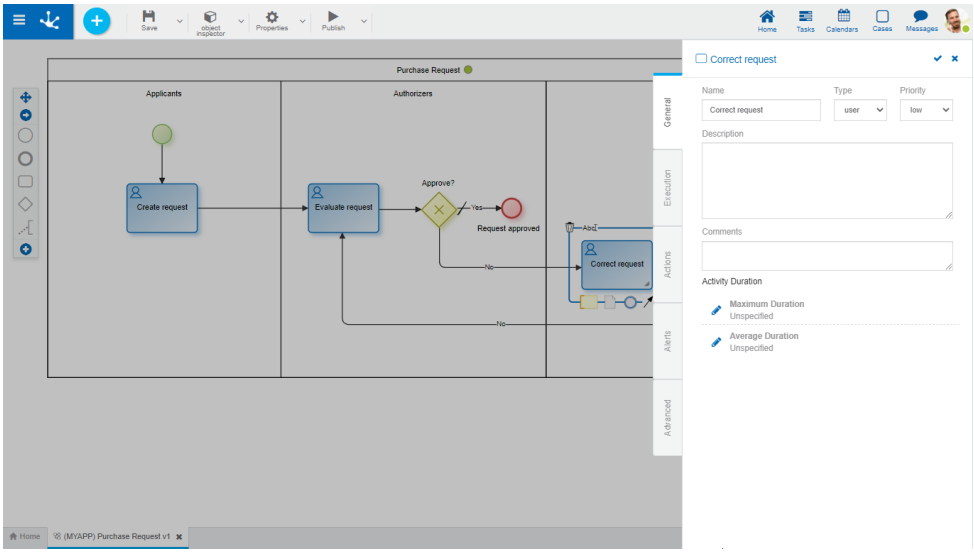
The icon is used to confirm the modifications made in the properties panel.

The icon is used to close the properties panel, if it was not previously saved, changes are discarded.

3.6.3.5. Activity Properties

There are different ways to enter the properties panel of an activity:

- Double click on the graphic element corresponding to the standard activity or task.
- Access the context menu of the activity.



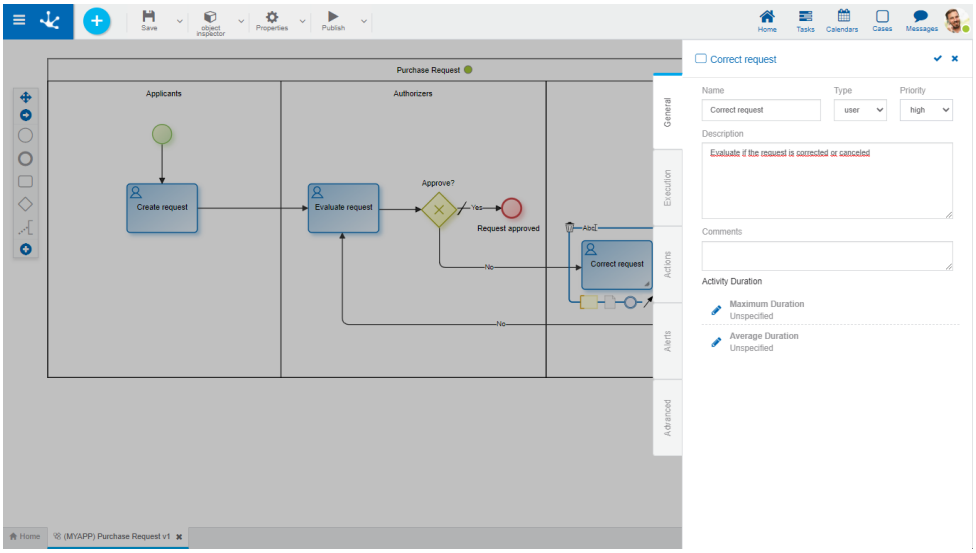
Tabs

- General
- Execution
- Actions
- Alerts
- Advanced

The business modeler user can have access to all tabs except for "Advanced". The IT modeling user can access all of them.
For each of the "General", "Execution", "Actions" and "Alerts" tabs, access permissions are defined from the security module. Users can see only the tabs they are allowed to access.

3.6.3.5.1. General

The properties panel is displayed on the right side of the activities modeler, where the first tab corresponds to general information.



Properties

Name
Activity name. It refers to the action performed by the activity.

Type
Defines the type of activity.

Priority
Indicates the activity priority. Possible values are URGENT, HIGH, MEDIUM, LOW.
The user can display this property in the to-do list and use it as a sort criteria, according to their preference.

Description
Description of the activity that can be used as an online manual.
The user who executes the activity can see it by clicking on the icon .

Comment
In this field, the designer can document in detail what the task consists of.

Activity Duration
Allows to set the Maximum Duration and the Average Duration of the activity.

When the execution of the activities begins, these durations are transformed into a maximum due date and an expected expiration date, respectively. The activity due date calculation is made in different ways, depending on whether the definition of maximum duration is made according to duration or due date.

In the case show or in the tasks show, the due date and time of the task are displayed and in addition, on these durations, alerts can be set.

Edits the maximum and average duration.

- By duration
  A deadline in days, hours, minutes and seconds is reported.
- By due date
  A date type variable is selected from a form associated with the process, using the wizard icon .

In the activity execution, such variable value is used as due date. If this variable value is modified, the due date of the activity is automatically modified.

Actions

The icon is used to confirm the modifications made in the properties panel.

The icon is used to close the properties panel, if it was not previously saved, changes are discarded.

3.6.3.5.1.1. Activity Due Date Calculation

The activity due date is calculated in different ways, depending on whether the maximum duration was modeled from a duration or by selecting a form field related to the process.

> *The calculation detail is made for the due date but it is also valid for the average date, depending on the modeling of the average duration of the activity.*

Due Date Based on a Duration

Elements involved in the calculation:

- Activity start date and time.
- Activity maximum duration.
- Calendar of the participant who executes the activity.

- Configuration of holidays and special dates.

Calendar of the Participant Who Executes the Activity

When the participant is a user:

- The calendar established in the user's profile is considered.
- If not indicated, the calendar established in the user's organizational unit is considered.
- If the unit does not indicate a calendar, it moves up in the hierarchy, searching for a higher unit to establish a calendar.
- If no higher unit indicates a calendar, the standard calendar is considered.

When the activity is executed in an organizational unit:

- The calendar established in the organizational unit is considered.
- If the unit does not indicate a calendar, it moves up in the organizational structure, searching for a higher unit to establish a calendar.
- If no higher unit indicates a calendar, the standard calendar is considered.

When the activity is assigned to a generic role:

- The standard calendar is considered.
- When role players assign themselves an activity, the due date is recalculated considering the calendar of the executing participants.

Steps for Activity Due Date Calculation

Step 1: The maximum duration of the activity is calculated, expressed in seconds.

The number of days, hours, minutes and seconds, indicated in the activity definition, is considered.

The number of hours and minutes of a working day is retrieved from the executing participant's calendar.

The maximum duration of the activity is calculated as:

Maximum Duration in Seconds =
Activity.Days * (Calendar.Hours * 3600 + Calendars.Minutes * 60) +
Activity.Hours * 3600 +
Activity.Minutes * 60 + Activity.Seconds

Step 2: The user's working times are defined, through the executing participant's calendar, together with the definition of general holidays and specific dates.

Each of the work bands is perfectly defined by the calendar, and has a start date and time and an end date and time, therefore, this specification can be converted to a number of work seconds.

The definition of "Holidays" means that certain periods are considered non-working days.

The definition of "Specific Date" allows to establish specific working times for a specific date.

The definition of specific dates that is performed at the calendar level takes precedence over the general specific dates.

Step 3: Express the Start Date and Time of the activity, in the time zone of the executing participant's calendar.

Based on that date, working times are assigned until all the seconds that correspond to the activity duration are assigned.

In this way, the activity due date is calculated, according to the user's working times.

Due Date Based on a Form Field

The due date takes the value of a form field.

If the field data type is Date or Local Date, if the value entered is MM/DD/YYYY, the due date is MM/DD/YYYY at 23:59:59.999

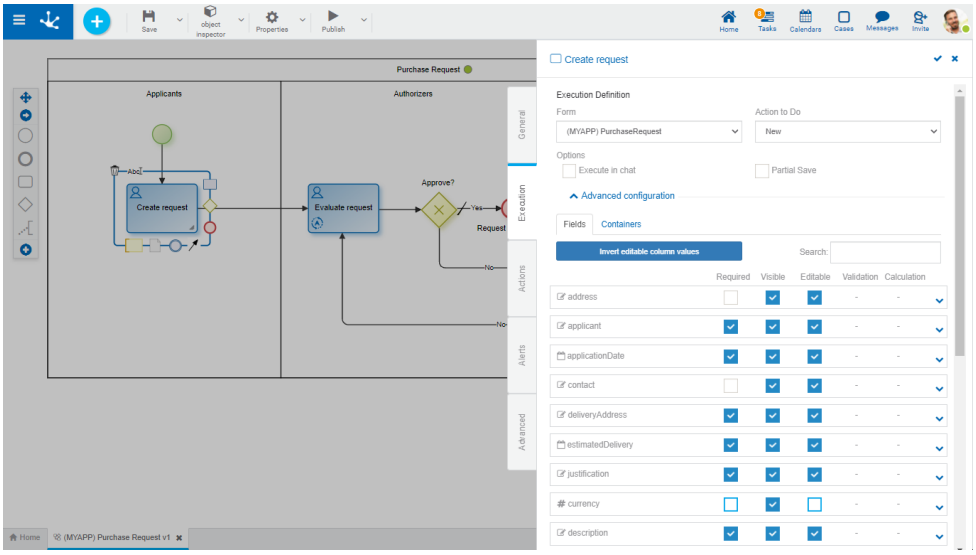
If the field data type is Date and Time, the DD/MM/YYYY HH:MM:SS value is considered as the due date.

If the field type is Local Date and Time, the DD/MM/YYYY HH:MM:SS value is transformed to the standard calendar time zone and is considered as the due date.

3.6.3.5.2. Execution

Process modeling > Modeling the execution of a process

The second side panel tab of an activity corresponds to execution information and it is displayed only for manual, user or undefined activities.
Allows modeling users to define behaviors of the activity and form fields, according to the action performed. There must be at least one form previously associated with the process.



Execution Definition

In this section, the general properties of the activity execution are defined.

Form

Displays the name of the form related to the process used in the activity.

Action to Do

Action performed on the form in this activity.

- **Open**
  Displays the form instance in show mode. Fields cannot be edited.
- **Create**
  Creates a new form instance that is related to the case. When there is an instance already created, it works like a "Modify" action.
- **Modify**
  Displays the existing form instance and allows to modify the fields defined as editable.

Execute on Chat

This option allows the task to be executed not only from the task icon on the top toolbar, but also from the **Deyel** chat.
Selecting this check expands the properties panel to set the actionable message.

- Who will send the message?
  Indicates a **Deyel** user that sends the message. Allows to use the autocomplete function to select it.
- Message content
  The text that makes up the body of the message is reported, this text can contain variables and functions. To select them you can use their wizards, of variables and of functions.
- What fields can the user complete?
  Using the "Drag and Drop" function, fields are moved from the form fields area to the fields to complete area.
  The fields to be completed must be entered by the user in the chat.

Save Partially

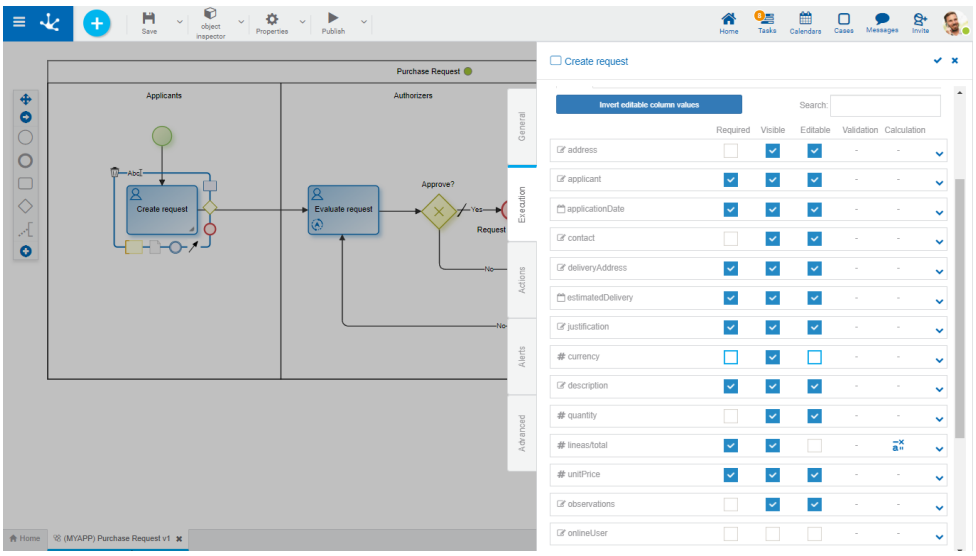
This option indicates that when executing the activity, the "Save" button is added, which allows to keep the values entered by the user, with no need to execute the activity.

Advanced Settings

To show this section, an action to do must have been previously defined.

Fields

In this tab, a grid with the form fields is displayed. The properties of these fields can be redefined for the execution of each activity.



Inverts the values of the Editable property, for all the fields displayed in the grid. If it is narrowed by search filter, the modification affects only the filtered fields.

As characters are entered to the search field, it dynamically filters the fields whose Label or Name properties, indistinctly, match the text being entered.

Fields Grid

It is made up of form fields according to the filter applied. Each field is identified by its property Name .

The grid columns correspond to the modeled behavior properties, whether they have been defined from the form modeler, or from the activity itself.

Properties are updated simply by checking or unchecking them and if they have embedded rules defined, they are displayed by icons. Checks and icons are displayed in blue if properties or rules were modeled in the form, while they are in light blue if they were modeled in the activity.

If there are properties or rules modeled from the form and any of them are modified from the activity, these modifications overwrite the definitions of the form only for the task execution.

> *If any property or rule of the form is overwritten from an activity, subsequent modifications made from the form have no impact on the behavior already defined in the activity.*

> *If a field is added to the form, it is added to the activities that use the form, as non-required and non-editable, regardless of how it has been modeled on the form.*

Required

Indicates if the field is required during the activity execution and if any required rule is defined, it displays the icon .

Visible

Indicates if the field remains visible in the activity and if any visibility rule is defined, it displays the icon .

Editable

Indicates if the field is editable for the activity and if any editability rule is defined, it displays the icon .

Validation

Indicates if there is a validation rule for the field with the icon .

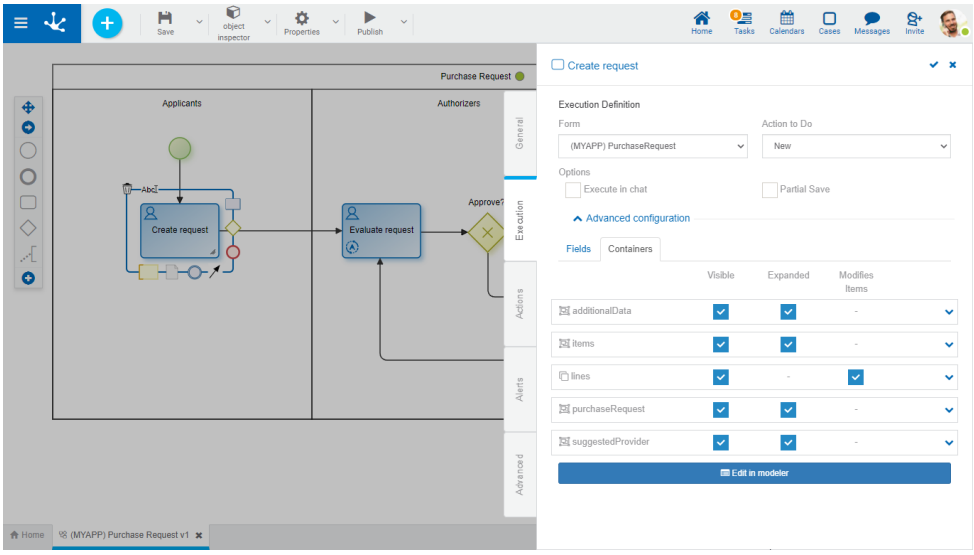
Calculation

Indicates if there is a calculation rule for the field with the icon .

> *Pressing the icon expands the area of values and rules modeling of each field.*

Containers

In this tab, a grid with the form containers is displayed. The properties of these containers can be redefined for the execution of each activity.



The grid columns correspond to the properties of containers.

Visible

This check indicates that the container remains visible when executing the activity. If it has associated visibility rules, it is represented by the icon .

Expanded

This check indicates that the container is displayed when executing the activity.

Modifies Item

Only for iterative containers. This check indicates if iterative occurrences can be added or deleted. This property is displayed by default as checked, however if an iterative container is added to the form in this tab, this property is unchecked.

> *Pressing the icon the embedded rules modeling area is expanded for each container.*

Edit in Modeler

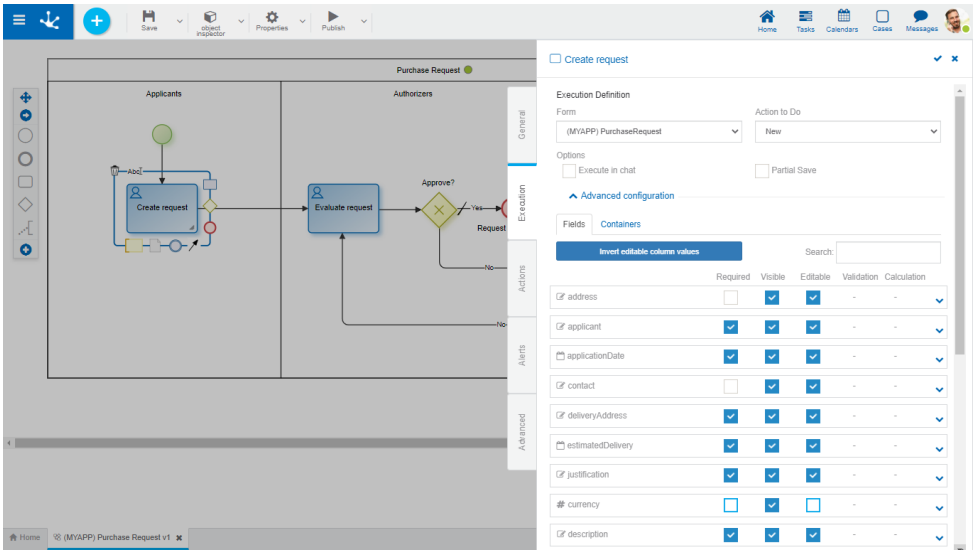
Pressing this button Edit in modeler the form associated with the activity is opened in another tab of the modeler, where the properties panel of each form element can be opened.

> *Pressing the icon go back of the top toolbar, returns to the activity execution tab.*

3.6.3.5.2.1. Values and Rules Modeling Area

From this area, behavior, validation and calculation rules are defined and default values of the fields are established.
Properties and rules that are defined in the activity overwrite those that were defined in the form fields.

Field Properties Panel



Properties

Rules

It is possible to define rules of behavior, validation and calculation, associated with a field for the activity.

Required

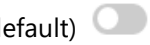
Indicates whether the field is required during the execution of the activity.

Required Not required (default)

Rule + Opens an editing area where a rule to determine the required condition can be defined by using the wizard (ctrl + space). If a rule is defined, the icon is displayed with light blue borders.

Visible

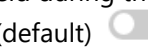
Indicates whether the field is visible. If this property is not selected, the field is not displayed during the execution of the activity.

Visible (default)  Not visible

Rule + Opens an editing area where a rule to determine the visibility condition can be defined by using the wizard (ctrl + space). If a rule is defined, the icon is displayed with light blue borders.

Editable

Indicates if the field is editable. If this property is not selected, the user cannot enter or modify values in the field during the execution of the activity.

Editable (default)  Not editable

Rule + Opens an editing area where a rule to determine the editability condition can be defined by using the wizard (ctrl + space). If a rule is defined, the icon is displayed with light blue borders.

Validation

Rule + Opens an editing area where the condition that determines if the field value is correct can be defined. It is possible to define more than one rule. If rules are defined, the icon is displayed with light blue borders.

Calculation

Rule + Opens an editing area where the expression to be executed to calculate the field value can be defined. If a rule is defined, the icon is displayed with light blue borders.

Persists

Indicates whether the information entered in the field is kept in an activity. This property is only available to IT modeler users.

Persists (default) Does not persist

The value of this property is presented as Does not persist when the field meets all the conditions mentioned below:

- It is not editable and has no associated editability rule
- It has no default value property defined
- It has no associated calculation rules
- It has no relation defined

If the IT modeler user needs to use scripting to update the field content in the activity, they must indicate this property value as Persists. In this way, the field value updated by scripting is not lost when the case moves on to the next activity.

> *If form rules or properties are overwritten from an activity, subsequent modifications from the form for the* ***overwritten properties****, have no impact on the behavior already defined in the activity.*

Undo changes

Allows to delete rules or behavior properties modeled from the activity. In this way, behavior properties and rules applicable in the execution of the activity are again those originally defined in the form.

Shows syntax examples to write rules.

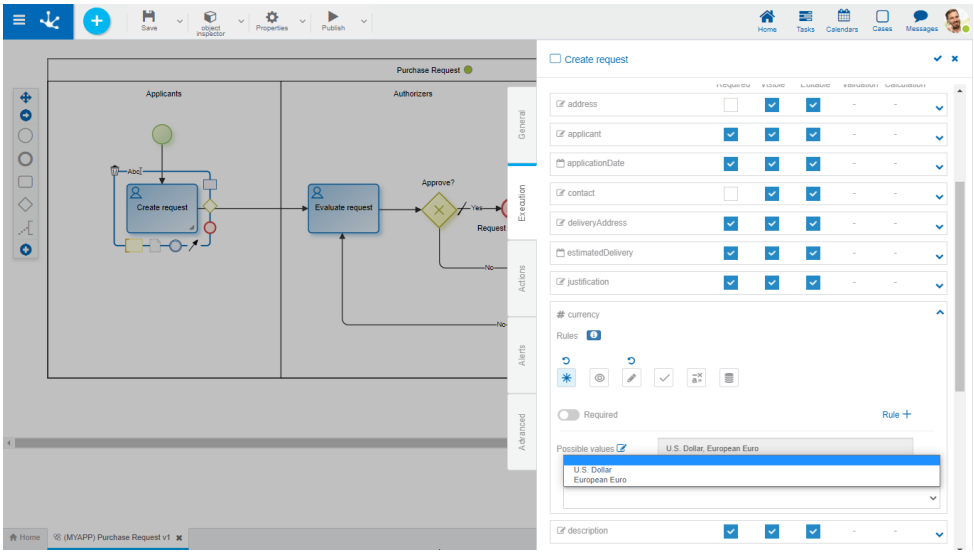
Default Value

It is an area to define the field default value.

Allows to select functions from a list to incorporate dynamic information into the field.

Allows to incorporate information of entities that participate in the process execution to the field content. For example, in the Refund Request process, the request date can be incorporated, this data is available in the date field of the Refund Request form.

Field Properties Panel with Relation



If the form field has a relation the possible values and the default value of the field in the activity are displayed.

Possible Values
Clicking on the icon enables a panel to filter data from the list in the activity.

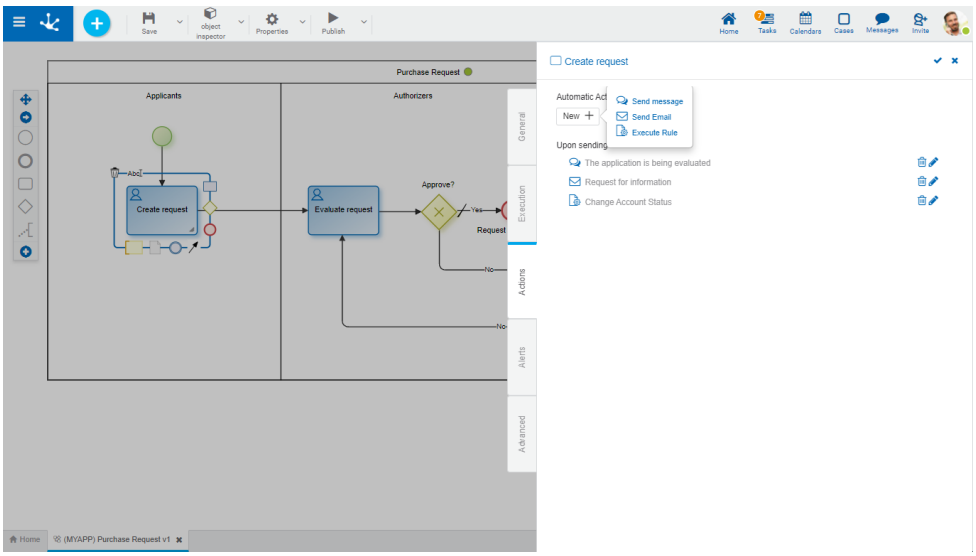
Invert Selection Inverts the selected values.

Select all Selects all values.

Exclude Values: Allows to exclude the selected values from the selection.

3.6.3.5.3. Automatic Actions

In the "Actions" tab, automatic actions executed in the activity are defined.



New + Pressing this button you can create new automatic actions of different types.

Types

Send message

Send Email

Execute rule

For existing actions, the icon allows editing a previously defined automatic action and the icon deletes it.

Properties

The different types of automatic actions share properties and have other specific ones.

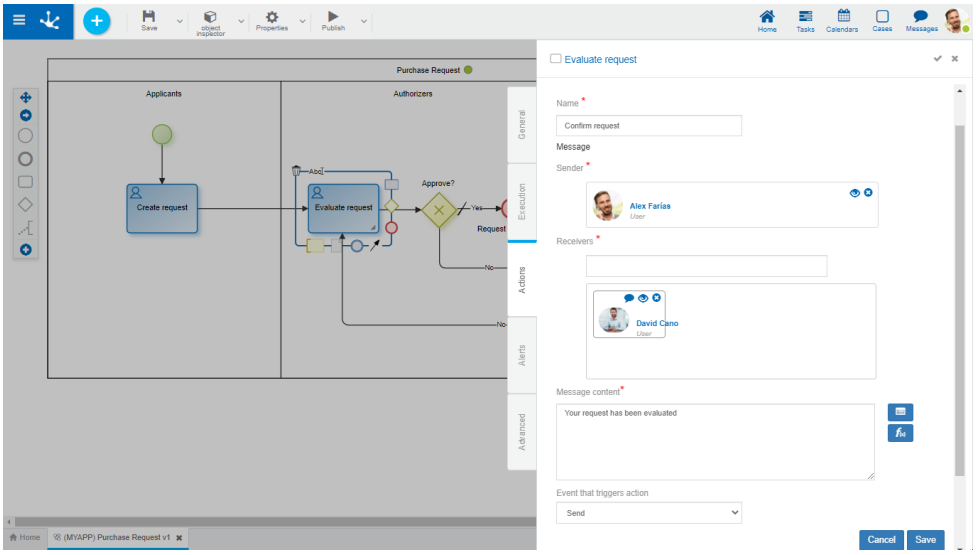
Name
Name that identifies the automatic action in the activity and is used in the description of the activity execution in the graphic show of a case.

Event that triggers the action
Moment when the automatic action is executed, defined by one of the following events:

- Send: Indicates that the action is executed when the task execution finishes.
- Receive: Indicates that the action is executed when the task execution starts.
- Disabled: Execution of the action is disabled.

3.6.3.5.3.1. Send Message

Allows to define the sending of a message via the chat of **Deyel**. The message may include process-related form variables and functions that allow to incorporate links to case shows, to do lists, forms, application access links, among others.



> *An asterisk "*" on the label indicates that the property is required.*

Properties

In addition to the properties shared with email sending and execution of rules, message sending actions have specific properties.

Sender

Indicates a **Deyel** user that sends the message. Allows to use the autocomplete function to select it.

Receivers

Indicates **Deyel** users that will receive the message. It is possible to select users or agents that return users at the time of execution. In both cases the autocomplete facility can be used to select them.

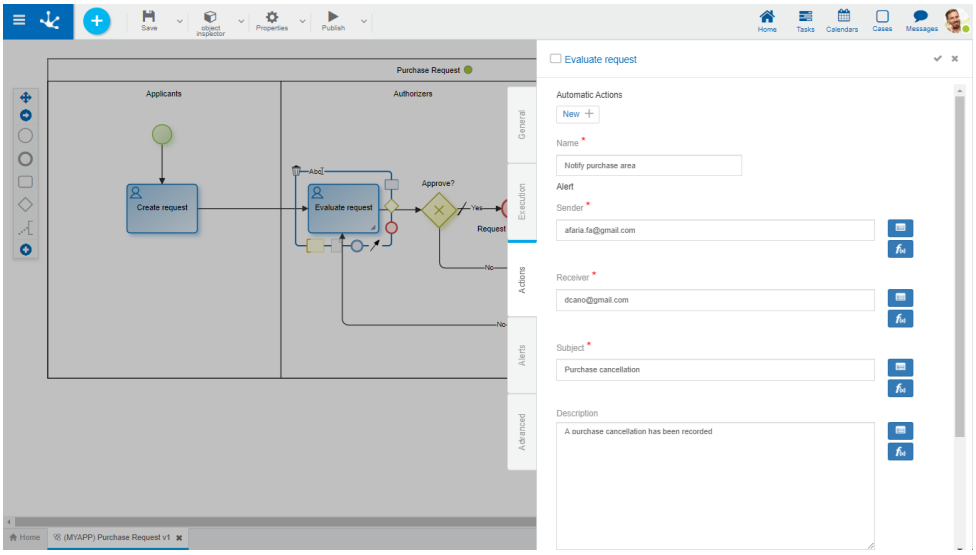
Message content

The text that makes up the body of the message is reported, this text can contain variables and functions. To select them, use their wizards of variables and functions.

> *For both senders and receivers, once the user has been selected, the chat facility can be used if the user is different from the one that is connected and the facility to display the user's profile .*

3.6.3.5.3.2. Send Email

Allows to define email sending. They can include process variables and functions that allow to incorporate links to case shows, to do lists, forms, application access links, among others.



An asterisk "" on the label indicates that the property is required.*

Properties

In addition to the properties shared with email sending and execution of rules, message sending actions have specific properties.

In all properties, text can be combined with values obtained through the variables wizard or the function wizard of **Deyel**.

Sender

Indicates the email address from which the message is sent. If the email server is Google or Microsoft, it is not necessary to complete this property and the account entered in the environment property User or account to establish connection will become the sender.

Receiver

Indicates the email address to whom the message is sent. More than one address can be informed separated by ";".

Subject

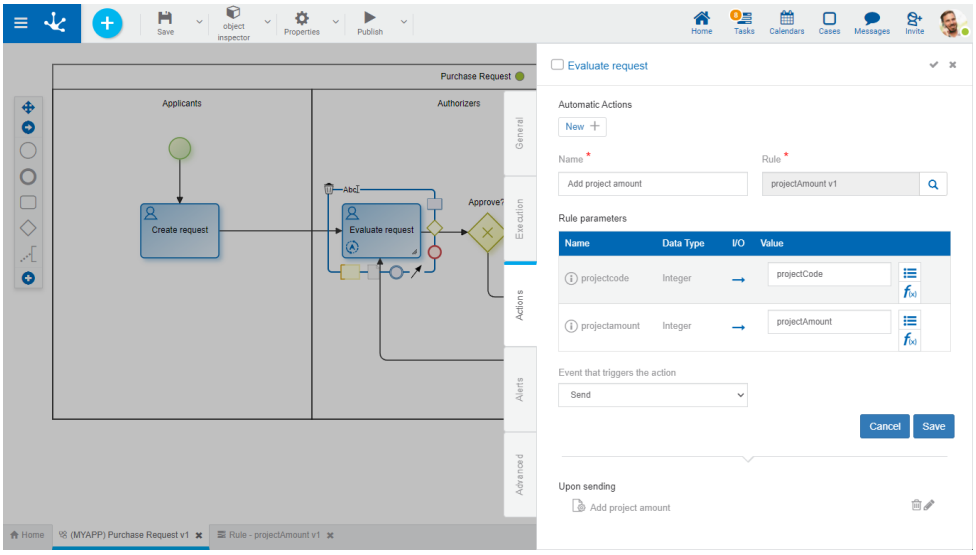
Contains the subject of the email that is being sent.

Description

The text that makes up the body of the email is reported. Html tags can be included to format the message.

3.6.3.5.3.3. Execute Rule

Allows to define the execution of an advanced rule of **Deyel**.



> *An asterisk "*" on the label indicates that the property is required.*

Properties

In addition to the properties shared with message sending and email sending, rule execution actions have specific properties.

Rule

Name of the advanced rule assigned to the automatic action. When creating the action, the rule is selected from a grid that contains the previously published rules.

Rule parameters

Depending on the selected advanced rule, a grid with its modeled parameters is enabled. For each parameter, report the properties:

Name

Parameter name as defined when modeling the advanced rule.

Data Type

Indicates the data type with which the parameter was defined. It can be any type of Java object allowed in **Deyel**. The Java object type can be displayed in full form, as a help, by placing the mouse over the data type in each grid element.

I/O

Specifies whether the parameter is input → or output ←.

Value

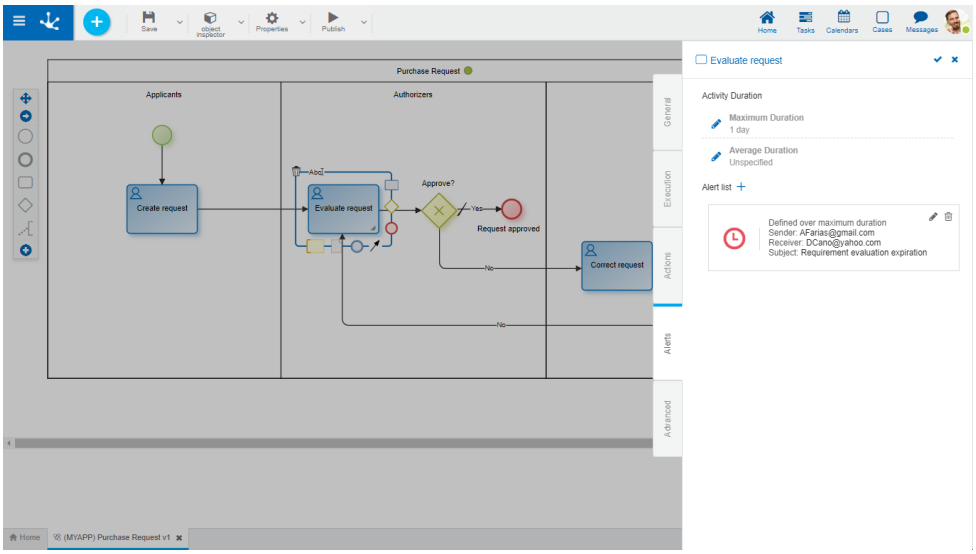
If the parameter is input, it corresponds to the initial value taken by the parameter, by entering a fixed value or selecting it from the wizard of variables or functions. This value must match the one defined for the parameter.

If the parameter is output, it corresponds to the form field that will contain the value returned by the rule and can be selected by using the wizard of variables or functions.

> *The parameter show is available from the "i" icon.*

3.6.3.5.4. Alerts

For each standard activity or task, the IT modeler can define the issuance of electronic notifications by sending emails based on the activity execution times. They are defined in the "Alerts" tab of the activity properties panel.



Properties

Activity Duration

This section displays the Maximum Duration and the Average Duration that have been informed in the "General" tab. Alerts definition is performed considering any of these durations, as indicated in the property Define alert on.

Click on the icon to edit the duration and modify it.

Alerts List

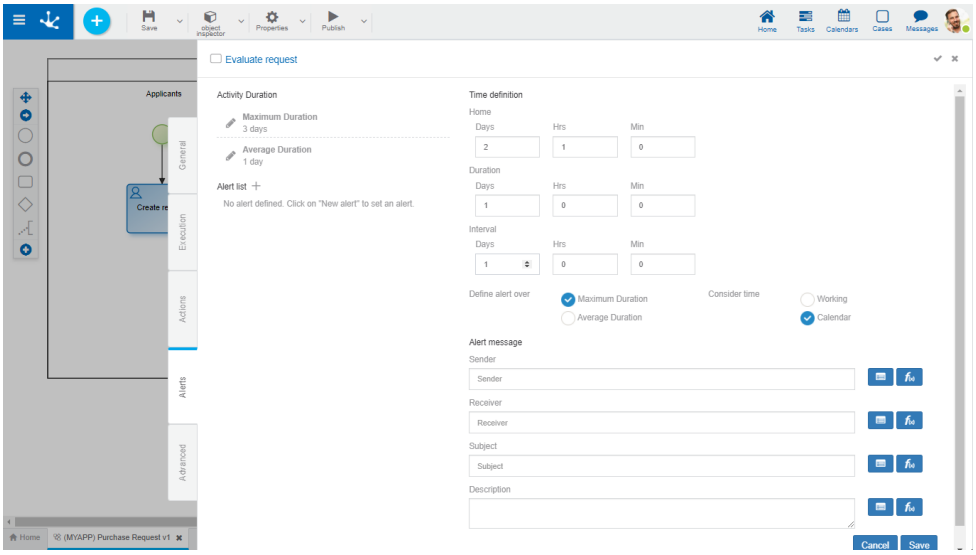
Displays the list of alerts set for the activity.

Indicates that the alert is set based on average duration.

Indicates that the alert is set based on maximum duration.

In order to define alerts, it is necessary to have entered the maximum or average duration of the activity.
To incorporate an alert, press the icon +. To modify it, press the icon and to delete it, press the icon .

Definition of Times



Alert start

Period of time that determines the start of the alert issuance, depending on the Maximum Duration or Average Duration established for the activity.
The time period can be measured either on a calendar time or working time basis, as specified in the property Consider time.

Days, Hours and Minutes values may:
• Not be informed: The alert is sent when the activity duration is fulfilled.
• Be informed with positive values: The alert is sent after the activity duration fulfillment.
• Be informed with negative values: The alert is sent before the activity duration fulfillment.

Alert duration

Period of time during which the alert is sent. The number of days, hours and minutes informed here allows to calculate the moment in which the alert sending must be finished as from the Alert start. The alert duration must be informed.
The time period can be measured either on a calendar time or working time basis, as specified in the property Consider time.

Alert interval

Time interval or frequency with which the alert is sent. Values are informed in Days, Hours and Minutes. If not informed, the alert will be sent every time the scheduled task is executed Execution of Activity Alerts.
The time period can be measured either on a calendar time or working time basis, as specified in the property Consider time.

Define alert on

The alert may be defined in relation to the Maximum Duration or Average Duration set in the "General" tab or from the same "Alerts" tab.

Consider time

Indicates that the periods set in the Start, Duration and Interval properties should be measured as calendar time or working time, taking into account the definition of environment calendars.
If working time is informed, the calendar defined for the participant responsible for executing the activity, if there is one, is considered first, or as a second option, the calendar defined for their organizational unit. When there are no special calendars, the standard environment calendar is considered.

Alert message
Sender

Email address from which the email is sent.
A value can be entered as long as the email server is not Google nor Microsoft.
Another possibility is to select the emailWorkflow() function, it is suggested when the email server is Google or Microsoft, and in this case it is not necessary to assign it a value in the property Value of the emailWorkflow() function because it will always take the address entered in the property User or account to establish connection.

Receiver
Receiver's email address.

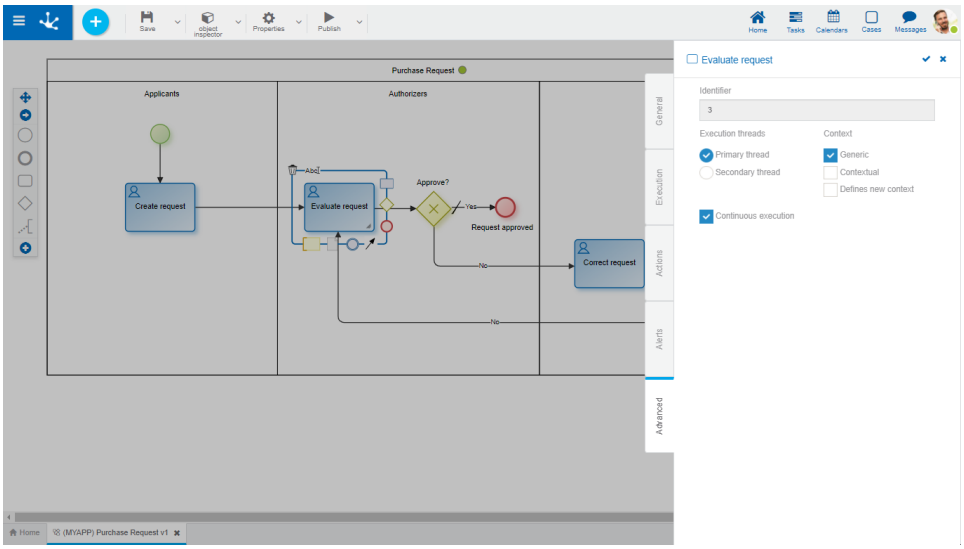
Subject
Contains the email subject.
The field can be completed manually by combining it with variables available in **Deyel** using the icon .

Description
It is the body of the email. The field can be completed manually and also combine it with variables available in **Deyel**, clicking on the icon. Html tags can be used to format the message.

Click on the "Save" button to store the new alert that is incorporated into Alerts List, from where it can be modified or deleted.
Click on the "Cancel" button to close the new alert panel, discarding the changes.

3.6.3.5.5. Advanced

IT modeler users can set advanced level modeling properties in this tab.



Properties

Identifier
Activity code that identifies the activity univocally in the process. This attribute is automatically generated by the modeler and cannot be modified.

Execution Threads
Indicates if the activity is executed on the Main Thread of the case or on the Secondary Thread.

Context
A process execution context determines the universe of users the case can receive to execute its current task.
When those responsible for the activities are expressly indicated users, it is not necessary to select any of the context attributes.

Generic
This attribute can be selected when the lane responsible defined for the activity is arole or an agent.

☑ Generic

When the case reaches the activity defined with Generic context, all the role/agent users receive the case in a generic way, without the possibility of execution, so that one of them can assign the case or the coordinator of the role can assign it to a particular user.
The moment the case is assigned to a user, it automatically disappears from the other users' task list. To assign the case, press the "Assign" button.

☐ Generic

When the next activity has no Generic attribute defined, the list of role/agent users is submitted from the current activity, to select who the case is given to.
Note:If the current executor is included in the role/agent of the next activity and also the continuous execution, property is selected in the current activity, the next activity execution interface is automatically presented to the current user.

Contextual
When the first case activity is executed, the organizational unit of the user that executes it is established as the "contextual unit" of the case.

☑ Contextual

It can be selected in cases where the lane responsible in which the activity is located is defined with role per unit.
When the case reaches the activity with the Contextual attribute selected, it can only be executed by those users with role per unit and who also belong to the context unit or to some unit that depends on it. The Generic attribute should not be selected.

☐ Contextual

When the activity has no Contextual attribute selected, it can be executed by the expressly defined responsible users or by the role/agent, without applying any other restriction.

Defines New Context

☑ Defines new context

When the activity execution begins, a new case execution context is established. This means that the context unit becomes the organizational unit to which the user who is executing the activity belongs. This option is used in very specific situations with a high degree of modeling experience.

☐ Defines new context

The case execution context is not modified. It is the most used option.

Continuous Execution

This indicator is used to specify that after executing the activity it should automatically continue with the next one, whenever possible. In this way, the user can execute the case activities continuously, without accessing the to-do list.

Continuous execution is possible when:

- The responsible for the next activity is a user and it is the same user that executes the current activity.
- The responsible for the next activity is an organizational unit and it is the same organizational unit of the user that executes the current activity.
- The person responsible for the next activity is a role / agent and the user executing the current activity belongs to the role/agent.
  If in the next activity the Contextual attribute is checked, the organizational unit verifies that the current user is the context unit or a unit that depends on it.

☑ Continuous execution

According to the definition of subsequent activity, if the current user can execute it, then its execution starts automatically.

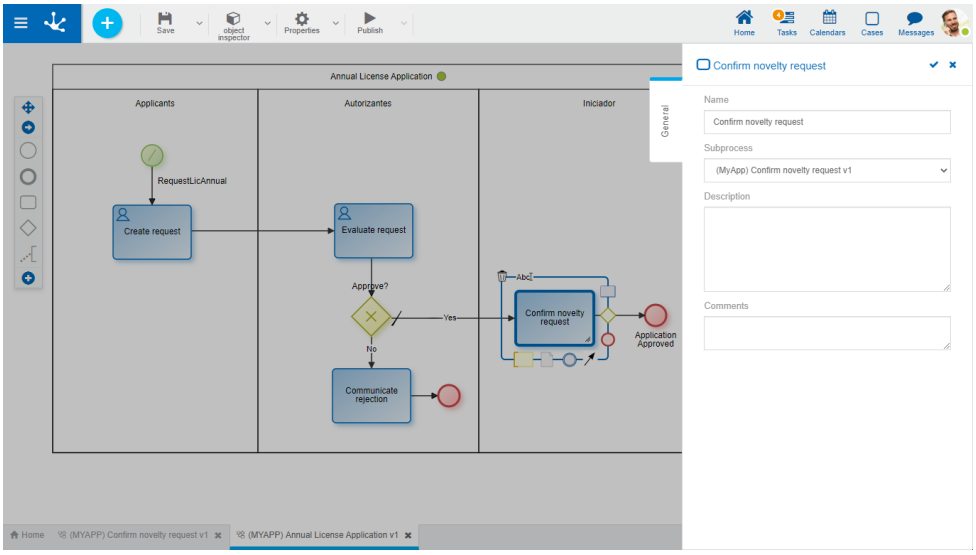
☐ Continuous execution

In this case, even though the current user can execute the next task, they must take the case from "My Tasks".

3.6.3.6. Subprocess Properties

There are different ways to enter a subprocess properties panel:

- Double click on the graphic element corresponding to the subprocess or abstract activity.
- Access the context menu of the subprocess graphic element.



General Tab

Properties

Name
Activity name. Describes in a generic way the objective of the process that is being represented.

Subprocess
Process represented by this abstract activity. Presents a list of all processes defined to select one of them.

Description
Conceptual description of the activities performed.

Comments
In this field the designer can document in detail what the subprocess execution consists of.

Actions

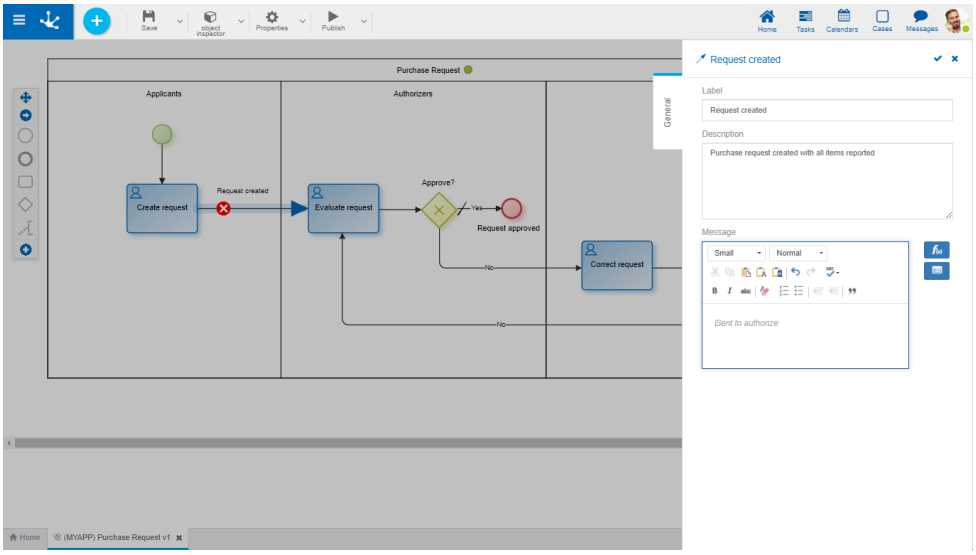
The icon ✔ is used to confirm the modifications made in the properties panel.

The icon ✖ is used to close the properties panel, if it was not previously saved, changes are discarded.

3.6.3.7. Flow Properties

There are different ways to enter a flow properties panel:

- Double click on the graphic element corresponding to the flow.
- Access the context menu of the flow graphic element.



General Tab

Properties

Label
Flow label.

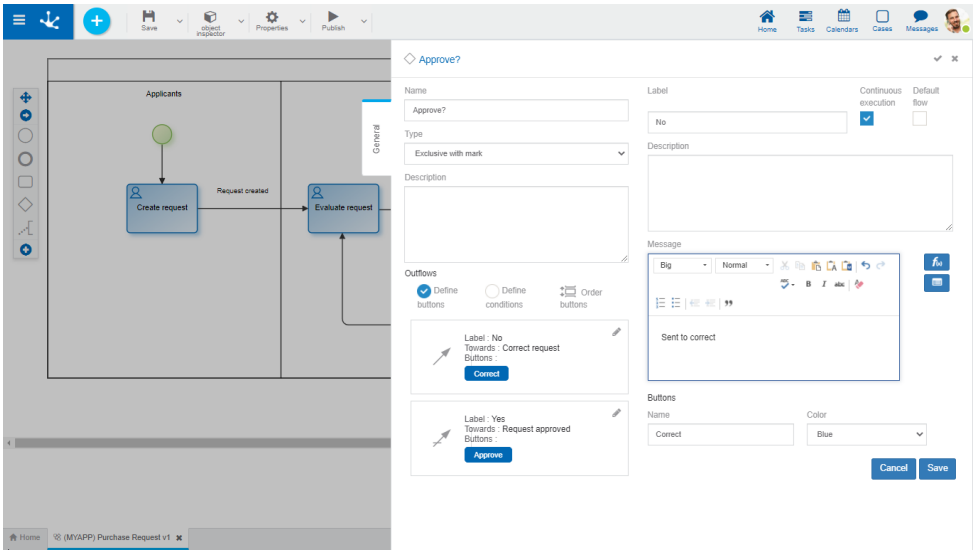
Description
Text field used to document the meaning of the flow.

Message
Text displayed when moving to the destination activity of the flow.

Allows to select functions of **Deyel** from a list to incorporate dynamic information into the message.

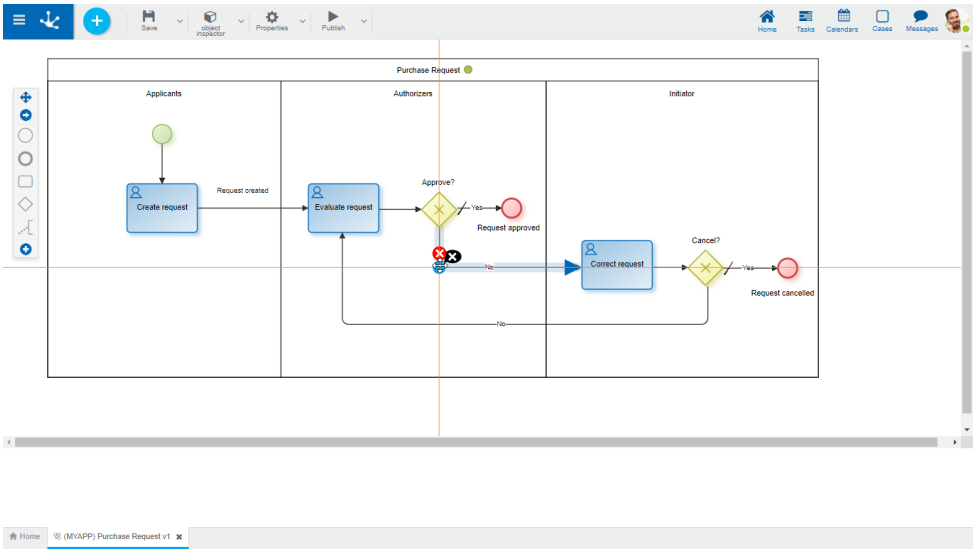
Allows to incorporate information on entities used in the process execution to the message.

When flow cases are gate outputs, in addition to flow properties, the panel displays the gate properties.



Visual Flow Options

These options are displayed when hovering over the graphic element.



Deletes the flow.

Indicates there is a vertex. Positioning the cursor over it enables a black cross that, when selected, will delete such vertex.

Deletes the vertex. Position the cursor over the blue circle to display the icon and then click on it.

To add a new vertex to the flow, keep the mouse button pressed until a blue point (vertex) is displayed. Once generated a new vertex, it can be moved with the Drag&Drop function.

Actions

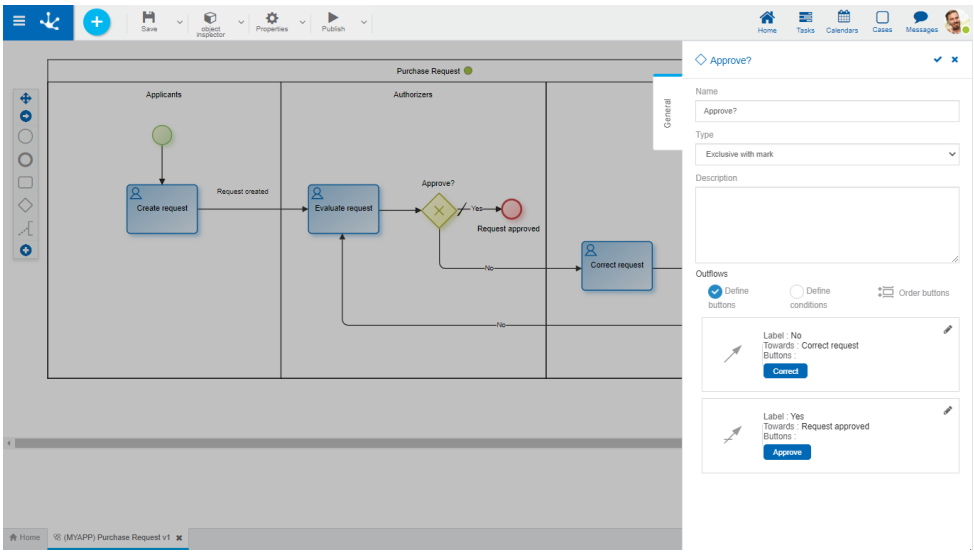
The icon ✔ is used to confirm the modifications made in the properties panel.

The icon ✖ is used to close the properties panel, if it was not previously saved, changes are discarded.

3.6.3.8. Gateway Properties

There are different ways to enter the gateway properties panel:

- Double click on the graphic element corresponding to the gateway.
- Access the context menu of the gateway graphic element.



General Tab

Properties

Name
Gateway name that should reflect what is being evaluated.

Type
Indicates the gateway type.

Description
Text that describes the gateway purpose.

Output flows

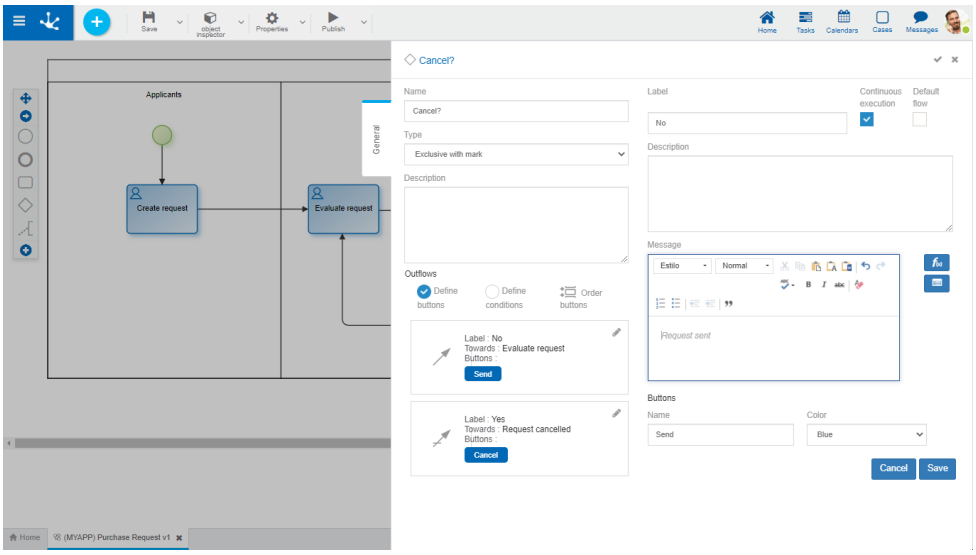
Default Flow
Allows to define a default flow indicating which flow to follow when other flow conditions are not met. This property is required.

Continuous Execution

Allows to define the continuous execution of each output flow of the exclusive gateways.

Define buttons

Allows to assign a button for the user to push at each gateway output flow. If the gateway is inclusive, more than one button can be assigned to each flow, and each button may have more than one flow assigned.
Buttons become visible in the execution activity previous to the gateway.
A name and a color are defined for each button .



Displays a properties panel that allows to establish the button name and color assigned to the flow, and also modifies other flow properties.
Within this panel, if it is an inclusive gateway, the following possibilities are enabled.

New button

Allows to add a new button to the flow.

Select assistant bot

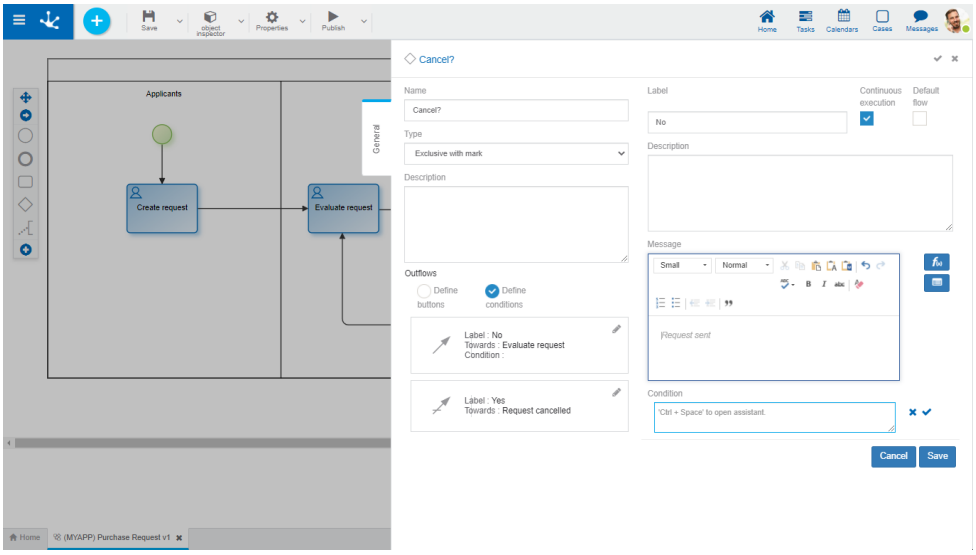
Allows to assign an existing button to the flow.

Order buttons

This functionality is enabled when you select Define buttons. Clicking on it, displays a panel to order buttons with the Drag&Drop function.

Define conditions

Allows to assign conditions to be met so that the process continues its execution following the flow.



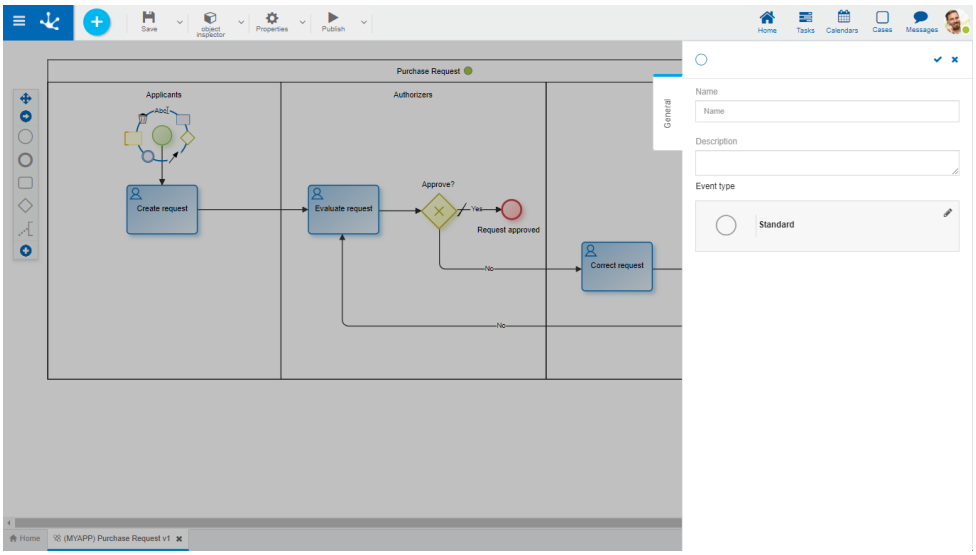
Displays a properties panel where the flow conditions are modeled by means of the embedded rules wizard that allows to select fields of entities that participate in the process and operations to be applied. Additionally, it is also possible to modify other flow properties.

> *If the process is related to several forms, the wizard allows to select the form fields indicated as main entity in form properties.*
> *If an entity is not defined as the main entity, the first form fields displayed when expanding the list in the Execution Definition property can be selected in the process activities.*

3.6.3.9. Event Properties

There are different ways to enter the properties panel an event:

- Double click on the graphic element corresponding to the event.
- Access the context menu of the event graphic element.



Actions

The icon ✔ is used to confirm the modifications made in the properties panel.

The icon ✖ is used to close the properties panel, if it was not previously saved, changes are discarded.

3.6.3.9.1. Starts Events

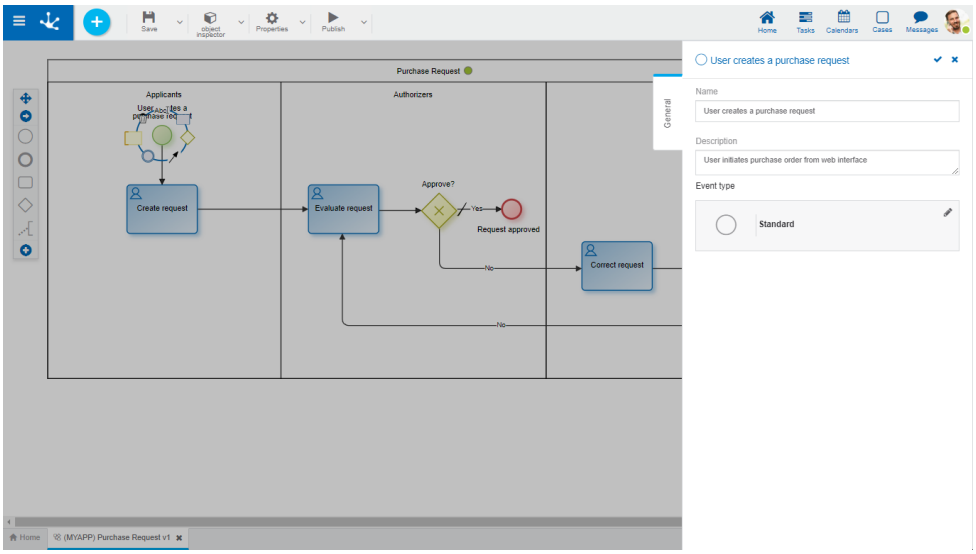
General Tab

In the "General" tab, common properties for all start events and the special ones for each case are defined, which are enabled when assigning the type of start event.

- Standard
- Command
- Email
- Signal
- Timer
- File
- Rule

3.6.3.9.1.1. Standard

The properties panel of the standard start event is displayed on the right side of the process modeler.



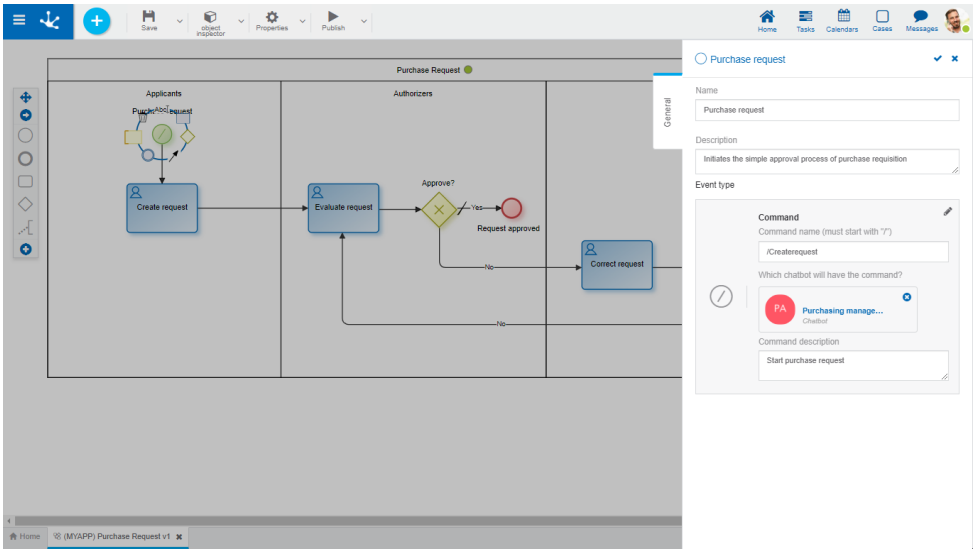
Properties

Name
Text that is displayed in the diagram next to the graphic element of the event.

Description
Field that allows to document the event.

3.6.3.9.1.2. Command

The properties panel of the start event by command is displayed on the right side of the process modeler.



Properties

Name
Text that is displayed in the diagram next to the graphic element of the event.
It is recommended to provide a text that identifies the command.

Description
Field that allows to document the event.

Command
Text assigned by the user to identify the command that starts the process. MIt must start with a slash "/" and followed by the name the user informed, with no blank spaces.
By default it presents the command established in the process creation. It can be modified.

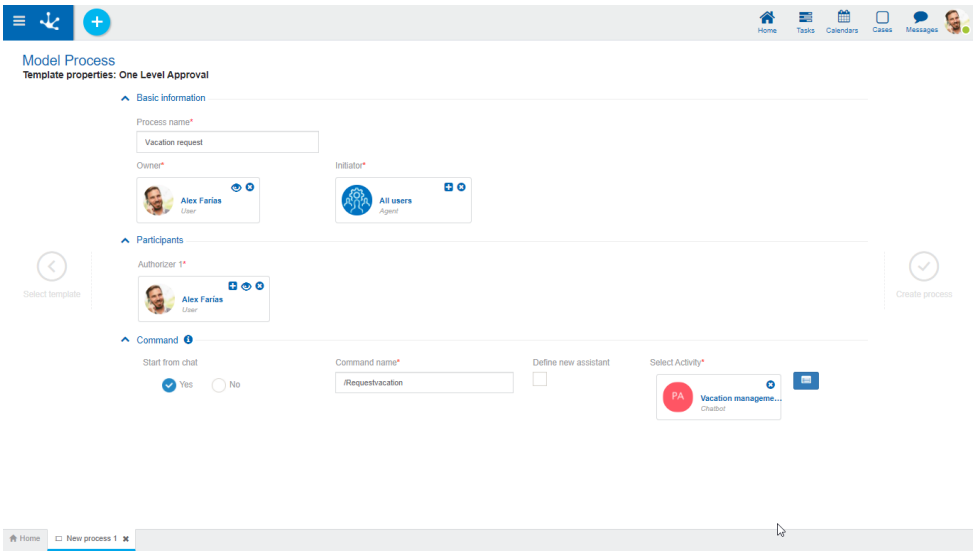
Chatbot
Indicates the chatbot used to execute the command.
By default, the chatbot informed in the process creation is displayed, and besides, if the Define new assistant process property is checked, a new chatbot is automatically created, configuring itself as wizard of the user that is modeling the process.
To modify the assigned chatbot, click on the delete icon. This property has autocomplete function, therefore as characters are entered, the first five matches are dynamically displayed to select one of them.

Command Description

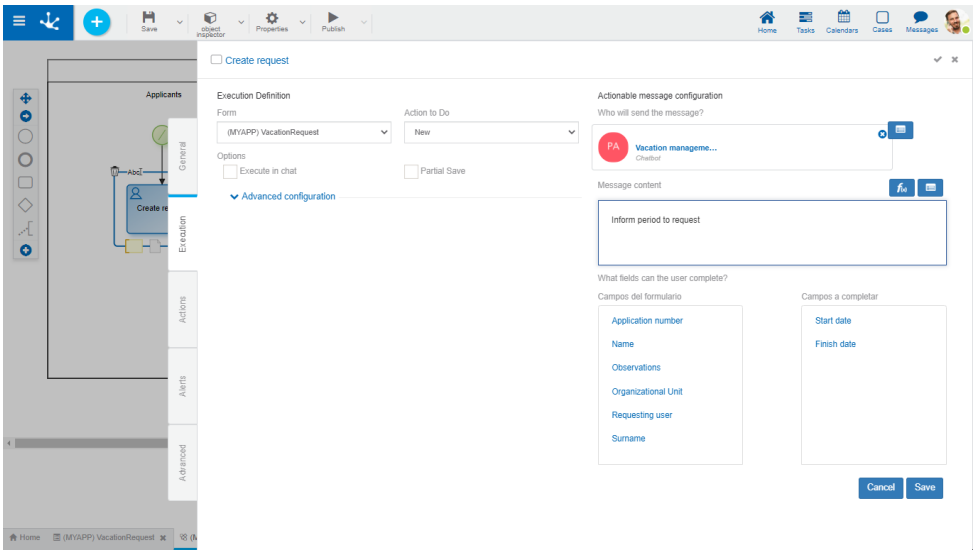
Explanatory text that is displayed when selecting the command from the chat window.

Example

The command start event is automatically defined when the process property is checked Start from Chat, when modeling a new process, whether from a template or not.



To modify the type in an already defined start event, access the context menu of the graphic element and select the type of start event from the options that are presented.
The first activity of a process that starts with a command event, must have the Execute on Chat property checked.



Properties

Who will send the message?

The user or chatbot that sends the message from the chat is informed. It generally matches the chatbot assigned in the command start event.
Note that if a user is informed, that user does not see their own message.

Message content

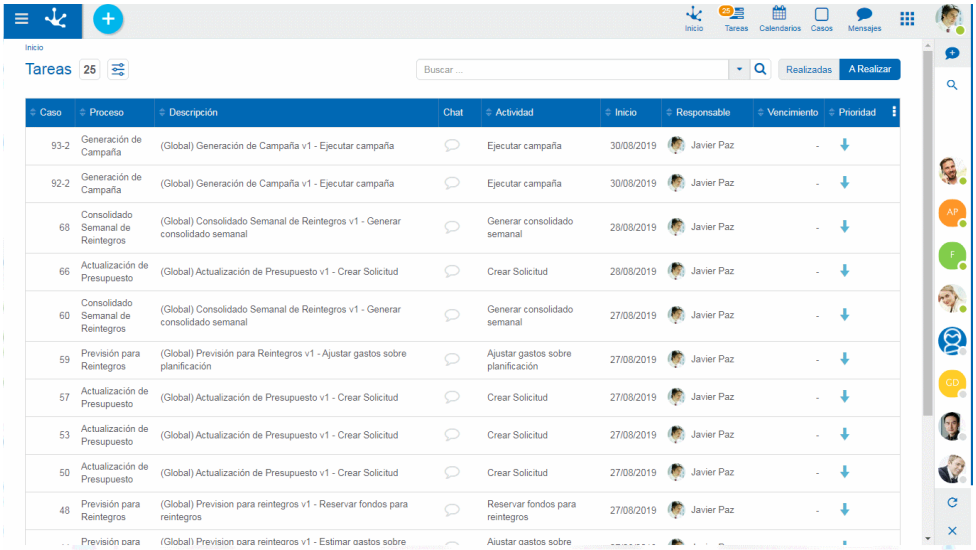
Inform the text displayed in the chat at the beginning of the process.

What fields can the user complete?

Move the fields that the user must complete from the chat to the fields box to complete, in the first process activity.

Use of the Process

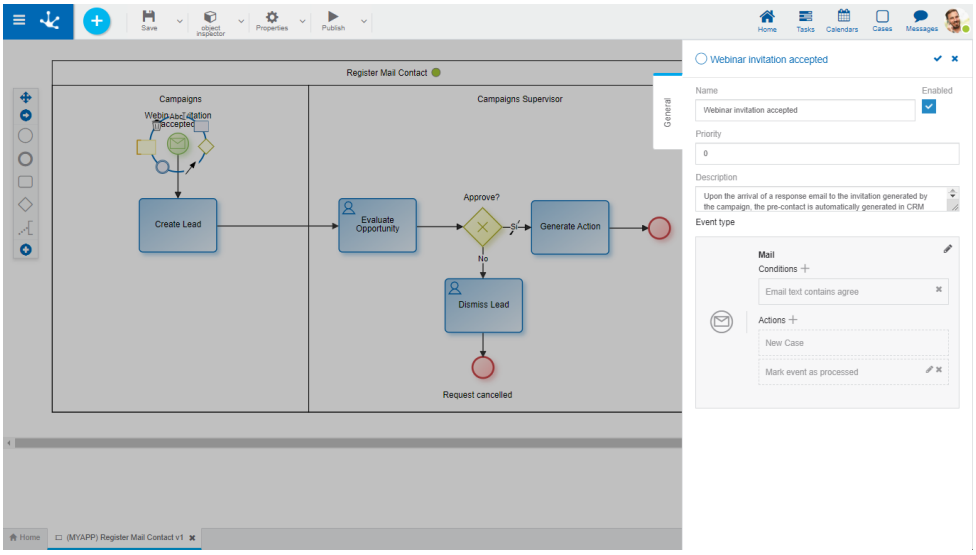
- To start the process by chat, the user must have permission to start the process.
- If the command is assigned to a chatbot configured as "visible", users can initiate the command directly from the chatbot in a chat.
- If the chatbot with the command is configured as "not visible", then necessarily some user must be using it as wizard. A chat with that user must be opened to access the command.



In the chat window, the data required in the form must be informed and one of the proposed buttons of the actionable message must be pressed.

3.6.3.9.1.3. Email

The properties panel of the start event by email is displayed on the right side of the process modeler.



Properties

Name

Text that is displayed in the diagram next to the graphic element of the event.
It is recommended to inform a text in reference to the email expected to be received.

Enabled

Indicates that the event is enabled to receive emails and execute if applicable.

Priority

Execution priority assigned to the event. It is reported as an integer, with 0 being the highest priority. If the event performs the action mark event as processed, events with lower priority are not executed.

Description

Text that allows documenting detailed information about the event.

Conditions

They are the conditions that must be met for the event to be executed. If there are defined conditions, a list of them is displayed.
To add a condition, press the icon + and from the panel that opens, select Property and Operator in the corresponding drop-down lists. Report Value and where applicable, indicate in Parameter the field to be evaluated of the form associated with the process.

Property	Example
• Email subject • Email text	Email subject **contains**"Information request"
• Email sender • Email recipient	Email sender **Equals**AtencionConsultas@Deyel.com *More than one address can be reported separated by ; (semicolon)*

Property	Example
	without leaving spaces
• Name without the attachment extension in n position (starts at 1) • Attachment extension in n position (starts at 1)	Name without the attachment extension in n position **starts with**"Sheet", indicated in parameter 1 (first position)
• Number of email days elapsed	Number of days elapsed **Equals** 3

Actions

They are the actions that are carried out automatically when the event occurs.

- Create Case

For all start events, a case is compulsorily created for the process and a version in which the event is executed.

- Attach file

Allows to attach one or more files to the case.

Parameter	Description
Email attachment	Position number of the email attachment that is attached to the case. (*)
Depends on the previous execution	Indicates if the action is executed only when the previously defined actions were executed successfully.(*)

Note: The parameters selected with (*) are mandatory.

- Attach form

Allows to relate a form to the case.

Parameter	Description
User Code	If it is not reported, the current user is assigned.
Values map	Report <field1><value1><field2><value2>…, where fieldN corresponds to form field names and valueN to the values that are assigned to each field. It has higher priority than DS_XML_CONTENT_PATH

Parameter	Description
XML	File path that contains the XML. It is mandatory if DS_VALUE_MAP is not reported
Annex Type	Annex type attached (*).
Create version if it exists and if it is unique	If when attaching the form, it already exists and is unique, a historical version of it is previously created.
Reset previous form and it is unique	Indicates that if there is a form associated with the case and it is unique, data is deleted from it.
Depends on the previous execution	The action is executed only if the previously defined actions were executed successfully.(*).

> *Note: The parameters marked with (*) are mandatory.*

- Check event as processed

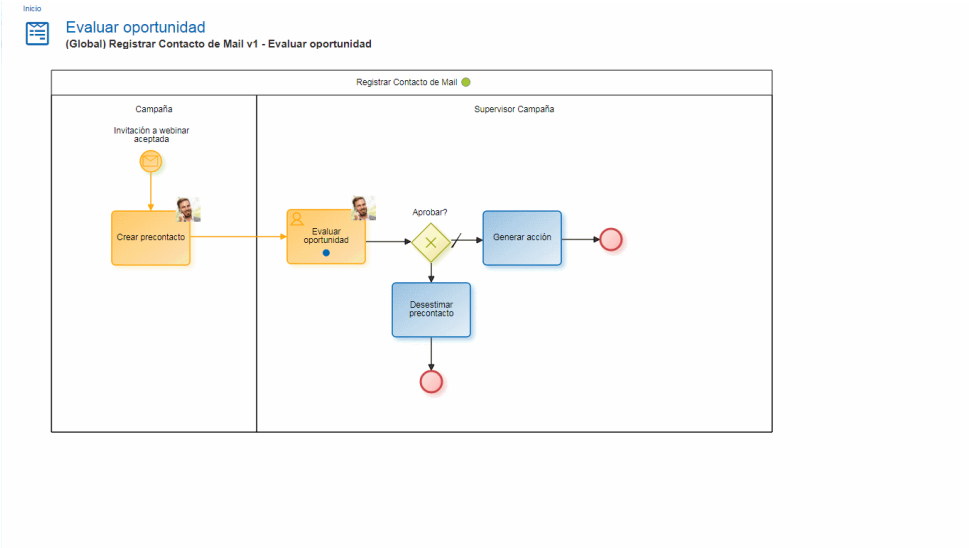
An automatic response is sent to the email received indicating that it has been processed.

Parameter	Description
Depends on the previous execution	Indicates if the action is executed only when the previously defined actions were executed successfully (*).

> *Note: The parameters marked with (*) are mandatory.*

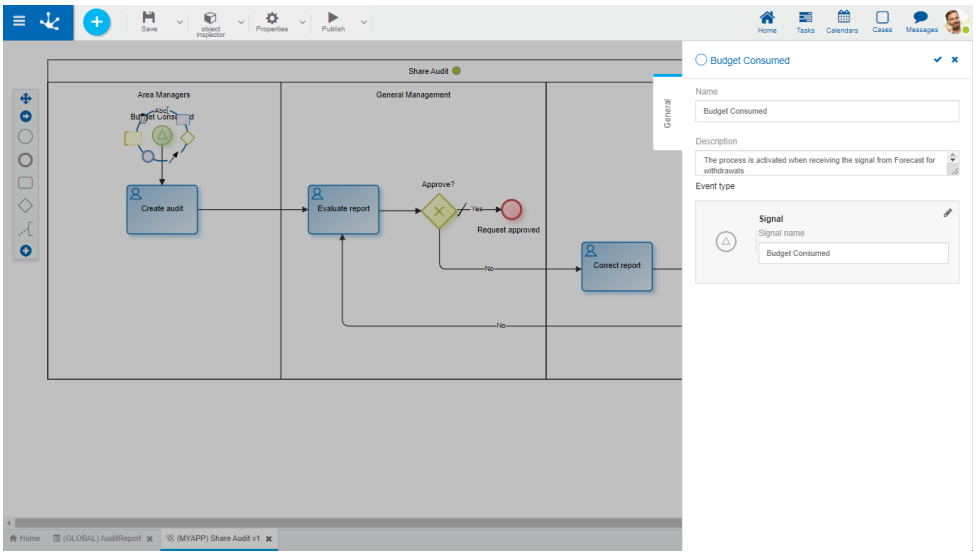
Example of Use

The "Register Email Contact" process starts its execution automatically when an email arrives to the email account established in the environment configuration, under the title "Invitation to **Deyel** Webinar" and the word "Agree" is in the content.



3.6.3.9.1.4. Signal

The properties panel of the start event by signal is displayed on the right side of the process modeler.



Properties

Name

Text that is displayed in the diagram next to the graphic element of the event.
It is recommended to inform a text in reference to the signal expected to be received.

Description

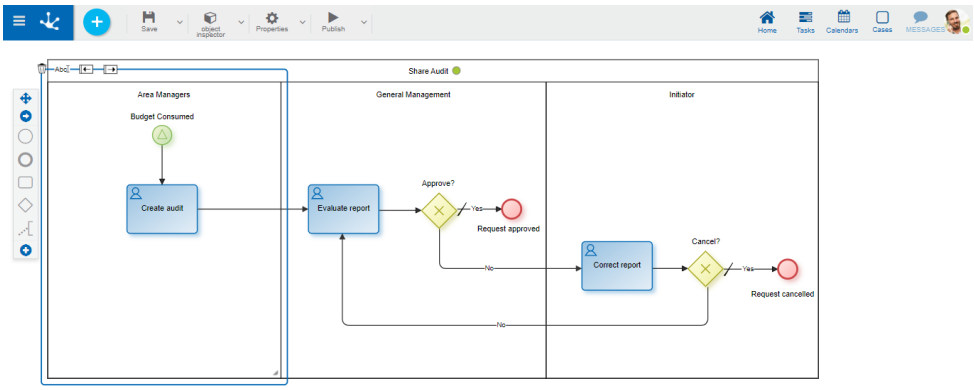
Text that allows documenting detailed information about the event.

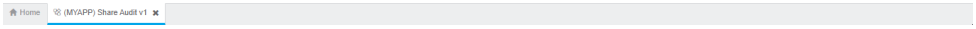
Signal Name

Signal that the event expects to receive to start execution automatically. The signal is sent by a signal end event or a throw signal intermediate event.
There may be more than one process waiting for the same signal.

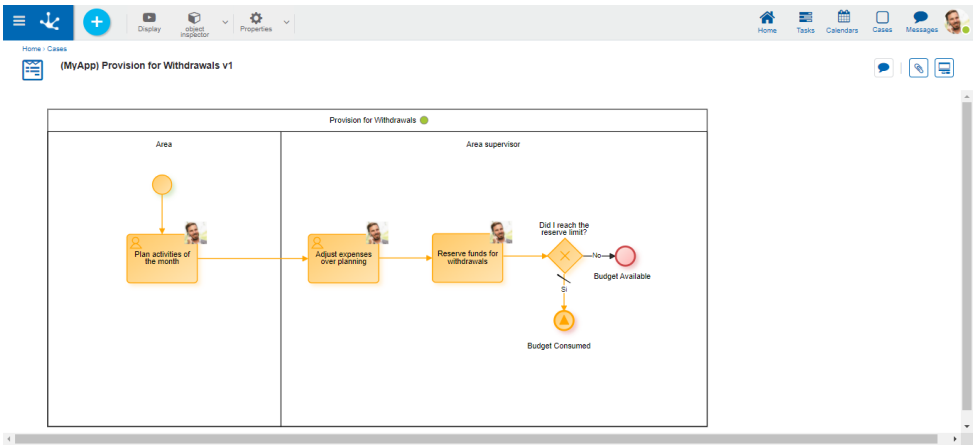
Example of Use

The audit process starts automatically upon receiving the "Budget Consumed" signal sent by the "Provision for Refunds",process in one of its possible endings.



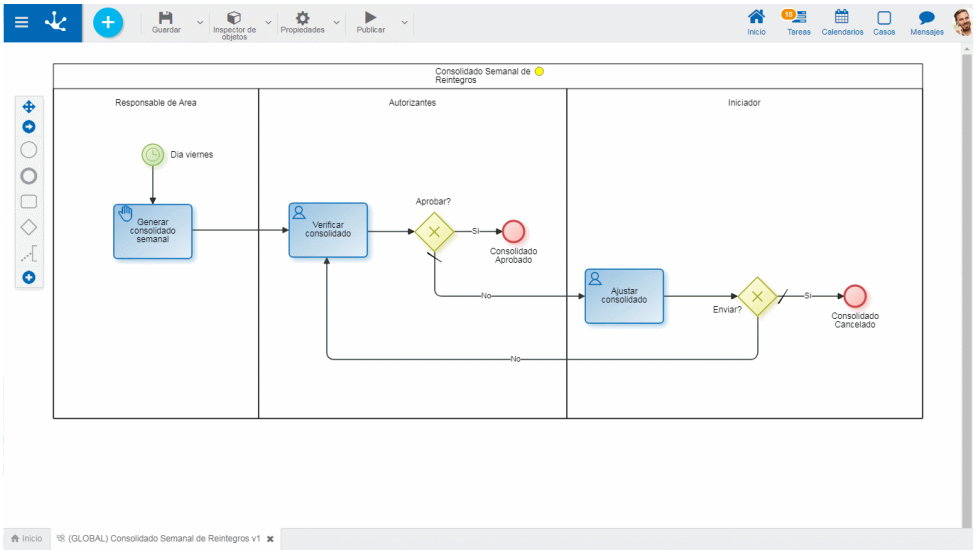


"Provision for Refunds" process in which the "Consumed Budget" signal is generated.



3.6.3.9.1.5. Timer

The properties panel of the timer start event is displayed on the right side of the process modeler.



Properties

Name
Text that is displayed in the diagram next to the graphic element of the event.
It is recommended to inform a text in reference to the event scheduling.

Enabled
This property is checked to indicate that an event is active and can be automatically executed according to its scheduling.

Description
Text that allows documenting detailed information about the event.

Conditions
They are the conditions that must be met for the event to be executed. If there are defined conditions, a list of them is displayed.
To add a condition, press the icon + and from the panel that opens, select Property and Operator in the corresponding drop-down lists. Report Value and where applicable, indicate in Parameter the field to be evaluated of the form associated with the process.

For this type of event, apply the following conditions:
- Current time
- Current day
- Current month
- Current year
- Current hour
- Current minute
- Current seconds
- Current milliseconds
- Weekday

How often should it execute?
Type
Allows to define the execution schedule for the event

- Regular intervals
- Daily at a specific time

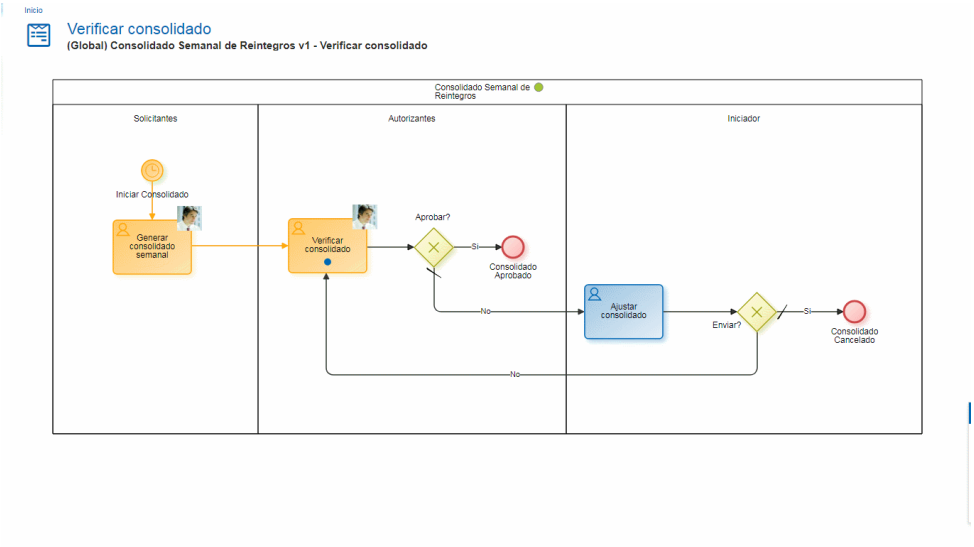
Value

Allows to define the interval value in seconds or at a specific time, according to the previously defined type.

The "Save schedule" button must be pressed to keep the established schedule. It can be seen on the Scheduled Tasks Monitor.

Example of Use

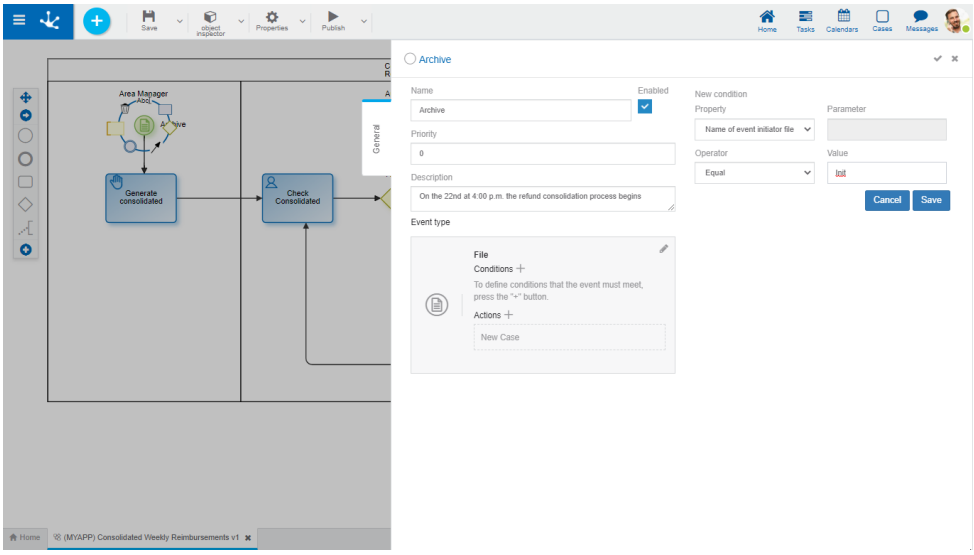
The "Consolidated Weekly Refund" process begins its execution automatically on Fridays at 10:00.



3.6.3.9.1.6. File

This type of event is available only for on-premises installations, as they require access to resources on the client's computer.

The properties panel of the start event per file is displayed on the right side of the process modeler.



Properties

Name

Text that is displayed in the diagram next to the graphic element of the event.
It is recommended to inform a text in reference to the email expected to be received.

Enabled

This property is checked when the event activates, waiting for the file.

Priority

Execution priority assigned to the event. It is reported as an integer, with 0 being the highest priority. If the event performs the action mark event as processed, events with lower priority are not executed.

Description

Text that allows documenting detailed information about the event.

Conditions

They are the conditions that must be met for the event to be executed. If there are defined conditions, a list of them is displayed.
To add a condition, press the icon + and from the panel that opens, select Property and Operator in the corresponding drop-down lists. Report Value and where applicable, indicate in Parameter the field to be evaluated of the form associated with the process.

Property	Example
Name of the event initiator file, without the extension	Name of the event initiator file **Contains**"Ticket"
Name of the folder that contains the event initiator file	Name of the folder that contains the event initiator file **Equals**"Receipt of Tickets"
File extension (without dot)	File extension **Equals** XML

Property	Example
Event initiator file path	Event initiator file path **Equals**"C:\Claims\Tickets Attention"
Folder path that contains the event initiator file	Folder path that contains the event initiator file **Equals**"D:\Claims Attention"
Number of file days elapsed	Number of file days elapsed **Greater than or Equal to** 9
File type (File or Folder)	

Actions

They are the actions that are carried out automatically when the event occurs.

- Create Case

For all start events, a case is compulsorily created for the process and a version in which the event is executed.

- Attach file

Allows to attach a file to the case.

Parameter	Description
Depends on the previous execution	Indicates if the action is executed only when the previously defined actions were executed successfully.(*)

> *Note: The parameters selected with (*) are mandatory.*

- Attach form

Allows to relate a form to the case.

Parameter	Description
User Code	If it is not reported, the current user is assigned.
Values map	Report <field1><value1><field2><value2>…, where fieldN corresponds to form field names and valueN to the values that are assigned to each field. It has higher priority than DS_XML_CONTENT_PATH

Parameter	Description
XML	File path that contains the XML. It is mandatory if DS_VALUE_MAP is not reported
Annex Type	Annex type attached (*).
Create version if it exists and if it is unique	If when attaching the form, it already exists and is unique, a historical version of it is previously created.
Reset previous form and it is unique	Indicates that if there is a form associated with the case and it is unique, data is deleted from it.
Depends on the previous execution	The action is executed only if the previously defined actions were executed successfully.(*).

> *Note: The parameters marked with (*) are mandatory.*

- Check event as processed

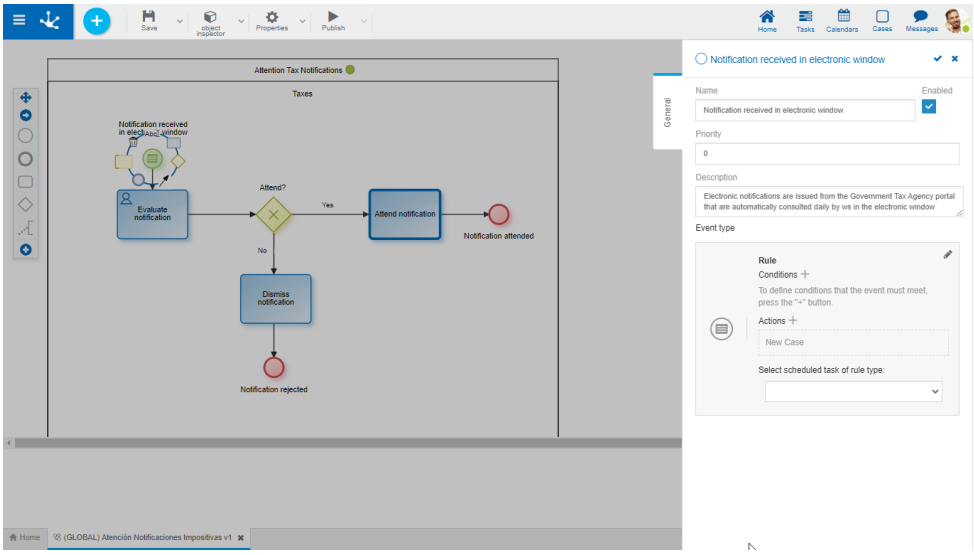
An automatic response is sent to the email received indicating that it has been processed.

Parameter	Description
Depends on the previous execution	Indicates if the action is executed only when the previously defined actions were executed successfully (*).

> *Note: The parameters marked with (*) are mandatory.*

3.6.3.9.1.7. Rule

The properties panel of the start event by rule is displayed on the right side of the process modeler.



Properties

Name

Text that is displayed in the diagram next to the graphic element of the event.
It is recommended to inform a text in reference to the rule whose execution is expected.

Enabled

This property is checked to indicate that the event activates, waiting for the rule.

Priority

Execution priority assigned to the event. It is reported as an integer, being 0 the top priority.

Description

Text that allows documenting detailed information about the event.

Conditions

They are the conditions that must be met for the event to be executed. If there are defined conditions, a list of them is displayed.

To add a condition, press the icon + and from the panel that opens, select Property and Operator in the corresponding drop-down lists. Report Value and where applicable, indicate in Parameter the field to be evaluated of the form associated with the process.

Actions

They are the actions that are carried out automatically when the event occurs.

- Create Case

For all start events, a case is compulsorily created for the process and a version in which the event is executed.

- Attach file

Allows to attach a file to the case.

Parameter	Description

Parameter	Description
File to attach	File attached to the case.
Depends on the previous execution	Indicates if the action is executed only when the previously defined actions were executed successfully.(*)

> *Note: The parameters selected with (*) are mandatory.*

- Attach form

Allows to relate a form to the case.

Parameter	Description
User Code	If it is not reported, the current user is assigned.
Values map	Report <field1><value1><field2><value2>…, where fieldN corresponds to form field names and valueN to the values that are assigned to each field. It has higher priority than DS_XML_CONTENT_PATH
XML	File path that contains the XML. It is mandatory if DS_VALUE_MAP is not reported
Annex Type	Annex type attached (*).
Create version if it exists and if it is unique	If when attaching the form, it already exists and is unique, a historical version of it is previously created.
Reset previous form and it is unique	Indicates that if there is a form associated with the case and it is unique, data is deleted from it.
Depends on the previous execution	The action is executed only if the previously defined actions were executed successfully.(*).

> *Note: The parameters marked with (*) are mandatory.*

Select scheduled rule task

The executed business rule is selected. It must be a previously published advanced rule.

Example of Use

The "Attention to Tax Notifications" process starts automatically when the business rule associated with the rule start event detects a notification in the government portal, following the WS specifications published in it.

3.6.3.9.2. Intermediate Events

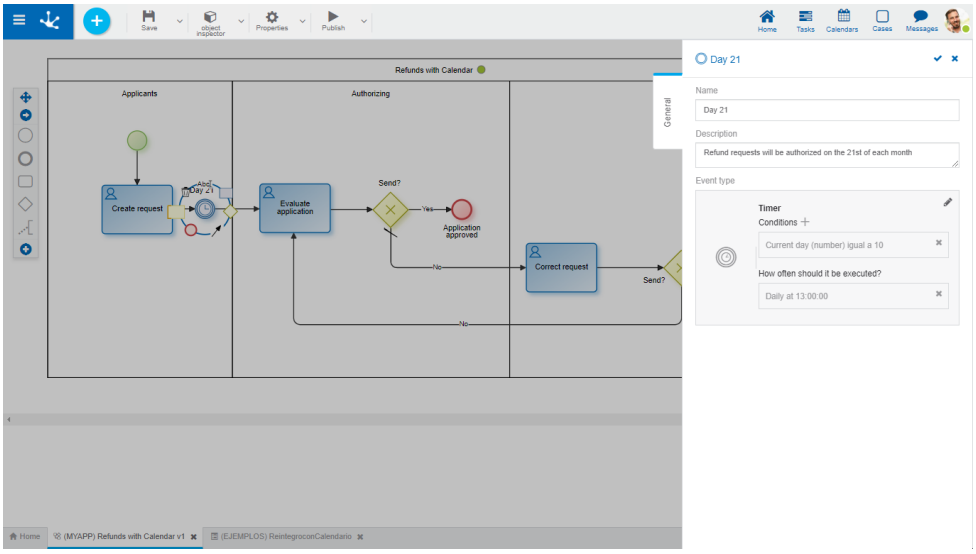
General Tab

In the "General" tab, common properties for all intermediate events and the specific ones for each case are defined, which are enabled when assigning the type of intermediate event.

- Timer
- Throw and Catch a Signal
- Throw and Catch Link

3.6.3.9.2.1. Timer

The properties panel of the timer intermediate eventis displayed on the right side of the process modeler.



Properties

Name

Text that is displayed in the diagram next to the graphic element of the event.
It is recommended to inform a text in reference to the event scheduling.

Description

Text that allows documenting detailed information about the event.

Conditions

They are the conditions that must be met for the event to be executed. If there are defined conditions, a list of them is displayed.

To add a condition, press the icon + and from the panel that opens, select Property and Operator in the corresponding drop-down lists. Report Value and where applicable, indicate in Parameter the field to be evaluated of the form associated with the process.

Property	Example
• Current time • Current day (numeric) • Current month (numeric) • Current year (numeric)	Current time**Equals** 13:30 Current month **Less than or equals** 6 *Time format is HH:MM. Supports one-digit values for the time.* *When showing the current month, a number between 1 and 12 must be informed in value.*
• Current hour (numeric) • Current minute (numeric) • Current seconds (numeric) • Current milliseconds	Current time (numeric) **Equals** 1.50 Current milliseconds **Greater** 48600000
• Day of the week (Sunday= 1)	Day of the week equals 6 (6 = Friday) *An integer between 1 and 7 must be informed in value, where 1 corresponds to Sunday.*
• DateTime parameter (*) • Date parameter (*) • Time parameter (*)	Inform in Parameter a form field that is converted to date/time/date time for its evaluation *The date format is DD/MM/YYYY.* *The datetime format is DD/MM/YYY HH:MM, with a blank space.* *Time format is HH:MM.* *Both formats support one-digit values for day and month.*
• Working days elapsed since the date (*) • Years elapsed since the date (*) • Months elapsed since the date (*) • Days elapsed since the date (*)	Inform in Parameter a specific date in DD/MM/YYYY format or a form field that contains a date *For these properties an integer value must be informed.*

Property	Example
• Hours elapsed since the “time” hour (*) • Minutes elapsed since the "time" hour (*) • Seconds elapsed since the "time" hour (*)	Inform in Parameter a specific time in DD/MM/YYYY format or a form field that contains a time *For these properties an integer value must be informed.*

(*) Properties available only for intermediate and border events

How often should it execute?

Type

Allows to define the execution schedule for the event

- Regular intervals
- Daily at a specific time

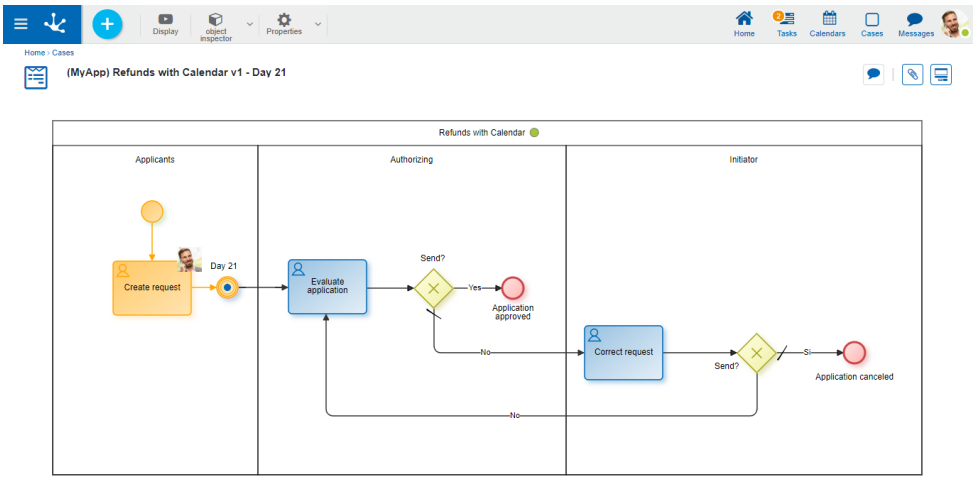
Value

Allows to define the interval value in seconds or at a specific time, according to the previously defined type.

The "Save schedule" button must be pressed to keep the established schedule. It can be seen on the Scheduled Tasks Monitor.

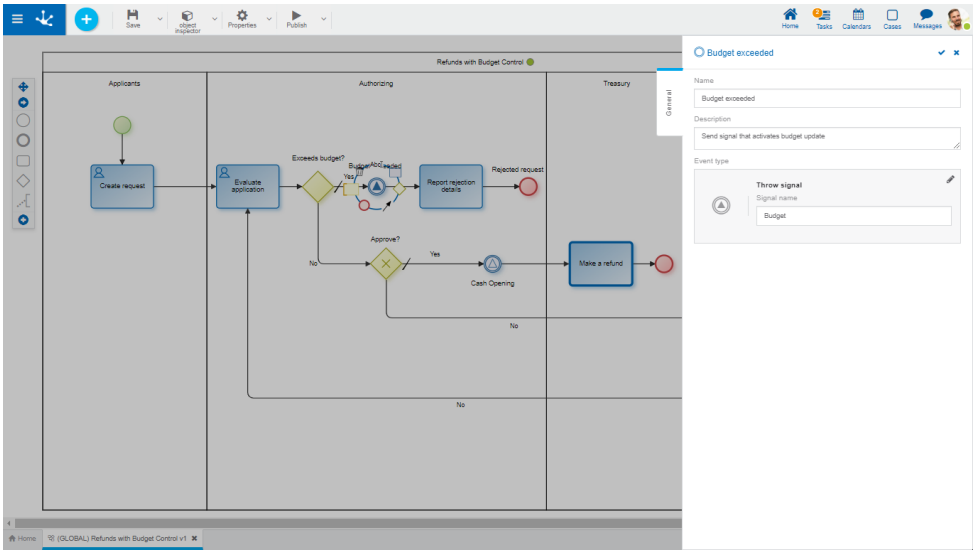
Example of Use

The "Refund Request with Calendar" process begins at the request of users. The authorizer receives requests not until day 21.



3.6.3.9.2.2. Throw and Catch a Signal

The properties panel of the throwand catch signal intermediate events, is displayed on the right side of the process modeler.



Properties

Name

Text that is displayed in the diagram next to the graphic element of the event.
It is recommended to inform a text in reference to the signal expected to be received or sent.

Description

Text that allows documenting detailed information about the event.

Signal Name

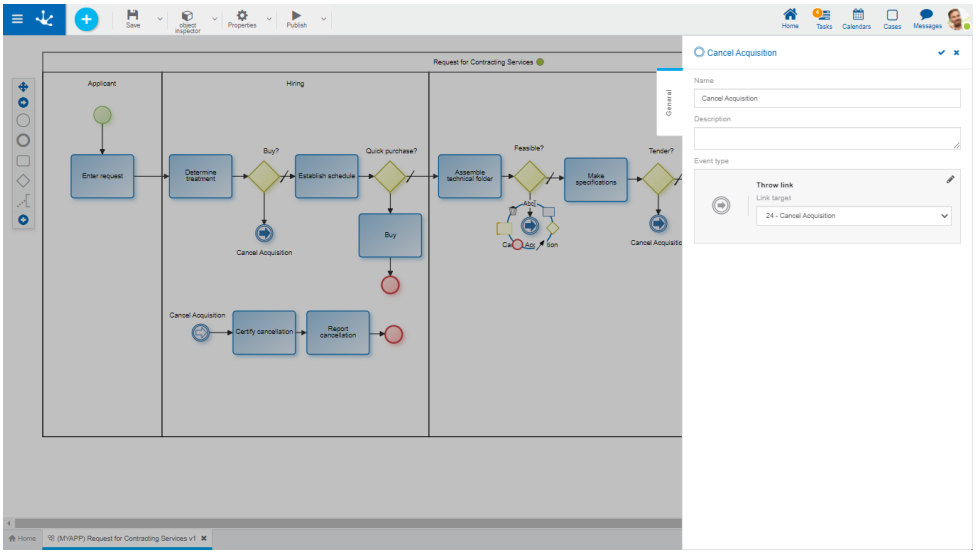
Signal identifier that the event expects to receive or sends in order to continue executing the case automatically, depending on whether it is a catch or a launch event. A character string is specified. There may be more than one process waiting for the same signal.

Example of Use

The "Refund with Budget Control" process sends a signal to automatically start the "Review Budgeting" process, when the established budget has been exceeded.
Approved refund requests await a treasury signal that enables the cash flow.

3.6.3.9.2.3. Throw and Catch Link

The properties panel of the throw linkand capture link intermediate events, is displayed on the right side of the process modeler.



Properties

Name

Text that is displayed in the diagram next to the graphic element of the event.
If it is a catch link event, it is also the identifier used to reach it, from the throw link event.

Description

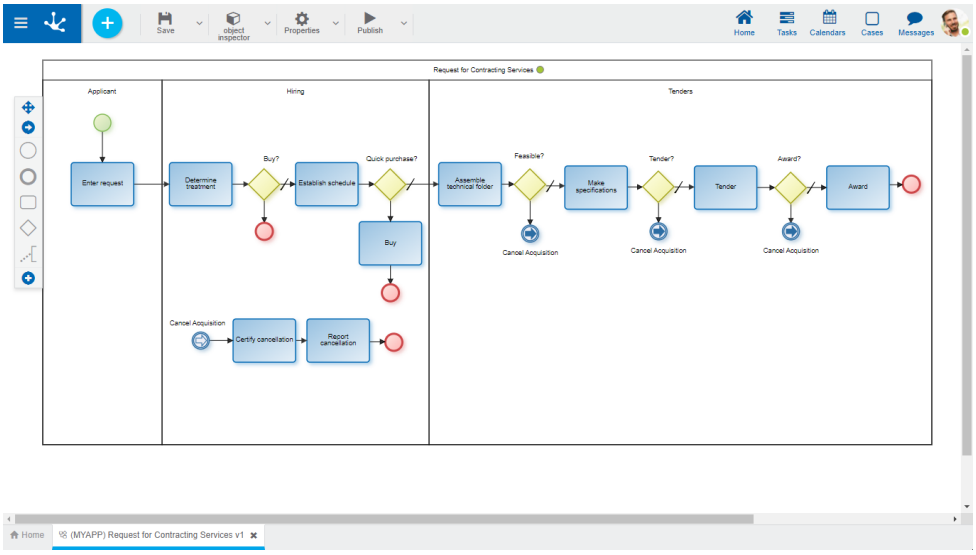
Text that allows documenting detailed information about the event.

Link Destination

This attribute corresponds to the throw link event. Allows to select the event type catch link to go to.

Example of Use

The "Request for Contracting Services" process, to facilitate its interpretation, presents several links to the "Cancel Acquisition" event. These events are particularly useful in complex diagrams.



3.6.3.9.3. Border Event

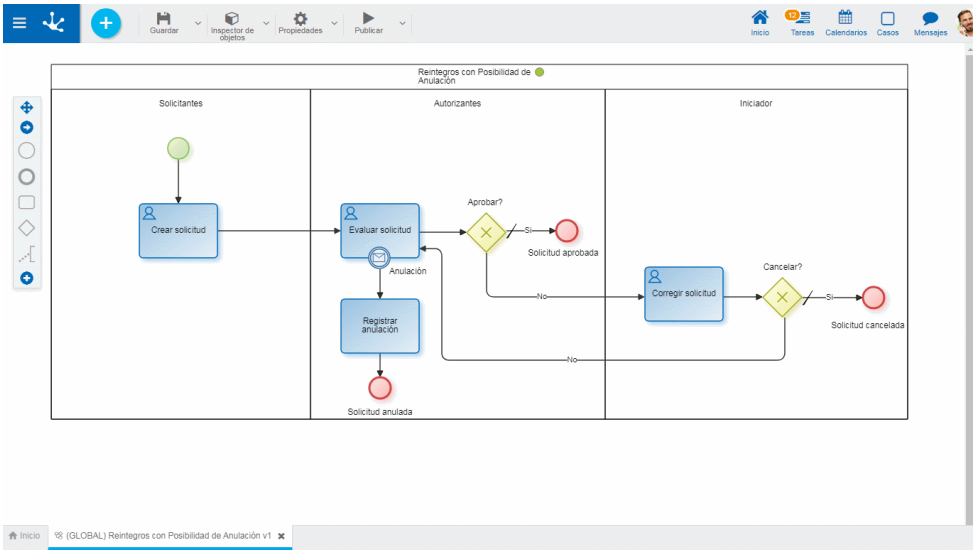
General Tab

In the "General" tab, common properties for all border events and the specific ones for each case are defined, which are enabled when assigning the type of border event.

- Email
- Throw and Catch a Signal
- Timer
- File
- Rule

3.6.3.9.3.1. Email

The properties panel of the email border event is displayed on the right side of the process modeler.



Properties

Name
Event name. It is displayed in the process next to the graphic element that represents the event.

Description
Field that allows documenting the event.

Priority
Execution priority assigned to the event. It is reported as an integer, with 0 being the highest priority. If the event performs the action mark event as processed, events with lower priority are not executed.

Interrupting
The email border event is always interrupting. This means that if the established conditions are met, the event actions are executed and the case follows the event output flow.

Enabled
Indicates that the event is enabled to receive emails and execute if applicable.

Conditions
Allows to define conditions to be met for the email event to be executed.

Actions
They are the actions that are carried out automatically when the event occurs.

- Move Case

The case affected by the event moves on to the next activity.

- Attach file

Allows to attach a file to the case.

Parameter	Description

Parameter	Description
Email attachment	Position number of the email attachment that is attached to the case. (*)
Depends on the previous execution	Indicates if the action is executed only when the previously defined actions were executed successfully.(*)

> *Note: The parameters selected with (*) are mandatory.*

- Attach form

Allows to relate a form to the case.

Parameter	Description
User Code	If it is not reported, the current user is assigned.
Values map	Report <field1><value1><field2><value2>…, where fieldN corresponds to form field names and valueN to the values that are assigned to each field. It has higher priority than DS_XML_CONTENT_PATH
XML	File path that contains the XML. It is mandatory if DS_VALUE_MAP is not reported
Annex Type	Annex type attached (*).
Create version if it exists and if it is unique	If when attaching the form, it already exists and is unique, a historical version of it is previously created.
Reset previous form and it is unique	Indicates that if there is a form associated with the case and it is unique, data is deleted from it.
Depends on the previous execution	The action is executed only if the previously defined actions were executed successfully.(*).

> *Note: The parameters marked with (*) are mandatory.*

- Check event as processed

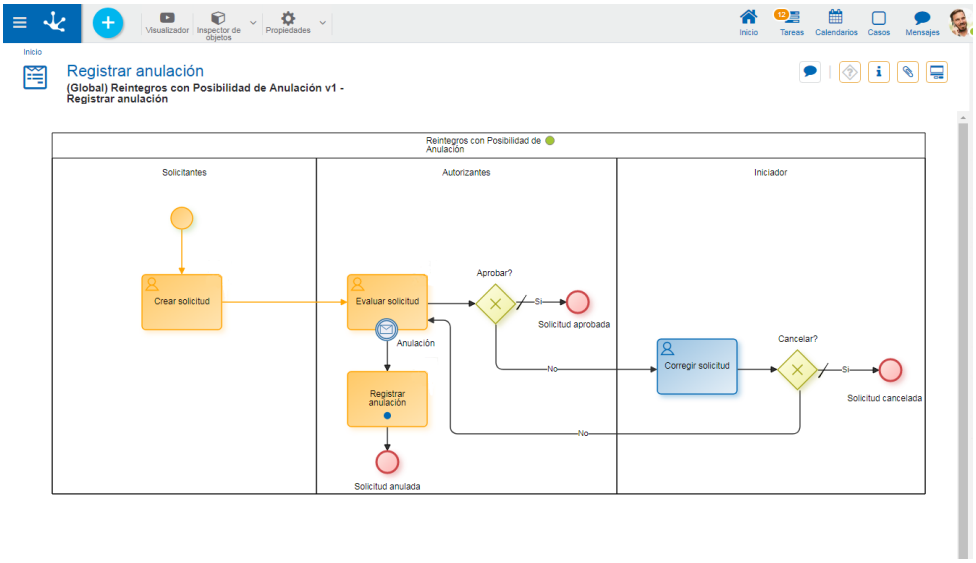
An automatic response is sent to the email received indicating that it has been processed.

Parameter	Description
Depends on the previous execution	Indicates if the action is executed only when the previously defined actions were executed successfully (*).

> *Note: The parameters marked with (*) are mandatory.*

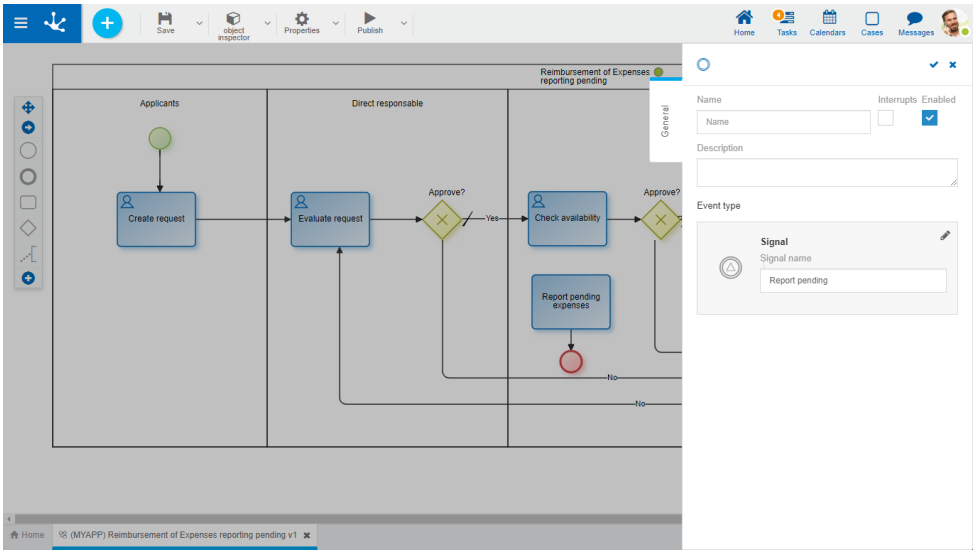
Examples of Use

In the "Refunds with Possibility of Cancellation" process, if the applicant notices a mistake in the request sent, they can send an email to cancel it, as long as it has not been approved yet.



3.6.3.9.3.2. Signal

The properties panel of the signal border event is displayed on the right side of the process modeler.



Properties

Name
Event name. It is displayed in the process next to the graphic element that represents the event.

Description
Field that allows documenting the event.

Interrupting
Defines if the event is interrupting or non-interrupting. In both cases the signal must be received when the current activity of the case is the one that has the border event.

- Interrupting: When the signal is received, the execution of the current activity is interrupted and the case follows the event output flow.
- Non-interrupting: When the signal is received, the execution of other tasks begins, defined as event output flow. The current activity is not interrupted and the case continues after it finishes.

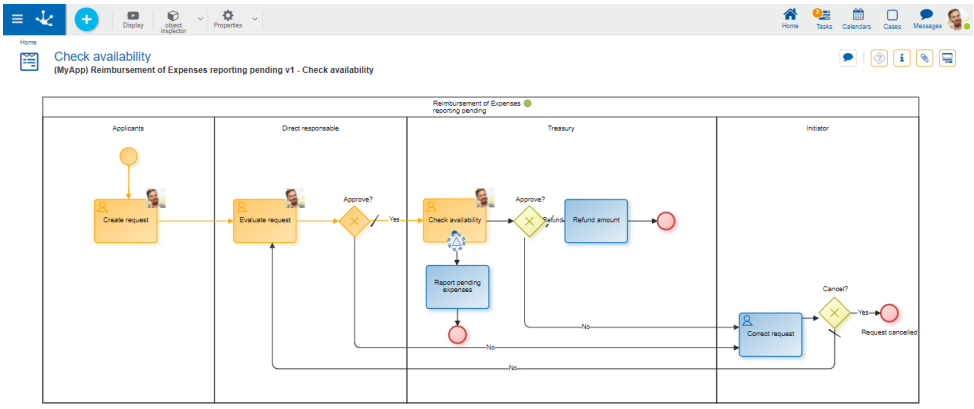
Enabled
Indicates if the event is activated to wait for the signal and execute.

Signal Code
It is the signal identifier. A character string is specified.

Examples of Use

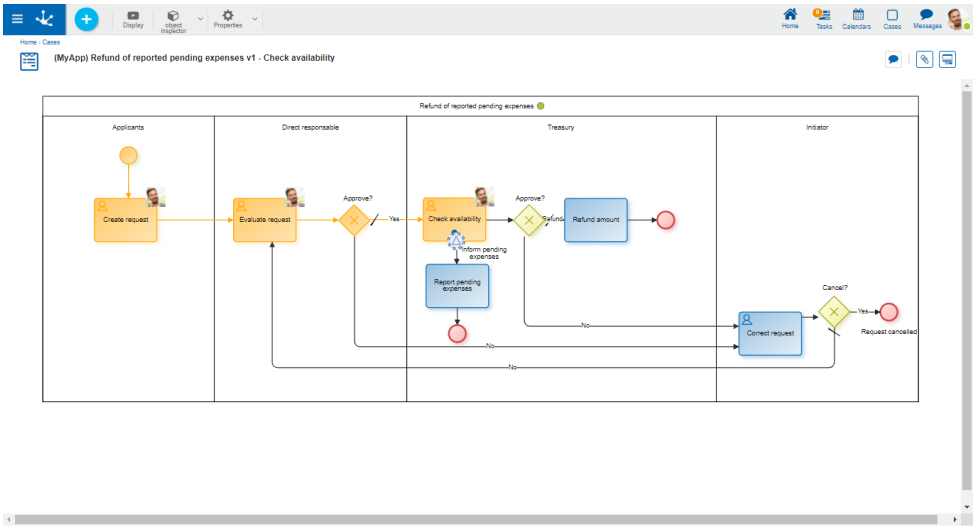
Interrupting Signal Border Event

In the "Refund with possible payment suspension" process, if the "Suspension of refunds" signal is received, the cases in the "Check documentation" activity are suspended and the process ends.



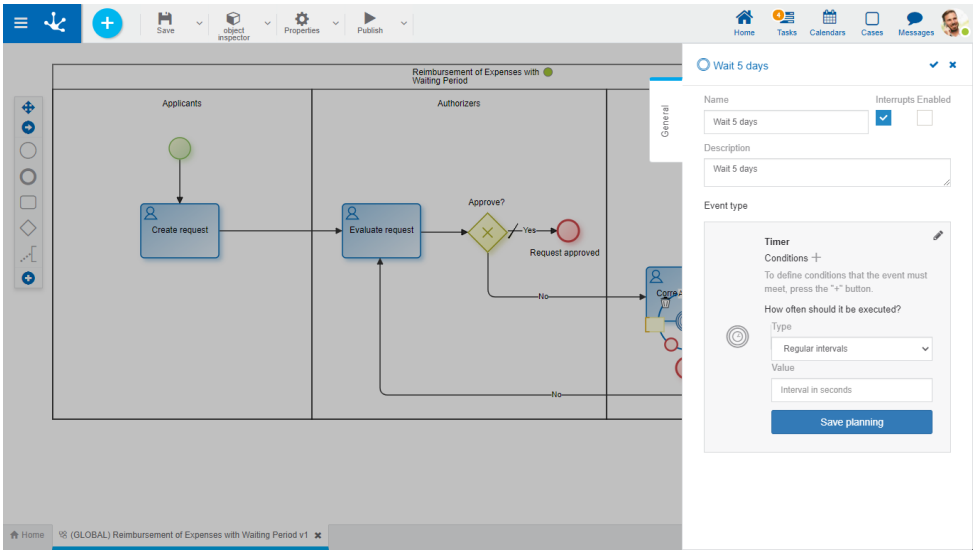
Interrupting Signal Border Event

When the "Inform pending expenses" signal is received, in the "Refund of reported pending expenses" process, a report of the cases in the "Check availability" activity is generated. The cases continue through the circuit defined in the process.



3.6.3.9.3.3. Timer

The properties panel of the timer border eventis displayed on the right side of the process modeler.



Properties

Name
Event name. It is displayed in the process next to the graphic element that represents the event.

Description
Field that allows documenting the event.

Interrupting
Defines if the event is interrupting or non-interrupting.

- Interrupting: When the conditions of the event are met, the execution of the current activity is interrupted and the case follows the event output flow.
- Non-interrupting: When the event conditions are met, the execution of other tasks begins, defined as event output flow. The current activity is not interrupted and the case advances after it finishes.

The event conditions are evaluated when the current activity of the case is the one that has the border event.

Enabled
Indicates if the case is enabled to be executed as scheduled, when the established conditions are met.

Conditions
They are the conditions that must be met for the event to be executed. If there are defined conditions, a list of them is displayed.

To add a condition, press the icon + and from the panel that opens, select Property and Operator in the corresponding drop-down lists. Report Value and where applicable, indicate in Parameter the field to be evaluated of the form associated with the process.
For this type of event, apply all the conditions.

How often should it execute?
Type
Allows to define the execution schedule for the event

- Regular intervals
- Daily at a specific time

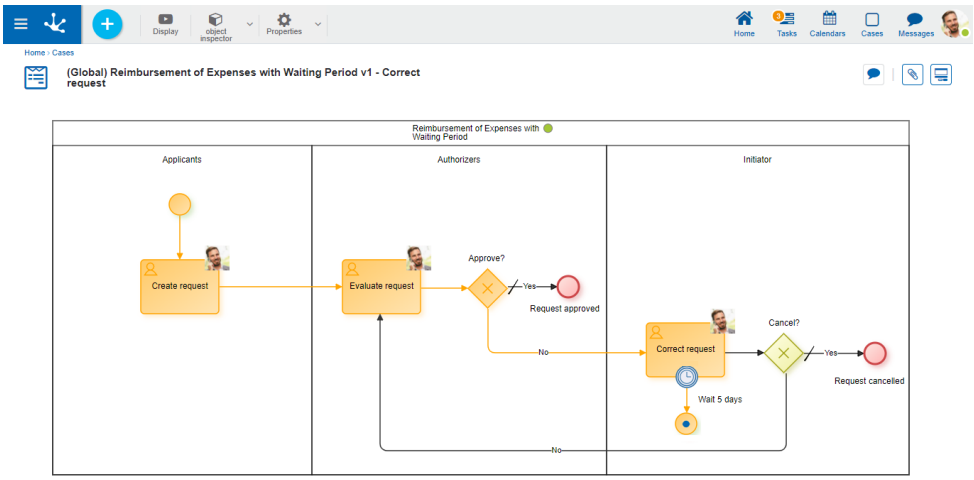
Value

Allows to define the interval value in seconds or at a specific time, according to the previously defined type.

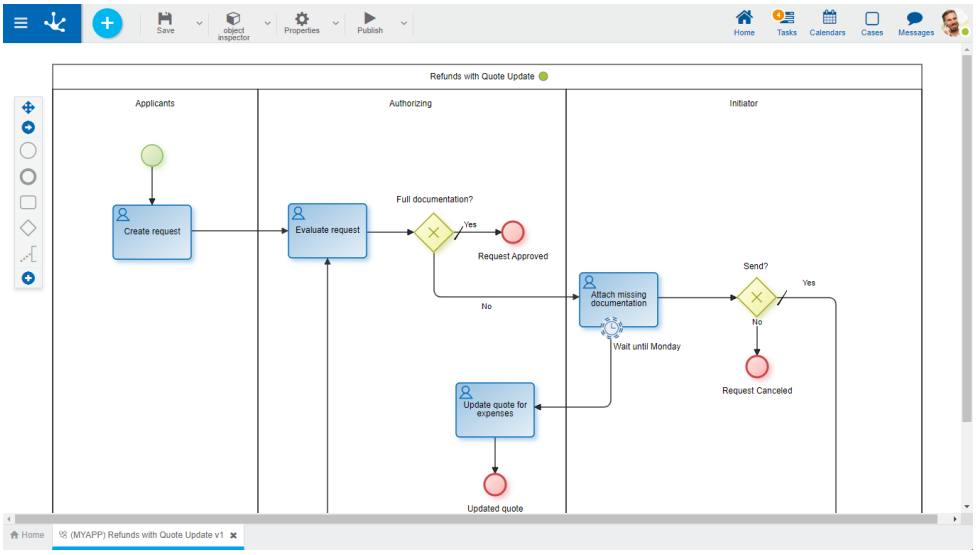
The "Save schedule" button must be pressed to keep the established schedule. It can be seen on the Scheduled Tasks Monitor.

Examples of Use

In the "Expenses Refund with Waiting Period" process, if the applicant receives the request for its modification, they have 5 days to correct it. If the deadline expires and the request was not resubmitted, the process ends automatically and the request expires.

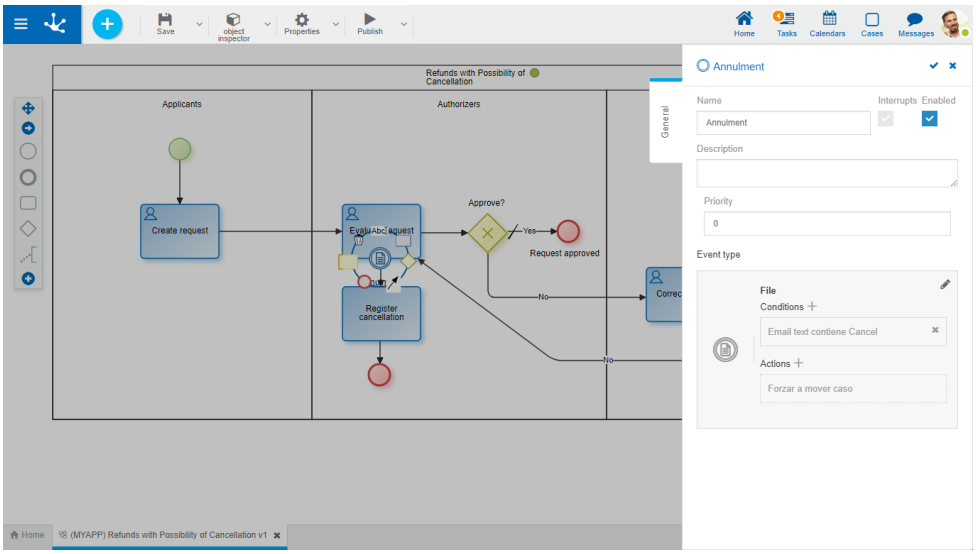


If in the "Refunds with Quote Update" process, while the case is in the "Attach missing documentation" activity, the conditions established in the timer border event are met, the quote is updated. When the applicant attaches the missing documentation, the case returns to the "Evaluate request" activity.



3.6.3.9.3.4. File

The properties panel of the file border eventis displayed on the right side of the process modeler.



Properties

Name

Event name. It is displayed in the process next to the graphic element that represents the event.

Description

Field that allows documenting the event.

Priority

Execution priority assigned to the event. It is reported as an integer, with 0 being the highest priority. If the event performs the action mark event as processed, events with lower priority are not executed.

Interrupting

The file border event is always interrupting. This means that if the established conditions are met, the event actions are executed and the case follows the event output flow.

Conditions

Allows to define conditions to be met for the event to be executed.

Actions

They are the actions that are carried out automatically when the event occurs.

- Move Case

The case affected by the event moves on to the next activity.

- Attach file

Allows to attach one or more files to the case.

Parameter	Description
Depends on the previous execution	Indicates if the action is executed only when the previously defined actions were executed successfully (*).

> *Note: The parameters marked with (*) are mandatory.*

- Attach form

Allows to relate a form to the case.

Parameter	Description
User Code	If it is not reported, the current user is assigned.
Values map	Report <field1><value1><field2><value2>…, where fieldN corresponds to form field names and valueN to the values that are assigned to each field. It has higher priority than DS_XML_CONTENT_PATH
XML	File path that contains the XML. It is mandatory if DS_VALUE_MAP is not reported
Annex Type	Annex type attached (*).

Parameter	Description
Create version if it exists and if it is unique	If when attaching the form, it already exists and is unique, a historical version of it is previously created.
Reset previous form and it is unique	Indicates that if there is a form associated with the case and it is unique, data is deleted from it.
Depends on the previous execution	The action is executed only if the previously defined actions were executed successfully.(*).

> *Note: The parameters marked with (*) are mandatory.*

- Check event as processed

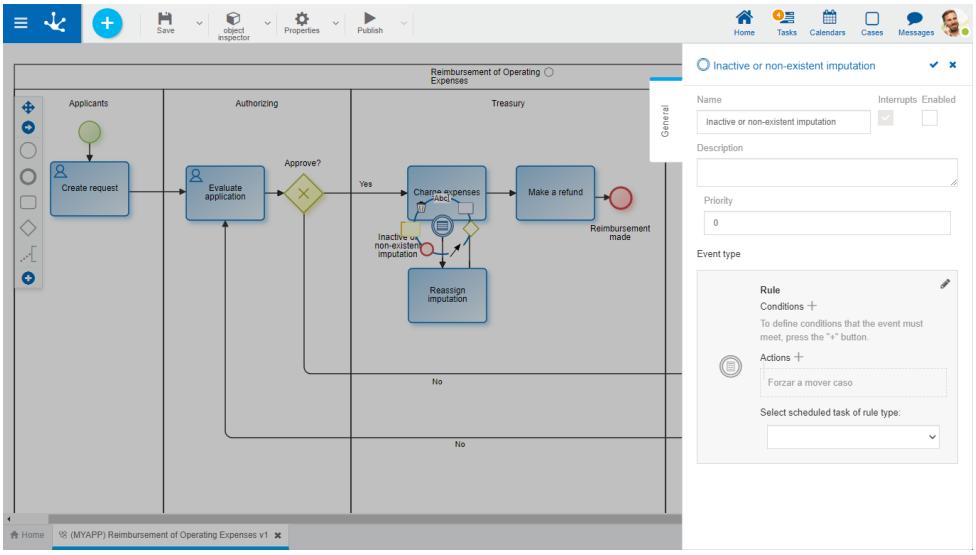
An automatic response is sent to the email received indicating that it has been processed.

Parameter	Description
Depends on the previous execution	Indicates if the action is executed only when the previously defined actions were executed successfully (*).

> *Note: The parameters marked with (*) are mandatory.*

3.6.3.9.3.5. Rule

The properties panel of the rule border eventis displayed on the right side of the process modeler.



Properties

Name
Event name. It is displayed in the process next to the graphic element that represents the event.

Description
Field that allows documenting the event.

Priority
Determines the priority of the event.

Interrupting
The rule border event is always interrupting. This means that if the established conditions are met, the event actions are executed and the case follows the event output flow.

Enabled
Indicates that the event is enabled to be executed according to the rule.

Conditions
They are the conditions that must be met for the event to be executed. If there are defined conditions, a list of them is displayed.

To add a condition, press the icon + and from the panel that opens, select Property and Operator in the corresponding drop-down lists. Report Value and where applicable, indicate in Parameter the field to be evaluated of the form associated with the process.

Actions
They are the actions that are carried out automatically when the event occurs.

- Move Case

The case affected by the event moves on to the next activity.

- Attach file

Allows to attach one or more files to the case.

Parameter	Description
File to attach	File attached to the case. (*)
Depends on the previous execution	Indicates if the action is executed only when the previously defined actions were executed successfully.(*)

> *Note: The parameters selected with (*) are mandatory.*

- Attach form

Allows to relate a form to the case.

Parameter	Description
User Code	If it is not reported, the current user is assigned.
Values map	Report <field1><value1><field2><value2>…, where fieldN corresponds to form field names and valueN to the values that are assigned to each field. It has higher priority than DS_XML_CONTENT_PATH
XML	File path that contains the XML. It is mandatory if DS_VALUE_MAP is not reported
Annex Type	Annex type attached (*).
Create version if it exists and if it is unique	If when attaching the form, it already exists and is unique, a historical version of it is previously created.
Reset previous form and it is unique	Indicates that if there is a form associated with the case and it is unique, data is deleted from it.
Depends on the previous execution	The action is executed only if the previously defined actions were executed successfully.(*).

> *Note: The parameters marked with (*) are mandatory.*

Select scheduled rule task

Allows to select the rule to use, which must be an advanced rule and must be published.

Example of Use

In the "Refund of Operating Expenses" process, if the "Charge expenses" activity is executed and a rule that detects inactive or non-existent expense charges is executed, charges are reassigned so that the case can move to the "Make a refund" activity.

3.6.3.9.4. End Events

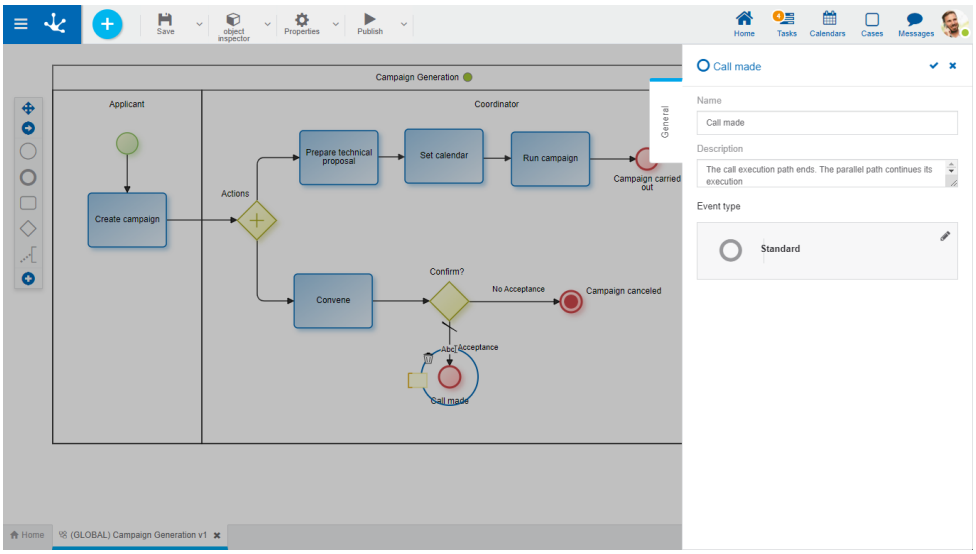
General Tab

In the "General" tab, common properties for all end events and the specific ones for each case are defined, which are enabled when assigning the type of end event.

- Standard
- Signal
- Terminal

3.6.3.9.4.1. Standard

The properties panel of the standard end event is displayed on the right side of the process modeler.



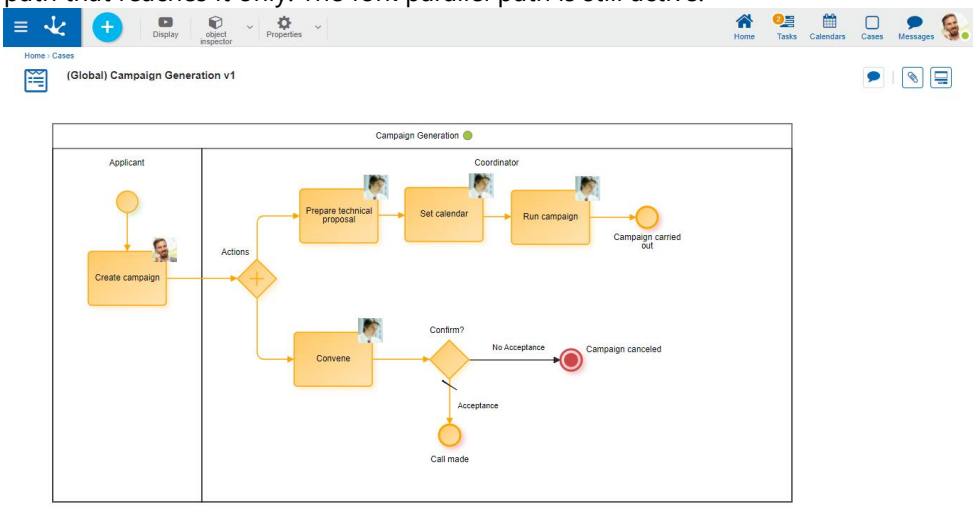
Properties

Name
Text that is displayed in the diagram next to the graphic element of the event.

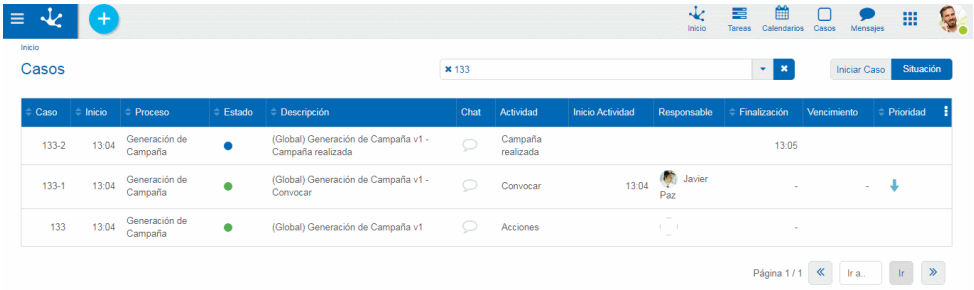
Description
Field that allows documenting the event.

Example of Use

The "Generate Campaign" process has two paths. The standard end event finishes the execution of the path that reaches it only. The fork parallel path is still active.

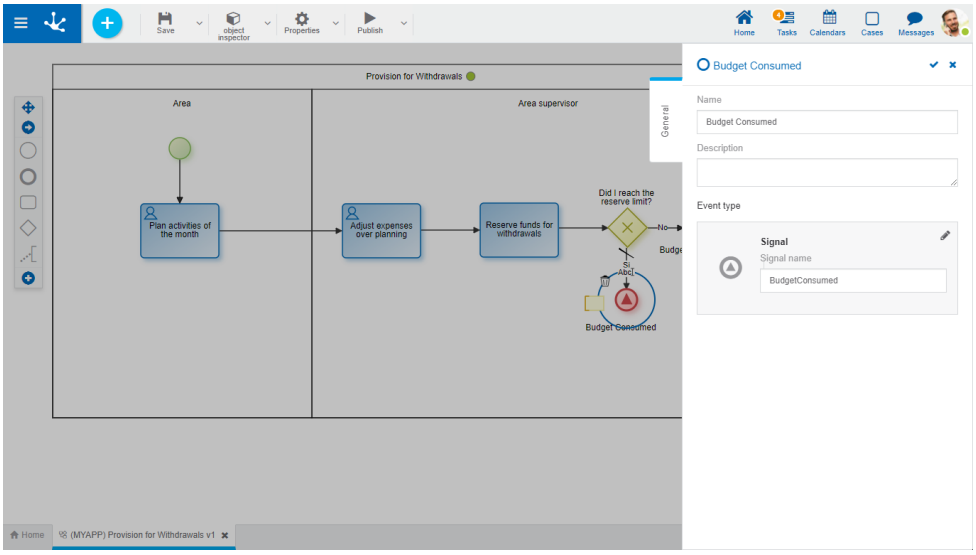


In the case show, the path end in which the standard end event is located can be seen. The parallel path is still active.



3.6.3.9.4.2. Signal

The properties panel of the signal end event is displayed on the right side of the process modeler.



Properties

Name

Text that is displayed in the diagram next to the graphic element of the event.
It is recommended to refer to the waiting signal.

Description

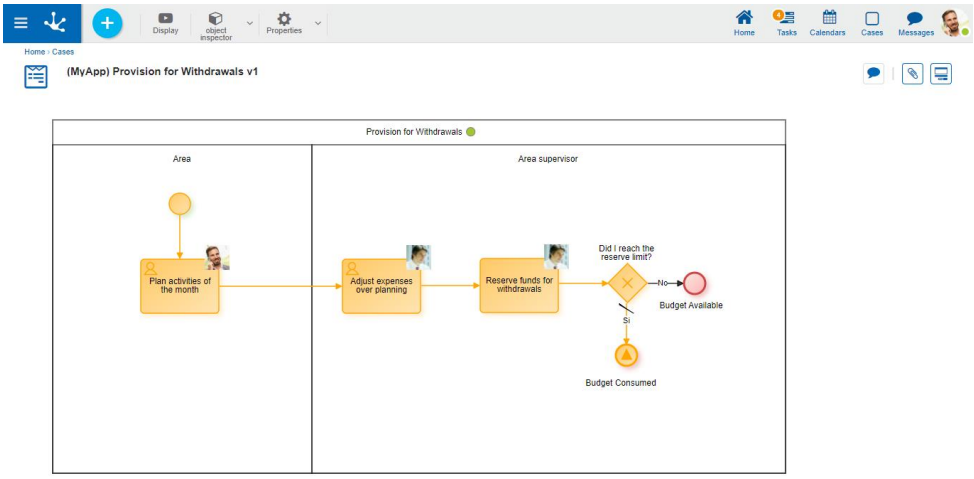
Text that allows documenting detailed information about the event.
The processes that send the signal can be informed.

Signal Name

Enter a string of characters representing the signal code to send.
The name entered must match the one configured when the signal is received.

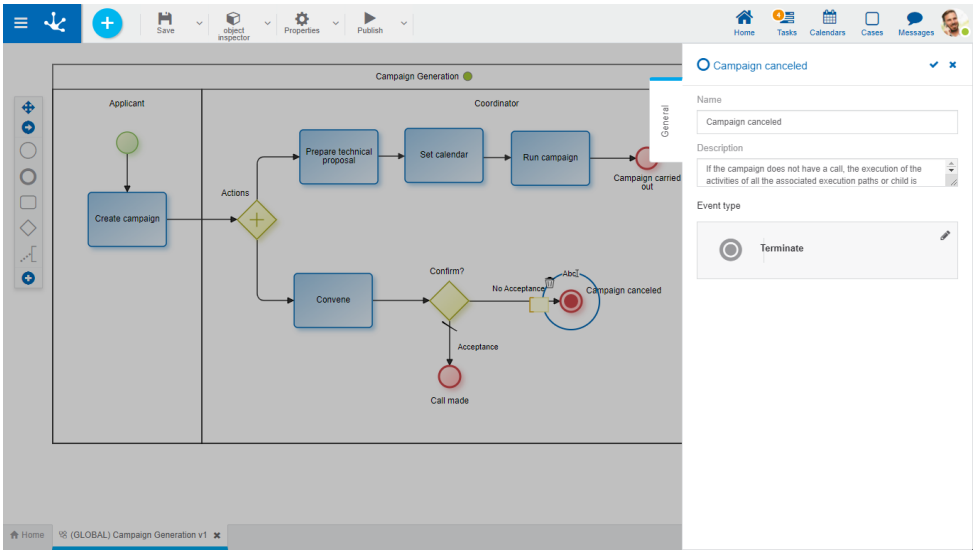
Example of Use

If the "Provision for Refunds" process reaches the availability limit, it sends the "Consumed Budget" signal. The "Actions Auditing" and "Budget Update" processes start when the signal start events receive the aforementioned signal.



3.6.3.9.4.3. Terminal

The properties panel of the terminate end event is displayed on the right side of the process modeler.



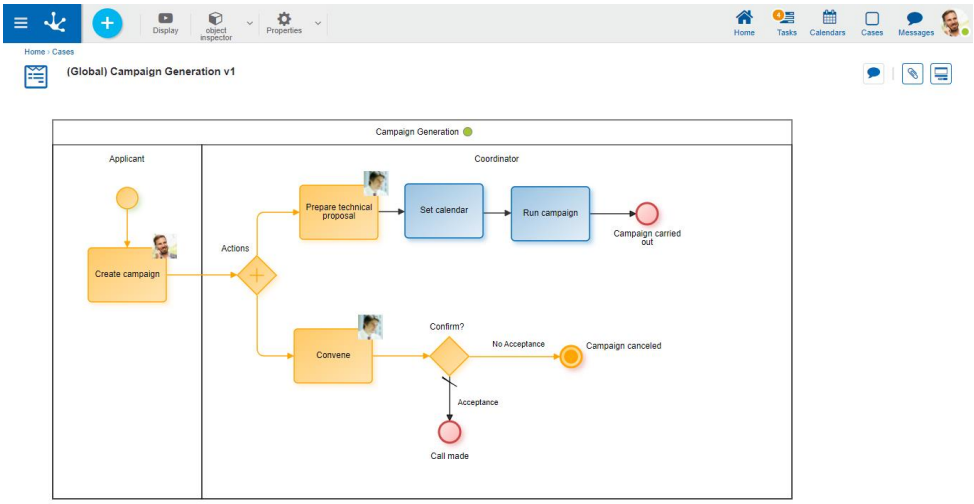
Properties

Name
Text that is displayed in the diagram next to the graphic element of the event.

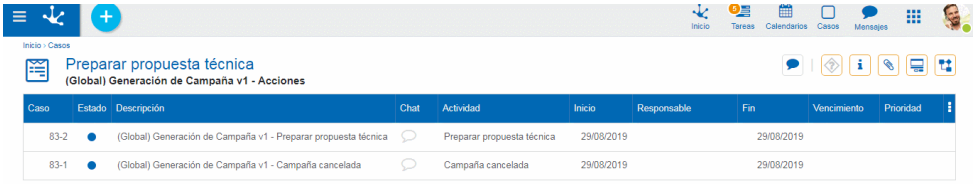
Description
Text that allows documenting detailed information about the event.

Example of Use

The "Campaign Generation" process, all paths are canceled when the evaluation of the call is not satisfactory.



In the case show, the termination of all fork paths can be verified.



3.6.3.10. Good Practices

Good practices for processes modeling in **Deyel** consist of a set of recommendations, to be met whenever possible.

They allow to build clearer and more understandable processes models, standardizing the use of the graphic elements that compose them.

- Processes Modeling
- Lanes Modeling
- Activities Modeling
- Events Modeling

- Flows Modeling
- Gateways Modeling

3.6.3.10.1. Processes Modeling

Good Practices

Name

- The name of the processes and sub processes must clearly describe their main purpose.
- Short names or abbreviations must not be used.

Diagrams Size

- Long diagrams do not allow to give a global perspective to the readers, they make reading difficult and don't communicate clearly the purpose of the processes.

Number of Activities

- Although various factors affect the complexity of a process, the number of activities is one of the most relevant factors.
- As a general criterion, diagrams with a maximum of 15 to 20 activities are recommended.

Unification of Activities

- If a set of consecutive tasks can be executed sequentially by the same person, at the same time, then these activities must be integrated into one.

Subprocesses

- The use of subprocesses is recommended to group activities with the same purpose.

Activities Alignment

- It is recommended to keep the activities aligned with each other, horizontally and vertically. This practice allows ordering the diagrams, keeping them neat visually.

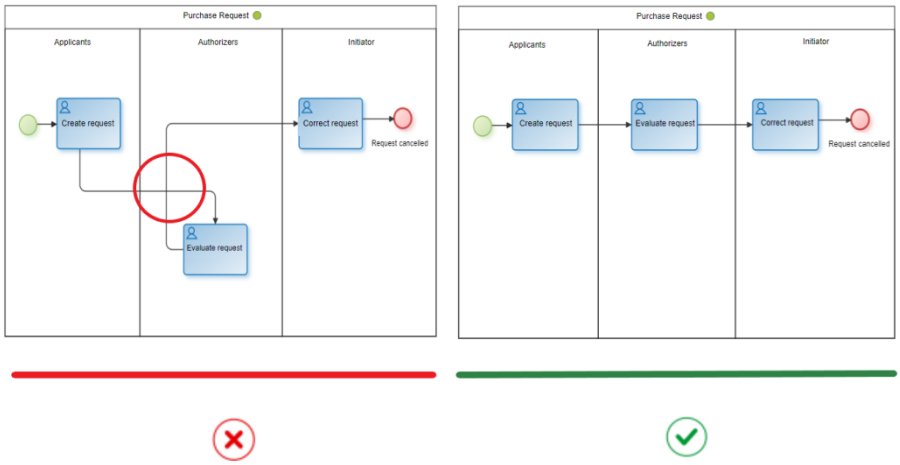
Standard Size of Elements.

- It is recommended not to change the size of diagram elements.
- Keeping the standard sizes avoids discrepancies and keeps the visual elements homogeneous.

Flow Orientation

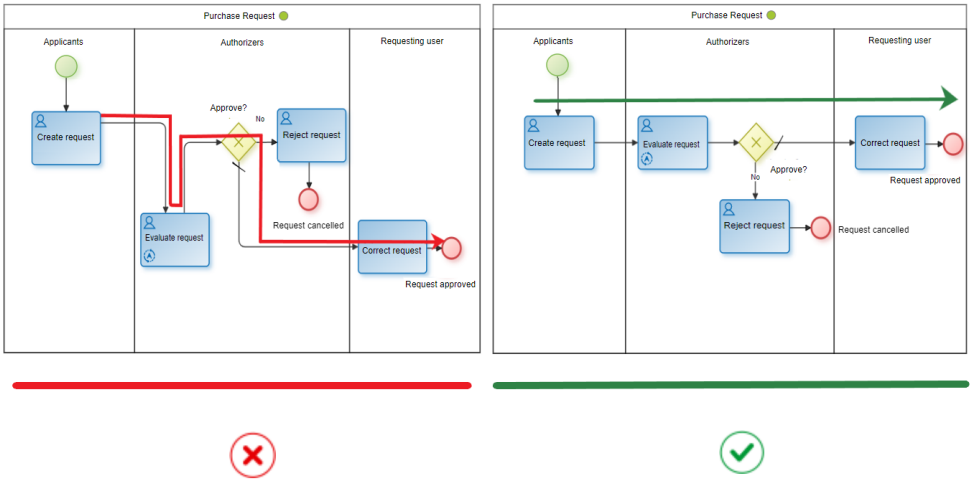
- For people who read from left to right, it is intuitive to follow a logical flow from left to right or from top to bottom. It is recommended to diagram the process flows in this direction.

- The crossing of connectors must be avoided, maintaining a chronological sequence and a consistent flow direction.



Primary Scenario Clarity

- The main road should be easily identified when reading the diagram, so it is recommended to diagram the main road first and then add the alternative roads.



Links

- It is recommended to use this type of event to avoid complexity in the flows.

Throw Link

Catch Link

Artifacts

- It is recommended not to overuse the objects that serve to document the process. Notations, associations, groups and data objects are elements that can clarify the diagram, but the excessive use of these types of elements can increase the visual complexity of the model.
- In order not to use a large number of artifacts, it is recommended to document the details within each activity or each element of the model.

3.6.3.10.2. Lanes Modeling

Good Practices

Name

- The lane name must be chosen carefully, so that it accurately represents the participant of the process.
- It is recommended to use the description field to document the responsibility.

Participant Type

- Avoid using responsibles of type user as much as possible.
- It is recommended to use roles or agents, that give greater possibilities to changes.

Need

- A lane must be created only if at least one intermediate task or event is executed on it.
- It is recommended not to create lanes to represent participants who only run automatic activities.

3.6.3.10.3. Activities Modeling

Good Practices

Name

- The activities name is made up of an infinitive verb and an object.

- The name must clearly identify the objective of the activity. For example, "Create request", "Authorize request", "Notify rejection", and so on.

- The first letter must be used in uppercase and the others in lowercase. Short names or abbreviations must not be used.

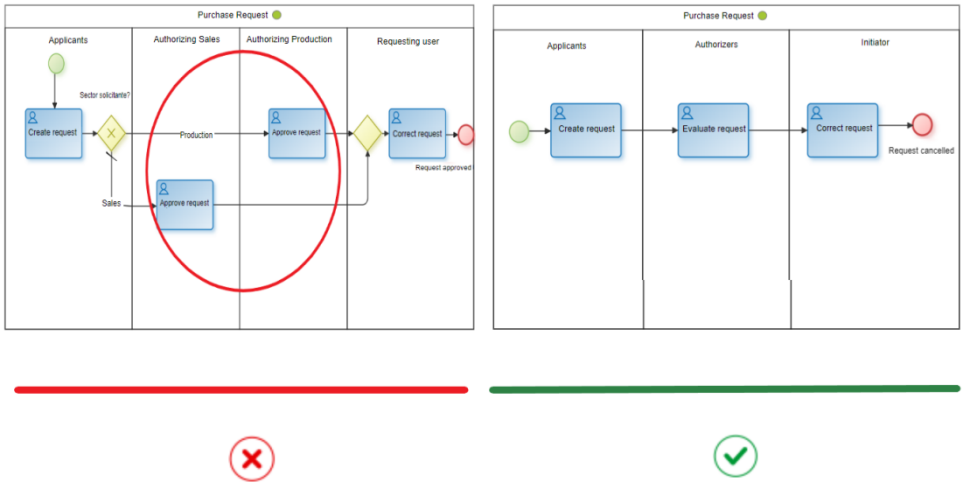
- It is recommended to avoid long names, they usually denote that certain details of the activity affect their name. The name must represent the objective while details are documented within the activity.

- The use of "and" and "or" in the names of activities must be avoided. These types of compound names can mean that they are actually two different activities.
  For example, if the activity is called “Review and approve request”, you must consider which is the correct option. If the same participant performs both actions at the same time, then the most relevant action must be chosen and used to name the activity, then it can be assumed that the primary objective is approval and the review is a subtask. The correct name is “Approve request”. On the other hand, if it is considered that both actions have the same relevance and are made at different times, then it is convenient to separate them as “Review request” and “Approve request”.

Repetition

- Multiple instances of the same task should not be diagrammed to represent multiple participants.

- It is recommended to use roles or agents so that the participant assignment is made accordingly, when executing the process.



Typing

- It must be indicated each activity type.

- It is recommended to be careful selecting the appropriate type, which allows a correct interpretation of the diagram.

Automatic Tasks vs Automatic Actions

- There are certain actions, such as sending an email, executing a rule, etc. that can be modeled in different ways in **Deyel**:
  a) Representing them explicitly in the model, through shipping type activities, business rule, services, etc.
  b) Specifying automatic actions that are executed when starting or ending a user activity or manual activity.

- The recommendation is that the tasks to be viewed individually have to be modeled as different activities and the type of each one is selected.

- Those others that do not need to be visually explicit, should be modeled as automatic actions of user activity. For example, a rule that retrieves the initial value of a field before executing the user activity,

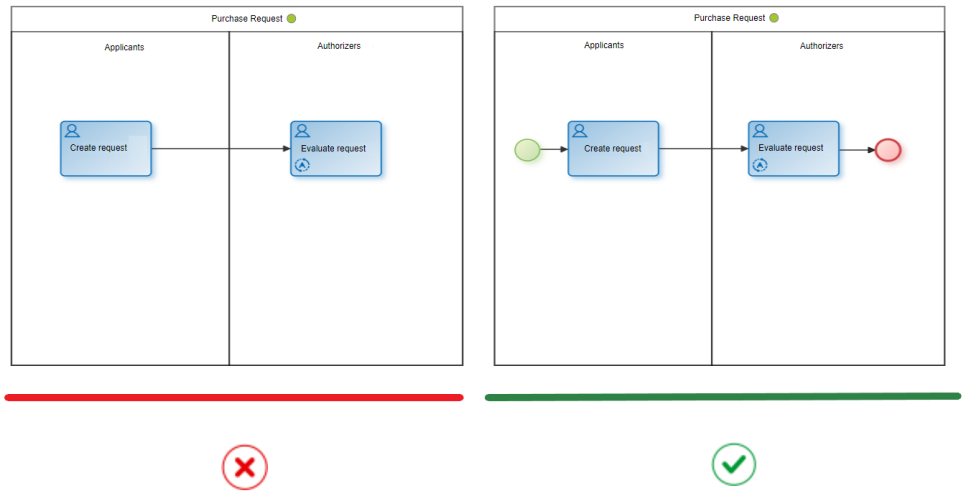
- It is important to maintain a uniform criterion in this regard and also consider that long diagrams are not convenient.

3.6.3.10.4. Events Modeling

Good Practices

Start and End Events

- It is recommended to always use the start and end events in each process and subprocess to represent its beginning and end.



Name

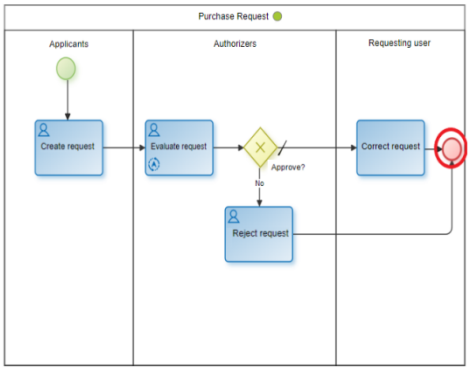
- The start and end events must not be tagged when they are unique. It is very common to name them as "Start of the process" and "End of the process" but this is redundant and unnecessary.

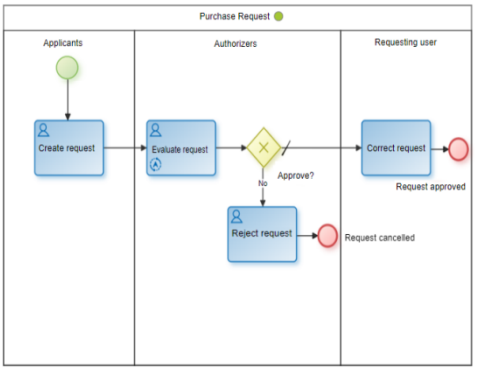
- Events must be identified when multiple start or end events are used.

- The start events must have labels indicating the condition or fact that triggers the start of the process.

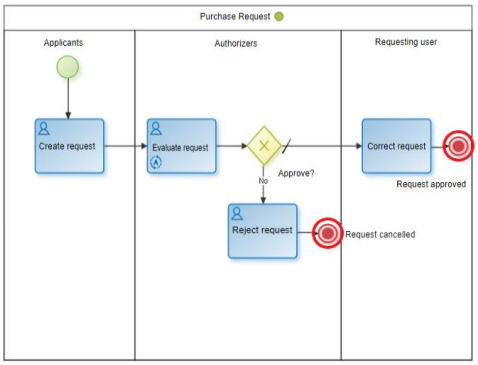
- The end events must be named indicating the final state of the process. In general the name must use a noun, which is the business object, along with a verb in the past tense. For example: "Request approved", "Request canceled", etc.

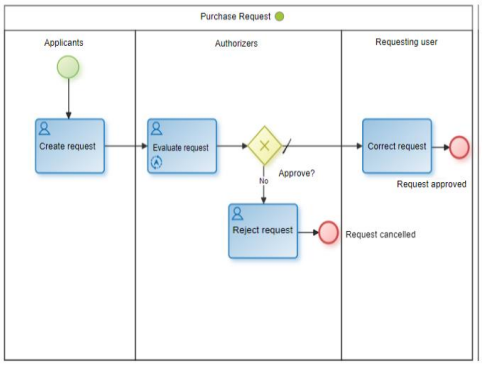
- It is recommended to identify the final successful and unsuccessful states.





- Terminal end events must only be used when strictly necessary. They are used to model situations where multiple alternative paths are enabled and only the completion of one of them is required to complete the entire process.





3.6.3.10.5. Flows Modeling

Good Practices

Use of Legends

- It is recommended to use the legends to represent the condition through which a flow is passing.
- Legends have to be visually associated with the flow and are not confused with other legends.

Avoid Crossings

- As far as possible the crossings of flows should be avoided.
- If necessary, let it be in places where the legends of the flows are not confused.

Avoid Diagonal Flows

- It is always preferable to use "L" shaped flows.
- There can be multiple vertices in the flow and it is recommended to minimize their number.

Anchor Points or Ports

- Central anchor points have to be used in activities.
  - Flows enter from the top side and the left side.
  - Flows come out from the bottom side and the right side.
- In the gateways it is recommended to use the anchor points of the vertices.
  - Flows enter through the top or left vertices, using the same entry point.
  - It is recommended to distribute the outgoing flows neatly.
  - The gateway name must not be below the flows.

3.6.3.10.6. Gateways Modeling

Good Practices

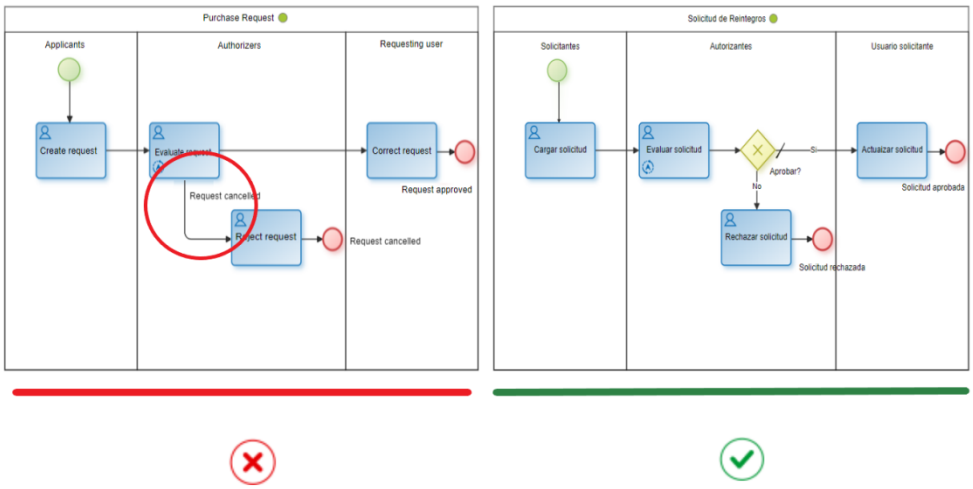
Name

- All divergence gateways must be named.
- A question is generally used that represents the decision or logical condition evaluated. For example: "Authorized Request?".

- All outgoing flows must be named.
- Their name must represent the responses or possible values of the condition evaluated.
  For example, if a gateway is called "Request Authorized?", Its outgoing flows should be called "Yes", "No".

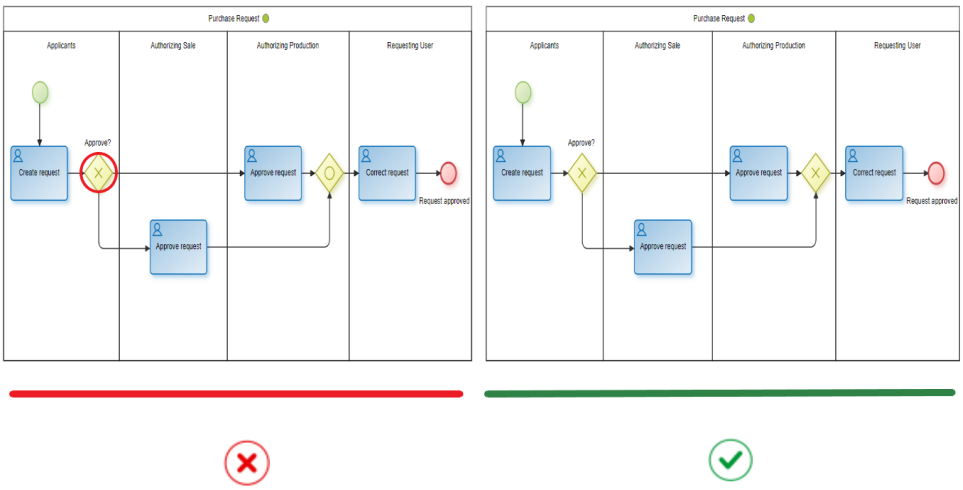
Gateways for Forks

- Flows must not be branched using tasks.
- Gateways should always be used for this purpose.



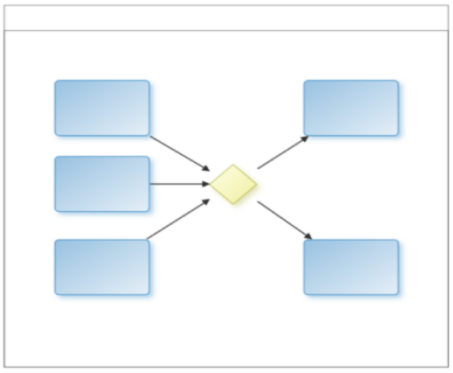
Gateways for Synchronization

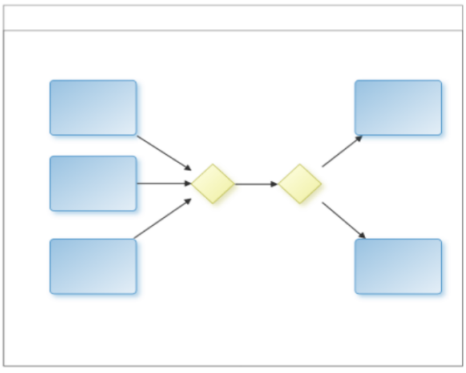
- To synchronize flows, it is recommended to always use the same type of gateway that is used to fork them.



Separate the forks and Synchronizations

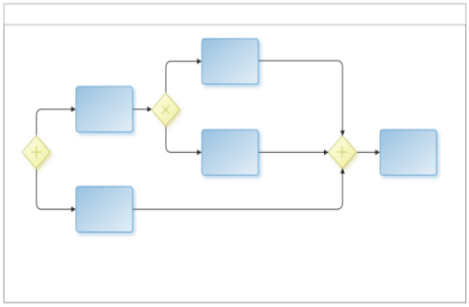
- It is recommended not to use gateways to join and separate flows at the same time.

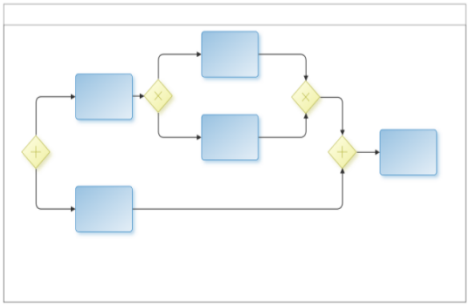




Balancing

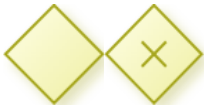
- Flow branches must be equivalently synchronized.
- The number of flows leaving a fork must be equal to the number of flows that are received in a synchronization.





Exclusive Gateway

Has two representations. The same representation must be used for the entire definition of a process.



- It can be used as a fork or synchronization element.

- It is used to represent a fork in the flow, where there are several possible alternatives but only one should be considered.
  They are mutually exclusive alternatives.

- These gateways require the definition of a default flow.
  This flow represents the path taken when none of the other conditions are met.
  We recommend that the most common path, the most probable, is the one that is modeled as the default flow.

- The exclusive gateway can be defined in pairs, one as a divergent element to activate several mutually exclusive paths and if these paths converge, a gateway of this type must be used as a convergent element, to graphically represent the concurrence of the flows.

Parallel Gateway

Its graphic representation is as follows:

- It can be used as a fork or synchronization element.

- When used as a fork, the output flows do not require conditions as all paths must be followed.
  Likewise, it is recommended to put a legend in each flow that represents the condition that was met in order to be able to transit such flow.

- When the gateway is used as a convergent element (synchronization), all the predecessor activities are expected to finish their execution.

- The parallel gateway must be defined in pairs, one as a divergent element to activate several parallel paths, and the other as a convergent element, to synchronize the previously activated paths.

Inclusive Gateway

Its graphic representation is as follows:

- It can be used as a fork or synchronization element.

- The inclusive gateway as a fork element is used at a point in the process where one or more alternatives may be viable based on a condition.

  For example, after creating a document, it must be approved by the corresponding offices according to the type of document created.
  To implement the multi-decision pattern it is necessary to use the inclusive gateway.
  This gateway allows one or more paths to be enabled according to the evaluation of the conditions of each flow.

- If none of the conditions are met, a default flow can optionally be defined, that takes into account this situation.
  If this default flow is not defined, then an error is obtained stating that there are no viable alternatives.

- The inclusive gateway as a convergence element is activated if at least one case reaches the inclusive gateway and if some other case of the processes execution that could reach the gateway (the canceled threads are not considered) arrived previously.

- The inclusive gateways can be defined in pairs. That is, one as a divergent element that generates parallel activities, and then that converge in an inclusive gateway of convergence, which synchronizes the previous paths.

3.6.3.10.6.1. Patterns

Patterns for processes design are examples that show how to connect activities to solve a common problem.

Parallel Paths

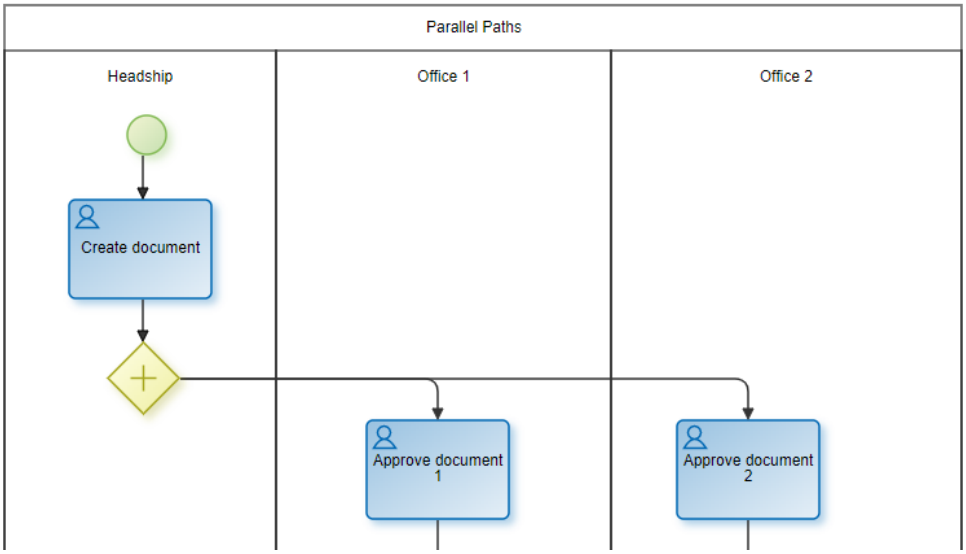
A parallel path is a point in the process execution where the execution flow is divided into two or more flows that are executed in parallel.

Example

A document is created in one office, which requires the approval of two other offices.

Implementation

To implement parallel paths, it is necessary to use a parallel gateway. This gateway creates all alternative paths without evaluating conditions.



Synchronization

Synchronization is a point in the process where two or more different flows of the process are joined into a single flow. It is called synchronization because it waits for all the flows to be joined to complete before proceeding to the next activity.

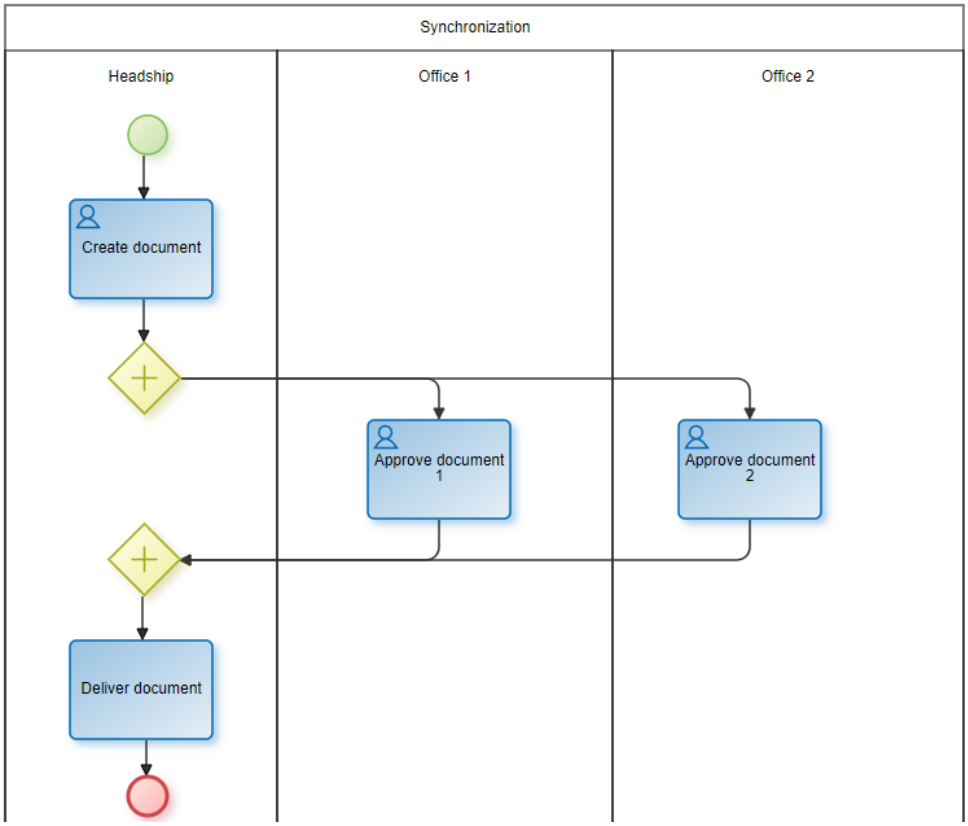
Example

The document created by one office must be delivered after it has been approved by two other offices.

Implementation

In the example we use the parallel gateway as convergence which means that it synchronizes the previously activated paths.

The synchronization pattern can also be modeled with the exclusive and inclusive gateway, according to the needs of the business.



Exclusive Decision

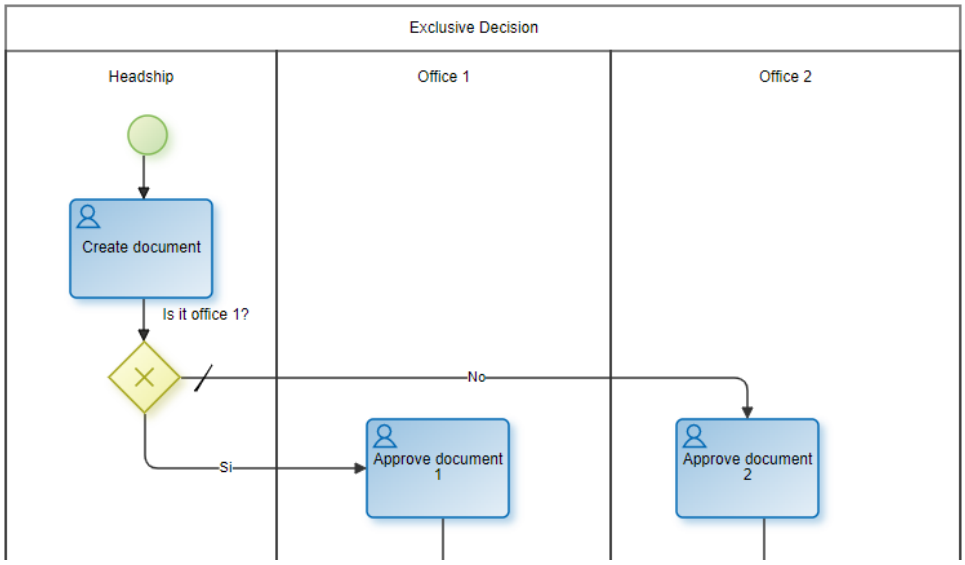
Exclusive decision is a point in the process where a path is chosen from several paths based on a condition or information from the process.

Example

When creating a document, it must be approved by an office that corresponds to the type of document created.

Implementation

The exclusive decision pattern can be modeled with the exclusive gateway. The exclusive decision has multiple output flows, but only one of them can be made based on the conditions of each stream.



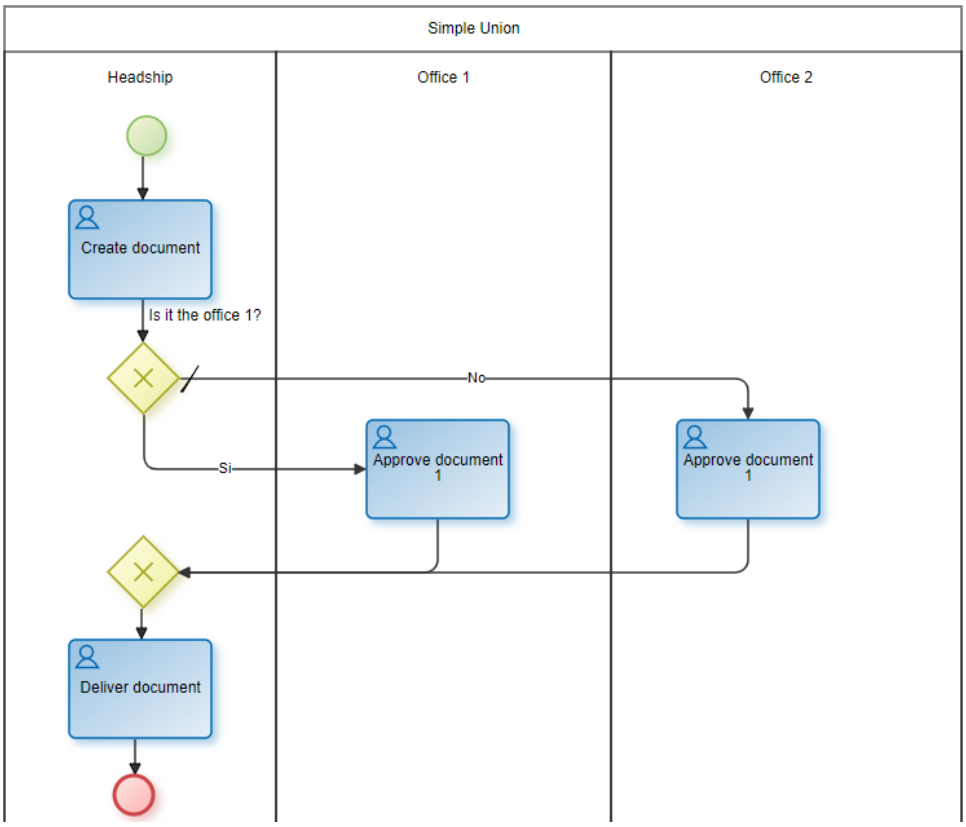
Simple Union

Simple union is a point in the process where two or more alternate paths meet without synchronization. This pattern assumes that of the different paths only one is executed.

Example

After creating a document, it must be approved by the corresponding offices according to the type of document created. Once approved by the corresponding office, it must be delivered by the office that created it.

Implementation



Multi-Decision

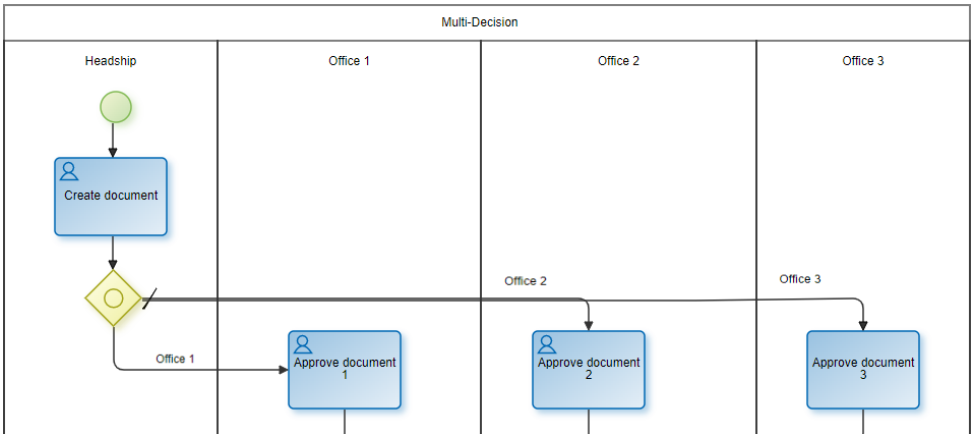
The multi-decision pattern is used to model a point in the process where a set of paths are chosen based on a condition.

Example

After creating a document, it must be approved by the corresponding office or offices according to the type of document created.

Implementation

To implement the multi-decision pattern it is necessary to use the inclusive gateway. This gateway allows one or more paths to be enabled according to the evaluation of the conditions of each flow.



Synchronized Structured Union

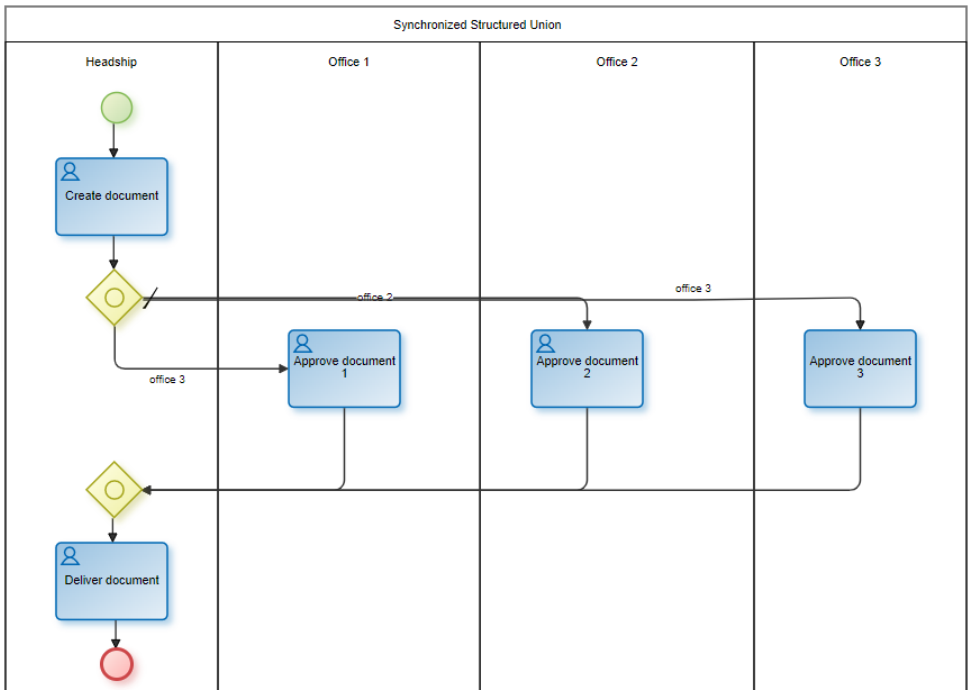
It is a point in the process where multiple paths that were previously activated in the process converge in a single thread of execution.

Example

Once the document has been approved by the appropriate office or offices, it must be delivered by the office that created the document.

Implementation

To implement this pattern it is necessary to use two inclusive gateways, one for divergence (activate some paths) and the other for synchronization (synchronize the activated paths).



Multi-Merge

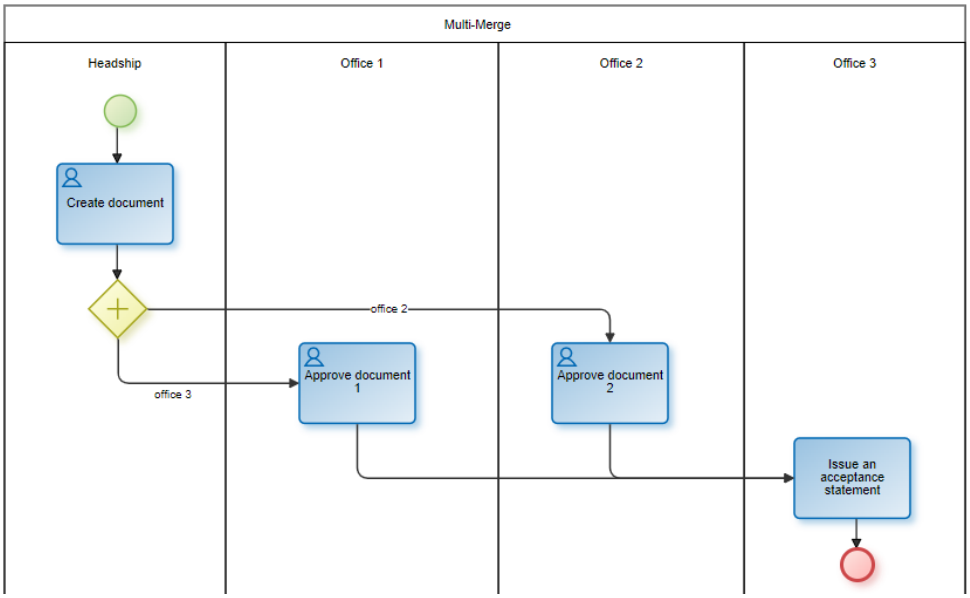
The multi-merge pattern is used to model the convergence of two or more paths in a single path. Each time an input path is activated it activates the next activity in the execution flow.

Example

After creating a document, it must be approved by two offices. Each time an office approves it, a third office must send an approval release.

Implementation

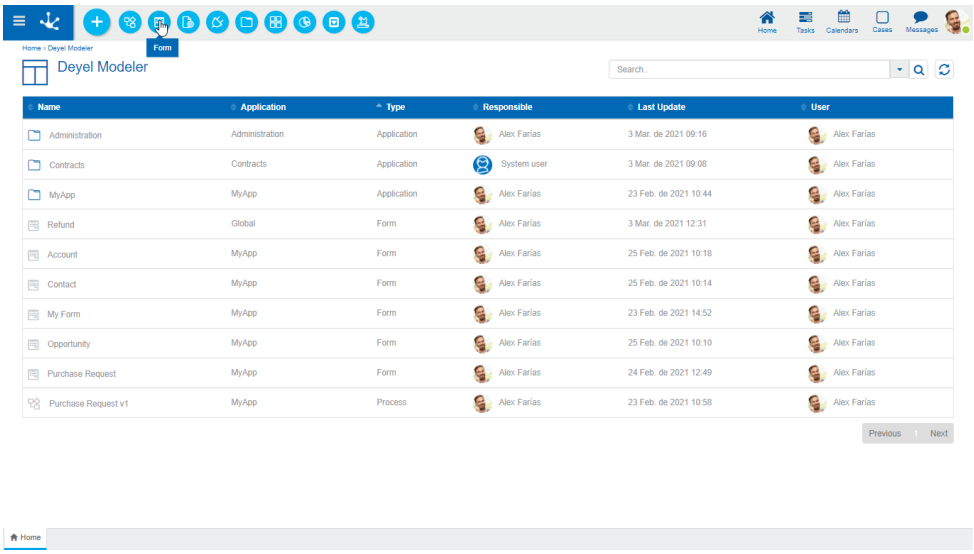
The pattern uses a parallel gateway to enable two paths. Each time the activities on one path are executed, the next activity will be as well. In this case, the "Send approval statement" activity is executed twice.



3.6.4. Forms Modeling

Phase 2: Forms Modeling

The form modeler is a tool that allows to graphically design forms, as well as define the characteristics of their fields and behavior in the associated processes.



Its main characteristic lies in the simplicity to manage the functionality associated with the forms, facilitating their progress through the different states that indicate its modeling situation and productive use.

The general characteristics of the forms modeling environment and the main elements that compose it are described in the topics:

- Modeling Facilities
- Form Properties
- Element Properties

A new form can be created from the icon corresponding to the context menu.

- From the icon
- From the menu displayed with the option "Form"

An existing form is modified by selecting it from the grid with the "Open" button.

3.6.4.1. Modeling Facilities

Phase 2: Forms Modeling > Forms modeling

New Form

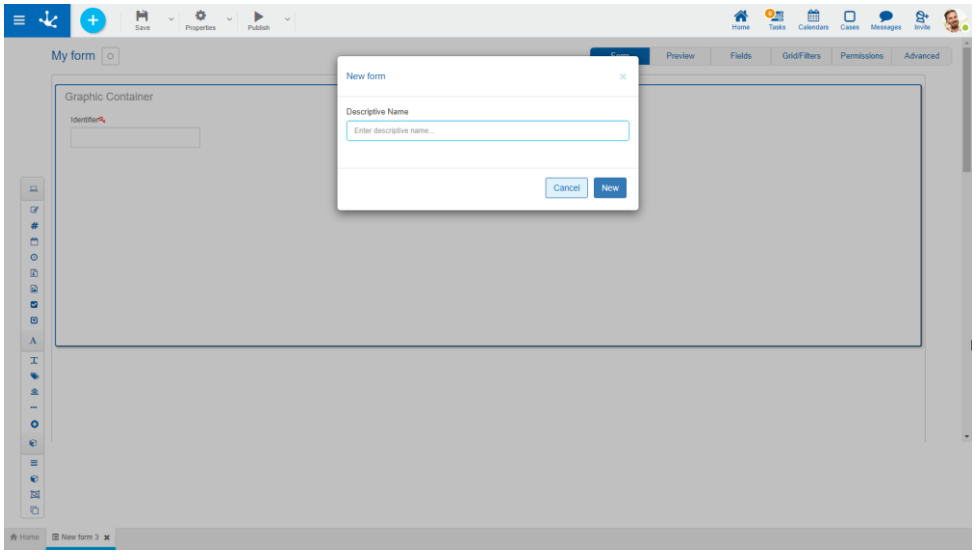
The modeler user defines a new form, which after being published is available to be used in the portal.

It can be modeled simultaneously:

- Data model layer
  Involves modeling all the fields with associated data types, field identifier, fields used as filters in searches, field data lengths, behavior and other validations related to business rules.

- Form display layer
  As its name implies, it is entirely related to the visual aspect of the form. This implies that you have control and decision about the arrangement, order, and width of the different fields and containers, interactive buttons, titles, labels, and informational messages, among other topics. There is also the option of displaying or not, the different sections or fields of a form and of using accordions.

When the form is created, a panel opens, allowing the entry of its name.



Workspace Sections

- Top Toolbar

- Form Information
  - Name
  - State
  - Locking

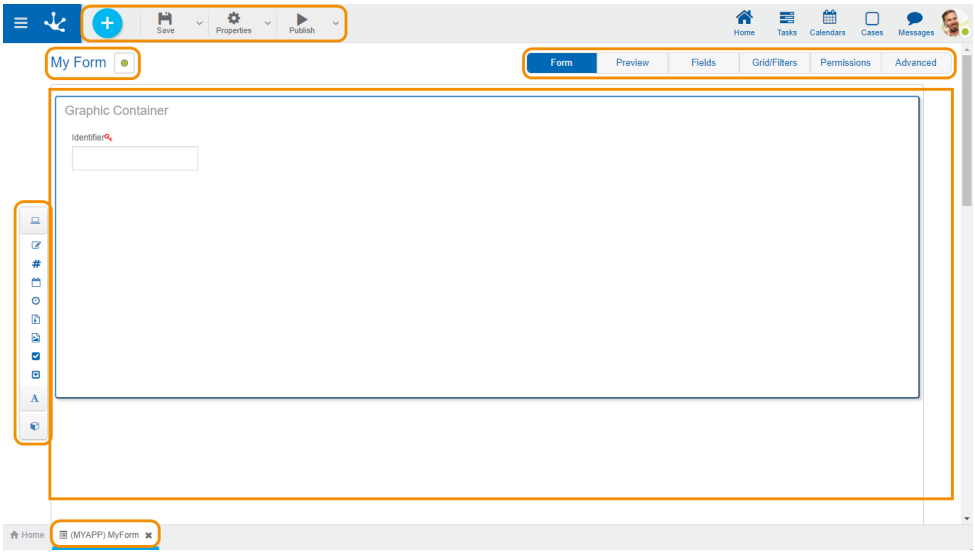
- Graphic Modeling Area
  - Graphic Container
  - Identifier

  Every new form contains an identifier field defined by default within the area of graphic modeling. It has the following characteristics:

  - Its data type is integer with a length of 10.
  - It is autonumeric, increasing value, starting from 1.
  - It cannot be deleted.
  - Only behavior properties can be modeled Visible and Validation.
  - It cannot be included within a "Multiple occurrence" type group.
  - Properties cannot be modeled Default Value, Type of content and Security.
  - Relations cannot be modeled.

- Design Options

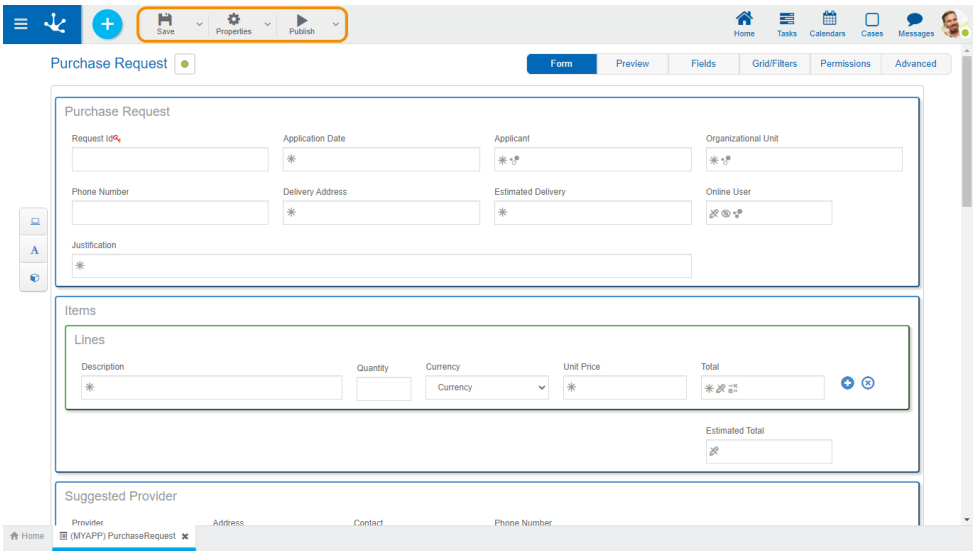
- Side Toolbar



3.6.4.1.1. Top Toolbar

This toolbar contains icons and submenus from which you can perform operations on the object. Depending on its state, some options may be disabled.

The bar is displayed on the toolbar of **Deyel**.



Save

This icon allows to save the object in the repository of **Deyel**, leaving its state as "Draft" or "Modified". If certain conditions are met, the modeler user receives a message indicating that the operation was performed correctly, otherwise they receive an explanatory message.

Main Conditions

- The application of the object must exist.

- The name must be unique in the application.
- The object must not be locked by another user.
- The object's permissions must exist.
- Modeled fields in process activities should not be eliminated as mandatory, whether they are used as parameters in automatic activities, in flow conditions or messages, in embedded rules of other fields in the form or in process activities, or asrelated attributes on entities.

To the right of the icon the option to open the submenu.

Properties

Open the form properties panel.

To the right of the icon the option to open the submenu.

Publish

By means of this icon the object goes to "Published" state, after verifying a set of additional conditions to those of the "Save" operation. The modeling user receives the corresponding message, indicating the result of the operation.

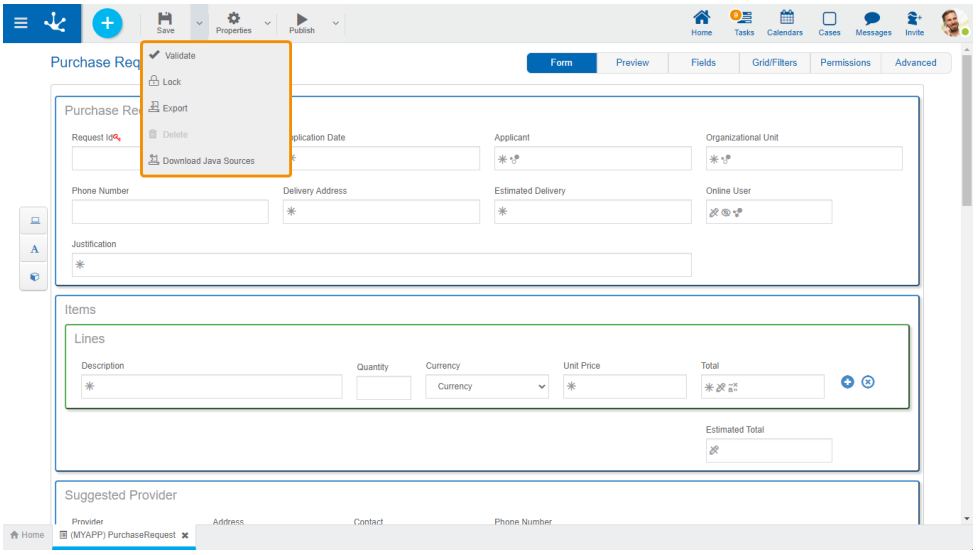
Main Conditions

- The entities and related rules must be published.
- Relations to published processes should not be removed.
- If new instances are created during the modeling of an already published form but without data, the form cannot be published again.
- If instances of a form are created while it is being modeled, it cannot be saved or published without opening it again in the modeler. Another option to save or publish the modeling in progress can be to delete instances.

To the right of the icon the option to open the submenu.

3.6.4.1.1.1. Save Submenu

This submenu is opened by pressing the icon immediately to the right of the icon corresponding to "Save".



Validate

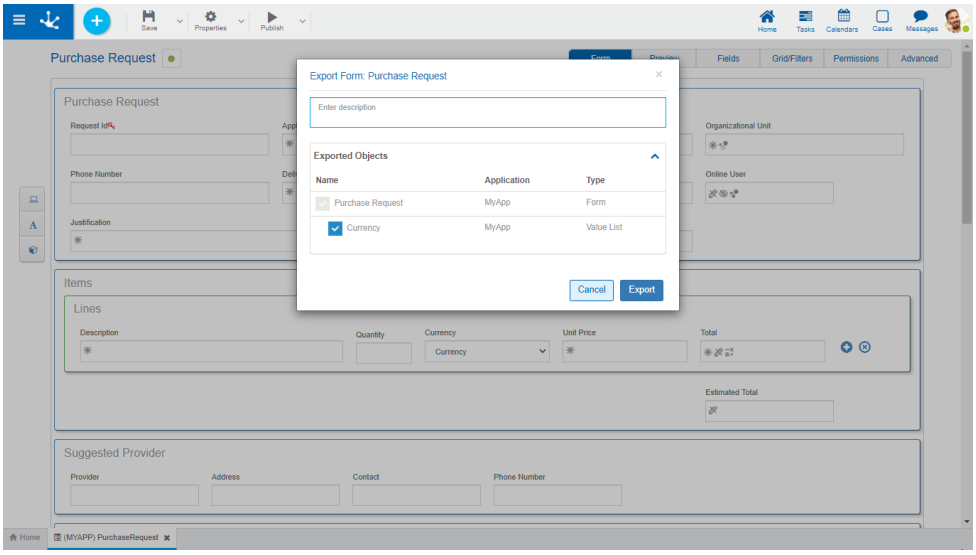
This icon validates if the form is ready to be published, that is, the validations performed are applied at the moment of publication and the result of the same is reported.

/ Lock unlock

- It allows blocking a form to ensure that no one can modify it until the person who is using it unlocks it, that is, releases it.
- Allows you to unlock a form so that another user can modify it.

Export

This icon opens a window for the user to select and confirm the export of the object and the related objects included in it.



Description

In this property a text explaining the reason for the operation can be entered.
This text can be modified upon import and is displayed in the description column of the export record.

Exported Objects

By expanding the container, the objects related to the form being exported are shown. Objects not meant to be exported are unchecked.
The related objects that can be exported along with the form are:

- Value lists associated with fields.
- Advanced rules used in fields, from the "Relation" tab of the field properties or included in embedded rules.

You must press the "Cancel" button to leave the export without effect or the "Export" button to finish the operation.

> *Forms corresponding to related entities are not included in the export.*

Delete

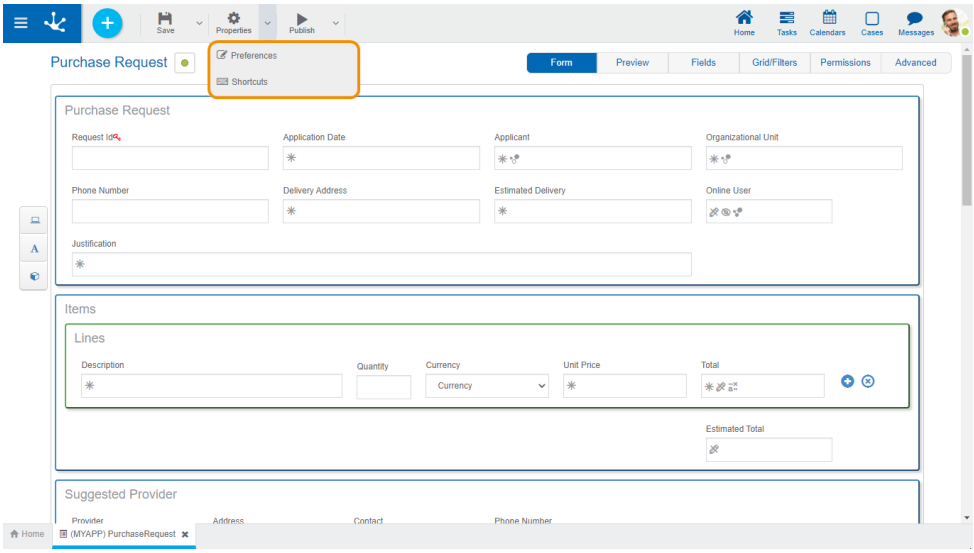
Allows to delete the object only if it is in "Draft"state, closing the tab in which it is located and deleting it from the modeler's grid.

Download Java Sources

This icon allows to download the Java files that represent the object's model and service, so that it can be used in advanced rules.
Pressing the icon displays a message to confirm file download.

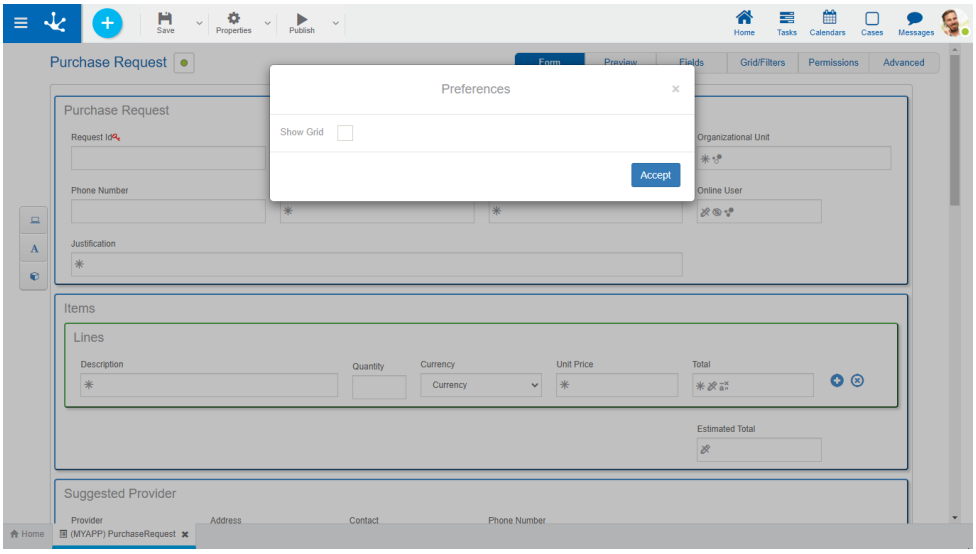
3.6.4.1.1.2. Properties Submenu

This submenu can be opened by pressing the icon immediately to the right of the icon corresponding to "Properties".

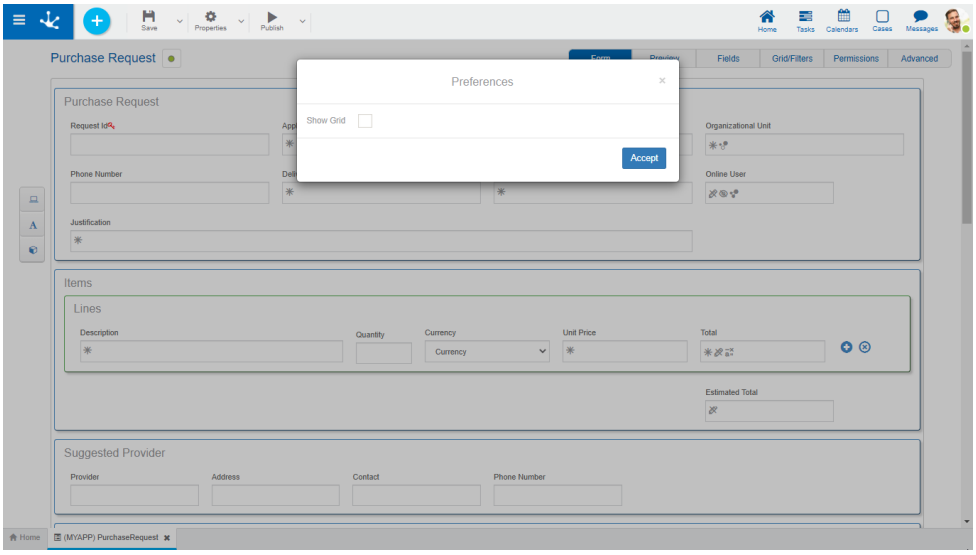


Preferences

Defines display preferences for forms modeling.

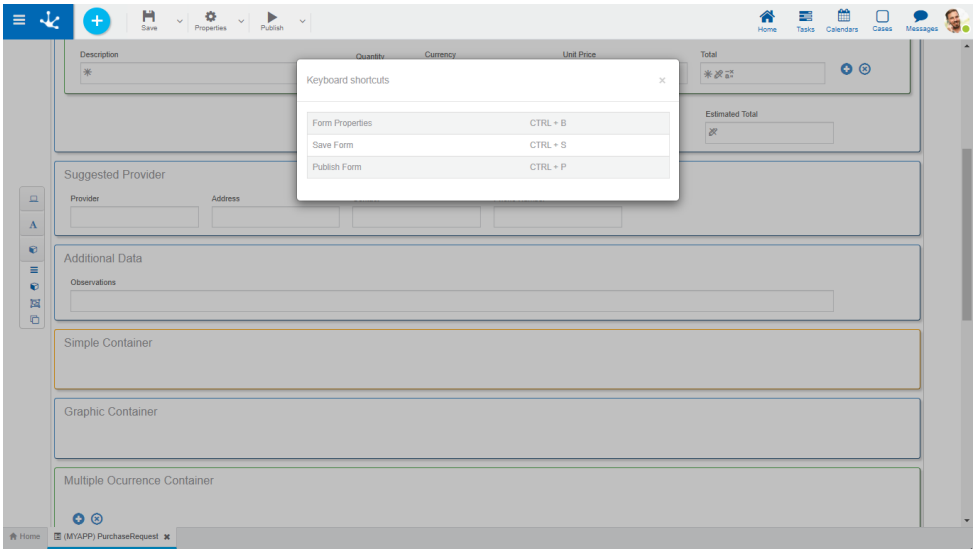


- See grid
  Indicates the division into rows of the graphical modeling area. The default value is "No".



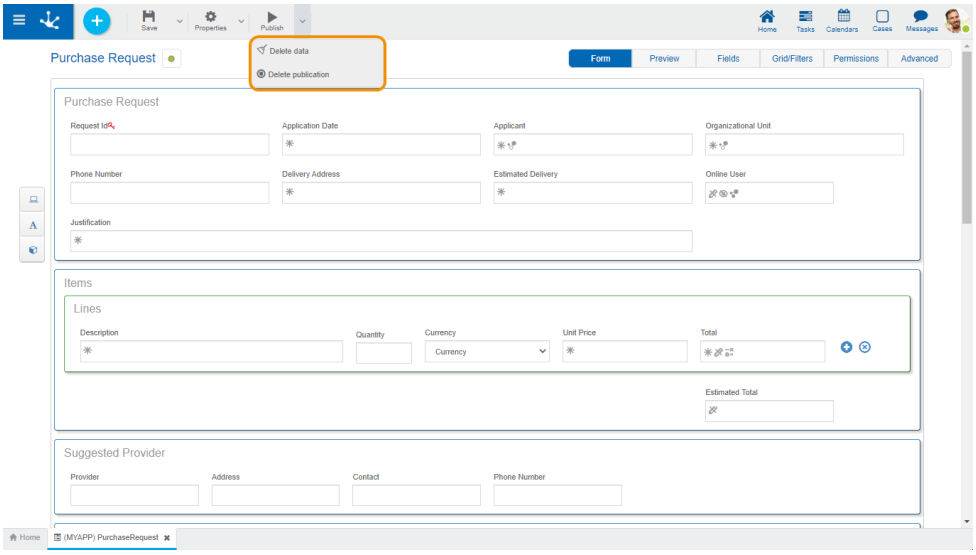
Shortcuts

Opens a panel with all available keyboard shortcuts to use in forms modeling.



3.6.4.1.1.3. Publish Submenu

This submenu can be opened by pressing the icon immediately to the right of the icon corresponding to "Publish".



Delete Data

Instances that would have been generated when using the form are deleted.

> *If the instances of a form are linked to cases of processes related to the form, first these cases must be deleted from the option Delete Data from the top bar of the process modeler.*

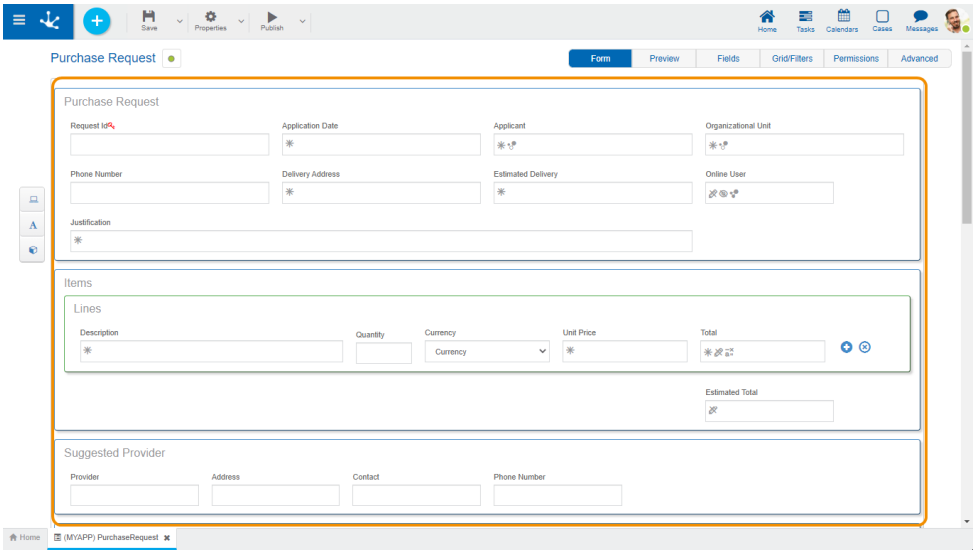
Delete publication

This icon allows to leave the form unavailable for use by returning it to the "Draft"state, in addition to deleting the data.

3.6.4.1.2. Graphic Modeling Area

The graphic modeling area is the space where the different elements are dragged from the side toolbar, allowing to model the graphical arrangement of the elements in the form.

The modeling area is divided into rows, and each element on the form occupies at least one cell within the row.



Operations on the Elements

When you drag an element from the left toolbar to the graphic modeling area, it is inserted in the chosen place. Hovering the mouse over the element displays a set of icons that allow to perform different operations. The operations available depend on the type of item you are working with.

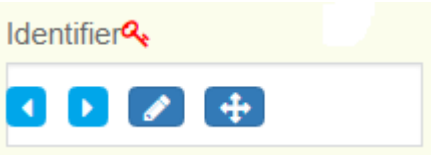
Allows to delete an item. Clicking on this icon displays a confirmation message that, when pressed, deletes the modeled element.
In case the form is published and it already has data loaded, this icon is disabled.

Allows to modify the graphic width of the element. The right icon enlarges the width in a unit and the left icon decreases it by the same amount.

Allows to enter the section of element properties.

Allows to move the field to any other sector in the modeler area. You can drag the field by keeping the click on this symbol and dropping it on a row, or on the cell separators, so the field moves to the indicated area.

Every form is created containing an identifier field by default, and is represented by a key to the right of the label.



Behavior Indicators

In each form field, you can visualize the icons corresponding to the modeled behavior rules, the relations to entities and the related attributes.

Required / Required with Rule

Not visible / Visible with Rule

Not editable / Editable with Rule

Validates with Rule

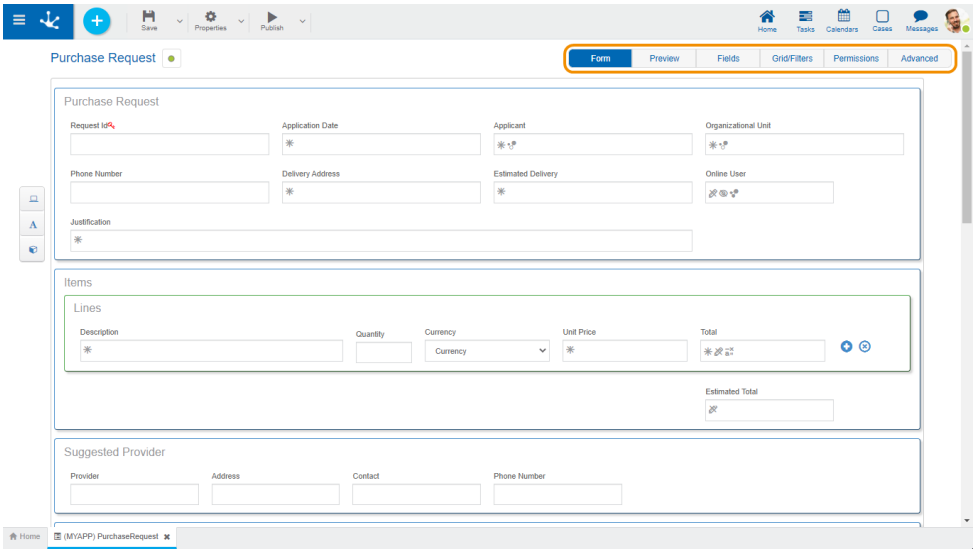
Calculates with Rule

Relation to an Entity

Related Attribute

3.6.4.1.3. Design Options

The mode of creation/editing of a form allows defining the set of fields that compose it, the type of information that is stored in each one, their distribution within the form and with which characteristics they are presented to the user. In this way you can work on the data model and its visualization at the same time.

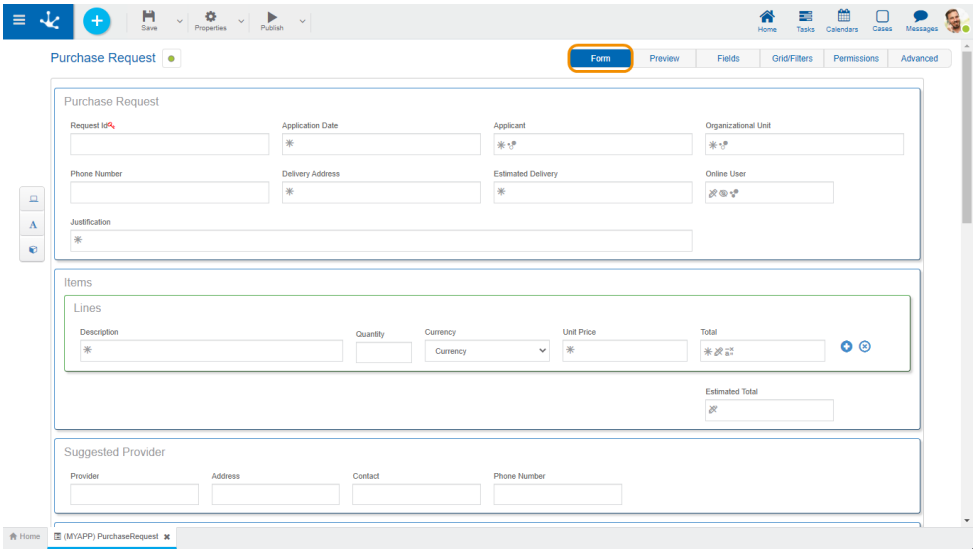


- Form
- Preview
- Fields
- Grid and Filters
- Permissions
- Advanced Edition

3.6.4.1.3.1. Form

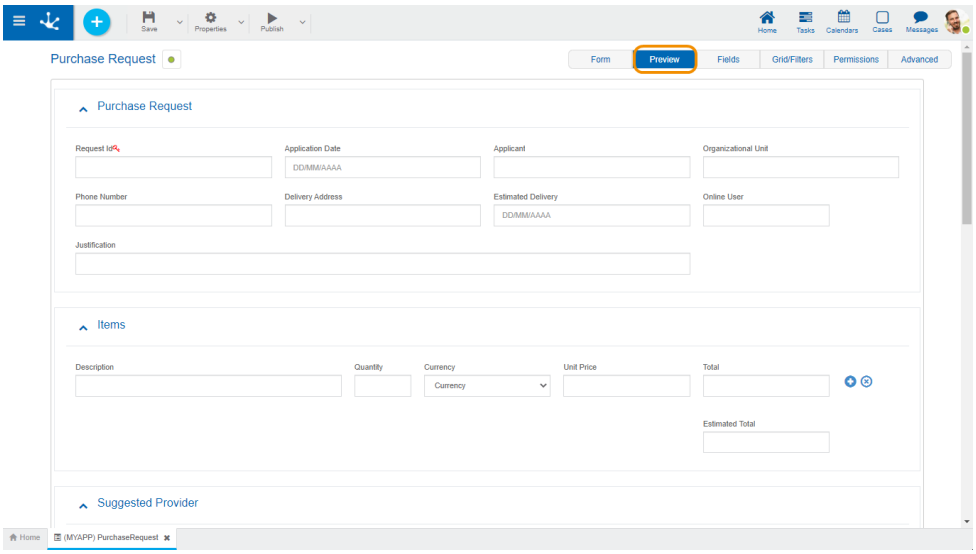
The graphical modeling area is presented when the modeler starts, where fields, graphical elements and containers can be defined.

These items are available in the left side toolbar, where its structure and operation are explained in detail.



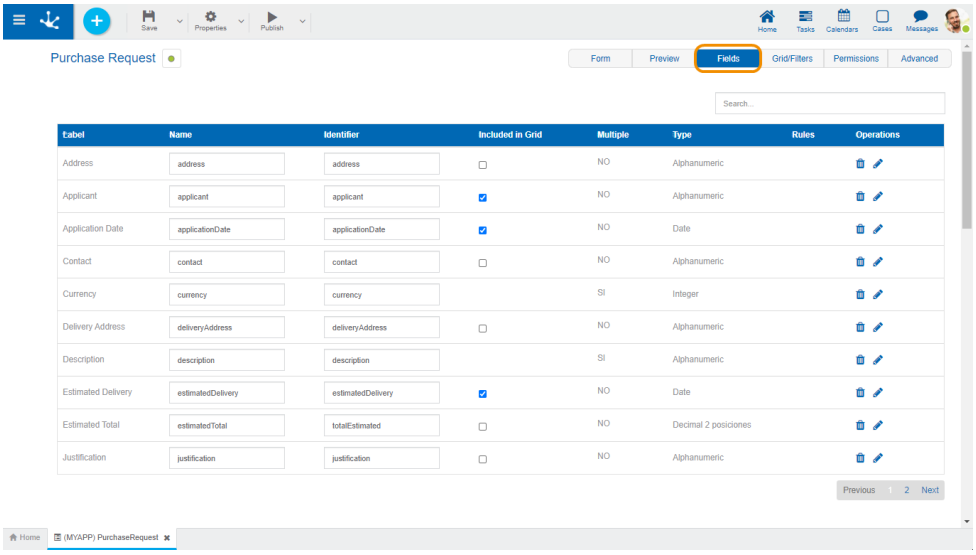
3.6.4.1.3.2. Preview

It allows to have a vision of the modeled form, showing how it would look in the user portal. For example, buttons for editing the "Form" tab are hidden and the containers (accordions and tabs) are displayed, the containers that have the hide property active are hidden, etc.



3.6.4.1.3.3. Fields

Allows you to see the list of all form fields in a paginated table, with quick search capabilities, filters, order, viewing the properties of each one and allowing operations to be carried out.



Columns

Label

Allows to enter the text that is displayed on the field. It works together with the prefix to reference the field in validation messages and supports blank spaces.

Name

Name assigned to reference a field in the modeling, allowing the field to be uniquely identified within the form. Used in rule wizards to refer to field within conditions. It generates automatically from the Label property, it can be modified by the user and does not allow spaces or special characters.
It allows its editing and it must be unique in the form.

Identifier

It is the name that is assigned to refer to a field in the programming code, it is used to refer to the field within the Java code in the "Execution Code" tab of the advanced rules and in the JavaScript code in the "Advanced Editing" tab of the form modeler. Allows to uniquely identify the field within the modeled form. It can be modified by the user, as long as no data has been loaded into the form and it does not allow spaces or special characters.
It allows its editing so that the user can modify it only when the form does not contain data.

Included in Grid

Indicates if the field is displayed in the results grid where the form instances are shown.

Multiple

Indicates if the field has multiple occurrences.

Type

Determines the data format that can be entered in the field. The supported data set varies, depending on the type of field that has been created from the elements of the left side toolbar.

Rules

In case the field has any associated rule of Required, Visibility, Editability or Validation, an icon is displayed indicating the type of rule.

Operations

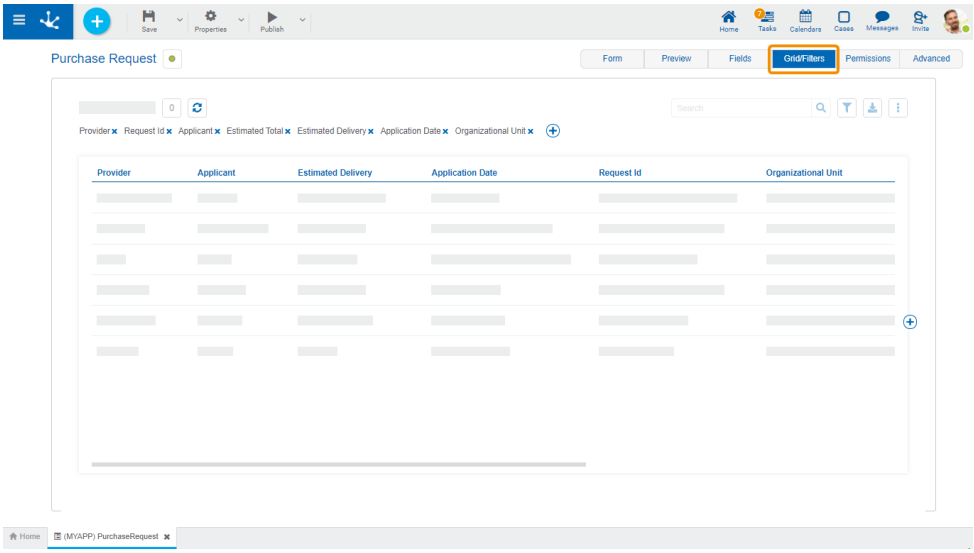
Allows to delete the field only in forms without data and requires confirmation by the user.

Attributes editing opens a new panel that allows to edit the field properties.

3.6.4.1.3.4. Grid and Filters

Grid

Allows to model the columns displayed in the results and search grids. The default columns of the grid correspond to those fields that have been configured with the Included in Grid property checked.



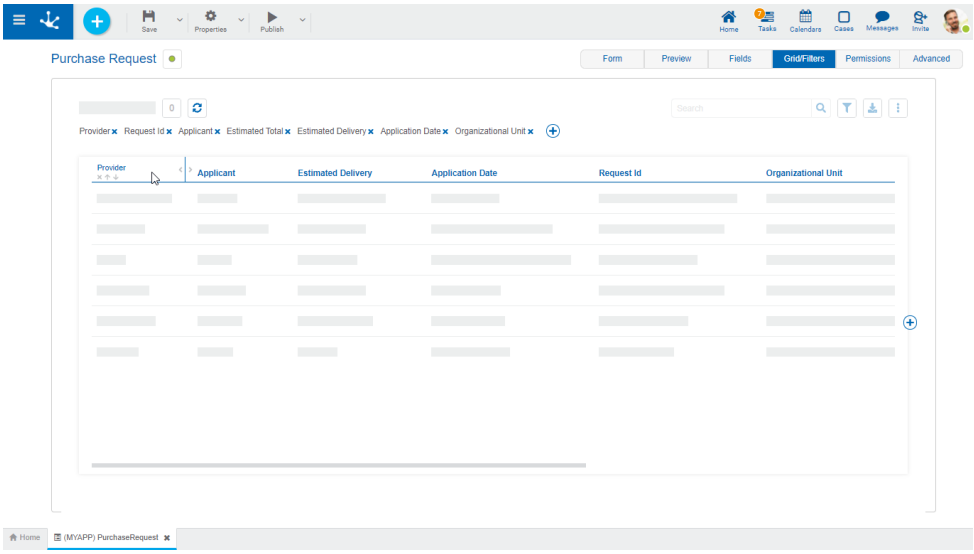
New fields can be added by pressing the icon ⊕ to the right of the last column of the grid and its order can be modeled by dragging the shaded rectangles of the columns to swap their positions.

When hovering over the title of each column, the following icons are displayed:

Allows to model the width of each column by moving the limit bar to the right to enlarge the column, or to the left to shrink it.

Remove the column from the grid.

To establish the column by which the default sorting will be performed, it is ascending or descending depending on the selected icon.



In forms with associated processes, columns related to the case execution can be added when modeling the result grid .

Case Activity

It is the activity of the process in which the case is.

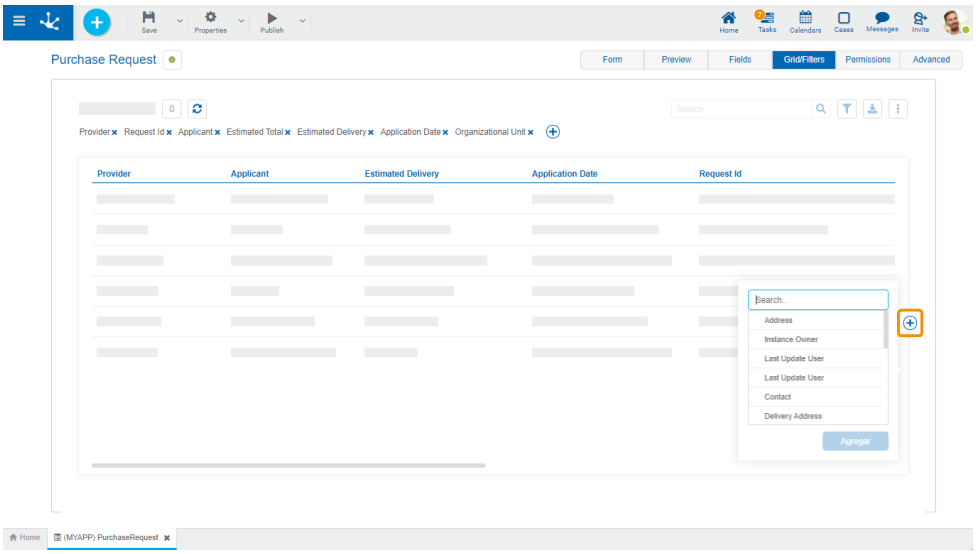
Case Responsible Agent

Refers to the user, role or office that the case has pending on its task list.

Case State

It is represented by a circle of different colors.

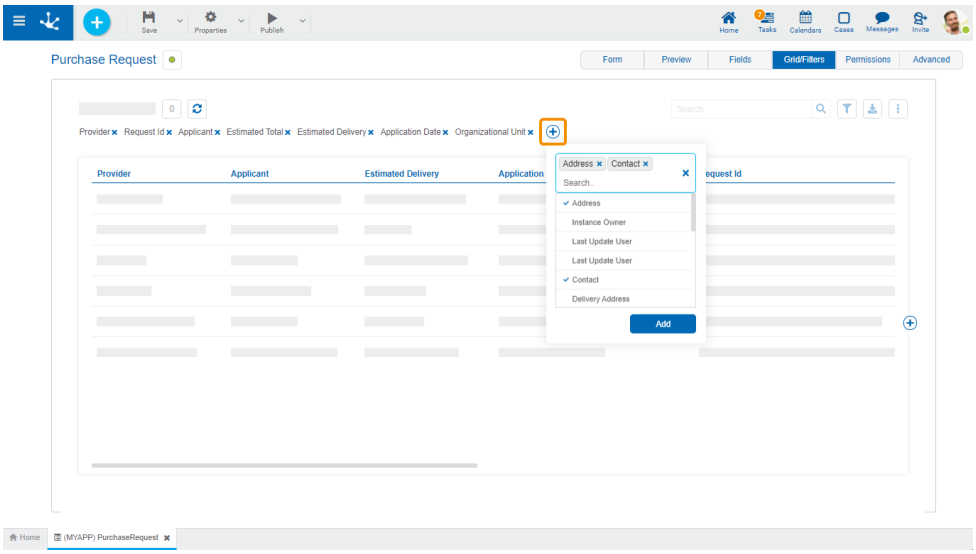
- Active: Green
- Finalized: Blue
- Cancelled: Red
- Discontinued: Gray



Filters

The definition of filters allows modeling search criteria on form instances. Each filter corresponds to a form field or to a field related to its execution.

Pressing the icon located in the filter line, a panel to select the fields opens.



3.6.4.1.3.5. Permissions

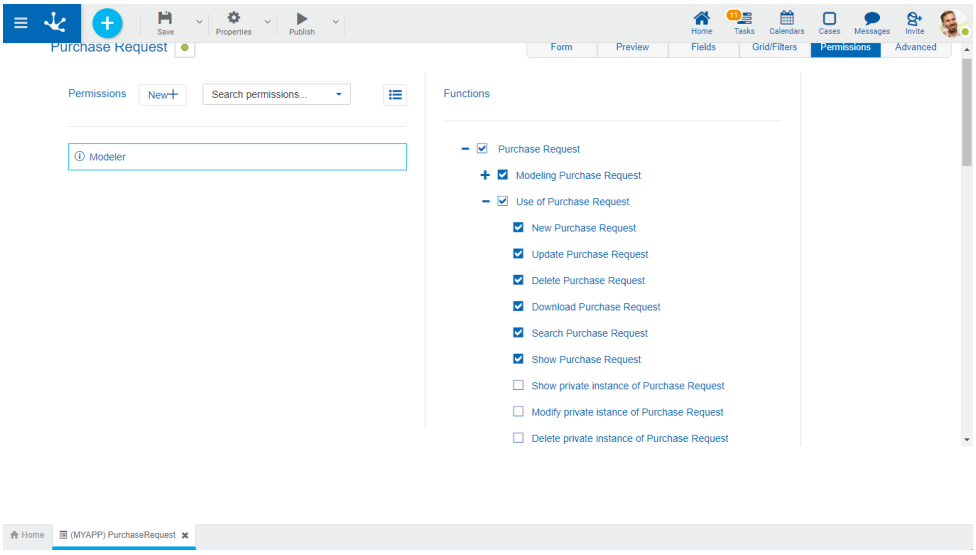
Allows to assign the security functions for use and modeling of the object to the existing permissions or to new permissions that the user creates, without having to go to the permission settings option from the menu.

Sections

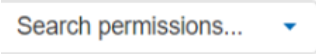
- Permissions: Permissions to which object functions are assigned.
- Security Functions: Represents the total set of security functions, modeling and use of the object. Those that are marked are the ones included in the selected permission.

By default, all security functions for a new object are assigned to the permission Modeler.

Users who are assigned the permissions have access to the functions included in it.



Opens a panel to create a new permission and once created, the security functions included in it must be selected .



Allows to select a permission from a list and enables the input of characters to filter the values in the list.

Opens the wizard to select a permission and once chosen, the necessary object security functions must be checked.

To unrelate a permission from the object, hover the cursor over the permission and press the icon ✖. If there are functions selected for that permission, they must be unchecked in order to delete it.

Security Functions to Model the Form

- Visibility in the modeler: Allows to display the object in the Deyel modeler.
- Open in modeler: Allows to show the object from the Deyel modeler.
- Copy to modeler: Enables the copy operation of the object.
- Save changes: Enables the operation of saving modifications made to the object.
- Publish form: Enables the operation of publishing the object leaving its state in "Published".
- Delete all data: Enables the delete data operation.
- Delete publication: Enables the operation of deleting the object publication leaving its state as "Draft".
- Delete draft definition: Enables the operation of deleting the object.
- Block form: Enables the lock or unlock operation, when locking a form only the user who locked the form can modify it.
- Export form: Enables the operation to export the object from the save submenu of the upper toolbar.

Security Functions to Use the Form

- Create: Enables the operation of creating an instance of the object.
- Modify: Enables the operation of modifying an instance of the object.
- Delete: Enables the operation of deleting an instance of the object.
- Download: Enables the operation of downloading the instances of the object in Excel.
- Search: Allows to use the show grid of the form instances.
- Show: Enables the operation of showing an instance of the object.
- Show/Modify/Delete Private Instance: They enable user operations on a private instance, regardless of the organizational unit they belong to and the rest of the criteria that define the privacy.

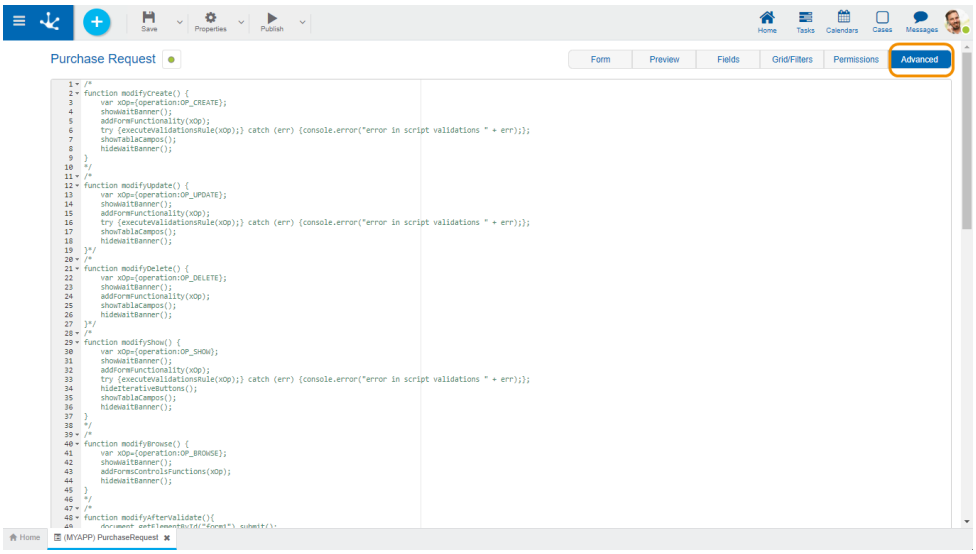
> *Private instance security functions are displayed if the properties are checked in the form* Hierarchical Privacy *and* Privacy by Permissions.

3.6.4.1.3.6. Advanced Editing

Being **Deyel** a development platform, it allows to make changes to the default functionality of a form. This functionality is aimed at IT users with knowledge of web development.

The "Advanced" tab presents a JavaScript code editing window that allows to apply programming logic in the web interface when using the form.

There is a set of JavaScript functions implemented in the platform that contain the default logic and behavior of the forms. Since business needs may require modifying this logic, these functions can be rewritten by programmers and apply different behavior to particular forms.



Predefined JavaScript Functions

Functions according to Operation

- **modifyCreate():** It is executed when entering the creation of a new form instance.
- **modifySubmitCreate():** It is executed when you click the "Accept" button on the creation of a form instance.
- **modifyUpdate():** It is executed when entering the modification of a form instance.
- **modifySubmitUpdate():** It is executed when you click the "Accept" button on the modification of a form instance.
- **modifyShow():** It is executed when entering to show a form instance.
- **modifySubmitShow():** It is executed when you click the "Submit" button showing a form instance.
- **modifyBrowse():** It is executed when entering the form's grid and filters tab.
- **modifySubmitBrowse():** It is executed when the "Submit" button is clicked in the form's grid and filters window.
- **modifySearch():** It is executed when entering the form's grid and filters tab.
- **modifySubmitSearch():** It is executed when the "Submit" button is clicked; runs when entering in the form's grid and filters screen.
- **modifyDelete():** It is executed when entering the deletion of a form instance.
- **modifySubmitDelete():** It is executed when you click the "Accept" button on the deletion of a form instance.
- **modifyAfterValidate():** It is executed after controlling the asynchronous validations, when confirming a registration or modification operation. Allows to perform operations after executing validations.
- **modifyPrint(pNode):** It is executed by pressing the "Print" button of the forms. Allows you to modify the default print display. The pNode parameter is the Jquery node of the entire html template of the form, any Javascript logic can be applied to modify such node, which is reflected only in printing.

JavaScript Functions Structure

The predefined functions have the following structure:

```
function...() {
      var xOp={operation:”OPERATION”};
showWaitBanner();
      addFormFunctionality(xOp);
      try {executeValidationsRule(xOp);} catch (err) {console.error("error
      in script validations " + err);};
```

```
showFieldTable();
hideWaitBanner();
}
```

The OPERATION variable takes the following values according to the operation being performed with the form, allowing different actions to be executed on the form instance.

- "CREATE";
- "UPDATE";
- "SHOW";
- "BROWSE";
- "DELETE";
- "SEARCH";

Internal Functions to Predefined JavaScript

showWaitBanner(): Shows the wait message.

addFormFunctionality(): Adds functionality to view elements such as containers, field controls, repeatable field buttons.

executeValidationsRule (): Associates JavaScript events to the form interface, related to modeled rules for visibility, calculation, and editability.

showFieldTable(): Allows to display the fields section of the form, which is hidden.

hideWaitBanner(): Hides the wait message if it is visible.

Other Useful Functions

GetCdActivity function

In some cases where the form is associated with a process, different actions are carried out depending on the running activity.
In order to add programmed logic, there is the getCdActivity () function, which returns the number of the activity in progress.

Example:

if**(getCdActivity() == 5){**

//show container
}{
//hide container
}

Container Functions

hideContainerById({idContainer:'id'}): This function hides in the interface the container indicated in the 'id' parameter.

showContainerById({idContainer:'id'}): This function displays in the interface the container indicated in the 'id' parameter (if hidden).

Graphic Container Functions

closeContainer({type:0,idContainer:'id'}): Defines that a graphic container is closed.

type: Indicates what type of container it is.

- 0 for Bootstrap-style containers.
- 1 for jQuery-style containers.

idContainer: Identifies the container to apply the functionality.

openContainer({type:0,idContainer:'id'}): Defines that a graphic container is displayed.

type: Indicates what type of container it is.

- 0 for Bootstrap-style containers.
- 1for jQuery-style containers.

idContainer: Identifies the container to apply the functionality.

Multiple Occurrence Container Functions

showIterativesHeaders(): Allows multi-occurrence containers to repeat their field labels in each row.

Example:

showIterativesHeaders (){

return ["idContainer"]

}

Supported Web Technology

In addition to native JavaScript, **Deyel** allows developers to use other web technologies on this tab.

- jQuery Version 2.1.4 official site https://jquery.com/
- Bootstrap 3.3 Official site https://getbootstrap.com/docs/3.3/
- Font awesome official site https://fontawesome.com/v4.7.0/

These technologies are already included in the platform, any functionality of the same technologies that the programmer user wants to use does not require anything additional.

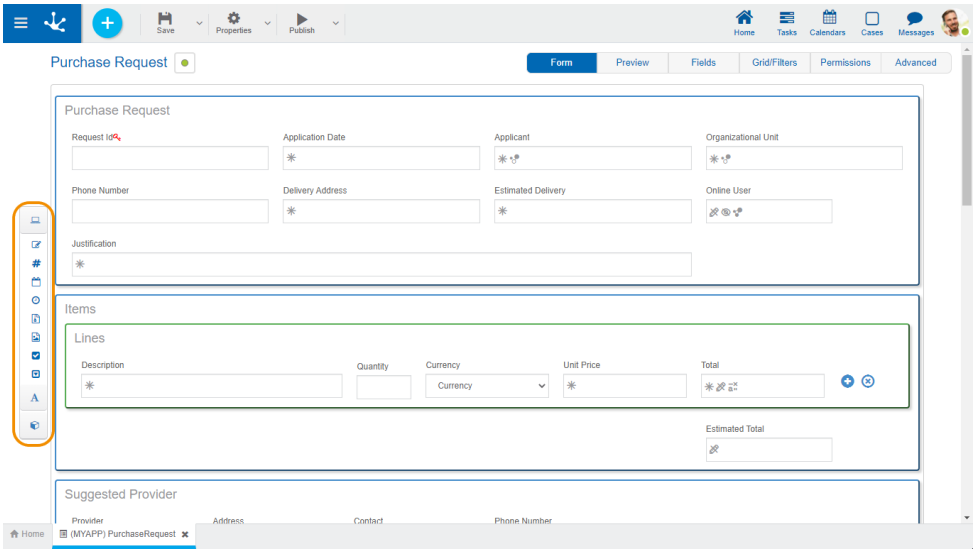
3.6.4.1.4. Side Toolbar

It is the toolbar that contains the icons corresponding to the different types of elements that can be defined in a form.

To use it, it is necessary to click on the corresponding section of the bar and select the type of element to add, dragging it to the desired place on the form.

Separators

They are light blue rectangular boxes that are displayed to the right and left, or above and below the elements defined in the modeling area, when dragging a field, graphic element or container with the mouse from the bar on the left. Indicate the positions within the modeling area where a field, graphic element or container chosen to be included in the form can be dropped.



Bar Elements

Fields Modeling
Pressing this icon displays a subset of elements that allow modeling fields represented by controls within the form. Each field can have a data type associated with it, depending on the con-

trol that represents it.

Text

Text

- Alphanumeric (Default Type)
  The values are saved keeping the entered uppercase and lowercase letters.
- Uppercase Alphanumeric
  Values are saved in uppercase.
- Long alphanumeric
  It works as a default alphanumeric, with the exception that it allows storing long texts (usually up to 4GB).
- Rich Text
  It has the characteristics of the alphanumeric type. It presents an extended editor that allows you to format the text by applying different styles, colors, sizes, etc. See detail of use of control.

Integer

Number

- Number
  It can contain integer values between -2147483648 and 2147483647 (they are stored in 32 bits).
- Large Integer
  It can contain integer values between -9223372036854775808 and 9223372036854775807 (they are stored in 64 bits).
- Decimal
  If the data type is decimal, the selection of the number of decimal places (2 to 5) is enabled.

Amount of Decimals

-

-
2 positions
3 positions
4 positions
5 positions

Date

Date

- Date

Allows you to view a date or enter it from a selector. See detail ofuse of control.

- Date and Time
  Allows you to view a date and time or enter them from selectors. See detail ofuse of control.
- Local Date
  Allows you to view a date or enter it from a selector, according to the time zone correspon-ding to the calendar of the user who uses it.
- Date and Local Time
  Allows you to view a date and time or enter them from selectors, according to the time zone corresponding to the calendar of the user who uses them.

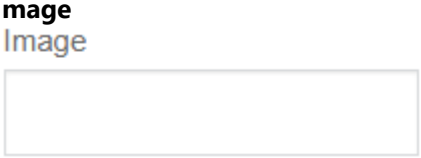
Time

- Time
  Allows you to view a time or enter it from a selector. See detail of use of control.
- Local time
  Allows you to view a time or enter it from a selector, according to the time zone correspon-ding to the calendar of the user who uses it.

File

- File in Database
  Allows user files to be used as attachments to the form. See detail of use of control.
- File in Folder
  The files are stored in the file structure of **Deyel**. This option is available only in the On-Premise version.

Image



- Image in Database
  Allows using image type user files as an attachment to the form. Allowed extensions are jpg, png, tif, bmp, and gif.
  See detail of control use.
- Image in Folder

Images are stored in the file structure of **Deyel**. This option is available only in the On-Premise version.

Check

Allows to model fields represented by a check control. The values are of the logical type "YES/NO".

It can be displayed in toggle format if the field Displayproperty was modeled.
See detail of use of control.

Value List

Value List

Example List

It allows modeling fields represented by a value list type control, whose possible values are predetermined based on a pre-existing list or are defined at the time the field is modeled, in the "Relation" tab of its properties. See detail of control use.

Modeling of Graphic Elements.

Pressing this icon displays a subset of graphic elements.

Title

It is used to include a text with a highlighted format.

My Title

Label

Used to include a free text and to be able to locate it in any section.

Label

Separator

Allows to divide sections of the form.

Space

Used to include blank lines.

Multiple Occurrence Button

Allows you to add and remove iterative. It can be used within iterative containers.

Containers Modeling

Containers are graphic elements that allow grouping other elements under certain criteria.

For more details on its configuration and attributes, seeContainer Properties.
Clicking on this icon displays a subset of container elements.

Row

It is the grouping used in the graphic modeling area. An unlimited amount of elements can be put within a row, although the window width is recommended not to be exceeded since the row expands by increasing the width with respect to the others.

Rows can be scrolled up or down by moving all their content.
To move a row, you must hold down the left mouse button with the cursor, focusing on the row you want to move and dragging it up or down another one.
Rows can only be ordered within the container they are in, either the general area or a created container.

To delete a row you must position the mouse over it, where the delete option is displayed on the right .

As an error prevention mechanism, a confirmation notice is displayed and if it is accepted, the row and all its content, including fields, are deleted.

If the form already contains data, deleting rows that contain fields is not allowed.

Simple Container
A rectangle containing any element of the form (fields, graphic elements and even other containers) is displayed.
It can be used to group sections of the form, to which operations are added (hide, relate rules, etc.).
The outline is not visible at the time of execution.

To differentiate it from other containers, its outline is orange.

Graphic Container
The difference with a simple container is that everything inside the graphic container is displayed in run mode as inside an accordion.
It can be used to group sections of the form, to which operations are added (hide, relate rules, etc.).
This container transformed into an accordion can be seen in the preview.

To differentiate it from other containers, its outline is blue.

Multiple Occurrence Container
It can contain any element of the form (fields, graphic elements and even other containers).

Its main functionality is that all content is defined as multiple, that is, when executing the form, copies can be generated and deleted from the icons to the right of the element. This container is created by holding down a couple of buttons, but they can be deleted if not required.

Unlike the other containers, all the fields contained within are defined as a set of multi-occurrence fields. It cannot include other iterative containers either, nor simple containers if the Field Group property is checked.

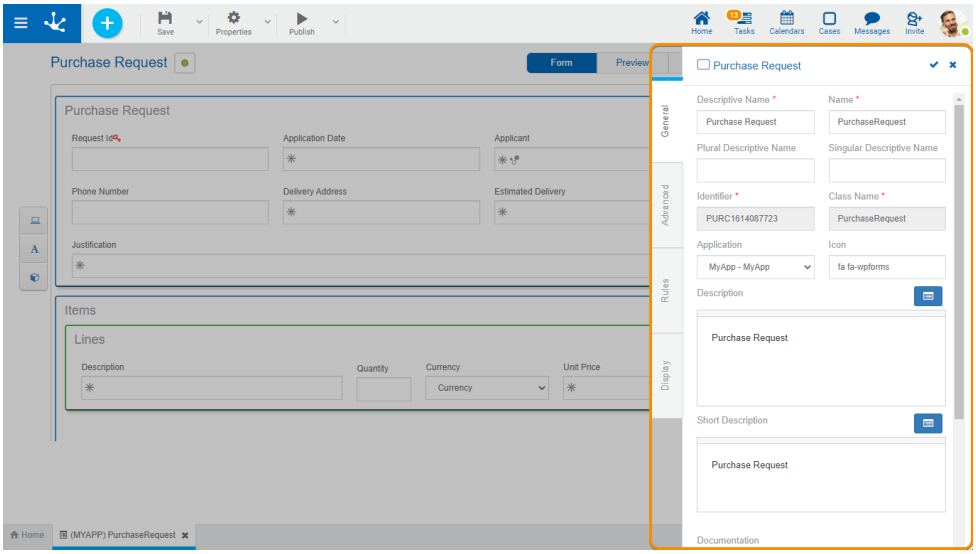
It can be used to group sections of the form, to which operations are added (hide, relate rules, etc.). The outline is not visible at the time of execution.

To differentiate it from other containers, its outline is green.

3.6.4.2. Form Properties

The properties of the forms can be entered both at the time of their creation and when modifying an existing one.

Entering the form properties panel is done using the icon which is in the top toolbar.



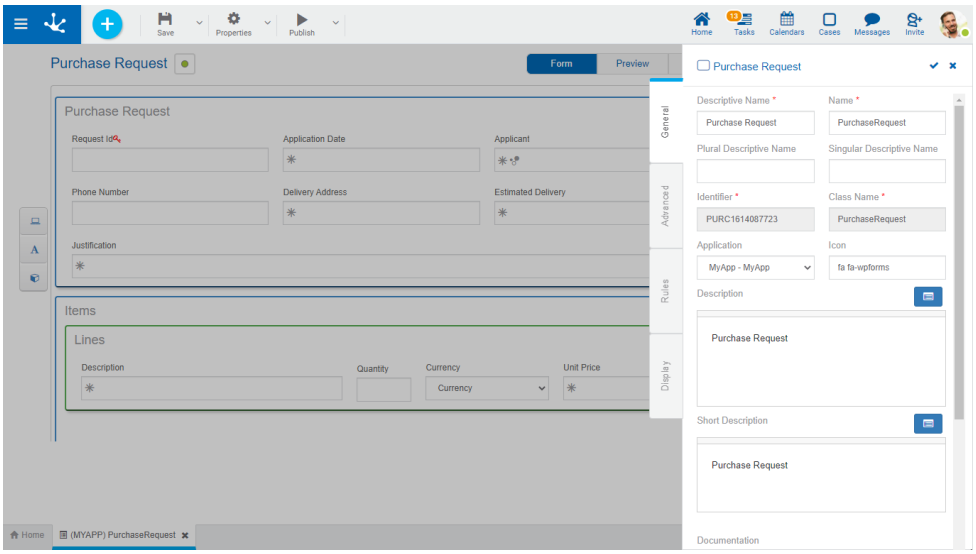
Tabs

- General
- Advanced

- Rules
- Display

3.6.4.2.1. General

The properties panel is displayed on the right side of the form modeler, where the first tab corresponds to general information.



> *An asterisk "*" on the label indicates that the property is required.*

Properties

Descriptive Name

It is the name used by users to reference the form, for example in the modeler's grid. Supports multi-language.

Name

It is used at the modeling level to reference the form, for example in rules or as a parameter. Supports multi-language.

Plural Descriptive Name

The text entered in this property is used as the title in the results grid of the form, while if it is not completed, the Descriptive Name property.

Singular Descriptive Name

The text entered in this property is used in the list of related entities, while if it is not completed, the Descriptive Name property.

Identifier

Uniquely identifies the form.

Application

Allows to define the application to which the form belongs.

Icon

Allows to assign a representative image of the form to the form.
The code to be entered corresponds to the Font Awesome standard. The identifier of the icon selected from the https://fontawesome.com standard must be preceded by the characters "fa".

Description

Defines the form describing its functionality and optionally its content.
It can include text and values of variables from each form instance, which are modeled using the variables wizard. The use of variables included in iterative containers and the use of file, image and check type variables are excluded.

In the execution of the form, the description entered is displayed in:

- The last updates, on the element that represents the form in the grid of "Forms and Tasks".
- The panel of a form instance, below the descriptive name.

Short Description

Text oriented to be a shortened description of a form instance. It can be used to identify the form in the related entities, in a friendlier way than using its identifier.
It can include text and values of variables from each form instance, which are modeled using the variables wizard. The use of variables included in iterative containers and the use of file, image and check type variables are excluded.

In the execution of the form, the identification entered in this property is displayed when using the referring entity field in:

- The form instance.
- The form grid.
- The filter panel of the form grid.

Help

Informative text displayed when pressing the icon ? using the form from the portal. Supports multilanguage.

Image

An image that is displayed in the grids of forms and tasks, templates and in the upper bar when creating forms. It is recommended to be related to the functionality or information that the form represents.

Reports on the usefulness of the image.
Allows to choose an image from a file selection window.
Allows to delete the current image associated with the form.

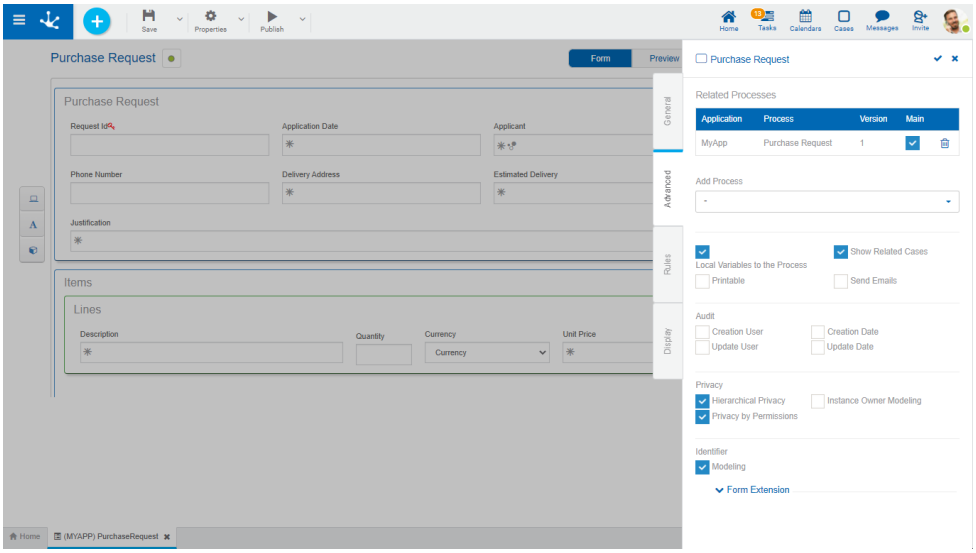
Actions

The icon is used to confirm the modifications made in the properties panel.

The icon is used to close the properties panel, if it was not previously saved, changes are discarded.

3.6.4.2.2. Advanced

The second tab of the side panel corresponds to advanced information of the form.



Properties

Related Processes

Every form can be associated with one or more processes. The user can select them from the set of available processes and define form relation properties. Once selected, the processes are displayed in a grid with their corresponding properties.

- Application
- Process
- Version
- Main

  In complex processes with a large number of variables, these variables can be defined in different entities, all related to the process. If this property is checked, it means that the modeled form is the most representative for the related process. This means that in the gallery of forms and tasks, the main form is the one where the tasks are displayed, even if other forms are used in the different activities.

  A process can be related to several forms, of which only one can be indicated as the main entity. If in the Related Processes grid, the process with this property selected is already defined as main in another form, then the user receives an error message when the form is published.

Add Process

Allows to select a process from the drop-down list of available processes, each of the chosen processes is added to the grid ofRelated Processes.

Local Variables to the Process

Checking this property indicates that the form fields have a local scope to the related process, that is, these variables can only be accessed and modified during the execution of the corresponding case. The variables of completed cases can only be accessed by them.
Leaving this property unchecked indicates that the variables have a global scope, that is, they can be accessed and modified by the corresponding form operations, by each case of the processes using the form or by business rules.

Show Related Cases
If this property is checked, each time a form instance is created or shown in the user portal, an additional section of data is displayed, to associate the form with a case.

Printable
Allows to print the form when it has already been instantiated. Checking this property, when showing form instances, the "Print" button is enabled.

Send Emails
This property enables the send emails icon at the time of using the form, allowing to send emails that become associated to the form instance.

Audit
To indicate that audit fields will be modeled in a form, the audit properties represented by the checks listed below must be used, one for each field.

- Creation User
- Update User
- Creation Date
- Update Date

Privacy
Phase 2: Forms Modeling > Advanced Tips>Entity privacy

Hierarchical Privacy
Check that indicates if the entity is private or not. If this check is indicated, the Privacy by Permissions property is displayed.
Privacy by Permissions
Mark indicating the creation of security functions: "Show Private Instance", "Modify Private Instance" and "Delete Private Instance" to be assigned in the design option "Permissions" of the form. These security functions are available if the form is in "Published" state, since they are use functionalities.
Instance Owner Modeling
Check indicating that the Instance Owner field can be modeled in the form modeling area. In case this check is not indicated, the data cannot be loaded by user interface, but by programming logic, for example through a business rule.
The instance owner can be modeled regardless of whether the form is private or not.

Identifier
Modeling
Indicates whether the identifier field is visible in the modeling area of the form. This property is displayed only for forms modeled with the default identifier. If the forms were modeled in previous versions using a key field, this property cannot be configured.

3.6.4.2.2.1. Audit Fields

The audit fields defined in a form are useful to keep track of its use. The information contained in such fields allows to know which user created or modified form data and the date and time of the operation.

Checking the audit properties causes the audit fields corresponding to the selected checks to be displayed in the design area.

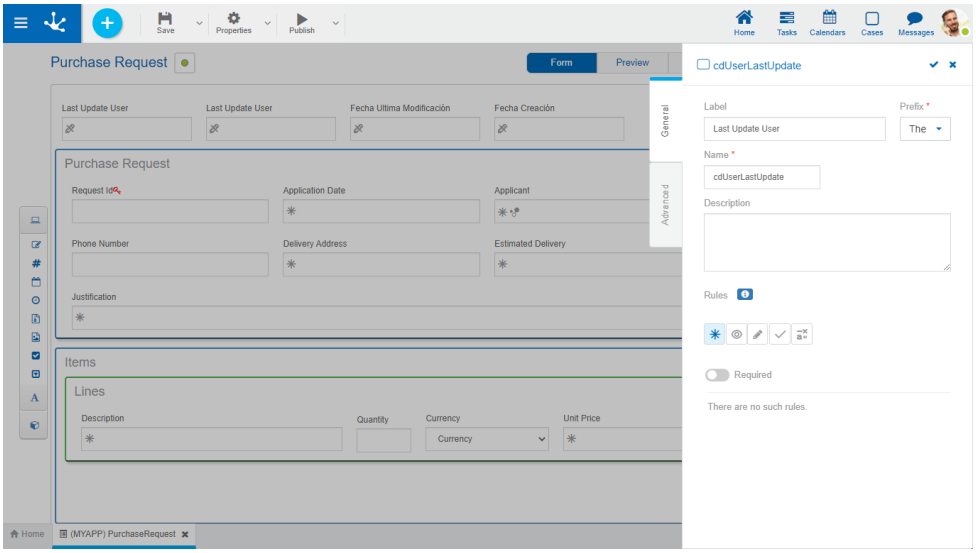
Characteristics

- The "General" and "Advanced" tabs are displayed, but not the "Relation" tab of the field properties panel.
- They can only be eliminated by unchecking the audit properties in the form properties panel, that is, they cannot be eliminated like the fields defined by the user, nor the rows or the containers where they are defined.
- They cannot be defined neither a multi-occurrence container.
- They cannot be used neither within validation rules defined within form properties nor within field properties.
- They can be used in the definition of columns in the display grid and as search filters in the "Grid and Filters" tab in the modeling area of the form.

General Tab

Audit fields can have the same set of properties defined as a user field, defined in the "General" tab , although with some limitations, since the properties listed below are not used.

- Field identification
- Editability
- Required
- Validation

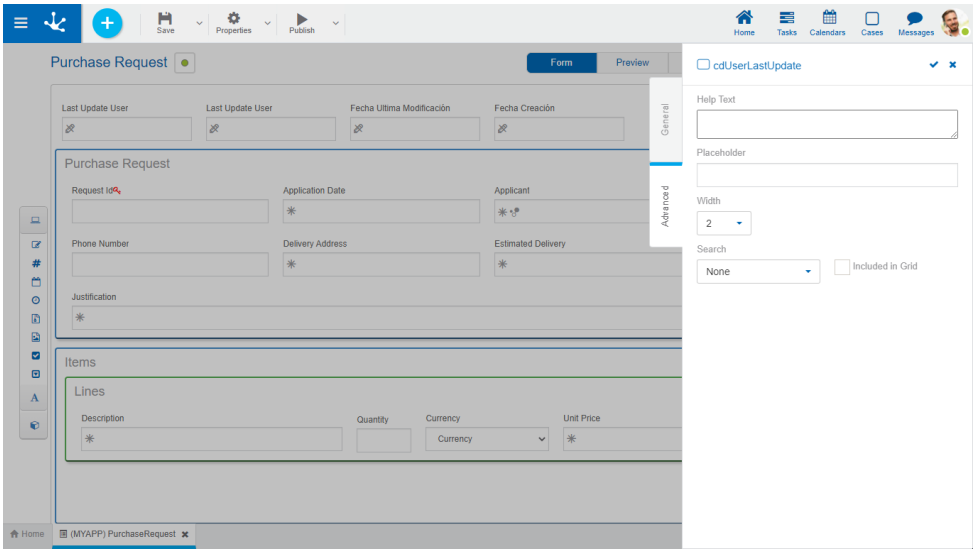


Advanced Tab

Audit fields can have the same set of properties defined as a user field, defined in the "Advanced" tab , although with some limitations, since the properties listed below are not used.

- Multiline

- Default Value
- Data Type
- Length
- Key

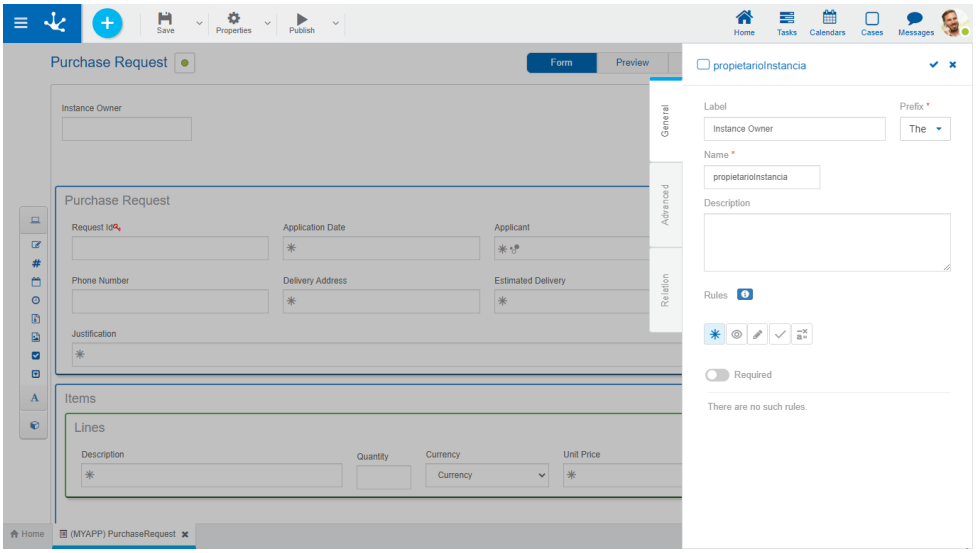


3.6.4.2.2.2. Instance Owner Field

If the property Instance Owner Modeling is checked in the "Advanced" tab of form properties, a new field labeled "Instance Owner" is included in the modeling area and it can be modeled like any other form field.

General Tab

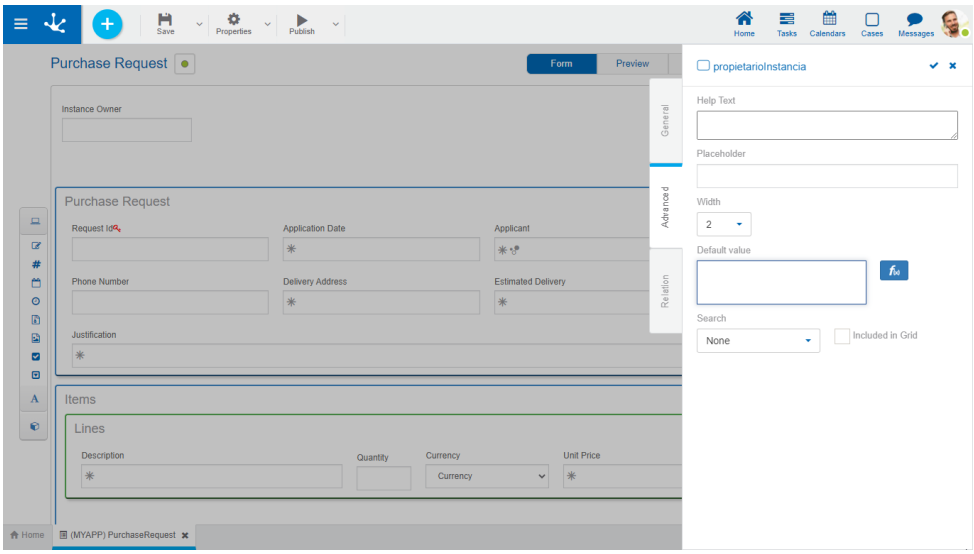
The field corresponding to the instance owner can have the same set of properties defined as a user field, in the "General" tab . The only limitation is that the Identifier property cannot be modified.



Advanced Tab

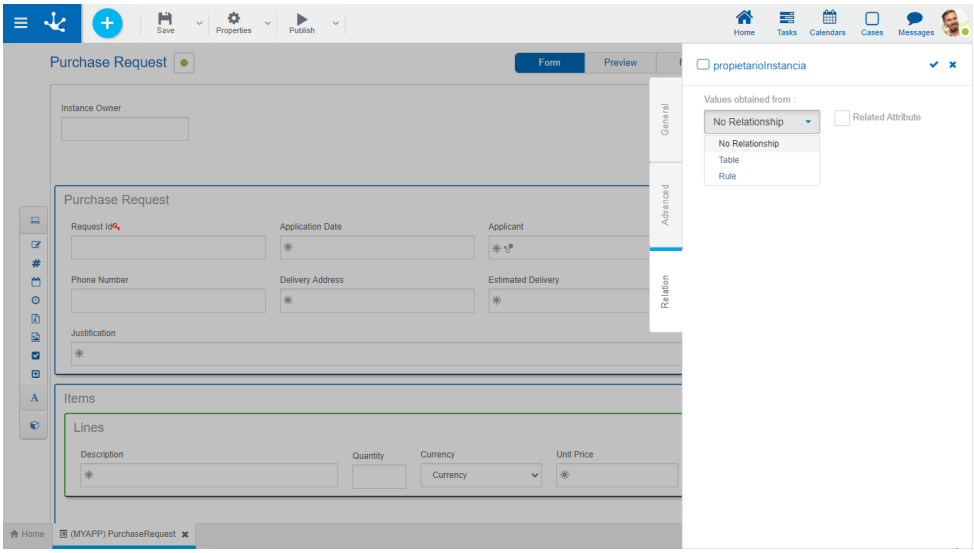
The instance owner field can have the same set of properties defined as a user field, defined in the "Advanced" tab, although with some limitations, since the properties listed below are not used.

- Multiline
- Data Type
- Length
- Key



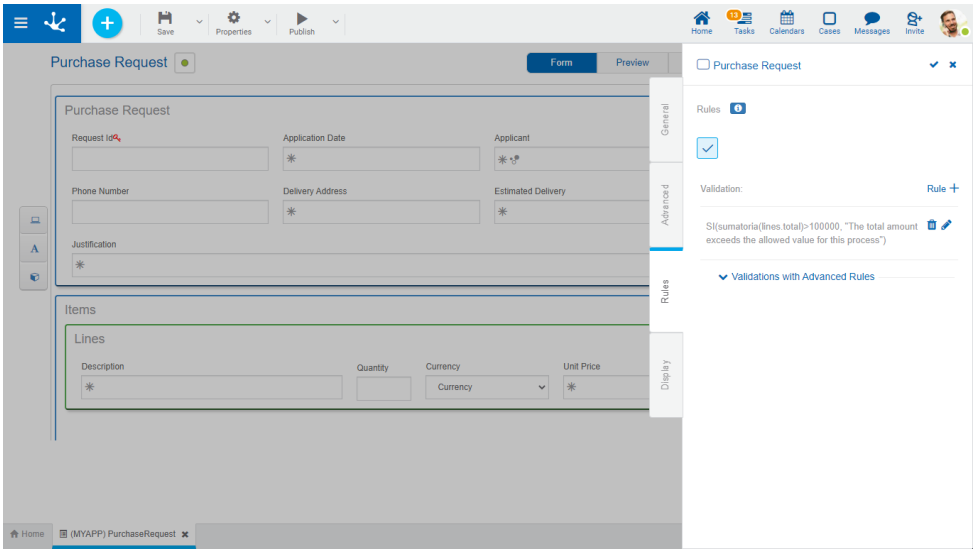
Relation Tab

The instance owner field can only define tables and rules as a source to obtain the field values, in the same way they are defined in the "Relation" tab of field properties.



3.6.4.2.3. Rules

The third tab in the side panel corresponds to the validation rules associated with the form.



Rules

It is possible to define validation rules associated with the form.

Validation

Rule + Opens an edit area where you can define the condition that determines if the form is correct or not. It is possible to define more than one rule. If rules are defined, the icon is displayed with light blue borders.

Shows syntax examples for writing the rules.

Operations

- Saves the new or modified rule
- Cancels the operation
- Edits the existing rule
- Deletes the rule

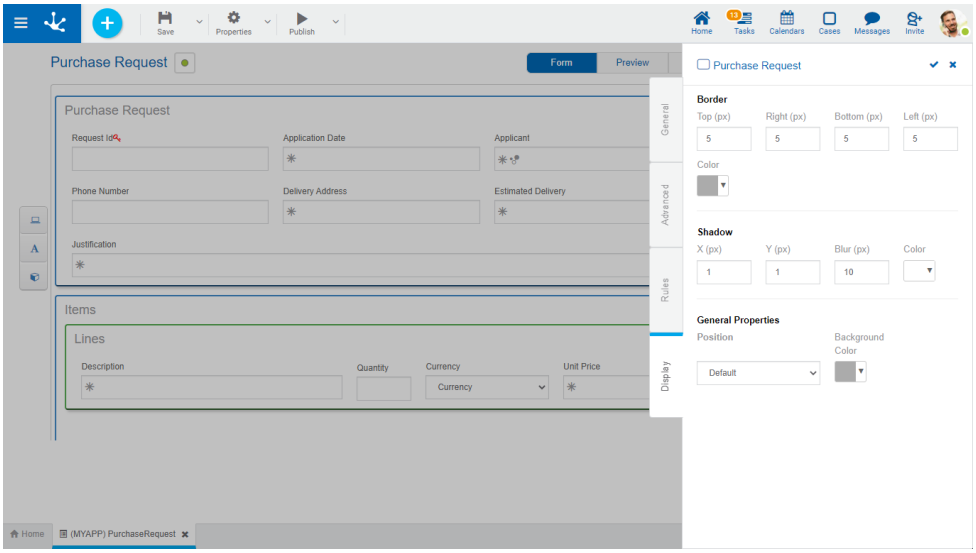
Validations with Advanced Rules

If a validation with an advanced rule needs to be incorporated, an existing rule must be selected from the drop-down list and press to add the rule to the grid. It can be selected using the options "Create", "Modify" and "Delete" when the validation is executed, either when creating a form instance, modifying it or deleting it. Through this icon you can unlink the rule from the grid.

3.6.4.2.4. Visualización

The last tab corresponds to information regarding the form display.

The result of indicating these properties can be seen in the design option "Preview".



An asterisk "" on the label indicates that the property is required.*

Properties

Border

Defines the width and color of the form borders. Width is measured in pixels, with 0 being the minimum value and 10 being the maximum value. The default value for each property is 0, with no border.

- **Top**
  Indicates the width of the top border.
- **Right**
  Indicates the width of the right border.
- **Bottom**
  Indicates the width of the bottom border.
- **Left**
  Indicates the width of the left border.
- **Color**
  Allows to select the border color from a palette.

Shadow

Allows to define a shaded area framing the form. The properties X, Y and Blur are measured in pixels, with 0 being the minimum value and 10 being the maximum value.

- **X**
  Indicates the displacement of the shadow on the vertical axis with respect to the frame of the form.
- **Y**
  Indicates the displacement of the shadow on the horizontal axis with respect to the frame of the form.
- **Blur**
  Defines how sharp the shadow is displayed.
- **Color**
  Allows to select the shadow color from a palette.

General Properties
Defines the content position and the background color of the form.

Position
Allows to display the form with margins on the left and right if the "Centered" option is selected.

Background Color
Allows to select the form background color from a palette.

3.6.4.3. Element Properties

The icon becomes visible when hovering the mouse over each element of the Graphic Modeling Area of the form, that allows to enter the side panel of properties that defines each of the different types of elements.

Types

- Fields
- Graphic Elements
- Containers

Side Panel

In the side panel, the properties corresponding to each type of element can be displayed and modified. In the upper right part of it you can select icons to perform the following actions:

Actions

The icon is used to confirm the modifications made in the properties panel.

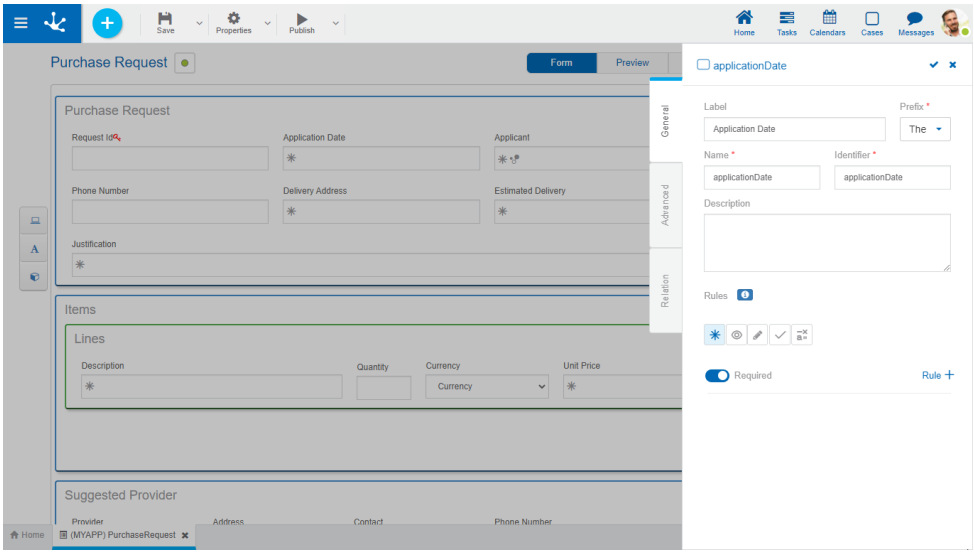
The icon is used to close the properties panel, if it was not previously saved, changes are discarded.

3.6.4.3.1. Field Properties

Phase 2: Form Modeling > Form elements modeling

Pressing the icon on the field opens the vertical panel on the right, which contains the following tabs:

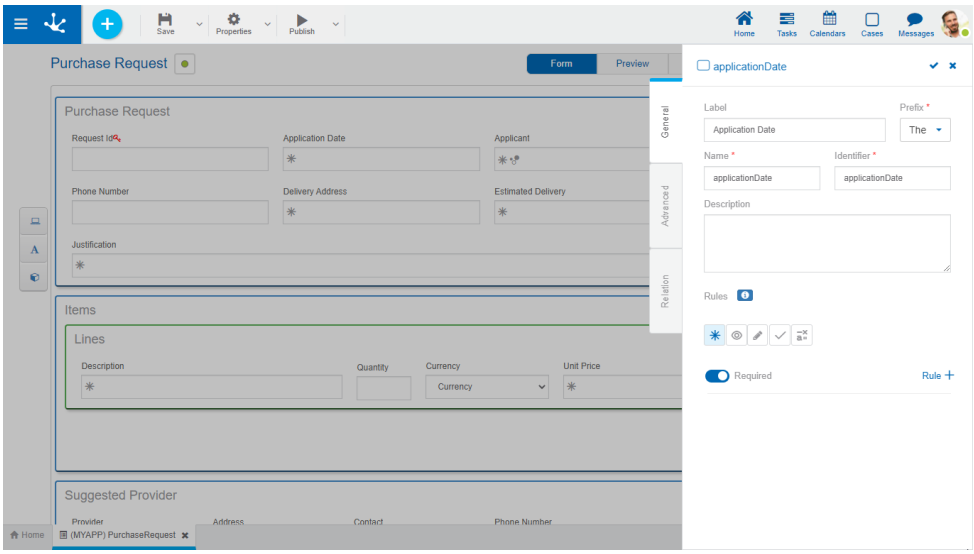
- General
- Advanced
- Relation



3.6.4.3.1.1. General

Phase 2: Forms Modeling > Behavior modeling

The properties panel is displayed on the right side of the form modeler, where the first tab corresponds to general information.



> *An asterisk "*" on the label indicates that the property is required.*

Properties

Label

Allows to enter the text that is displayed on the field. It works together with the prefix to reference the field in validation messages and supports blank spaces.

Prefix

This prefix is used to conform error messages during data entry. It is required. Allows to select the value: "The".

Name

Name assigned to reference a field in the modeling, allowing the field to be uniquely identified within the form. Used in rule wizards to refer to field within conditions. It generates automatically from the Label property, it can be modified by the user and does not allow spaces or special characters.

Identifier

It is the name that is assigned to refer to a field in the programming code, it is used to refer to the field within the Java code in the "Execution Code" tab of the advanced rules and in the JavaScript code in the "Advanced Editing" tab of the form modeler. Allows to uniquely identify the field within the modeled form. It can be modified by the user, as long as no data has been loaded into the form and it does not allow spaces or special characters.

Description

Text that defines the field and optionally its content. Supports multi-language.

Rules

To define rules on behavior, validation and calculation, associated with a field, by using the wizard (ctrl + space).

Shows syntax examples for writing the rules.

Required

Indicates whether the field is required when creating a new form instance or when modifying an existing one.

Required Not required (default)

Rule + Opens an edit area where a rule to determine the required condition can be defined. If a rule is defined, the icon is displayed with light blue borders.

Visible

Indicates whether the field is visible. If this property is not checked, the field is not displayed in the form instances.

Visible (default) Not visible

Rule + Opens an edit area where a rule to determine the visibility condition can be defined. If a rule is defined, the icon is displayed with light blue borders.

Editable

Indicates if the field is editable. If this property is not checked, the user cannot enter or modify values in the field.

Editable (default) Not editable

Rule + Opens an edit area where a rule to determine the editability condition can be defined. If a rule is defined, the icon is displayed with light blue borders.

Valid

Rule + Opens an edit area where you can define the condition that determines if the field value is correct or not. It is possible to define more than one rule. If rules are defined, the icon is displayed with light blue borders.

Calculation

Rule + Opens an edit area where you can define the expression to be executed to calculate the field value. If a rule is defined, the icon is displayed with light blue borders.

Operations when defining the rule:

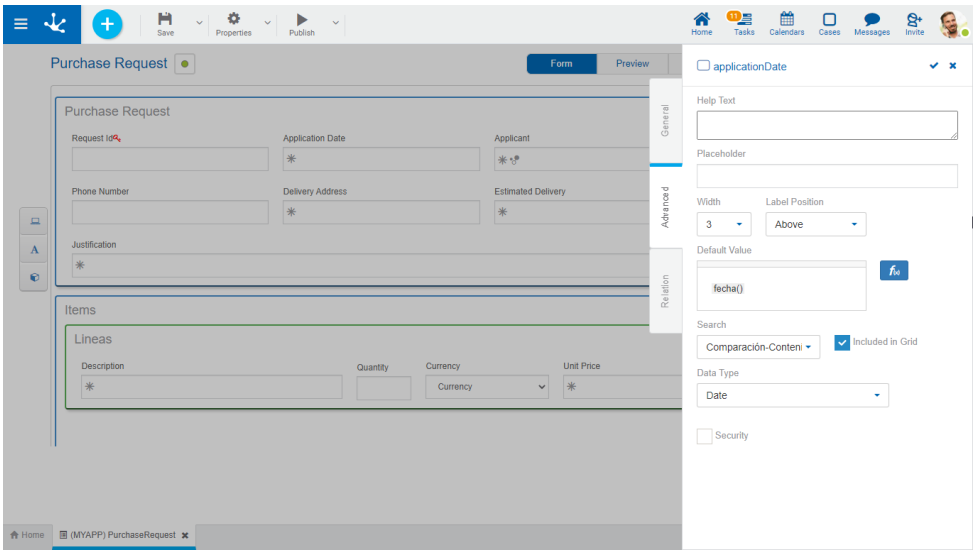
- Saves the new or modified rule
- Cancels the operation

Operations once the rule is defined:

- Edits the existing rule
- Deletes the rule

3.6.4.3.1.2. Advanced

The second tab in the side panel corresponds to the additional properties to define the behavior and functionality of a field.



An asterisk "" on the label indicates that the property is required.*

Properties

Help Text

Its function is to guide the user on the content to load in the field. The text entered as help is displayed when the user hovers the cursor over the field.

Placeholder

Its function is to guide the user on the content to load in the field. Unlike the Help Text property, the text entered as a placeholder is displayed within the field.

Width
It is used to modify the field width in the row. Values between 1 and 12 can be selected, where 12 is the total row width.

Label Position
Indicates if the label is located "Above", on the "Right" side or on the "Left" side of the field. The default value is "Above".
The result of indicating this property is displayed in the design option "Preview".

Multiline
This property determines if the field is displayed in text box mode when using the form, allowing the edition of the text in a larger area. This property is only visible if the field is text type .

Default Value
Allows to assign the value the field has by default at the time of the first data entry, to speed up the loading of values. The default values can be constant values or can be defined from the functions available for the field type. Each function returns a value when using the form and results of more than one function can be combined with text.

Search
Allows to use the field as filter when searching for instances of the form.

Included in Grid
Indicates if the field is displayed in the results grid where the form instances are shown.

Data Type
Determines the data format that can be entered in the field. The supported data set varies, depending on the type of field that has been created from the elements of the left side toolbar.

Length
Determines the maximum length of the value depending on the data type. This attribute is only visible if the field is of text or numeric type .

Key
This property is valid only for the identifier field, It allows to uniquely identify a form instance, that is, such value cannot be repeated between instances of the same form.

Autonumeric
This property is valid only for the identifier field and determines that the field value is automatically generated when creating the form instance.

Content Type
This property is valid only for text type fields. If email value is selected in this property, the field content becomes the recipient when sending an email from a form instance

Security
This property is used to define display and editing restrictions on the field that is being modeled.
If this property is checked, the security functions of field view and editing are created and they are displayed in the use functions panel of the design option "Permissions" of the form, if they are che-

cked, they are incorporated into the form's security functions assigned to the created or modified permission .

Display
This property is only visible if the field is of "Check'' type. Allows to select whether the field is displayed in "Check" or "Toggle" format''.

3.6.4.3.1.3. Relation

The third tab in the side panel corresponds to relation properties and the related attribute.

Relation Properties

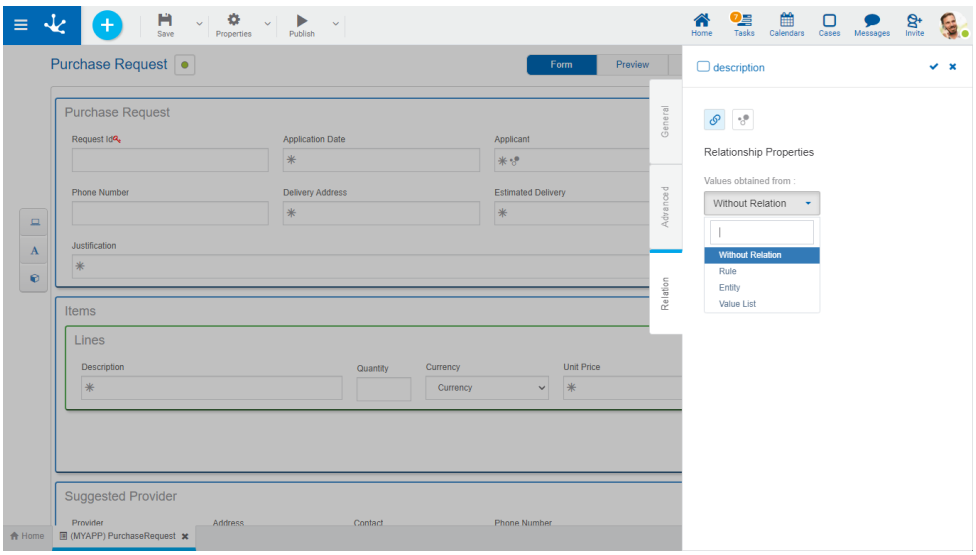
A relation allows to obtain values for the field being modeled from different sources, by selecting the Values obtained from property. If the field does not have a relation, the default option "No Relation" should be modeled.

- Rules
- Entities
- Value List

Related Attribute

This attribute indicates that the field value is retrieved from the attribute value of another entity, related when creating or updating a form instance. One of the relations defined in the form must be selected and then indicate which attribute of the related entity it is linked to.

A related attribute can be modeled regardless of the option selected in the Values obtained from property. This allows a field to have all the functionality of being related, and also to load its value automatically.



An asterisk "" on the label indicates that the property is required.*

Properties

Relation Name
The list of relations defined in the form is displayed, allowing the selection of one of them.

Entity Attribute
The list of attributes of the selected entity that have a format option compatible with the field type is displayed. Attributes can have a relation to an entity, a value list, or a rule modeled in the selected entity.

Relation Type
Reference
When this property is checked, the field value is always subject to the current value of the related attribute, that is, the updated value is automatically displayed.

Copy
When this property is checked, once the retrieved value is loaded, it is not automatically modified again even if the value of the related attribute is modified.

Content Type
The content of the field varies depending on the Value or Description property checked when:

- The entity attribute has a relation modeled.
- The "Entity" option of the field is selected in the Values obtained from (property).

Value
When this property is checked, if the relation is modeled with a:

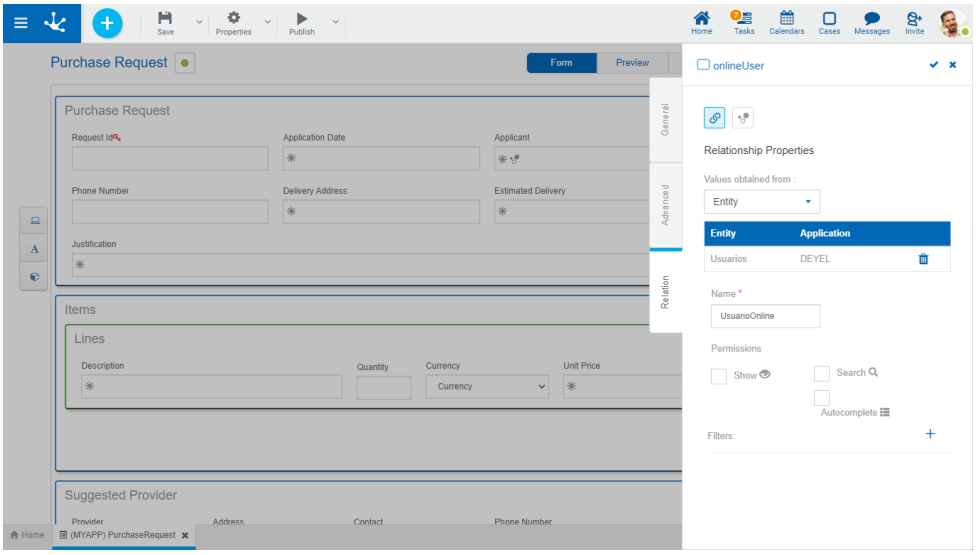
- Related entity, the identifier of the entity is displayed.
- Value list, the value list code is displayed.
- Rule, the code returned by the rule's execution is displayed.

Description
When this property is checked, if the relation is modeled with a:

- Related entity, the short description of the entity is displayed.
- Value list, the value list description is displayed.
- Rule, the description returned by the rule's execution is displayed.

Entities

Through the form modeler, the business entities are modeled. It is also possible to model relations among entities to represent their interaction. The relations defined in this panel can be modeled in the referenced entity so that they are reflected when using the corresponding form.



An asterisk "" on the label indicates that the property is required.*

To define a relation with an entity, the entity to establish a relation with must be selected first, and then complete a set of properties.

Properties

Name

Name of the relation between both entities, it is a modeling-oriented property. It does not allow spaces and must be unique per modeled form.

Permissions

The permissions of the relation define which functionalities are enabled in the relation control.

Show

By checking this property, the user is allowed to display the entity instance with which a relation is established in the form use.

Search

By checking this property, the user is allowed to access the results grid to find the entity instance to establish a relation with in the form use.

Autocomplete

If this property is checked, predictive text functionality is presented to users. Based on the characters that the user types, a subset of values is proposed, which coincide with all or part of the text entered. This facility is called autocomplete.

In order to use this property, the referenced entity must have the Short Description modeled. Such description must not contain date or time variables, nor can it have a variable related with another entity.

Filter

Allows to narrow the search results on the related entity.

The filter is applied both in the autocomplete of the field, as well as in the show through the magnifier that is displayed when you hover the mouse over the field.

To create a filter, click on the icon + and a panel opens to complete the following properties:

Filtered Attribute

Allows to select an attribute of the related entity.

Condition

Allows you to select a condition as part of the filter.

Type

The possible values to select are "Value" and "Field".

Value

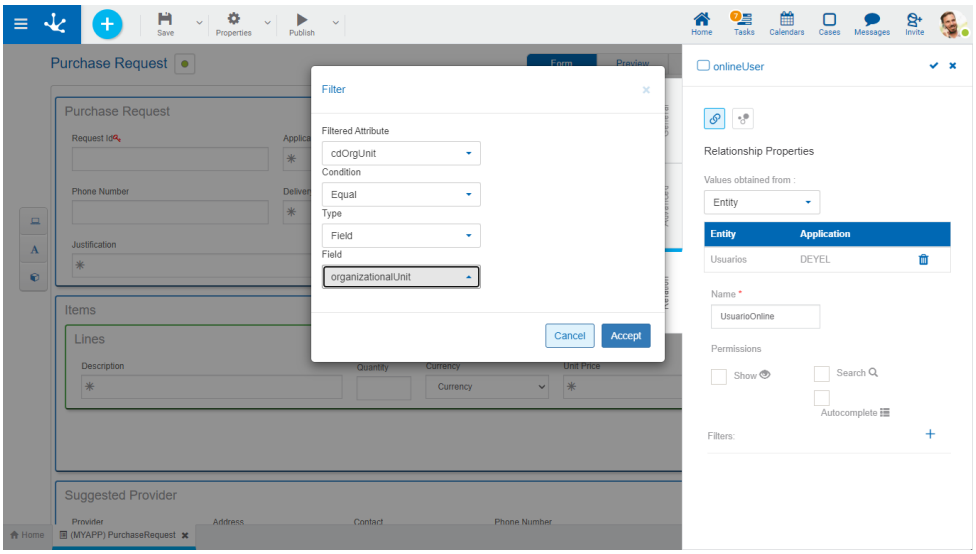
Allows to enter fixed values.

Field

Allows to select of a form field. It must be considered that its content depends on the Type of Content property of the related attribute

Example of Filtered Entities

A relation with the user entity is modeled.



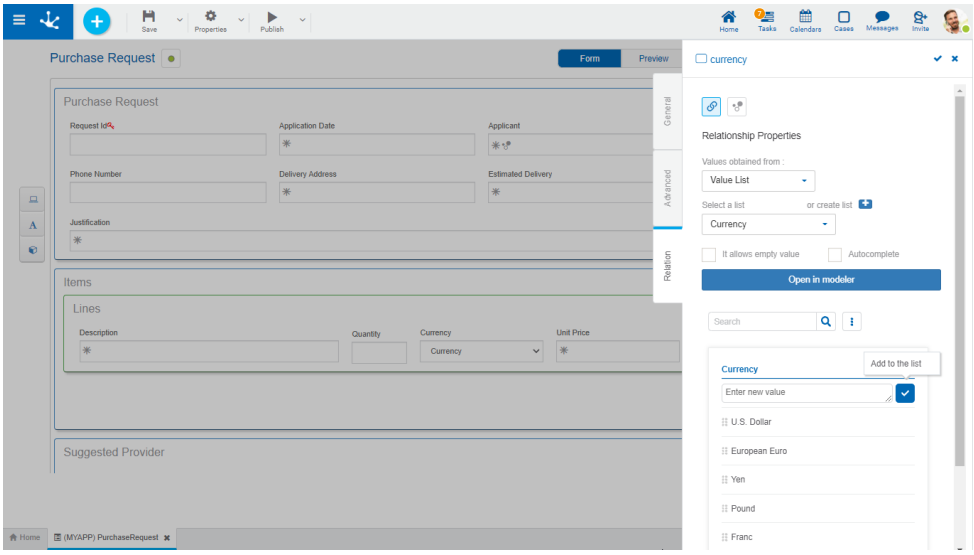
> *In this example the organizational unit field must have the Type of Content property defined as "Value". If during execution the description of this field needs to be displayed, the relation with the organizational unit entity must also be modeled.*

During execution, by clicking on the "User" field search, all users whose organizational unit is the one reported in the "Organizational Unit" field of the form are displayed.

Value List

Allows to define a set of values grouped under some criteria so that the application user knows the possible values a field can take.

New lists can be created from the modeler, then edit and relate them to form fields.



An asterisk "" on the label indicates that the property is required.*

To create a new list of values, click on the icon .
A panel opens to enter the name of the list, the application to which it belongs is selected, if alphabetical sorting is indicated, descending order can be selected. Pressing the "Accept" button creates the list of values, subsequently allowing the entry of new values.

The list of values can be used in any other field of the same or another form.

Properties

Select a list
Allows to select a particular list from the set of already existing lists, being able to filter such set by entering text into the search field that is above the list.

Allows empty
If this property is checked, the empty value is included as the first option in the list. Otherwise the first value in the list is displayed in the field.

Autocomplete
If this property is checked, predictive text functionality is presented to users. Based on the characters that the user types, a subset of values is proposed, which coincide with all or part of the text entered. This facility is called autocomplete.

Other Elements

Open in modeler
Allows using the list of values modeler for the definition, instead of the field properties panel.

Allows to filter values from the list based on the characters entered. If a list is very long it helps users to easily visualize the desired values.

Enables an option that allows to add the internal code to the list values.

Operations on Values

- Allows adding each entered value to the list of values.
- Double click: Allows to modify a value in the list.
- Move: Allows to change the position of a value within the list by dragging the value with the mouse.

Hovering the cursor over each of the values entered, a set of icons is displayed and this allows to perform different operations.

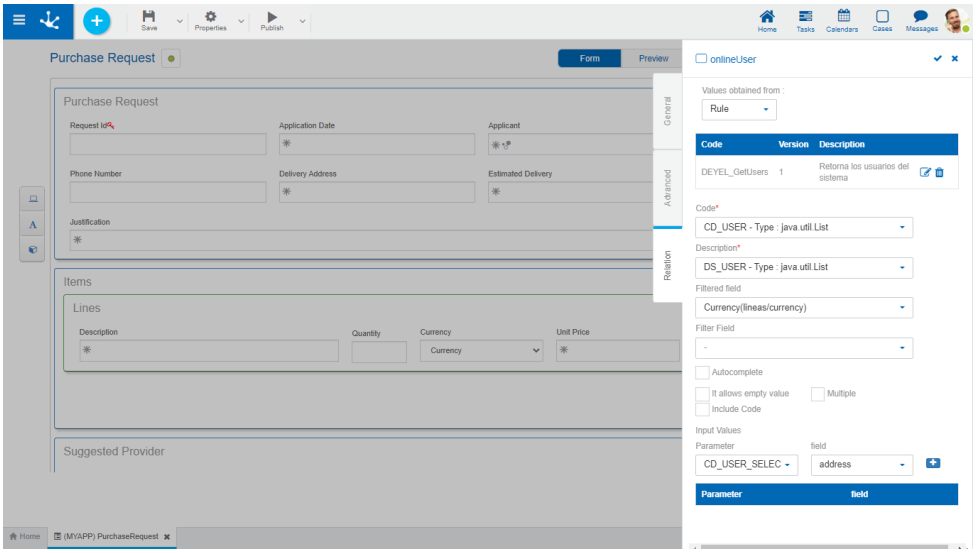
- Allows to delete a value from the list of values. Once deleted, it is displayed in gray and crossed out.
- Allows to restore a previously deleted value.

Display the Selected Line

- Hides the icons that are displayed.
- Shows hidden icons.

Rules

Through this relation, it is possible to limit the set of values to be loaded in a field to those from the indicated source.
When relating fields to rules, the possible values are retrieved from a predetermined business rule.When establishing a relation with a rule, a list of rules is displayed to select which of them to establish the relation with the field.



An asterisk "" on the label indicates that the property is required.*

Properties

Code
Represents a business rule parameter, which allows the description to be uniquely identified. Each parameter value is related with its corresponding description.

Description
Represents a business rule output parameter. Each parameter value is related with its corresponding code.

Filtered Field
Modeling filtered fields allows to make the value lists of fields associated with rules variable when using form instances. In many cases, depending on a value loaded in a form field, the possible values of another field must be restricted.
This property is completed with the form field on which the filter is applied. See detail of property use.

Filter Field
Name of the business rule parameter whose content is used to filter the field values indicated in the Filtered Field property.

Autocomplete
If this property is checked, predictive text functionality is presented to users. Based on the characters that the user types, a subset of values is proposed, which coincide with all or part of the text entered. This facility is called autocomplete.

Search by Content
It is only enabled if the property is checked Autocomplete. Allows selection of the number of characters on which the autocomplete search begins. In this way, if the "1 character" value is selected, the autocomplete search will start after the first character is typed.

Allow empty
If this property is checked, in addition to the values that come from the rule, a value may not be selected for this field. An entry with no value is added to the rule results list. See detail of property use.

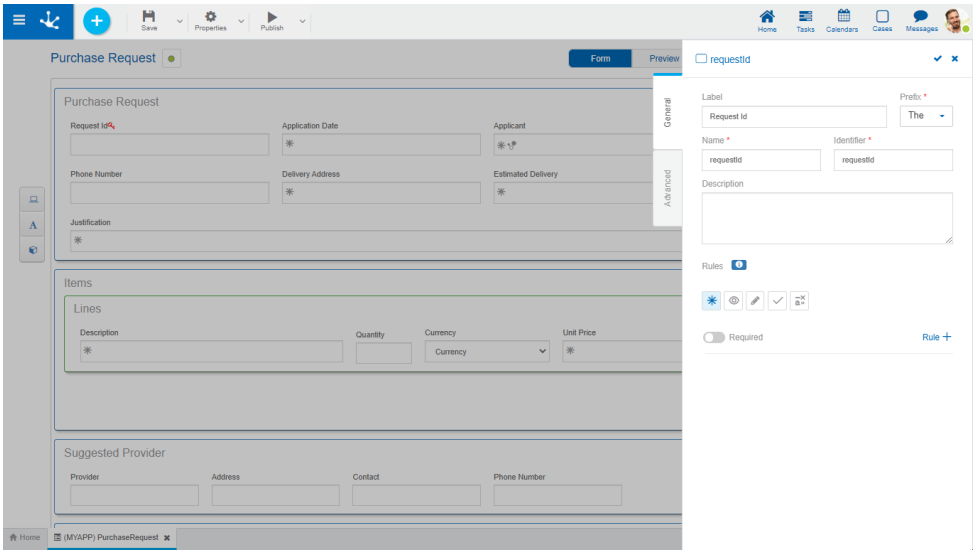
Multiple
If this property is checked, the user can select more than one value from the list, otherwise, a single element can be selected. See detail of property use.

Include Code
This property indicates how the list of possible values is displayed. The default view of the list shows only the value description. If this indicator is checked, the code for each element is also displayed.

3.6.4.3.2. Graphic Element Properties

This section explains the properties of the different types of graphic elements present in the form modeler.
Of all the elements of this type, it is only possible to edit "Title" and "Label" elements, since the rest do not have the icon available .

Pressing the icon on a graphic element opens the properties panel.



> *An asterisk "*" on the label indicates that the property is required.*

Properties

Text

Title or label text.

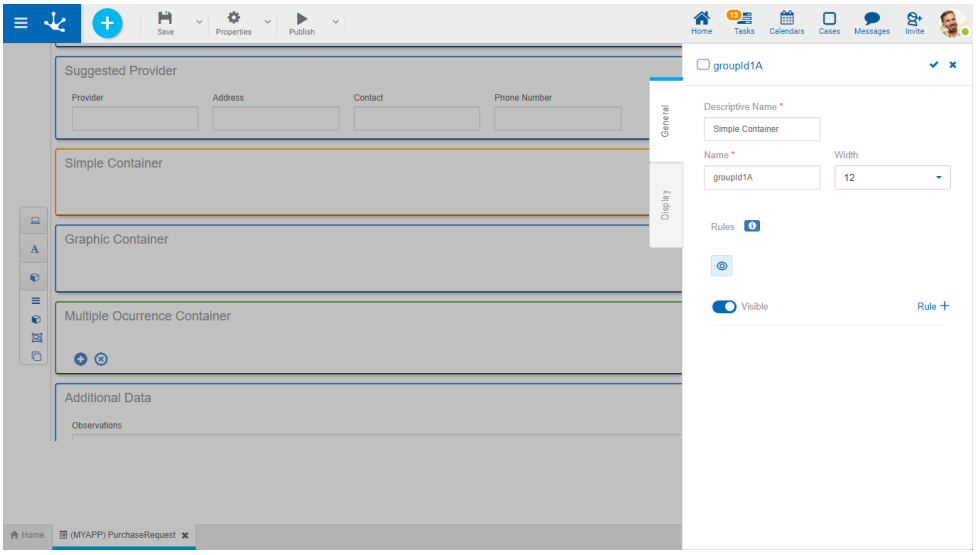
Width

Used to modify the width that the field uses in the row, being able to select values between 1 and 12, being 12 the total width of the row.

3.6.4.3.3. Container Properties

Properties are defined for different types of containers.

- Single Container
- Graphic Container
- Multiple Occurrence Container

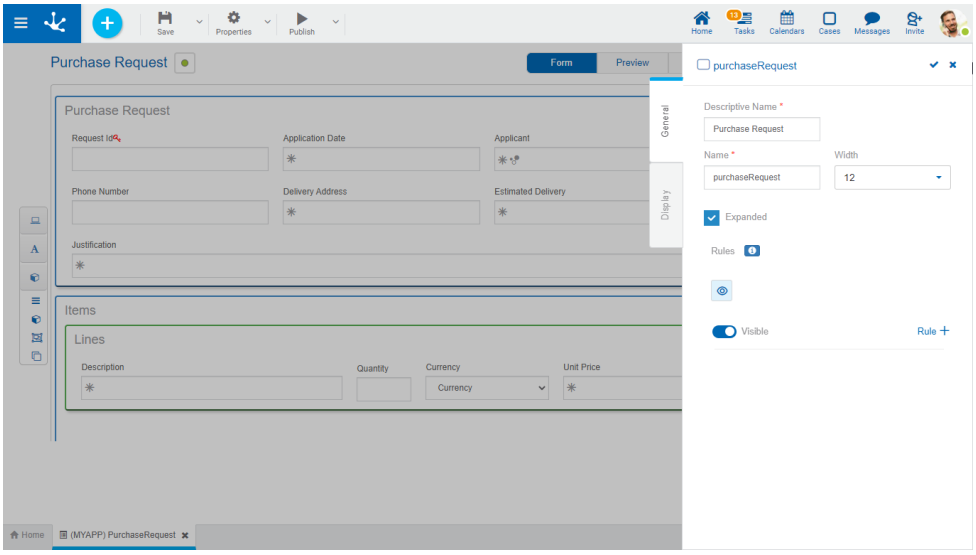


Pressing the icon on the field opens the vertical panel on the right, which contains the following tabs:

- General
- Display

3.6.4.3.3.1. General

The properties panel is displayed on the right side of the form modeler, where the first tab corresponds to general information.



An asterisk "" on the label indicates that the property is required.*

Properties

Descriptive Name
Allows to identify the container with a name oriented to the end user. Spaces and special characters are allowed.

Name
Allows to identify the container with a name oriented to the modeler user. Spaces and special characters are not allowed.

Field Group property
Indicates whether the fields that are in this container are part of a group. If a group has this property checked, it cannot include an iterative container.
It is only used for single type containers.

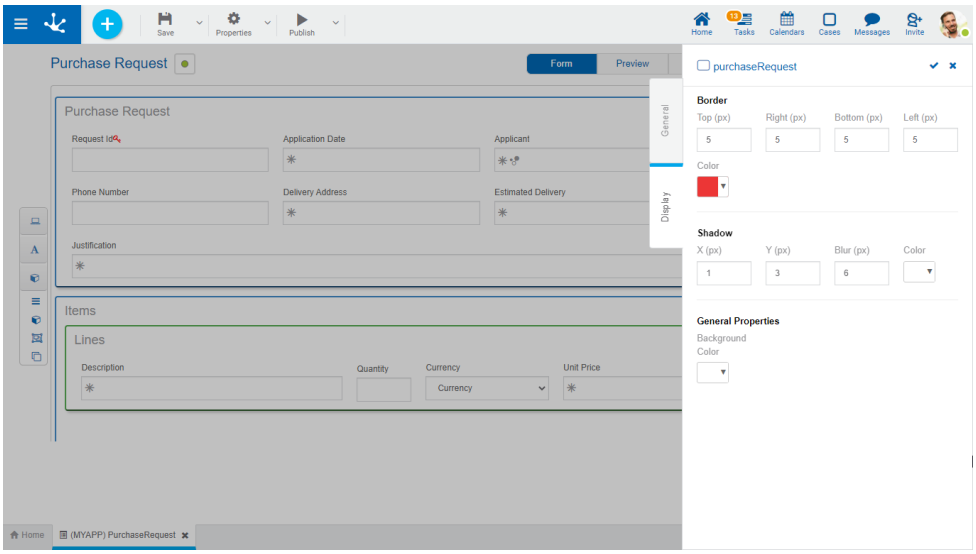
Width
Used to modify the container width, being able to select values between 1 and 12, being 12 the total width.

Expanded
Indicates whether the container is displayed expanded, showing its content, or closed when showing the form instance. It is used only for graphic containers.

Visible
This property indicates whether the container and its content are visible when showing the form instance. Allows the definition of visibility rules associated with the container as well as those defined for the fields.

3.6.4.3.3.2. Visualización

The second tab corresponds to information regarding the container display.
The result of indicating these properties can be seen in the design option "Preview".



An asterisk "" on the label indicates that the property is required.*

Properties

Border
Defines the width and color of the container borders. Width is measured in pixels, with 0 being the minimum value and 10 being the maximum value. The default value for each property is 0, with no border.

Top
Indicates the width of the top border.

Right
Indicates the width of the right border.

Bottom
Indicates the width of the bottom border.

Left
Indicates the width of the left border.

Color
Allows to select the border color from a palette.

Shadow
Allows to define a shaded area framing the container. The properties X, Y and Blur are measured in pixels, with 0 being the minimum value and 10 being the maximum value.

X
Indicates the displacement of the shadow on the vertical axis with respect to the frame of the container.

Y
Indicates the displacement of the shadow on the horizontal axis with respect to the frame of the container.

Blur
Defines how sharp the shadow is displayed.

Color
Allows to select the shadow color from a palette.

General Properties
Defines the background color of the container.

Background Color
Allows to select the container background color from a palette.

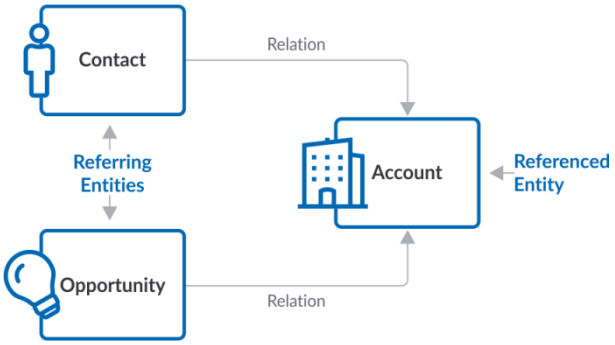
3.6.4.4. Modeling of Related Entities

Phase 2: Forms Modeling > Advanced Tips > Modeling related entities

Related entity modeling allows to define visibility and navigability among entities.

When defining relations among entities, the following concepts are used:

- **Referring entity** is the one that defines the relation with another modeled entity.
- **Referenced entity** is the one that has a relation defined from another modeled entity.
- **Relation** is the one that was defined in the referring entity.
- **Referring relation** is the one that was defined in the referenced entity to a referring entity.



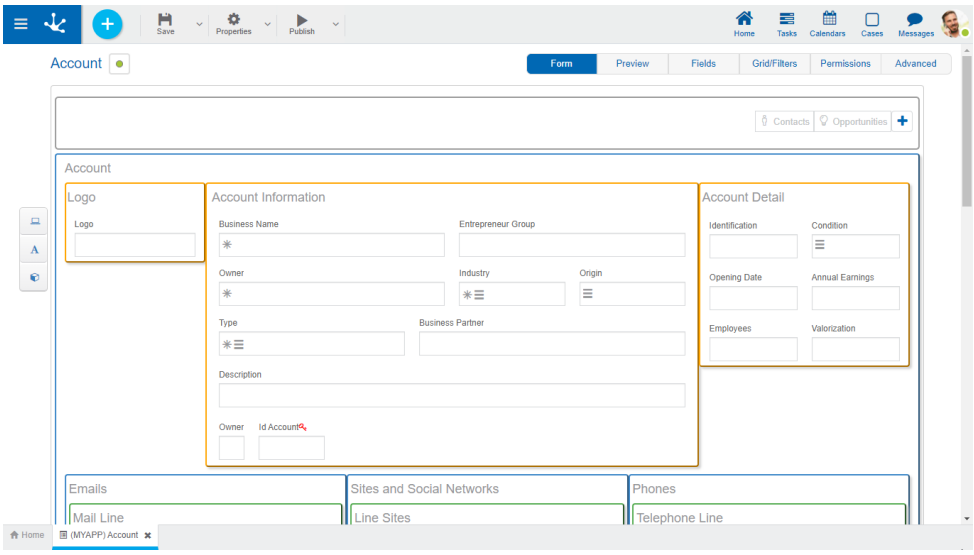
> *For the purpose of detailing the modeling of related entities, an example is used where the Contact and Opportunity entities are related to the Account entity. Such entities can be modeled from agile forms created from the templates provided as examples, after separating the forms from their respective processes.*

Relations in the Referenced Entity

Container

In the form modeler, referring relations are modeled in the first container of the referenced entity and are only displayed if there are relations defined in other forms. Within this container controls identified with the names of the relations defined in the referring entities can be included.

The icon + allows to add a list of controls to the form that represent the referring relations. By pressing this icon, a list of referring relations is displayed where you can select those you want to display at the top of the form.



The example shows the relation defined between Contact and Account and the relation between Opportunity and Account, as well as the operations and properties of both relations.

Operations

Hides the display of the relation, does not delete it.

Modifies the presentation order of relations.

Opens the properties panel of the relation.

Properties

Entity
Name of the entity where the relation was defined. Not editable.

Application
Name of the application where the entity that contains the relation was defined. Non-editable.

Relation Name
Descriptive name of the relation as defined in the referring entity when modeling the field that contains the relation. Non-editable.

Key Field
Identifier of the referring entity attribute used to define the relation. Non-editable.

Singular Relation Name
Represents the name in singular that is assigned to the relation. It is displayed if, when using the form, it has a single instance for the modeled relation.

Plural Relation Name
Represents the plural name that is assigned to the relation. It is displayed if, when using the form, it has more than one instance for the modeled relation.

Relation Behavior
Allows to define behavior when using the form, to assign values from a form instance to related form instances, by means of relations defined among entities.
The icon allows to open the value match wizard in an additional section to the right of the relation properties panel.

Correspondence

Destination
The form field where the value is received is selected. Once selected, the Value Type is displayed.

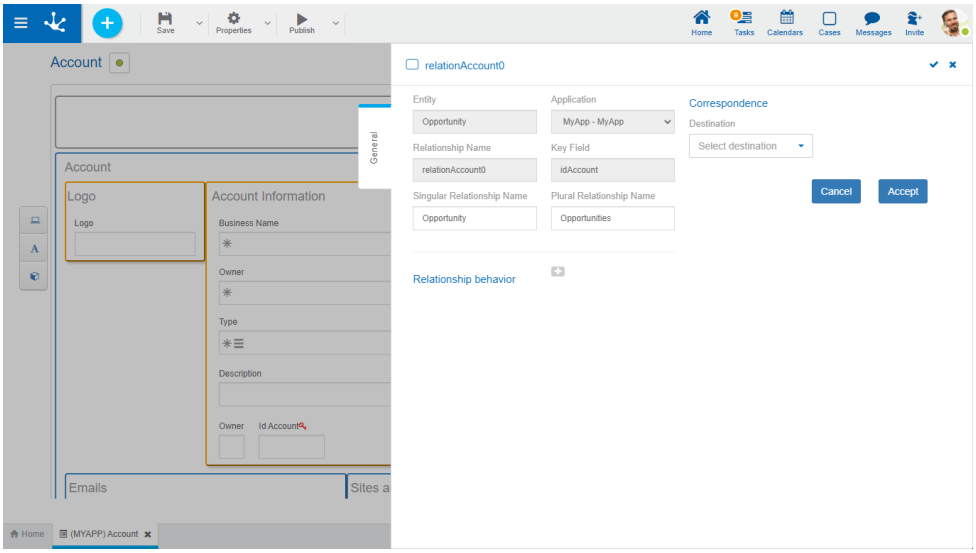
Value Type

- Constant: Indicates that the content of the field defined in the Destination property is a value entered by the IT modeler user.
- Field: Indicates that the content of the field defined in the Destination property is obtained from a form field that is being modeled.

Once the Value Type is selected, the Value is displayed.

Value

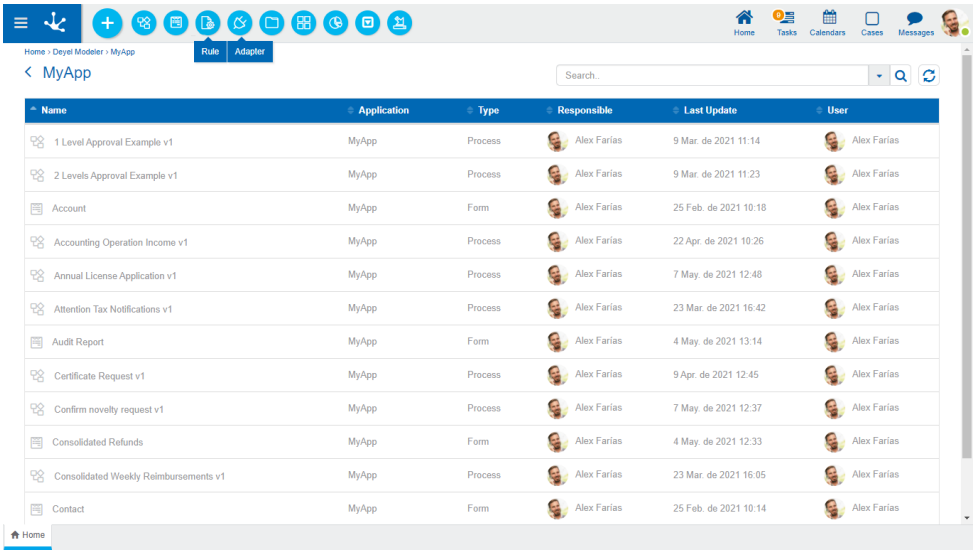
- If the value type is constant, it must be entered for matching.
- If the value type is a field, a list is displayed with all the form fields corresponding to the referring entity, for selection.



> *The example shows the properties of the Contact relation and the definition of the value match between the Industry field of the Contact entity and the Industry field of the Account entity. When the Contact entity is instantiated, the Industry field is automatically populated with the Industry field value of the Account entity.*

3.6.5. Rules Modeling

The rules modeler is a tool that allows IT users to model their own advanced rules, using different adapters.



Its main characteristic lies in its simplicity to model rules using different modeling facilities.

The general characteristics of the rules modeling environment and the main elements that compose it are described in the following topics:

- Modeling Facilities
- Rule Properties

3.6.5.1. Business Rules

A business rule allows to carry out a specific task, being able to receive information, perform certain processing and inform the results obtained.

Deyel uses business rules to define the behavior of processes and forms. Rules can be used in validations, specific business logic, process flow control, integration with other applications, and field display control in forms, among other functionalities.

Types

- Advanced Rules
- Embedded Rules

3.6.5.1.1. Embedded Rules

These rules are used to define form field behavior, validations, and calculations, as well as form validations. Embedded rules allow to define logical conditions and arithmetic expressions, which are evaluated using forms and their related processes.

The definition is done using a wizard integrated into the corresponding modelers. This wizard can be found in the property definition panels of fields and containers, forms and flows if they were conditional.

Unlike the advanced rules, the embedded ones are not displayed into the grid of the modeler because they can only be used in the **Deyel** object where they are defined.

Types

Behaviour

A behavior rule is used to verify the fulfillment of a condition. A logical expression is evaluated and a "True" or "False" value is returned depending on whether the condition is met or not.

Embedded rules are modeled in form fields and containers, both in the form modeler and in the process modeler when modeling the activities and in the flows if they have conditions.

They are classified according to the purpose of their use.

- Visibility
  The modeled condition defines whether a form field or section is visible to the user at the time of use.

- Editability
  The modeled condition defines whether a form field is editable for the user at the time of use.

- Required
  The modeled condition defines whether a field is required at the time of using the form.

The execution of these rules returns a logical value, "True" or "False".

Validation

A validation rule is used to ensure the proper information entry or to control incorrect situations. It can be set at field or form level.

This rule evaluates a condition and returns a message when it is met, indicating the wrong or invalid situation to the user. If no message returns, it means that the validation was successful.

Conditions involving one or multiple fields and the relations among them can be defined.

Calculation

They define arithmetic expressions that allow to perform calculations to fill in field values when using the form. The rule is defined with a calculation algorithm for the target field that is populated with the resulting value. This value must be of the same type as the field that contains it.

Conditions can also be defined to determine when to execute the calculation rule.

Use

Embedded rules are modeled within the definition panels of **Deyel** objects properties.

- Rules in Field Properties
- Rules in Form Properties
- Rules in Container Properties
- Rules in Activity Properties
- Rules in Field Properties of Agile Forms
- Rules in Properties of Agile Forms
- Rules in Container Properties of Agile Forms

The table below summarizes where the different types of embedded rules can be modeled.

Rule Type	Field	Form	Container	Flow	Activity
Behaviour	✔		✔	✔	✔
Validation	✔	✔			✔
Calculation	✔				✔

> *Only visibility behavior rules can be modeled in a container.*

3.6.5.1.1.1. Rule Elements

Embedded rules can be made up of different types of elements:

- Form fields
  Form fields are used by their names. In the wizard they are identified with ● on the left.

- Group containers
  Form containers are used by their names. Whenever a group container is used, indicate the field separated by a ".". In the wizard they are identified with ● on the left.
  Example: PersonalData.CompanyName

- Iterative containers
  Their name is displayed instead of their label. Whenever an iterative container is used, indicate the field separated by a ".". In the wizard they are identified with ● on the left.
  Example: Items.Quantity

- Functions. In the wizard they are identified with ● on the left.

- Advanced rules. In the wizard they are identified with ● on the left.

- Form field relation. In the wizard they are identified with ● on the left.

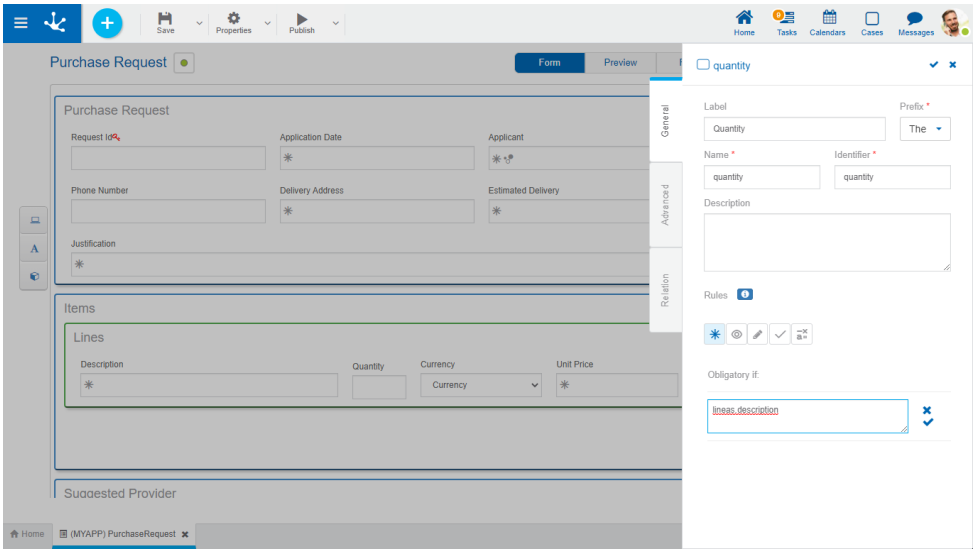
- Possible values of related fields. In the wizard they are identified with ● on the left.

When modeling a related field, either to a value list, rule, or entity, the wizard displays the field, and the possible values of its relation.

In all cases, the type of data they represent and, in the case of functions or rules, the value they return is indicated on the right.

In addition to the elements proposed by the wizard, the user can include, in the rules, texts enclosed in double quotes and integers or decimals, the latter with separator "."

In the event of a syntax error, a message is displayed at the top of the properties panel, indicating the problem.



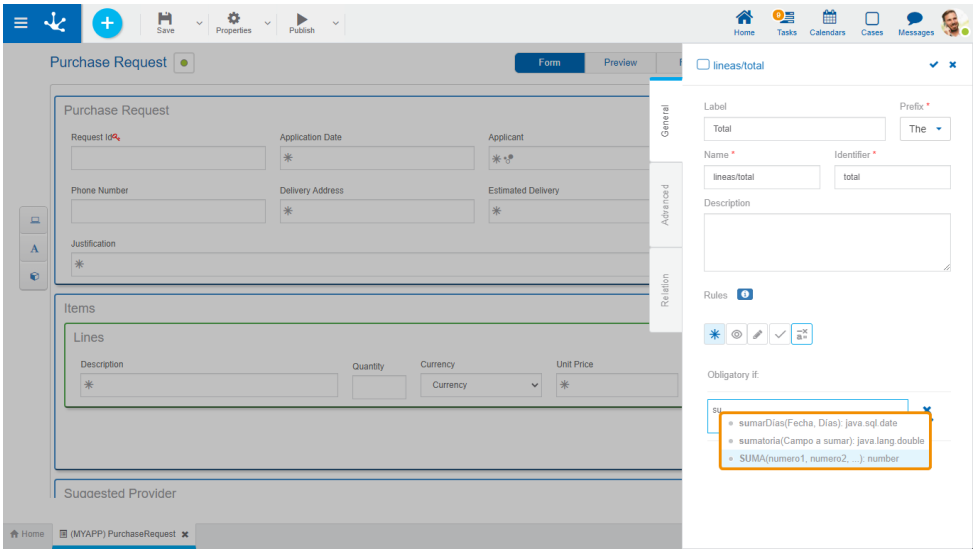
Available Operators

Operator	Description	Example
==	Compares if two operands are equal	5 == getDay()
+	Adds two operands	5 + field3
-	Substracts two operands	price - discount
*	Multiplies two operands	price * 1.21
/	Divides the first operand by the second one	10 / 2
!=	Compares if two operands are equal	last name != "Paz"
<	Compares if the first operand	1 < 3

Operator	Description	Example
	is less than the second one	
>	Compares if the first operand is greater than the second one	4 > 1
<=	Compares if the first operand is less or equal than the second one	1 <= 3
>=	Compares if the first operand is greater than or equal to the second one	3 >= 3
%	Calculates the division remainder of the first operand by the second one (mod)	5 % 3, the result is 2
**	Raises the first operator to the power indicated on the second operator	2 ** 3

3.6.5.1.1.2. Wizard

A wizard to model the rules can be used in the rule editing area. This wizard is activated by pressing the "Ctrl + Space" keys. As text is entered in the wizard, it uses the auto-complete functionality and shows only the elements whose names begin with the text typed by the user.



3.6.5.1.1.3. Functions

A function is a software unit. It performs a specific task, receives input parameters, and returns a result.

They can be used:

From the function selection wizard in:

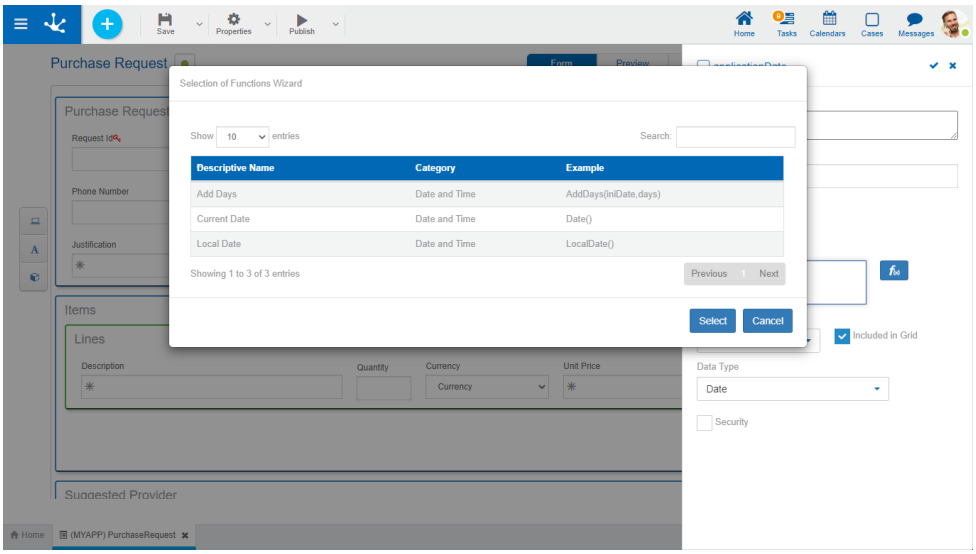
- Form fields, when defining the Default Value property.
- The processes, by defining the Case description property and in alerts and automatic activity parameters.
- The process flows, when defining theMessage property.

When defining conditions in:

- Rules in form properties.
- Field Properties to verify conditions of edition, mandatory nature, visibility, validation and calculation rules.
- Gateway properties when output flow conditions are evaluated.

Function Selection Wizard

The wizard is used from the icon . When pressed, a window with the list of functions opens and once the function has been selected, it is included in the description area of the form or case.



Elements

- Selection of number of records to display in the window.
- Search field to be able to filter functions, filtering can be performed by the properties Descriptive Name, Category or Example
- List of functions where properties are displayed in a grid mode.
- Buttons for paging.

Properties

Descriptive Name
It is used in the interface when it is referenced, in the functions list . Supports multi-language.

Category
Grouping by function type.

Example
Corresponds to how the function can be used in the calculation or condition areas.

Categories

Functions are grouped into categories.

- Execution
- System
- Date and time
- Math
- Validations
- Miscellaneous
- Exclusives for Rules

Execution

These functions retrieve information related to the form instance that is being used, to the process if it had one associated and from the connected user.

Descriptive Name	Description	Example
Current User Last Name	Returns the last name of the user that started the session.	currentUserLast-Name()
Current Activity Code	Returns the code of the activity in progress. It does not work in the first process activity because the case has not been created.	currentActivityCo-de()
Current Process Code	Returns the code of the process in progress. It does not work in the first process activity because the case has not been created.	processCode()
Current Activity Description	Returns the description of the activity in progress. It does not work in the first process activity because the case has not been created.	currentActivityDes-cription()
Current Process Description	Returns the description of the process in progress. It does not work in the first process activity because the	processDescription()

Descriptive Name	Description	Example
	case has not been created.	
Email of the Executing Participant's Organizational Unit	Returns the organizational unit email of the participant responsible for executing the activity in progress, if it is a role, it returns empty. It does not work in the first process activity.	userOrgUnitEmail()
Email of the User Administrator of the Executing Participant's Organizational Unit	Returns the administration user email of the organizational unit of the participant responsible for executing the current activity. It does not work in the first process activity.	orgUnitAdminEmail()
Email of the Role Coordinator Responsible for the Current Activity	Returns the email of the role coordinator user modeled as responsible for the activity in progress, only if the person responsible is a role, otherwise it returns empty. It does not work in the first process activity.	currentRoleCoordinatorEmail()
Executing Participant Email	Returns the email of the participant responsible for executing the current activity It does not work in the first process activity. If the responsible participant is a role or an organizational unit, it returns the emails of all users belonging to the role or organizational unit.	currentUserEmail()
Current Activity Name	Returns the name of the activity in progress. It does not work in the first process activity because the case has not been created.	currentActivityName()
Name of the Current User's Organizational Unit	Returns the organizational unit name of the user who started the session.	currentUserOrgUnitName()
Current Process Name	Returns the name of the process in progress. It does not work in the first process activity.	processName()
Current User Name	Returns the name of the user that started the session.	currentUserName()
Case ID	Returns the case id without applying the case display mask. It does not work in the first process activity because the case has not been created.	caseNumber()
Executing Parti-	Returns the organizational unit code of the participant	currentOrgUnitCo-

Descriptive Name	Description	Example
cipant's Organizational Unit	responsible for executing the current activity. If the participant responsible is a role, it returns empty. It does not work in the first process activity.	de()
Code of the Current User's Organizational Unit	Returns the organizational unit code of the user that started the session.	currentUserOrgUnit()
Executing Participant	Returns the participant responsible for executing the current activity It does not work in the first process activity. It can return a user, a role or a unit: - Returns a user when the participant responsible for the activity is a user, or when it is a role and the user is assigned the activity. - Returns a role only when the participant responsible is a role and the activity is generic, until one of the users is assigned the activity. - Returns an organizational unit when the participant responsible for the activity is a unit.	executingUser()
Executing User of the Current Process Activity	Returns the user that executed the activity informed by parameter. It requires the code of an activity already executed. It does not work in the first process activity.	activityExecutingUser(activityNo)
Current User	Returns the user that started the session.	currentUser()
Current Process Version	Returns the version of the process in progress. It does not work in the first process activity.	processVersion()
Last Button Pressed	Returns the last button pressed.	lastButtonPressed()

System

These functions retrieve information related to the environment where the form and the modeled application are being used, if there is one associated.

Descriptive Name	Description	Example
Web Server Address	Returns the web server address.	webServerAddress()

Descriptive Name	Description	Example
Workflow Engine Email	Returns the workflow engine email address.	workflowEmail()
Company Name	Returns the name of the company.	companyName()
Workflow Engine Name	Returns the workflow engine name.	workflowEngineNa-me()

Date and Time

Retrieve information related to dates and times and carry out operations with them.

Descriptive Name	Description	Example
Current Year	Returns the current year.	year()
Difference Bet-ween Work Days	Returns the difference in business days between 2 past dates as parameters.	difBetweenWorkDa-ys(dateFrom,dateTo)
Difference Bet-ween Hours	Returns the time difference between 2 times entered as parameters. The format must be HH:MM:SS.	diffInTi-me(timeFrom,timeTo)
Difference in Mi-nutes between Times	Returns the difference in minutes between 2 times en-tered as parameters. The format is HH:MM:SS.	difInMinu-tes(timeFrom,timeTo)
Difference bet-ween Days	Returns the difference in days between 2 dates entered as parameters. The format is DD/MM/YYYY.	diffBet-weenDays(dateFrom, dateTo)
Current Day	Returns the current day.	day()
In between Dates	Validates that a date falls within a valid date range. The format is DD/MM/YYYY.	inBetweenDa-tes(pDate,pStartDate ,pEndDate)
Current Date	Returns the server current date.	date()
Local Date	Returns the local date according to the user's time and time zone.	localDate()

Descriptive Name	Description	Example
Current Time	Returns the server current time.	time()
Local Time	Returns the local time according to the user's time zone.	localTime()
Current Month	Returns the current month.	month()
Month Name	Returns the name of the month according to the number entered.	monthName(monthnumber)
Add days	Returns the date obtained as a result of adding days to a date.	addDays(pStartDate,pDays)

Math

These functions perform mathematical operations based on the form fields.

Descriptive Name	Description	Example
Sum	Returns the result of the sum of all the field values in a multiple-occurrence container, taken as a parameter. The field data type must be integer or decimal.	sum(field)

Validations

The functions in this category allow to confirm a question about the information used in the form.

Descriptive Name	Description	Example
Is it an Email?	Verifies that the text entered as a parameter has an email format.	isEmail(text)
Only Letters?	Verifies that the value entered as a parameter consists only of letters.	onlyLetters(Text)
Only Numbers?	Verifies that the value entered as a parameter is a valid number.	onlyNumbers(Text)

Miscellaneous

This category groups functions that return information on different topics.

Descriptive Name	Description	Example
Organizational Unit Email	Returns the email address of an organizational unit whose code is entered as a parameter.	orgUnitEmail(orgUnitId)
Organizational Unit Users Emails	Returns the email address of an organizational unit users whose code is entered as a parameter.	orgUnitUserEmail(orgUnitId)
Organizational Unit Administrator Email	Returns the email address of an organizational unit administrator whose code is entered as a parameter.	orgUnitAdminEmail(orgUnitId)
Role Coordinator Email	Returns the email address of a role coordinator whose code is entered as a parameter.	roleCoordinatorEmail(roleid)
Email of the Activity Responsible Participant	Returns the email address of the participant responsible for executing an activity whose code is entered as a parameter. If the participant is a role or an organizational unit, it returns the emails of all its members.	activityResponsibleEmail(activityNo)
User Email	Returns the email address of a user whose code is entered as a parameter.	userEmail(userid)
Role Users Emails	Returns the email addresses of an organizational unit users whose code is entered as a parameter.	roleUserEmail(roleid)
Case Show Link	Returns the link to the case show in progress. Requires the user to be logged in.	caseShowLink()
Activity Execution Link	Returns the link to the execution of the activity in progress, requires the user to be connected.	activityExecLink()
My Tasks Link	Returns the link to the user's task list.	myTasksLink()
Login Link	Returns the link to the user portal home panel.	loginLink()

Exclusives for Rules

Functions in this category are used in embedded rules modeling by means of their wizard.

Descriptive Name	Description	Example

Descriptive Name	Description	Example
IF(logical_test,value_if_true,value_if_false)	Checks if the condition indicated in "logic_test" is met. If it is true, it returns the value informed in "value_if_true", if not, it returns the informed value in "value_if_false". The "value_if_false" parameter is optional, if the condition is not met and such parameter is not defined, the function does not return a value.	IF(amount > 1000000, "Amount exceeds limit") If the amount is greater than 1,000,000, the user receives the "Amount exceeds limit" message.
AND(logical_value1, logical_value2,...)	Evaluates that all parameters return "True", in which case it returns "True" value.	AND(day() > 5, month() == "April") Returns "True" only when the day of the month is greater than 5 and the month is April.
OR(logical_value1, logical_value2,...)	Evaluates all parameters and returns "False" value if all parameters return "False". If at least one of them returns "True", it returns "True" value.	OR(contact == "AFARIAS", currentUser() == "AFARIAS") Returns "False" only when the current user and contact are different from AFARIAS.
NO(logical_value)	IF"logical_value" is "True", it returns "False" value. IF"logical_value" is "False", it returns "True" value.	NO(5 > 1) Returns "False", since it is the logical value contrary to evaluation 5>1.
ADD(number1, number2,...)	Adds all the parameters.	ADD(AmountInvoiced, TaxAmount) Returns the value resulting from adding both amounts.
CONCAT(text1,text2,...)	Joins multiple text elements into one.	CONCAT("Hello,", currentUser()) Returns "Hello AFARIAS", where AFARIAS is the current user.
MAX(number1, number2,...)	Returns the maximum value from a list of parameters. Ignores logical values and text.	MAX(2, 5, customerNumber, 22) Returns 22.
ISBLANK(value)	Checks the value of the reported parameter and returns "True" if it has no content. If the parameter	ISBLANK(description) Returns "False", when

Descriptive Name	Description	Example
	is an iterative field, all its occurrences must be empty for it to return true.	description has content.
isURL(text)	Indicates whether the text entered corresponds to a valid URL format. Valid Examples: • www.sitename.com ; sitename.com; https://sitename.com • 192.1.1.111; 192.1.1.111:8090 ; 192.1.1.111:8090/myfunction	IF(NO(isUrl(field)), "Must inform URL") If field is equal to "Deyel.com", the user does not receive any messages.
REPLACE(text to search; text to replace; original text)	Replaces in "original text", the "text to search" matches with "text to replace".	REPLACE("Supplier XX", "New Supplier", contractText) As a result, the original content of the contractText field remains with all occurrences of Supplier XX replaced with New Supplier.
REGEXMATCH(text, regular_expression)	Returns "True" value if the first parameter meets the regular expression indicated in the second parameter. Regular expressions, also known as regex, are widely used, standardized writing patterns used mainly to process texts. They allow, for example, to validate formats, extract a part of text or replace occurrences of a character string. There are many free access sites that provide information on use and syntax and offer the possibility tu perform simulations as tests.	REGEXMATCH(myField,”^[0-9]*$”) Returns "True" if myField is made up of numeric characters between 0 and 9.
equalInAllValues(iterative_field, value)	Returns "True" value if all occurrences of "iterative_field" match with "value".	equalInAllValues(customerNo, 136) Returns "True" if all occurrences of the customerNo iterative field has a 136 value.
equalInSomeValue(iterative_field, value)	Returns "True" value if any of the occurrences of "iterative_field" matches with "value".	equalInSomeValue(customerNo, 136) Returns "True" if one or more occurrences of customerNo iterative field has a 136 value.
FIRSTVALUE(list)	Returns the first value in a list.	FIRSTVA-

Descriptive Name	Description	Example
		LUE(Items.quantity) Returns 5, the first value of the iterative quantity [5, 3, 2, ...].
QUANTITY(list)	Returns the length of a list.	QUANTITY(Items.price) Returns 3, the iterative price being [100,50,30].
EXTRACT(text, position, amount_characters)	Returns a subset of text characters, given an initial position and length.	EXTRACT("The user was successfully removed", 13, 13) Returns the characters "succesfully".
NUMBER(alphanumeric_value)	Transforms an alphanumeric into a number.	NUMBER("1988") Returns number 1988.
ROUND(numeric_value, decimals)	Rounds a number to a specified number of decimal places. If the digit after the specified position is greater than or equal to 5, the digit is rounded up. Otherwise, it is rounded down. This happens regardless of the sign. It is recommended to use this function when using 4- and 5-place decimals.	ROUND(49.9999, 0) Returns number 50.

Functions are allowed to be nested in parameters.

3.6.5.1.2. Advanced Rules

Advanced rules can be invoked from processes, forms and scheduled tasks, to incorporate business logic.

In forms and processes they can be included using the rules wizard, whereas in scheduled tasks the reference to the rule is done manually.

Types

- Standard Rules
  These are rules developed in the Java programming language, for which there is a development and execution environment given by the adapter being use.

- Integration Rules
  These are rules that use of adapters that provide integration with external components to **Deyel**.

Integration Classification

- Relational Databases
  Access to relational databases is done through the Java JDBC protocol. The database adapters via JDBC identify the driver to use and establish the connection.
  The integration rules that use JDBC adapters allow the execution of SQL statements that interact with any database.

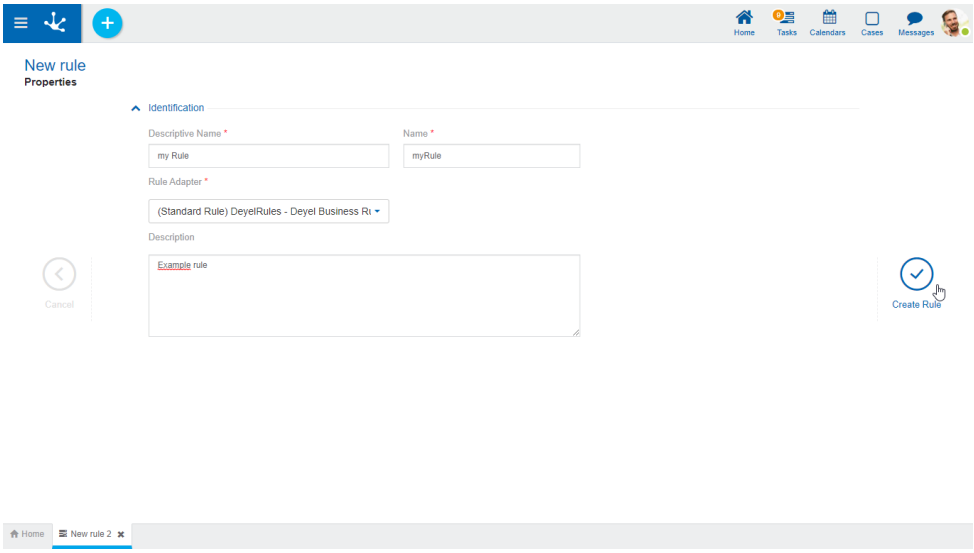
- Web Service
  The Web Service type adapters allow to define access to web services published by a specific provider.
  The integration rules that use Web Service adapters allow to use such web services.

3.6.5.1.2.1. Modeling Facilities

The advanced rules modeler is a tool that allows to easily design business rules, allowing the addition of the necessary logic to make them work when they are executed.

New Rule

The modeler user can define new rules, which after being published are available to be used in the portal.



Steps to Create a Rule

Step 1: Open the Rules Wizard

A new rule can be created from the icon corresponding to context menu.

- From the icon
- From the menu displayed with the "Rule" option

An existing rule is modified by selecting it from the grid with the "Open" button.

Step 2: Enter the Rule Properties

Properties are organized under the container Identification, which contains information common to all the rules.

Descriptive name
Name used by the modeler to refer to the rule, display it in the rules gallery and in the object tree. Supports multi-language.

Name
Used internally to refer to the rule within forms, embedded rules, processes or as parameters of other advanced rules. Unlike other objects, the rule name also acts as a unique identifier. Not multi-language.

Adapter
Used to determine the adapter to be used by the rule. A list of adapters available for selection is displayed.

Operation
Used to choose the operation to be executed and is only displayed when the chosen adapter type is web services.

Description
Text that defines the rule describing its functionality and optionally its content. Supports multi-language.

Step 3: Close the Rules Wizard

Cancel
Allows to return to the Deyel modeler to display the objects in the gallery.

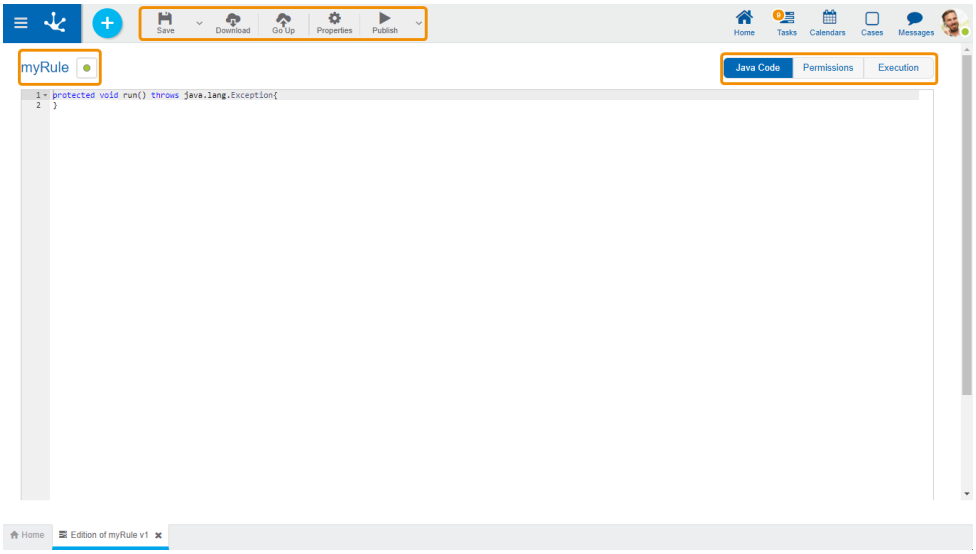
Create Rule
Moves to the rule workspace.

Workspace Sections

- Top Toolbar

- Rule Information
  - Name
  - State
  - Locking

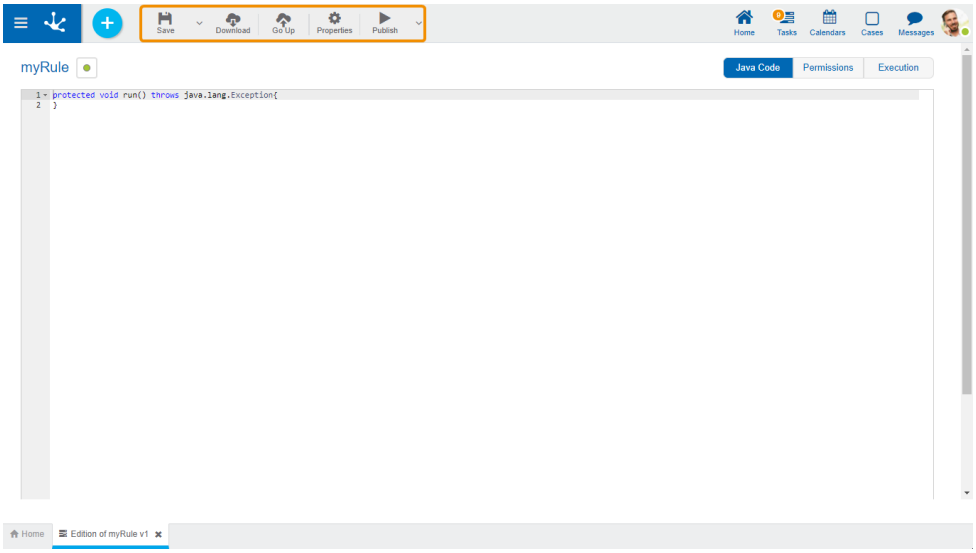
- Design Options



Top Toolbar

This toolbar contains icons and submenus from which you can perform operations on a rule. Depending on the state of the rule, some options may be disabled.

The bar is displayed on **Deyel's**.



Save

This icon allows to save the object in the repository of **Deyel**, leaving its state as "Draft" or "Modified". If certain conditions are met, the modeler user receives a message indicating that the operation was performed correctly, otherwise they receive an explanatory message.

Main Conditions

- The object must not be locked by another user,
- There must be an adapter that uses the rule.

To the right of the icon, the option to open the submenu.

Download

Allows to download the Java rule and the model and service classes of the objects related to the object wizard. A compressed file (.zip) containing the classes to edit later in an IDE for Java is generated. This icon is displayed for rules with all types of adapters with the exception of SOAP rules

Increase

Allows to upload a source file to update the Java rule. Such operation updates the window displaying the imported source code in the Java code editing area, but it is not saved until the user saves or publishes the rule. The code must be compilable, otherwise the load operation throws a compilation error. This icon is displayed for rules with all types of adapters with the exception of SOAP rules

Properties

Opens the rules properties.

Publish

By means of this icon the object goes to "Published" state, after verifying a set of additional conditions to those of the "Save" operation. The modeling user receives the corresponding message, indicating the result of the operation.
In the event of an exception due to a failed compilation of the rule code, the error is displayed in the design option "Compilation errors".

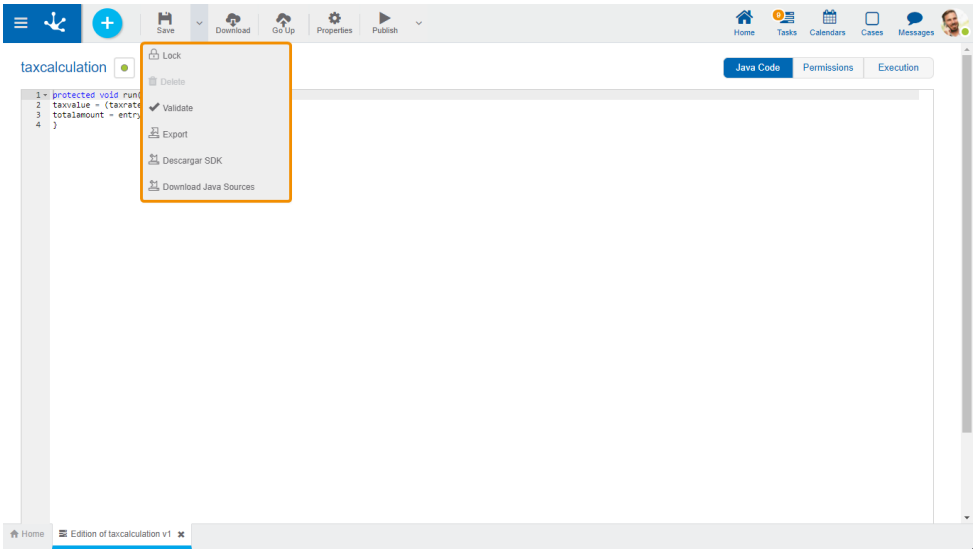
Main Conditions

- The packages corresponding to Java objects included in the rule code must be known to **Deyel**.
- The related objects in the rule must be previously published.

To the right of the icon, the option to open the submenu.

Save Submenu

This submenu is opened by pressing the icon immediately to the right of the icon corresponding to "Save".



/ Lock/Unlock

An advanced rule can only be modified by a modeler user.

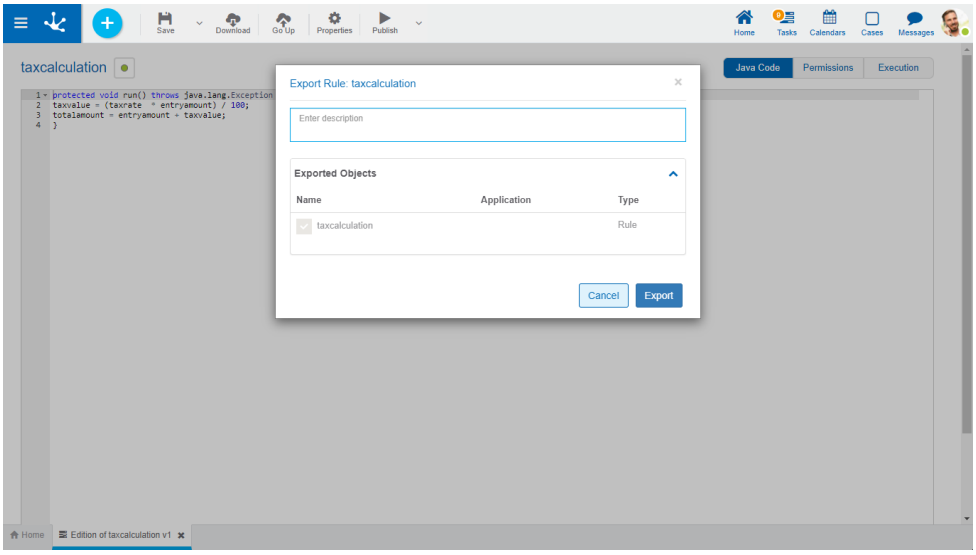
- It allows to block an advanced rule so that no one can modify it until the person who is using it unlocks it, that is, releases it.
- Allows to unlock an advanced rule so that another modeler user can modify it.

Validate

This icon validates if the rule is ready to be published, that is, validations performed at the moment of publication are applied and the result of the same is reported.

Export

This icon opens a window for the user to select and confirm the export of the object.



Description

In this property a text explaining the reason for the operation can be entered.
This text can be modified upon import and is displayed in the description column of the export record.

Exported Objects

By expanding the container, the object being exported is shown.

Press the "Cancel" button to undo export or press the "Export" button to finish.

Delete

Allows to delete the advanced rule only if it is in “Draft” state, closing the tab it is located in and deleting it from the modeler's grid.

Download SDK

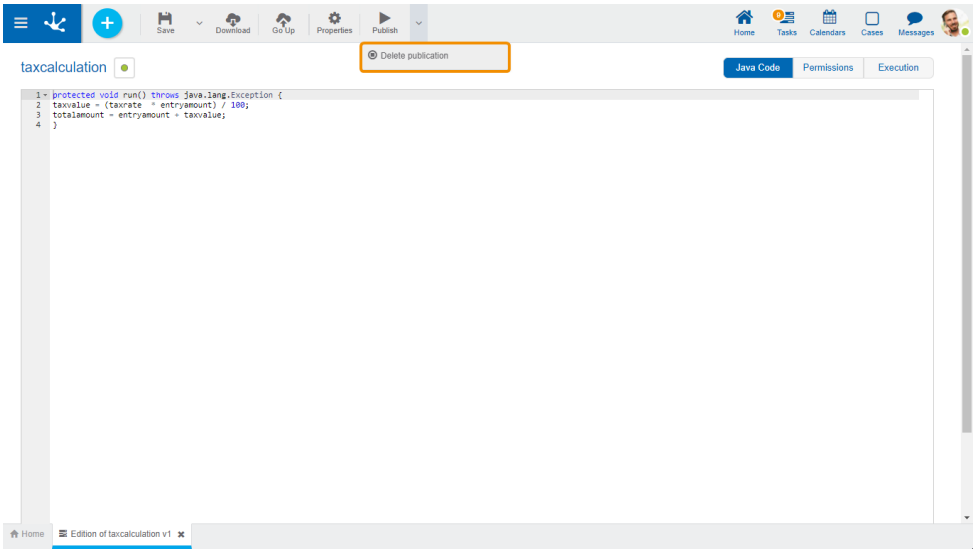
This icon allows to download the JAR file that represents Deyel’s SDK.

Download Java fonts

This icon allows to download the Java files that represent the object's model and service, so that it can be used in advanced rules.
Pressing the icon displays a message to confirm file download.

Publish Submenu

This submenu can be opened by pressing the icon immediately to the right of the icon corresponding to "Publish".



Delete Publication

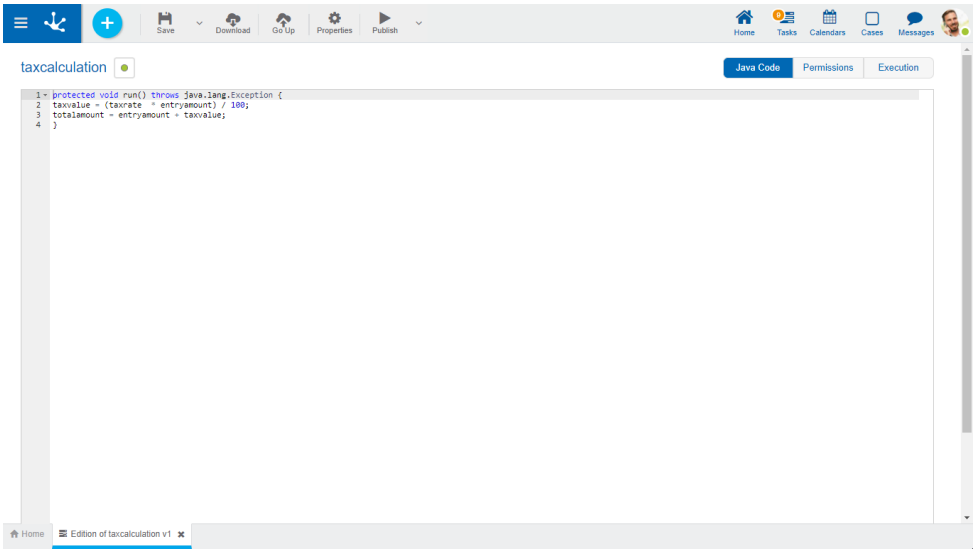
Allows to remove the advanced rule of use, returning it to the "Draft"state.

Modeling Area

The modeling area is the space where the rulers are designed and modeled. Within this area we can find the different design options which depend on the type of rule.

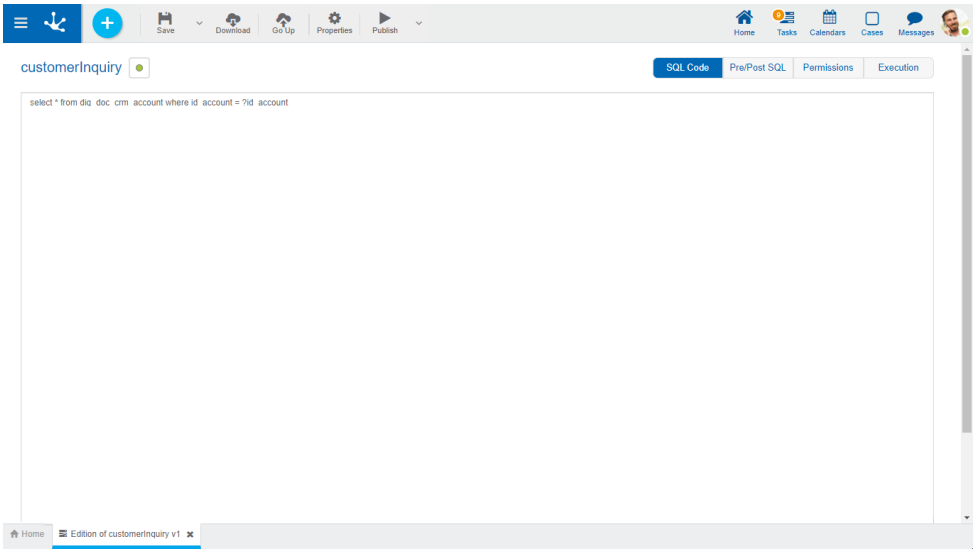
Standard Rule

Standard type rules use Java code to implement business logic. They use "Standard Rule" type adapters.



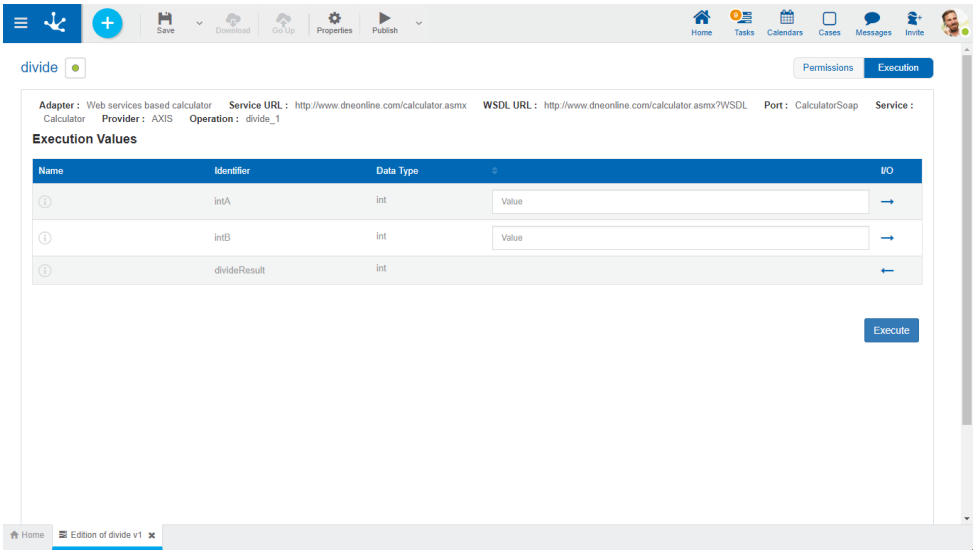
JDBC Rule

JDBC type rules are those used to perform operations related to databases, using a previously defined "JDBC" adapter.



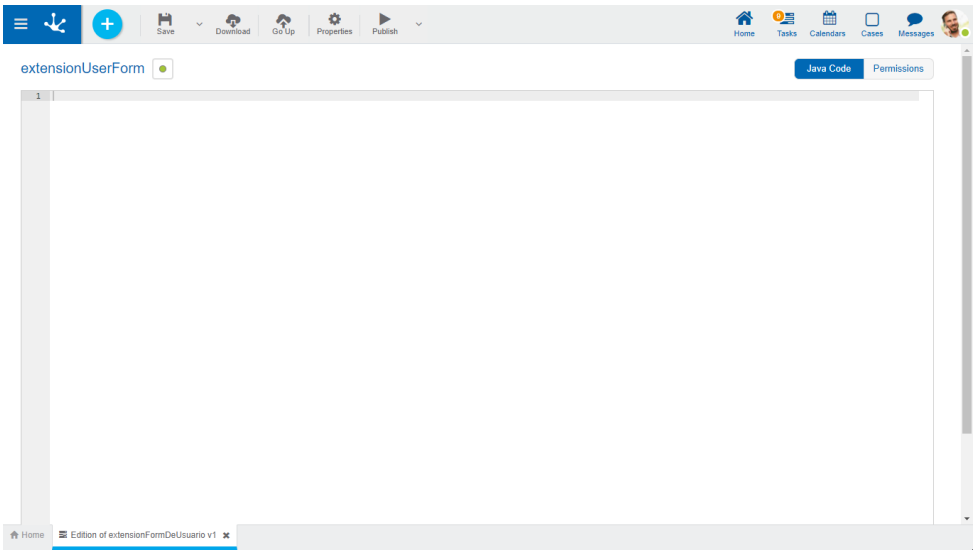
SOAP Rule

SOAP web service type rules are used to use external SOAP services in order to achieve integration with them. They use SOAP Rule" type adapters.



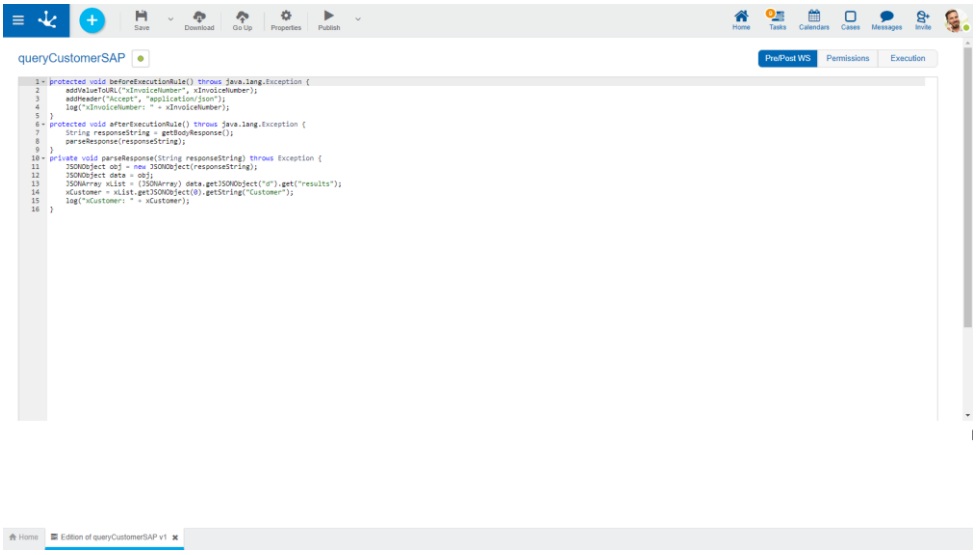
Forms Extension Rule

Forms extension type rules allow to use logic that cannot be done in the form modeler. They use the "Forms Extension" adapter.



Rest rule

REST type rules are used to invoke ApiRest services external to **Deyel**.
They use Java code in their Pre/Post section to execute business logic before and after the ApiRest service call. They use "Rest Rule" type adapters.



Design Options

Creating or modifying an advanced rule, allows to write the Java code or SQL code as applicable, to test the rule and review its compilation errors, viewing them in the same panel.

Depending on the type of advanced rule, one or more design options are displayed.

- Standard Rule
  The design options available are: "JAVA code", "Permissions", "Execution" and "Compilation errors".

- JDBC rule
  The design options available are: "SQL code", "Pre/Post SQL", "Permissions", "Execution" and "Compilation errors".

- SOAP rule
  The design options available are: "Permissions", "Execution" and "Compilation errors".

- Forms Extension Rule
  The design options available are: "JAVA code", "Permissions" and "Compilation errors".

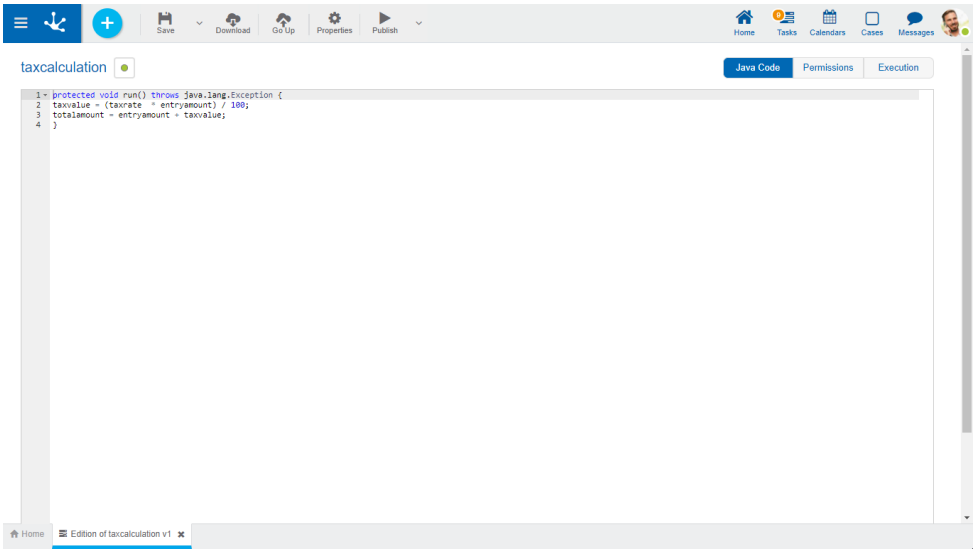
- Rest rule
  The design options available are: "Pre/Post WS", "Execution" and "Compilation errors".

> *In all cases, the "Compilation Errors" option is displayed only if compilation errors occur when executing or publishing a rule*

Java Code

The Java code area is displayed when the modeler is started for a standard type rule that uses a "Standard Rule"type adapter.

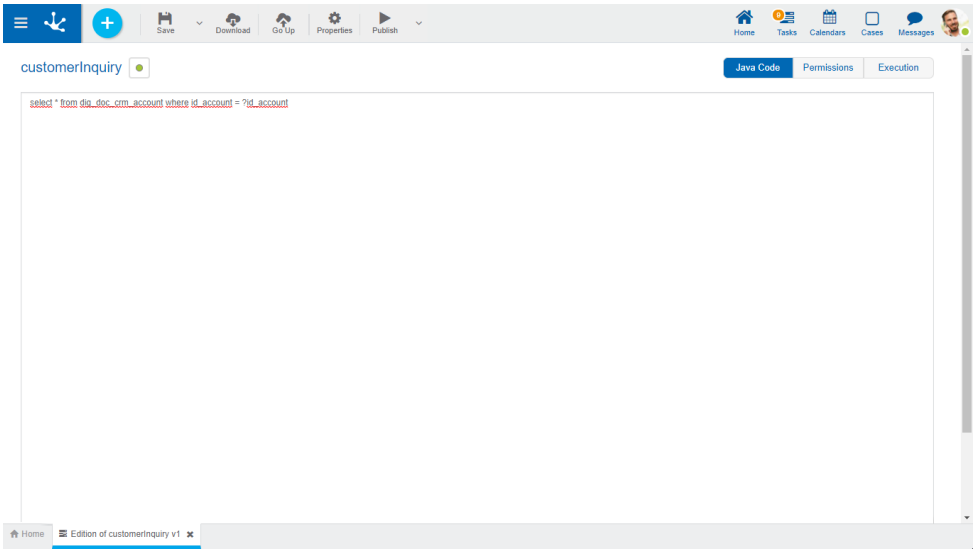
The Java code that executes the rule is written in this area.



Sql Code

The SQL code area is displayed when the modeler is started for a rule that uses a "JDBC" adapter.

The SQL show that the rule will execute is written in this area.



Permissions

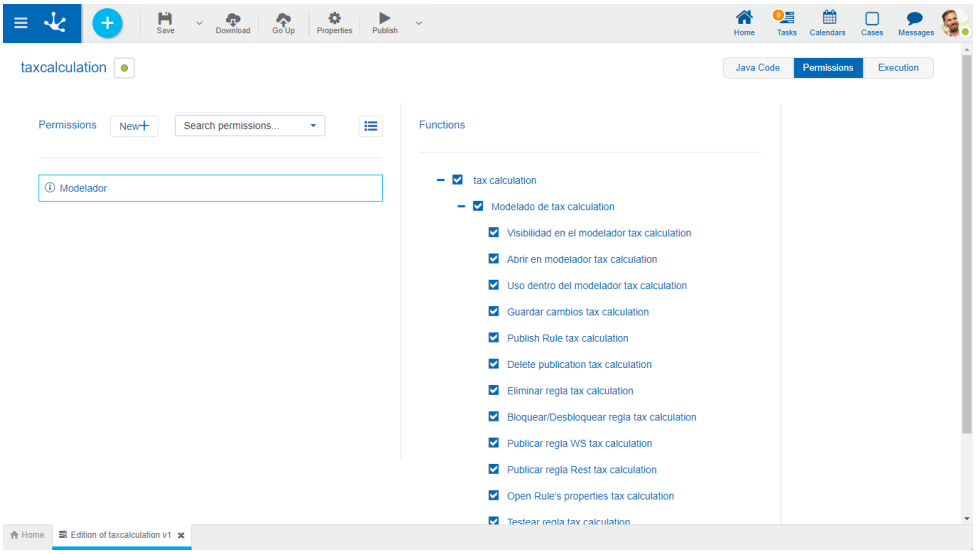
Allows to assign the security functions for use and modeling of the object to the existing permissions or to new permissions that the user creates, without having to go to the permission settings option from the menu.

Sections

- Permissions: Permissions to which object functions are assigned.
- Security Functions: Represents the total set of security functions, modeling and use of the object. Those that are marked are the ones included in the selected permission.

By default, all security functions for a new object are assigned to the permission Modeler.

Users who are assigned the permissions have access to the functions included in it.



New+ Opens a panel to create a new permission and once created, the security functions included in it must be selected .

Search permissions... Allows to select a permission from a list and enables the input of characters to filter the values in the list.

Opens the wizard to select a permission and once chosen, the necessary object security functions must be checked.

To unrelate a permission from the object, hover the cursor over the permission and press the icon ✖. If there are functions selected for that permission, they must be unchecked in order to delete it.

Security Functions to Model the Rule

- Visibility in the modeler: Allows to display the object in the Deyel modeler.
- Open in modeler: Allows to show the object from the Deyel modeler.
- Use within the modeler: Allows the use of the object from another modelable object.
- Save rule changes: Enables the save, download rule and upload rule operations.
- Publish rule: Enables the publish rule operation.
- Delete publication: Enables the operation of deleting the object publication leaving its state as "Draft".
- Delete rule: Enables the delete rule operation.
- Lock/unlock: Enables the lock/unlock operation, only the user who locks it can modify it.
- Publish WS rule: Enables the publish rule operation as SOAP Web Services.
- Publish Rest rule: Enables the operation to publish the API Rest type rule.
- Open rule's properties: Allows to consult the rule properties panel.
- Test rule: Enables the operation to test the rule execution from the "Execution" option.
- Export rule: Enables the operation to export the rule definition.

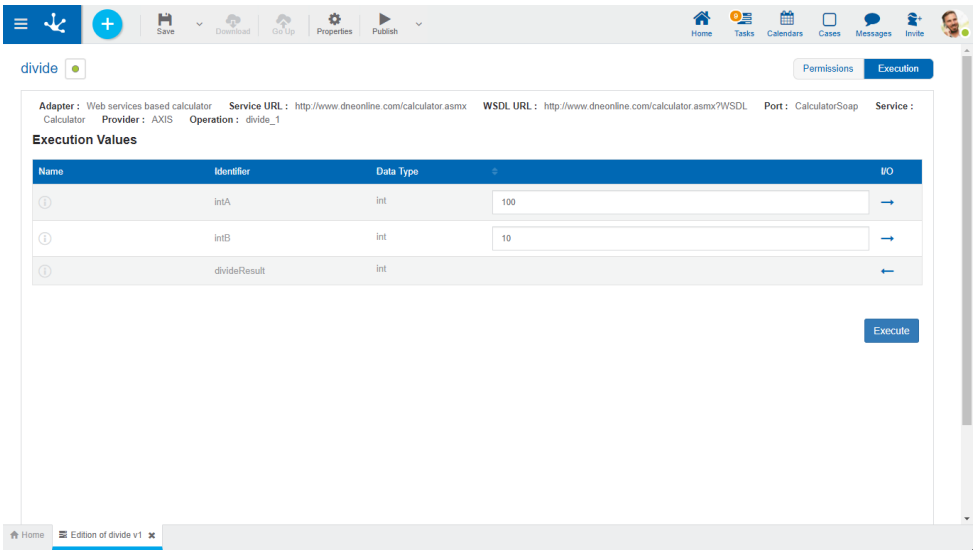
Security Functions to Use the Rule

- Execute rule via Rest: Allows the smart thing type user (API Rest type "Client") to execute the endpoints of the rules API Rest.

Execution

This area is displayed for all types of rules except for the forms functionality extension rules.
In the case of web services type rules, this area is displayed when starting the modeler.

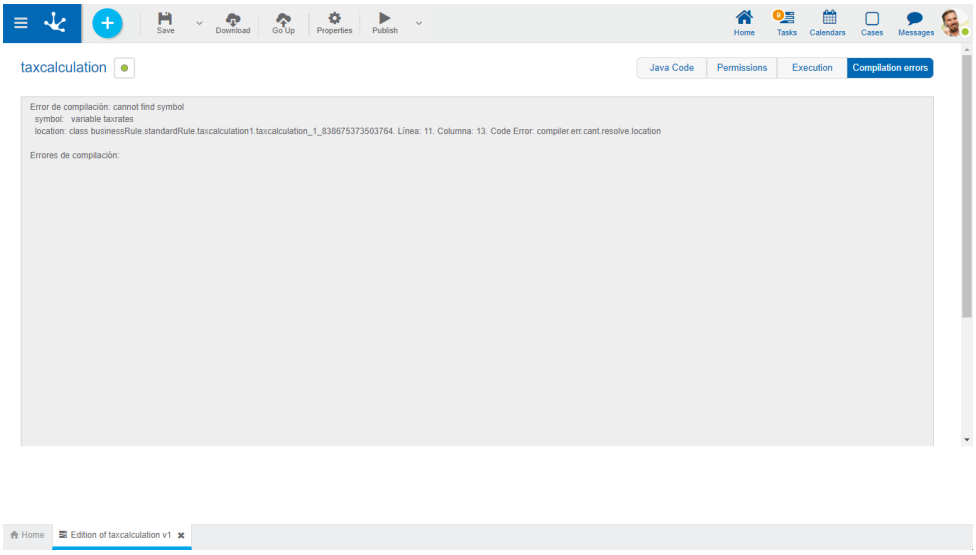
The parameters defined in the rules are displayed. For JDBC and web services SOAP type rules, the information of the adapter used is displayed.
Test values can be loaded on the defined input parameters to test the rule.



Compilation Errors

This area is displayed for all types of advanced rules, only if when executing a rule or publishing it, a compilation error occurs that prevents its correct operation.

The area has a space in which the corresponding compilation error is displayed.



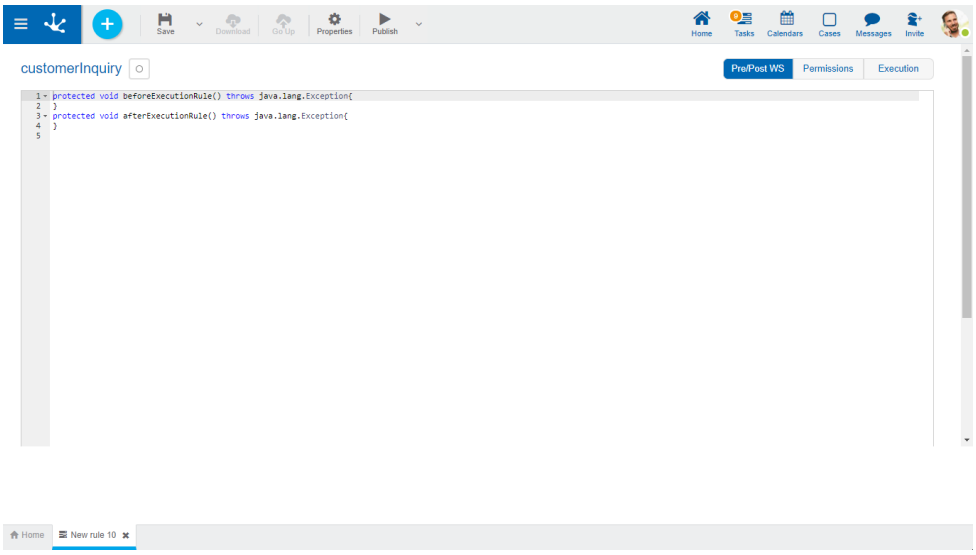
Pre/Post

When starting the modeler for a rule that uses the "Rest" type adapter, the area displayed is Pre/Post WS, and for a "JDBC" type rule is Pre/Post SQL.

Two required initial methods must be coded:

- **protected void beforeExecutionRule() throws java.lang.Exception{}**
  It allows adding logic before the ApiRest service call or the JDBC invocation.
  Example: An object can be read using the Deyel SDK and the result used as an input parameter in the JDBC invocation.

- **protected void afterExecutionRule() throws java.lang.Exception{}**
  It allows adding logic after the ApiRest service call or the JDBC invocation.
  Example: The result of the JDBC invocation can be received, processed, and saved in a form.



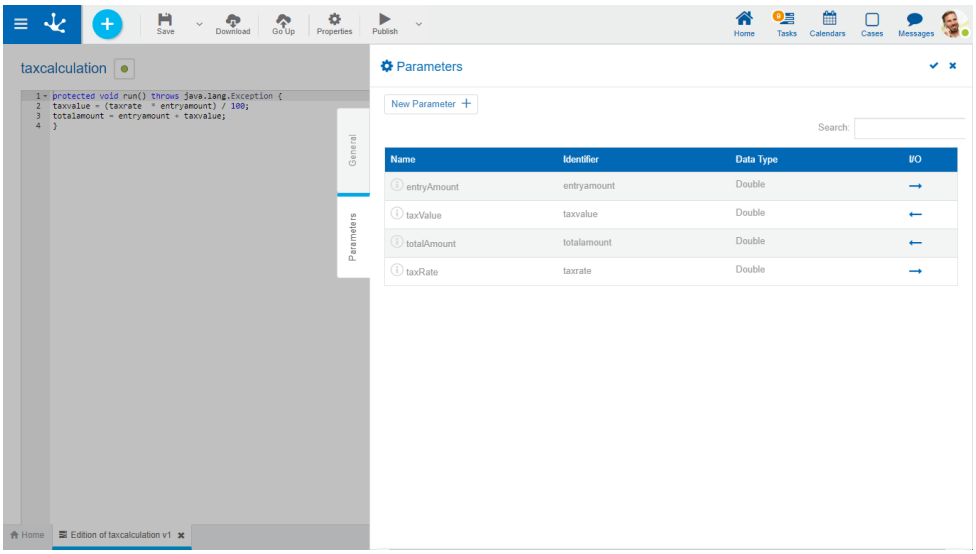
Additionally, in the “Rest” type rule, the following methods can be invoked:

Before Execution (beforeExecutionRule)	After Execution (afterExecutionRule)
setBody(String body): Allows to define the body parameter of the request.	String getBodyResponse(): Allows to get the body of the response.
addHeader(String key, String value): Allows to add a heading to the request.	Integer getHTTPStatus():Allows to get the HTTP state of the response.
addQueryParam(String key, String value): Allows to add a parameter query to the request.	List getHeadersResponse (); It allows to obtain a list of the headers of the HTTP request executed. This list is made up of objects of org.apache.http. header type.
addValueToURL(String key, String value): Replaces the text indicated in the first parameter with the value of the second parameter in the url of the service to call.	

3.6.5.1.2.2. Rule Properties

The advanced rule properties can be entered both when they are created while using the wizard, as well as when modifying an existing one.

To enter the rule properties panel, use the icon in the top toolbar.



Tabs

Entering the properties panel, one, two or three tabs are displayed, depending on the type of rule.

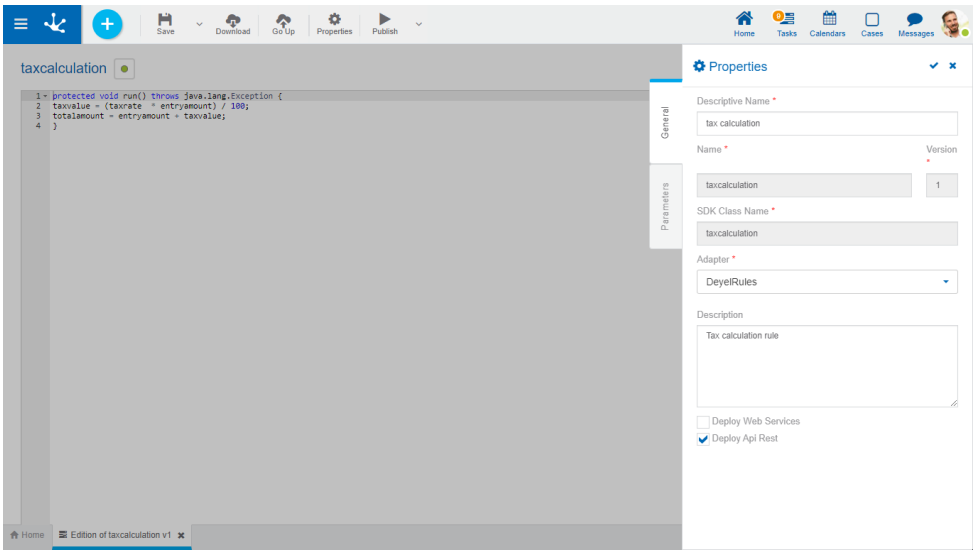
- General
  It is available for all types of rules.

- Parameters
  It is available for all rules except those of the form extension type.

- Advanced
  It is available for all rules except for standard rules, SOAP rules and form extension type rules.

General

The properties panel is displayed on the right side of the advanced rules modeler, where the first tab corresponds to general information.



> *An asterisk "*" on the label indicates that the property is required.*

Properties

Descriptive Name

Name used by the modeler to reference the rule, display it in the rules gallery and in the object tree. Supports multi-language.

Name

Used internally to reference the rule within forms, embedded rules, processes or as parameters of other advanced rules. Unlike other objects, the rule name also acts as a unique identifier to identify a rule. Not multi-language. Once saved, the rule cannot be changed.

Version

Used internally with the rule name to reference the rule within forms, embedded rules, processes or as parameters of other advanced rules. Once saved, the rule cannot be changed.

SDK Class Name
Name that represents the object in the SDK service and model classes.

Adapter
Used to determine the adapter to be used by the rule. Only it is displayed on this screen.

Operation
Used to choose the operation to be executed. This property is only displayed when the chosen adapter type is "Web Services". It is displayed in this section, it is necessary to go to the "parameters" tab in order to change it.

Description
Text that defines the rule describing its functionality and optionally its content. Supports multi-language.

Deploy Web Services
Used to publish the rule as a SOAP web service to be consumed from outside **Deyel**. This property is only displayed when the chosen adapter type is either "Standard Rule" or "JDBC".

Deploy Rest API
Used to publish the rule as a Rest API to be consumed from outside **Deyel**. This property is only displayed when the chosen adapter type is "Standard Rule".

Specific Properties

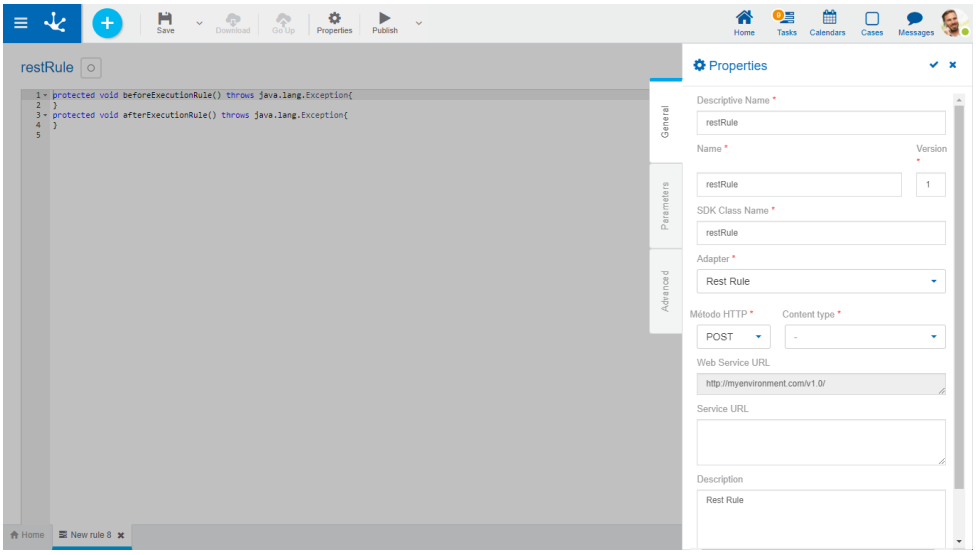
Rules with a "Rest" type adapter add specific properties for this type of execution.

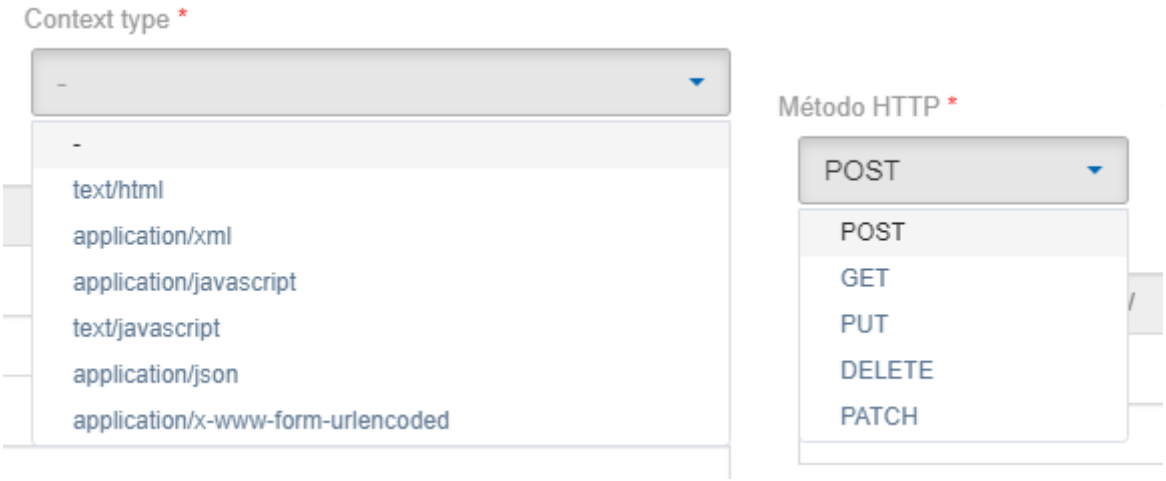
HTTP Method
Configures the HTTP method to use.

Context Type
Configures the content type of the request.

Web Services URL
Url entered in the adapter, displayed in a reading mode to facilitate the correct writing of the final url.

URL Service
Url of the service to consume, the final url is the concatenation of the Web Services URL and the URL Service. The addValueToURL(String key, String value) method can be used within the Pre/Post WS section to replace a part of the url (determined by the key) with a new value (determined by the value).





Actions

The icon ✔ is used to confirm the modifications made in the properties panel.

The icon ✖ is used to close the properties panel, if it was not previously saved, changes are discarded.

Parameters

The second tab of the side panel corresponds to the rules parameter. In this tab, the input and output parameters are defined, which are the communication channel between rules and the processes that use them.

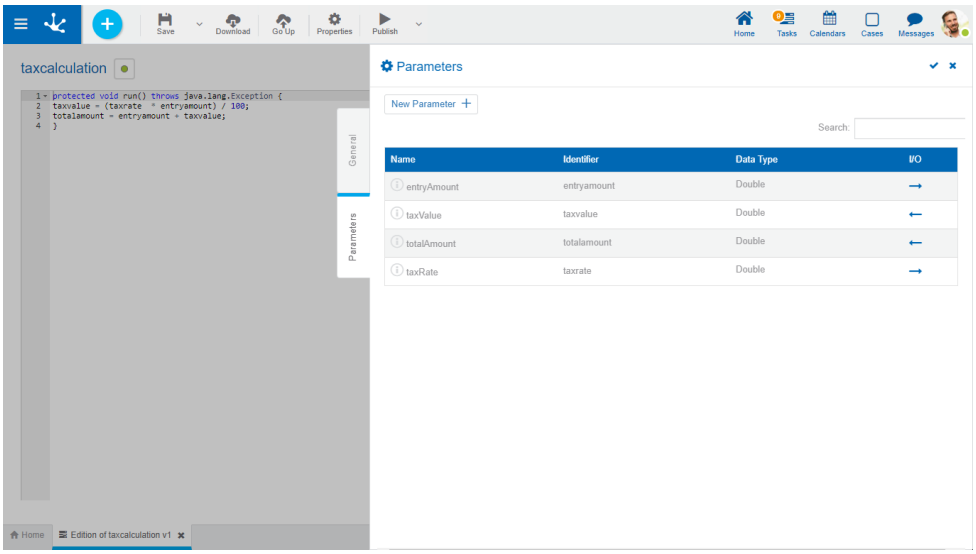
Parameters can be configured in any advanced rule except those of forms extension type.

Parameter Grid

The grid columns identify the properties of each parameter.

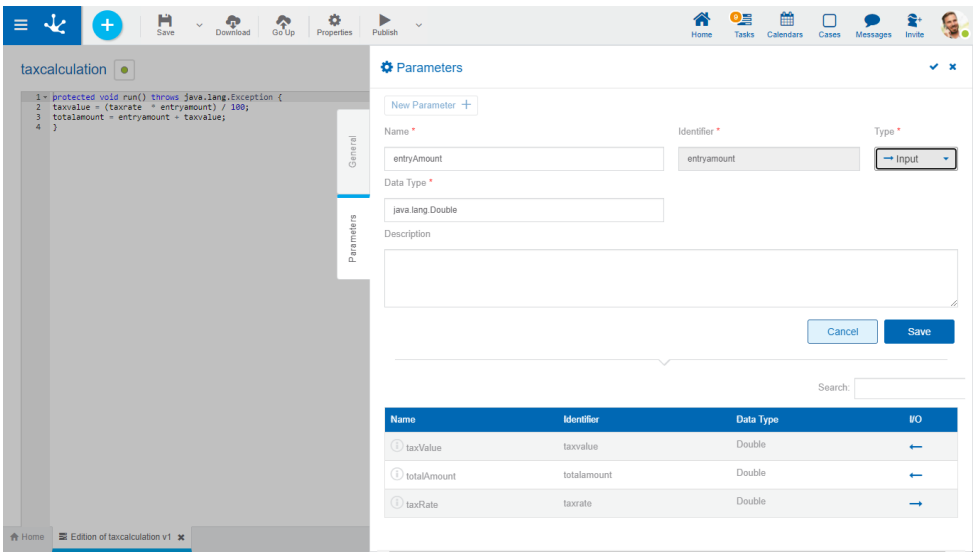
By placing the mouse over the grid, the following operations are enabled:

Allows modifying an existing parameter.
Allows deleting a parameter.



Allows Creating a Parameter

New Parameter + Allows adding a new parameter to the grid, completing all its values.



.

Properties

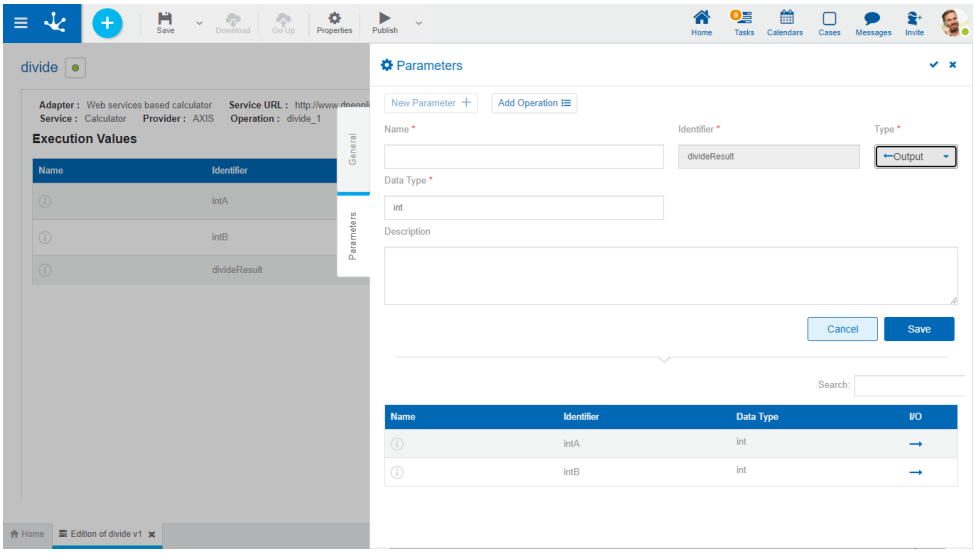
Name
Name the modeler uses to refer to the parameter.

Identifier
Uniquely identifies the parameter. It is used within Java programming code.

Type
Specifies if the parameter is input → or output ←.

Data Type
It can be any type of Java object allowed in **Deyel**, either a primitive (integer) or an array.
The type wizard, while typing, gives the most common data type values to use.
By placing the mouse over the data type in the grid, the complete data type can be displayed, to help.

Description
Describes the parameter use. Its writing when modeling the parameters is optional.
If it was entered, the colored icon is displayed in the grid, in the Name column ⓘ, otherwise the icon is colorless ⓘ.
Placing the cursor over the icon shows the help.



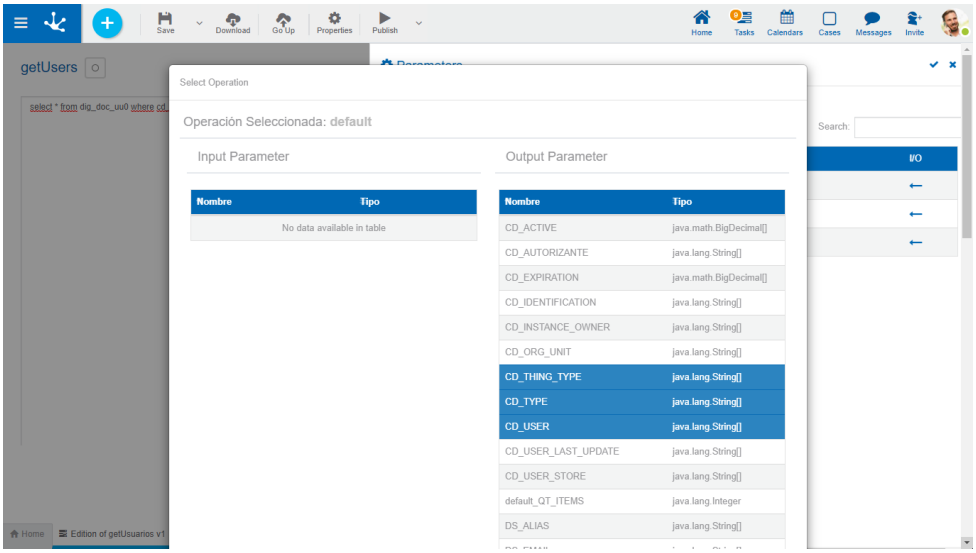
An asterisk "" on the label indicates that the property is required.*

Wizard to Create Parameters in JDBC Rules

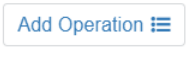
New Parameter + Allows to invoke the wizard available for JDBC type rules.

This wizard allows to select the parameters defined in the SQL statement as input parameters and the columns resulting from the query execution as output parameters.
In this way, the probability of errors is minimized by not having to manually transcribe the names of columns and parameters.

For the wizard to return results, it is necessary to define the SQL statement to execute before invoking it.



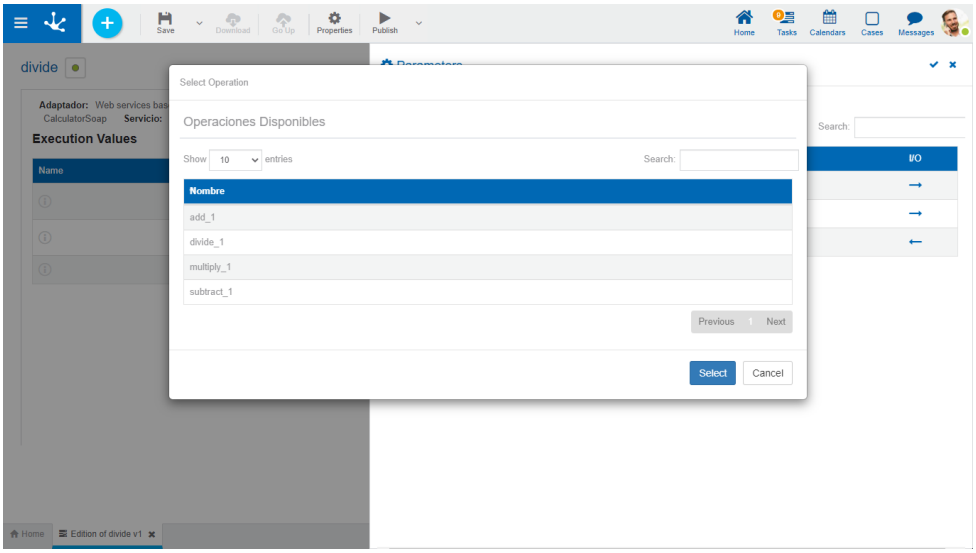
Wizard to Add Operations in SOAP Rules



Allows to invoke the wizard available for SOAP type rules.

The wizard analyzes the adapter used and shows the operations available for such object, along with the input and output parameters for each one.

It allows to select the operation that is executed by default and its parameters, to transfer them to the rule interface. In this way, the probability of errors is minimized by not having to manually transcribe the names of operations and parameters.



Advanced

The third tab of the side panel corresponds to the objects wizard. When entering the modeler, it is displayed for all rules except for standard rules, SOAP rules and form extension rules.

This tab lists the objects to be used in the rule.

Creating a relation using the object wizard allows to:

- Work with the model and service classes of the related object from the rule modeler without having to depend on a Java IDE.
- Always keep the Java imports statements of related objects up to date in the rule, regardless of whether the object has been modified after having created the relation.

If significant changes are made to the related objects used in the rules, when they are published:

- Java imports statements of the rules that use them are updated. Rules remain in the same state they were in, with updated modification date.
- Verify that the rule using them is not in a modified state. If modified, it will be requested to be published or unpublished.

Related Objects Grid

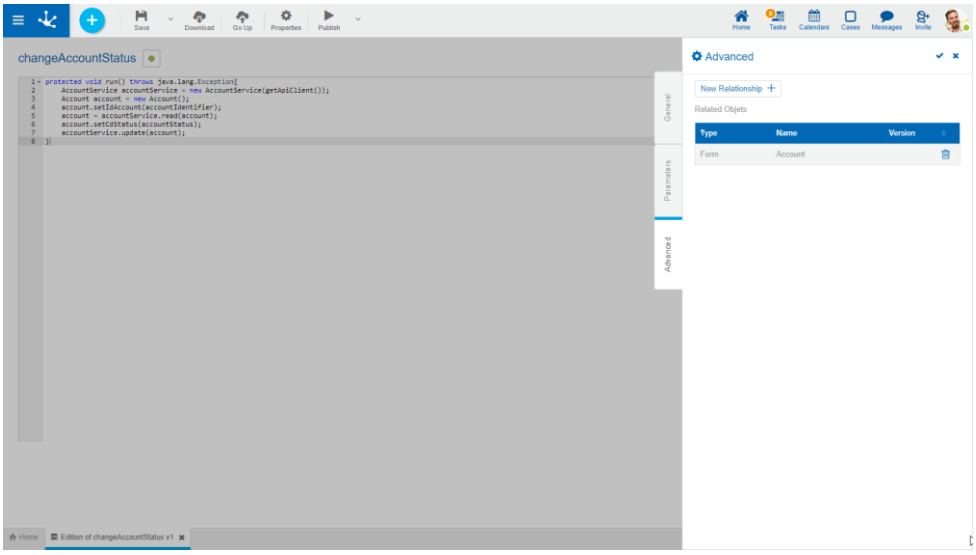
The grid columns identify the related object.

Type
Indicates the object type. It can be a form, process or rule.

Name
It is used at the modeling level to refer to the object.

Version
Indicates the object version for processes and rules.

By hovering the mouse over each related object in the grid, the delete operation is enabled.



New Relationship + Opens a panel with the forms, processes and rules that the user can select according to their security permissions. Once selected, it is displayed in the grid of related objects.

3.6.5.1.2.3. Deyel SDK for Java

By using the Deyel SDK (Software Development Kit) **Deyel** incorporates the ease of developing advanced and integration rules that interact with the platform objects and developed applications, in an easy and intuitive way.

These rules can be developed in the advanced rules modeler in the case of those that require low code or when they are more complex in a Java IDE, inheriting all its advantages.

Tools to Develop Advanced Rules with SDK

Deyel SDK allows to use **Deyel** objects in advanced rules, by using one class with the model and another one with the object's service.

Model Class

The model class corresponds to the properties that each object has.

For example, the "Account (CRM_ACCOUNT)" form model class of the CRM application contains the Identifier, Company, Opening, Industry, Source and Status fields, among others. It also contains the getter and setter methods for each of these fields.

The model includes:

- Object builder.
- Set of getter and setter methods for each object attribute.
- Internal classes if necessary. For example, the representation of iteratives for forms.

Service Class

A service class contains the operations to manage the model. For example, the service class of a form allows to perform the create, update, delete, query, and find operations.

The service class includes:

- CRUD (Create, Read, Update, Delete) of the object.
- Searches
- Specific methods to work on the model.

Each object service class has a common method available to initialize all objects.

Operation	Description	Parameters
getAPIClient()	Initializes the service to be able to use it.	

Interaction with Application Objects

In order to interact with **Deyel** objects from a Java IDE, the model and service classes corresponding to each object must be downloaded.

These classes are included when downloading the rule that interacts with such **Deyel** objects, in the "Download" option available on the top toolbar of the rule modeler.

By using the model and service classes, Deyel SDK allows to use the following objects:

- Forms
- Cases
- Rules
- Users
- Organizational Units
- Calendars
- Email Sending

Forms

The model class contains the properties with its getter and setter methods, while the service class contains the operations to be performed with the model.

For example, if the "Account (CRM_ACCOUNT)" form from the CRM application is used, when downloading the form Java sources, an account.java file is obtained, representing the model and another accountService.java file with the available operations of the service.

Model Class Content

A form model contains:

- Model builder.
- Set of getter and setter methods of its fields.
- Inner classes that represent the “Multiple occurrences” containers model, if any.

Data Type

Model properties for a form can have the following data types.

Equivalence between form field data types and Java model is shown.

Control	Data Type	Java
Text	Alphanumeric (length)	String
	Alphanumeric Uppercase (length)	String
	Large Alphanumeric	String
	Enriched Text	String
Integer	Number	Integer
	Large Integer	Long
	Decimal	Double
Time	Time	java.sql.Time
	Local Time	java.sql.Time
Date	Date	java.sql.Date
	Date and Time	java.sql.Timestamp
	Local Date	java.sql.Timestamp
	Date and Local Time	java.sql.Timestamp
Image	Image in Folder	String
	Image in Database	String

Control	Data Type	Java
File	File in Database	String
Check	Boolean	Boolean

Service Class Content

The service allows to perform the following CRUD (Create, Read, Update, Delete) and search (search) operations, as well as containing specific methods, for example to retrieve files from the form.

Operation	Description	Parameters
create(entity)	Creates a form instance.	**entity**: Form model class
read(entity)	Reads a form instance.	**entity**: Form model class
update(entity)	Updates a form instance.	**entity**: Form model class
delete(entity)	Deletes a form instance.	**entity**: Form model class
search(searchCriteria)	Allows to retrieve the form instances that meet the search criteria.	**searchCriteria**: Class that allows defining a set of criteria and search configuration parameters.

Examples of Use

The examples use the "Account (CRM_ACCOUNT)" form of the CRM application and each example contains the use of "Account" model class and "AccountService" service class.

1. Create service

This service is created only once in the rule and is reused in different operations.

```
AccountService accountService = new Ac-
countService(getApiClient());
```

2. Create a form instance

An instance of “Account” model class is created, values are assigned to its properties. Using the “AccountService” service class, the new account is saved with the assigned values through the create (account) method.

```
Account account = new Account() ;
ac-
count.setNuIdentifNumber("010000123");
account.setDsIndustry("1");// 1 = Agri-
culture
account.setDsSource("7");// 7 = Social
Networking
account.setDsCompany("Trees Company");
account.setDtOpening((Date) new ja-
va.util.Date());
account.setFlLogo(new Fi-
le("/photo.png"));
account.setCdStatus("1");// 1 = Active
Integer accountNumber = accountServi-
ce.create(account);
log("Account created with account num-
ber:"+ accountNumber);
```

3. Read a form instance

To obtain data from an existing account, an "Account" model class instance is created, an identifier chosen through the corresponding setter method is indicated and such instance is read using the read(account) method of the “AccountService” service class. The instance contains all its properties which can be queried using the corresponding getter methods. The example reads the dsCompany, dtOpening, dtStore, and dtLastUpdate properties.

```
Account account = new Account();
Integer readAccountNumber = 50000;
ac-
count.setIdAccount(readAccountNumber);
account = accountService.read(account)
;
```

```
// Once read, continue working
String dscompany = ac-
count.getDsCompany();
Date dtopening = ac-
count.getDtOpening();

// Audit data
Timestamp dtStore = ac-
count.getDtStore();
Timestamp dtLastUpdate = ac-
count.getDtLastUpdate();
```

4. Modify a form instance

To modify data from an existing account, an "Account" model class instance is created, an identifier chosen through the corresponding setter method is indicated and such instance is read using the read(account) method of the “AccountService” service class. The dsCompany property is then modified using the corresponding setter method and the update(account) method of the service is invoked to update the account.

```
Account account = new Account();
account.setIdAccount(50000);
account = accountService.read(account);

// Once read, update
account.setDsCompany("Trees Company");
accountService.update(account);
```

5. Delete a form instance

To delete an existing account, an instance of the "Account" model class is created, indicating its identifier by means of the corresponding setter method. The delete(account) method of the service is then invoked to delete the instance.

```
Account account = new Account();
account.setIdAccount(50000);
accountService.delete(account);
```

6. Use of file and image type fields

To retrieve a file related to a form instance, use the corresponding get+<fieldname>+(entity) method of the "AccountService" service class.
In this example, the getFlLogo(account) method of the "AccountService" service class is invoked, which allows to retrieve the file corresponding to a logo associated with a File object.

```
Account account = new Account();
account.setIdAccount(50000);

// Account reading
account = accountService.read(account);

// Logo image reading
File logo = accoountServi-
ce.getFlLogo(account);

// Assign a new logo to the account
account.setFlLogo(new File("Path"));
accountService.update(account);
```

7. Use of internal objects

Internal objects represent the set of fields modeled in iterative containers.

In this example, two new “Account.PhoneLine” internal objects are instantiated and values are assigned to their nrPhone and tpPhone properties.
A new list of “Account.PhoneLine” objects can be saved with the setPhoneLine(lsPhones) method or by adding two instances of the “Account.PhoneLine” object to the current list.

```
Account.PhoneLine phones= new Ac-
count.PhoneLine();
phones.setNrPhone("01149834589");
phones.setTpPhone("Line phone");

Account.PhoneLine phones= new Ac-
count.PhoneLine();
phones1.setNrPhone("0115123987");
phones1.setNrPhone("Cell phone");

List<Account.PhoneLine> lsPhones = new
```

```
ArrayList();
lsPhones.add(phones);
lsPhones.add(phones1);
account.setPhoneLine(lsPhones);

// Another option is
account.getPhoneLine().add(phones);
account.getPhoneLine().add(phones1);
```

Searchs

Searches on form instances can be performed using the following objects.

Criteria- represents search criteria on form data. It is made up of elements that can be fields and values, connected by operators.

Operator	Description
eq	Equal
neq	Different
gt	Greater
gte	Greater equal
lt	Less
lte	Less equal
between	Between
betweene	Between and admits equals
nbetween	Out of range
like	Contains
nlike	It does not contain
startsWith	Starts with

Operator	Description
nstartsWith	It does not start with
in	Included
nin	Not included

SearchCriteria: object that groups search criteria (Criteria object), it also allows parameterizing the order of the result, the size of the reading page and the number of pages to be retrieved. Reading by pages is used since the data volume on a form can be very large.
.
Form Service: service class of the form on which the search is being carried out. Contains the search(searchCriteria) operation that must receive the SearchCriteria object as a parameter and returns the SearchResult object.
Example:
SearchResult searchResult = accountService.search(searchCriteria);

SearchResult: object that contains the search results. Contains a list of form instances, which are instances of the corresponding model, for example "Account". This object allows to know the total number of reading pages resulting from the search, the size of each page and the number of pages to retrieve.

Examples

1. Searchs

This example retrieves a list of accounts with active state, where such state corresponds to code "1". A quantity of 15 lines per reading page and the number of pages to retrieve are defined. The results are ordered by company name in ascending order.

```
SearchCriteria searchCriteria = new
SearchCriteria();
 Criteria criteria1 =
          Criteria.eq("CdStatus","1");
 searchCriteria.addCriteria(criteria1);
 searchCriteria.setPageSize(15);
 searchCriteria.setPage(1);
 searchCrite-
ria.addOrderAsc("DsCompany");


SearchResult searchResult =
             accountServi-
```

```
ce.search(searchCriteria);
 List<Account> instancesResult = sear-
chResult.getResult();
```

2. Constructor types for search criteria

This example creates a Criteria object and defines different search conditions using the equal, different, greater, greater equal, less, less equal, between, among, and allows equals, out of range, and contains operators.

```
// Equal cdStatus= 1 corresponds to an
active account
 Criteria criteria = Crite-
ria.eq("CdStatus", "1");

// Different cdStatus= 1 corresponds to
an active account
// Criteria criteria =  Criteria.neq
("CdStatus","1");

// Greater
// Criteria criteria = Criteria.gt
("qtScore", 20) ;

// Greater equal
// Criteria criteria = Criteria.gte
("qtScore", 20) ;

// Less
// Criteria criteria = Criteria.It
("qtStore", 50) ;

// Less equal
// Criteria criteria = Criteria.Ite
("qtStore", 50) ;

// Between
// Criteria criteria = Criteria.between
("dtOpening",
// Date.valueOf("01/01/1990"), Da-
te.valueOf ("01/01/2020"));
```

```
// Between and admits equals
// Criteria criteria = Crite-
ria.between("dtOpening",
// Date.valueOf("01/01/1990"), Da-
te.valueOf("01/01/2020"));

// Out of range
// Format YYYY-MM-DD is used
// SimpleDateFormat format =
//      new SimpleDateFormat("yyyy-MM-
dd");
// Date dateFrom =
//      new Date (format.parse( "1990-
01-01" ).getTime());
// Date dateTo =
//      new Date(format.parse( "2020-
01-01" ).getTime());
//
// Criteria criteria =
//        Criteria.nbetween ("dtOpe-
ning", dateFrom, dateTo );

// Contains In an internal object
// Criteria criteria = Crite-
ria.like("eMailLine/dsEmail",
//         "yahoo.com.ar");
```

> ***Deyel*** *uses the data type java.sql.Date for "date" type fields, so the transformation from java.util.Date to java.sql.Date must be done, when it is required to create an object with this type of data.*

```
// Contains
// Criteria criteria = Crite-
ria.like("dsCompany","Inc.");

// It does not contain
// Criteria criteria = Crite-
ria.nlike("dsCompany","Inc.");
```

```
// Starts with
// Criteria criteria = Crite-
ria.startsWith("dsCompany","The");

// It does not start with
// Criteria criteria = Crite-
ria.nstartsWith("dsCompany","The");

// Included dsIndustry = 6 is Communi-
cations, 8 is Education
// Criteria criteria =
// Criteria.in("dsIndustry", Arra-
ys.asList("6", "8" ));

// Not Included dsIndustry 6 is Commu-
nications, 8 is Education
// Criteria criteria =
// Criteria.nin("dsIndustry", Arra-
ys.asList("6", "8" ));
```

3. Operations on the results object

The SearchResult object is an iterative object and as such can be scrolled through in different ways. The example uses a FOR statement to iterate over each of the instances that meet the search conditions of the dsIndustry and qtScore properties, retrieving the idAccount and dsCompany properties.

```
SearchCriteria searchCriteria = new
SearchCriteria();
 searchCriteria.setPageSize(15);
 searchCrite-
ria.addOrderDesc("idAccount");

 // Accounts with industry equal to
communications
 searchCriteria.addCriteria
 Criteria.eq("dsIndustry", "6"));

// Accounts with a score greater than
20
 searchCriteria.addCriteria
 (Criteria.gt("qtScore", 20)  );
```

```
// Scrolls through all the accounts in
the search result
 String stringData1 = "";
 for (Account account: searchResult) {
 stringData1 += " - " + ac-
count.getIdAccount()
                  + " - " + ac-
count.getDsCompany();
 }
// If I only need the first results
sheet
 SearchResult<Account> resultFirstPage
=
 accountService.search(searchCriteria);
 List<Account> firstList = resultFirst-
Page .getResult();

 String stringData = "";
 stringData += " Amount of data " +
firstList.size();
 stringData += " Pages " + resultFirst-
Page .getPages();
 stringData += " Actual Page " +
 resultFirstPage .getCurrentPage();
```

Navigation between Related Entities

In the CRM application, the “Contact (CRM_CONTACT)” form has a relation with the “Account (CRM_ACCOUNT)” form and the related account can be retrieved from a business rule, from a contact.

An instance of the “Contact” model class is created and the account identifier is retrieved using the corresponding getter method.
Next, an instance of the "Account" model class is created indicating the identifier retrieved from the contact in the setter method and the account instance is read using the read(myAccount) method of the "AccountService" service class.

```
Contact contact = new Contact() ;
Integer idAccount = con-
tact.getIdCompany();
// Retrieve the id of a contact account
```

```
// and I read the account with that id

Account myAccount = new Account();
myAccount.setIdAccount(idAccount);
myAccount = accountServi-
ce.read(myAccount);
```

Cases

The model class contains properties with their getter and setter methods, while the service class contains operations to be performed on the model.

For example, when downloading the Java sources of the process, if using the "Request Payment" process, a requestPayment.java file is generated with the model methods and also another requestPaymentService.java file with the available service operations.

Model Class Content

A case model contains:

- Model builder.
- Getter and setter methods of the case properties.
- Getter methods to get the related forms.
- Getter methods to get the activities in progress.
- Getter methods to get the process corresponding to the case.

Getter and setter methods of the case properties

The case model class contains a set of getter and setter methods for the case properties.

Operation	Description	Parameters
getCdCase()	Gets the identifier of the case.	
setCdCase(casenumber)	Allows to set the case identifier value.	String **casenumber**: Case number
getCdState()	Gets the state of the case. Possible values: **ACTIVE** - Active **CANCELLED** Cancelled: **ENDEDCASE** - Ended:	

Operation	Description	Parameters
getDsCase()	Gets the description of the case.	
getDtEnded()	Gets the end date of the case.	
getDtExpiration()	Gets the expiration date of the case.	
getDtInitiated()	Gets the start date of the case.	
getPriority()	Gets the priority of the case. Possible values: **1** - Urgent **2** - High **3** - Medium **4** - Low	
setLastPressedButton(button)	Allows to set the last button pressed.	String **button**: Name of the last button pressed.
getLastPressedButton()	Gets the last button pressed.	
get + <Form class name> + Entity()	Gets an instance of the form class related to the process.	
getLsVinculatedForms()	Gets a list of form instances related to the case.	
getVinculatedFormByIdEntity(idEntity)	Gets a form instance related to the case.	String **idEntity**: Identifier of the form instance
getProcess()	Gets the process corresponding to the case.	
getLsExecutedActivities()	Gets the executed activities of the case.	
getLsCurrentActivities()	Gets the activities in progress of the case.	

Getter method to get the related forms.

Given a case, the values of the form fields related to it can be used, in the same way as if the activities were executed manually.

The model class has a getter method for each form related to the process, with a “get + <Form class name> + Entity()” structure. Once an instance has been retrieved, it is used in the same way as for forms.

Example
If the "Request Payment" process related to the "Payment" and "Report" forms is modeled, the RequestPayment class has the following methods:

- getPaymentEntity(): Gets an instance of the Payment class that corresponds to the "Payment" form related to the case.
- getReportEntity(): Gets an instance of the Payment class that corresponds to the "Payment" form related to the case.

> *The case cannot have forms associated with the same class name, even if they belong to different applications.*

Getter methods to get the activities in progress

Given a case, the list of its activities in progress can be accessed, using the getLsCurrentActivities() case properties method of the model class, which returns a list of "ExecutedActivity" objects with their methods.

Operation	Description	Parameters
getCdActivity()	Gets the code of the activity.	
getDsNameActivity()	Gets the name of the activity.	
getCdState()	Gets the state of the activity. Possible values: **EXEC** - In execution **CANCELLEDACT** - Cancelled: **ENDEDACT** - Ended:	
getPriority()	Gets the activity priority. Possible values: **1** - Urgent **2** - High **3** - Medium **4** - Low	
getDtEnded()	Gets the end date of the activi-	

Operation	Description	Parameters
	ty.	
getDtExpiration()	Gets the expiration date of the activity.	
getDtInitiated()	Gets the start date of the activity.	
getCdUserExec()	Gets the code of the user that executed the activity.	
getCdUserInit()	Gets the code of the user that started the activity.	
getNuDurationSeconds()	Gets the duration in seconds of the activity execution.	
getTpActivity()	Gets the state of the activity. Possible values: **TP_GATEWAY** - Gateway **TP_STANDARD** - Standard **TP_ABSTRACT** - Abstract	
getTpParticipant()	Gets the type of participant for the activity. Possible values: **USER** - User **ORG_UNIT** - Organizational Unit **ROLE** - Role **THING** - Userthing	
getLsExecutedActions()	Gets a list of actions executed by the activity.	

Getter method to get the process corresponding to the case.

The process corresponding to a case can be obtained through the getProccess() method of the model class, which returns an instance of the "Process" class with its methods.

Operation	Description	Parameters
getCdProcess()	Gets the code of the process.	
getCdVersion()	Gets the version of the process.	
getIdApplication()	Gets the application identifier of the process.	
getCdFirstActivity()	Gets an Activity object with the first process activity,	
getDsComment()	Gets the comment of the process.	
getDsDescription()	Gets the description of the process.	
getDsName()	Gets the name of the process.	

Service Class Content

The service allows the following operations to be performed on the cases:

Operation	Description	Parameters
startCase(case) startCase(case, user)	Starts a case, executing the initial activity of the related process. Returns the code of the created case.	Case **case**: Model of the case to start. Related form fields must have values Case **case**: Case model with the values assigned to the related form fields String **user**:Code of the user that executes the activity. It must be specified when the participant responsible for the activity is a role
execute(case) execute(case, user)	Executes the current activity of a case.	Case **case**: Model of the case with assigned case number

Operation	Description	Parameters
	Returns the state of the updated case.	Case **case**: Model of the case with assigned case number String **user**:Code of the user that executes the activity. It must be specified when the participant responsible for the activity is a role
read(case)	Reads a case.	Case **case**: Model of the case with assigned case number

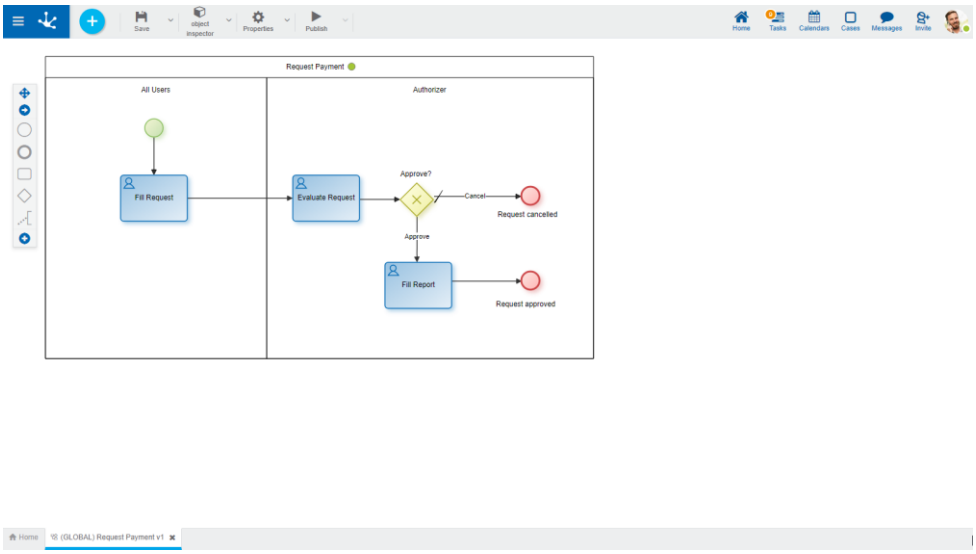
Examples of Use

The examples use the "Request Payment" process and each example contains the use of the "RequestPayment" model class and the "RequestPaymentService" service class.

In addition, "Request" and "Report" classes are used, which represent models of forms related to the "Request Payment" process.

The examples detail how to start a case, read it, execute its activities, and finish it.

Process Diagram



1. **Starting a case**

To start the case, the first activity "Fill Request" is executed. The case moves to the “Evaluate Request” activity under the responsibility of “Afarias” user, configured in the “Authorizer” lane

Executing the “Fill Request” activity performs the “Create” operation on the “Request” form with the values assigned to its fields.

An instance of the "RequestPayment" model class is created, in the "Request" form obtained through the getRequestEntity() method, values are assigned to its fields with the corresponding setter methods. The case is started using the "RequestPaymentService" service class with the startCa-se(myFirstPaymentCase) method.

```
RequestPayment myFirstPaymentCase = new
RequestPayment();
myFirstPaymentCa-
se.getRequestEntity().setAmount(new
Double(200));

SimpleDateFormat format = new SimpleDa-
teFormat("yyyy-MM-dd");
myFirstPaymentCa-
se.getRequestEntity().setDueDate(new
Date(format.parse("2020-06-
28").getTime()));
myFirstPaymentCa-
se.getRequestEntity().setMessage("This
is my first payment request .. i need
to buy more coffee !!");

RequestPaymentService myService = new
RequestPaymentService(getApiClient());
String cdCase = myServi-
ce.startCase(myFirstPaymentCase);
```

2. Reading the case

An instance of the “RequestPayment” model class is created, it is assigned a case number generated by the startCase(myFirstPaymentCase) method of the “RequestPaymentService” service class, in the pre-vious example.

The case is read with the read(myFirstPaymentCase) method of the "RequestPaymentSercice" service class and the case number.

```
RequestPayment myPaymentCase = new Re-
questPayment();
myPaymentCase.setCdCase(cdCase);
myPaymentCase = myServi-
ce.read(myFirstPaymentCase);
```

3. Activity execution defining modeled buttons

The case is in the “Evaluate Request” activity, which has two modeled buttons. The next activity to be executed depends on the button selected, in this example “Approve” is selected. If no button is specified, the modeled flow is followed as default.

The “Approve” button is assigned to the “Request Payment” model class using the corresponding setter method. The “Evaluate Request” activity is executed with execute(myPaymentCase) method of the “RequestPaymentService” service class.

As a result, the case is in the “Fill Report” activity and its state is “ACTIVE”,

```
myPaymentCa-
se.setLastPresedButton("Approve");
myService.execute(myPaymentCase);
```

4. Activity execution not defining modeled buttons

The case is in the “Fill Report” activity. This activity execution creates the “Report” form. This form has a field called "request" that is related to the "Request" entity.

The “request” field is completed by means of the corresponding setter method, with the identifier of the “Request” entity related to the case, which is obtained by means of the getRequestEntity() method.

The rest of the fields are completed and the “Fill Report” activity is executed with the execute(myPaymentCase) method of the “RequestPaymentService” service class.

The case is sent to the “Request approved” end event and its state is ended.

```
Integer requestId = myPaymentCa-
se.getRequestEntity().GetRequestId();

myPaymentCa-
se.getReportEntity().setRequest(request
Id.toString());
myPaymentCa-
```

```
se.getReportEntity().setPaymentMethod("
Credit");
myPaymentCa-
se.getReportEntity().setSummary("Fillin
g the report using the SDK.");
myService.execute(myPaymentCase);
```

Rules

The model class contains getter and setter methods for each of the parameters defined in an advanced rule, while the service class contains the operation to be performed with the model.

The model name as well as the service name are determined by the property SDK Class Name specified when modeling the advanced rule.

For example, if the "taxCalculation" advanced rule is used when downloading the Java sources of the rule, you get a taxCalculation.java file that represents the model and another taxCalculationService.java file with the available operation of the service.

Model Class Content

An advanced rule model contains:

- Model builder.
- General getter method.
- Set of getter and setter methods of its parameters. The output parameters only have getter methods.

General getter method

Operation	Description	Parameter
getCdRule()	Returns the identifier of the rule.	

Data Type

The parameters of an advanced rule have their equivalence with the data types of form fields.

Service Class Content

Service allows the following operations to be performed:

Operation	Description	Parameter
execute(rule)	Executes an advanced rule.	Rule **rule**: Model of the rule to be executed

Example of Use

The example invokes the "taxCalculation" advanced rule, using the model and service classes.

Values for "amount" and "taxRate" input parameters are defined in the model using its setter methods. Next, the execute(myRule) method of the service is executed and a VAT value is obtained from the getter method of the "taxValue" parameter.

```
// Calculation of 21% VAT on the Amount
Received
taxCalculationService myService =
new taxCalculationService (getApi-
Client());
taxCalculation myRule = new taxCalcula-
tion() ;
myRule.setAmount(new Double(100.12));
myRule.setTaxRate(new Double(21.00));

myRule = myService.execute(myRule);
// Gets the Amount of VAT Applied
Double doubleResult = myRu-
le.getTaxValue();
log("The result of the tax value is:"
 + doubleResult);
```

Users

The model class contains the properties with its getter and setter methods, while the service class contains the operations to be performed with the model.

Content of the Model Class

A user's model contains:

- Model builder.
- Set of getter and setter methods of its attributes.

Getter methods to get user attributes

The model class for a case contains a set of getter methods for the attributes of the case.

Operation	Description	Parameters
getFirstName()	Gets the user name.	
getLastName()	Gets the user last name.	
getEmail()	Gets the user email.	
getUserCode()	Gets the user code.	
getAlias()	Gets the user alias.	
getOrganizationalUnit()	Gets the user organizational unit.	
getAuthorizedUserCode()	Gets the authorizing user code.	
getCalendar()	Gets the user calendar.	
getJobs()	Gets the job positions.	
getDelegates()	Gets the user delegated users.	
getDelegatesOf()	Gets the users where the user is defined as a delegate.	
getActive()	Gets the user state. Possible values: **1** - Active **0**- Inactive	
getExpiration()	Gets the user expiration date.	
getUserDurationDays()	Gets the number of days the	

Operation	Description	Parameters
	password expires since the user was created. Possible values: **0** - Never **30** - 30 days **60** - 60 days **90** - 90 days	
getLicenses()	Returns a list of licensed products for the user.	
getPermissions()	Returns a list with the user permissions.	
getRoles()	Returns a list with the user roles.	
getIdentificationType()	Gets the identification type of the user personal data.	
getIdentificationNumber()	Gets the identification number of the user personal data.	
getPhone()	Gets the user phone number.	
getExtensionNumber()	Gets the user phone extension number.	
getNationality()	Gets the user nationality.	
getBirthday()	Gets the user birthdate.	
getCountry()	Gets the user country.	
getState()	Gets the user state	
getCity()	Gets the user city.	
getZipCode()	Gets the user zip code.	
getStreetName()	Gets the user street name.	
getStreetNumber()	Gets the user street number.	

Operation	Description	Parameters
getDepartament()	Gets the user department code.	
getLinkedinAccount()	Gets the linkedin user code.	
getTwitterAccount()	Gets the twitter user code.	
getFacebookAccount()	Gets the facebook user code.	
getYouTubeAccount()	Gets the youtube user code.	
getSkypeAccount()	Gets the skype user code.	
getObservation()	Gets the user observations.	

Service Class Content

The service allows the following operations to be performed:

Operation	Description	Parameters
read(user)	Reads an instance.	User **user**: User code to read
getUserImage(user)	Gets the user profile image.	User **user**: User code to read
getThumbnail(user)	Gets the thumbnail image of the user profile.	User **user**: User code to read
verifySecurity(user,function)	Returns true if the user has the security function specified as a parameter in his permissions, and false if otherwise.	User **user**: User code to read String **function**: Code of the specified safety function
search(searchCriteria)	Returns a list of instances of the model class with specified search criteria	SearchCriteria **searchCriteria**: Search criteria

Example of Use

The examples use the **Deyel** user object and each example contains the use of the "User" model class and the "UserService" service class.

1. **Service creation**

This service is created only once in the rule and is reused in different operations.

```
UserService userService = new UserSer-
vice(getApiClient());
```

2. **User reading**

In this example, a user with the userCode property with "JPAZ" value is read, using the read(user) method of the "UserService" service class. The value of the lastName property is obtained with the getter method of the "User" model class.

```
User user = new User();
user.setUserCode("AFARIAS");
user = userService.read(user);
```

3. **Get the user profile image**

In this example, a user with the userCode property with "AFARIAS" value is read and its profile image is obtained through the getUserImage(user) method of the "UserService" service class.

```
User user = new User();
user.setUserCode("AFARIAS");
userService.getUserImage(user);
```

4. **Get user profile image**

In this example, a user with the userCode property with "AFARIAS" value is read and its profile image is obtained through the getUserImage(user) method of the "UserService" service class.

```
User user = new User();
user.setUserCode("AFARIAS");
userService.getThumbnail(user);
```

5. **Security check**

In this example it is verified if a user with the userCode property with “AFARIAS' value has the “DUU0C001” security function defined. The verifySecurity((user, “DuU0C001”) method of the “UserService” service class is used with a user that has been read using the read(user) method and the specified security function, as parameters.

```
User user = new User();
user.setUserCode("AFARIAS");
user = userService.read(user);
boolean userHasAccess = userServi-
ce.verifySecurity(user,"DUU0C001");
```

6. Search

This example retrieves a list with users whose dsEmail property contains the word "gmail" and the cdOrgUnit property has the "0000000007" value. The results are sorted by last name in ascending order. The list is scrolled displaying the first and last names of the users. Same criteria as for form searches are used.

```
SearchCriteria searchCriteria = new SearchCriteria();
Criteria criteria1 = Criteria.like("dsEmail", "gmail");
Criteria criteria2 = Criteria.eq("cdOrgUnit", "0000000007");
searchCriteria.addCriteria(criteria1);
searchCriteria.addCriteria(criteria2);
searchCriteria.addOrderAsc("dsLastName");
List list = userService.search(searchCriteria);
Iterator it = list.iterator();
while (it.hasNext()) {
 User user = (User) it.next();
 log(user.getFirstName() + "" + user.getLastName());
}
```

Organizational Units

The model class contains the properties with its getter and setter methods, while the service class contains the operation to be performed with the model.

Model Class Content

A user's model contains:

- Model builder.
- Set of getter and setter methods of its attributes.

Getter methods to get the organizational unit attributes

The model class of a case contains a set of getter methods for the organizational unit attributes.

Operation	Description	Parameters
getDisplayName()	Gets the descriptive name of the organizational unit.	
getName()	Gets the name of the organizational unit.	
getManager()	Gets the manager user of the organizational unit.	
getSuperiorOrganizationalUnit()	Gets the superior organizational unit of the organizational unit.	
getDescription()	Gets the description of the organizational unit.	
getEmail()	Gets the email of the organizational unit.	
getCalendar()	Gets the calendar of the organizational unit.	
getManagerCanReadPrivateData()	Returns true if the organizational unit manager has permission to read private data, false otherwise.	
getManagerCanUpdatePrivateData()	Returns true if the organizational unit manager has permission to update private data, false otherwise.	
getManagerCanDeletePrivateData()	Returns true if the organizational unit manager has permission to delete private data, false otherwise.	

Operation	Description	Parameters
getUserCanReadPrivateData()	Returns true if the organizational unit users have permission to read private data, false otherwise.	
getUserCanUpdatePrivateData()	Returns true if the organizational unit users have permission to update private data, false otherwise.	
getUserCanDeletePrivateData()	Returns true if the organizational unit users have permission to delete private data, false otherwise.	

Service Class Content

The service allows the following operations to be performed:

Operation	Description	Parameter
read(orgUnit)	Reads an instance.	User **OrgUnit**: Code of the organizational unit to read

Example of Use

The **Deyel** organizational unit object is used. It contains the use of the "OrganizationalUnit" model class and the "OrganizationalUnitService" service class.

1. Service creation

This service is created only once in the rule and is reused in different operations.

```
OrganizationalUnitService organizationalUnitService = new OrganizationalUnitService(getApiClient());
```

2. Reading of the organizational unit

This example retrieves the manager property value of an organizational unit identified with the "000000008' code, using the read method (organizationalUnit) of the "OrganizationalUnitService" service class. The manager property value is obtained through the corresponding getter method of the "OrganizationalUnit" model class.

```
OrganizationalUnit organizationalU-
nit = new
OrganizationalUnit();
organizationalU-
nit.setOrganizationalUnit("00000000
08");
OrganizationalUnit xOrganizatio-
nalUnitReaded =
organizationalUnitServi-
ce.read(organizationalUnit);
log("Manager: " + organizationalU-
nitReaded.getManager());
```

Calendars

For the calendar object, only service class operations are defined to work with dates, the model class is not necessary because the **Deyel** data types are used. The service class is predefined in the Deyel SDK and does not need to be downloaded.

Service Class Content

The service allows the following operations to be performed:

Operation	Description	Parameters
worksDays(start, end)	Obtains work days between dates, using the current user's calendar or another one specified as a parameter.	Timestamp **start**: Period start date Timestamp **end**: Period end date
worksDays(start, end, calendar)	Obtains work days between dates, using the calendar specified as a parameter.	Timestamp **start**: Period start date Timestamp **end**: Period end

Operation	Description	Parameters
		date Integer **calendar**: Code of the specified calendar
convertToTimeZone(date, calendar)	Converts a date into a specified time zone.	Timestamp **date**: Date to convert Integer **calendar**: Calendar code with specified time zone

Example of Use

Examples contain the use of "CalendarService" service class.

1. Service creation

This service is created only once in the rule and is reused in different operations.

```
CalendarService calendarService =
new
    CalendarServi-
ce(getApiClient());
```

2. Calculation of work days between dates

In this example, a budget number is read and the number of work days between the budget entry date and the budget due date is calculated.

A budget with the dsNumber "100234" property value of the "Budget" model class is read, using the read(budget) method of the "BudgetService" service class.
With the workDays(start, end) method of the "CalendarService" service class, the number of work days between the dtQuote and dtDue properties values is obtained, from the corresponding getter methods of the "Budget" model class.

```
BudgetService budgetService = new Bud-
getService(getApiClient());
 String dsNumber = "100234";
 Budget budget = new Budget();
 budget.setDsNumber(dsNumber);
```

```
budget = budgetService.read(budget);
Timestamp start = new Times-
tamp(budget.getDtQuote().getTime());
Timestamp end = new Times-
tamp(budget.getDtDue().getTime());
calendarService.worksDays(start, end);
```

3. Time calculation with another time zone

This example converts local time to Mexico time, set by calendar 2 time zone, using the convertToTimeZone(localTime, 2) method of the "CalendarService" service.

```
Timestamp localTime = Times-
tamp.valueOf(LocalDateTime.now());
    Timestamp timeMexico =
calendarServi-
ce.convertToTimeZone(localTime, 2);
   log("Local Time: " + localTime + "
Time Mexico: " + timeMexico);
```

Email Sending

For email sending, only the service class operation is defined, the model class is not necessary because **Deyel** data types are used. The service class is predefined in the Deyel SDK and does not need to be downloaded.

Service Class Content

Service allows the following operation to be performed:

Operation	Description	Parameters
sendmail(message, from, listTo, subject)	Sends an email.	String **message**: Email text String **from**: Sender email String[] **listTo**: List of recipient emails String **subject**: Email subject

Example of Use

The example contains the use of "EmailService" service class.

Standard email sending

In this example an email is sent where the body of the email, the subject, who sends it and those who receive it is specified.

The "EmailService" service is created to then execute the mail sending method sendMail(), using the required parameters.

```
EmailService emailService = new Email-
Service(getApiClient());
emailService.sendMail ("Message of
email",
"emailFrom@optaris.com",
new String[]{"emailTo@optaris.com"},
"My Email" );
```

SDK Programming Guide

To develop an advanced rule using an IDE, a series of steps must be followed.

Once the rule is created in the advanced rules modeler, it is downloaded to be included in the IDE project, it is developed and, once finished, it is imported from the context menu into the Deyel modeler.

Example of Use

This example describes the steps to develop the "getCostsWithTax" rule in the IDE.

A rule named "getCostsWithTax" is created. It receives a list of amounts as input parameter. Such list is added and the VAT cost is obtained. Returns the total VAT and the total amount of the sum as output parameter.

The "getCostsWithTax'' rule invokes the "taxCalculation" rule, which receives the sum of amounts as an input parameter, in the "inputAmount" field. Returns the VAT value as an output parameter, in the "taxValue" field.

> *For the example, the "taxCalculation" rule is assumed to be available within the environment.*

Step 1: Create the "getCostsWithTax" rule

In the rule modeler, a rule is created with the "getCostsWithTax" descriptive name.
The previously created adapter is selected, enter a description and press "Create Rule".

An input parameter is defined:

Name	Data Type
"amountList"	Double[]

Defined as output parameters:

Name	Data Type
"totalAmount"	Double
"AmountOfTaxApplied"	Double

This rule should be related to the "taxCalculation" rule using the wizard to include related objects, from the advanced properties tab.

Once the parameters have been entered and the new relation has been added, select the "Save" option from the top toolbar.

Step 2: Download the "getCostsWithTax" rule

From the top toolbar of the rule modeler, select the “Download” option.
The rule download includes the Java sources for developing and testing the rule.

Step 3: Setup the local environment

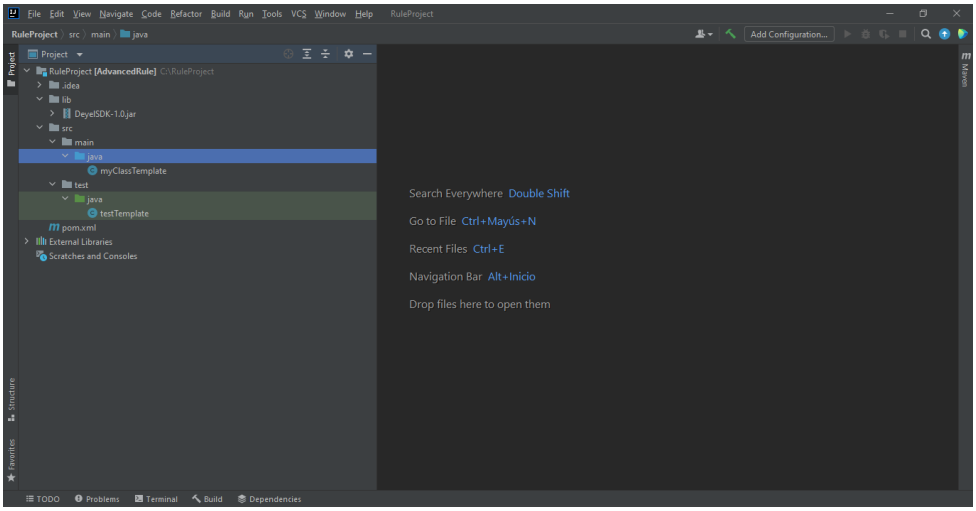
To develop the rule you need to have a local environment to work with Java.

Step 4: Include the "getCostsWithTax" rule in the Java IDE project

The rule download generates a file with a zip extension containing a "src" folder that respects the structure of Maven projects.

The file with zip extension should be copied to the project folder and decompressed in the same folder.

The structure of the project already configured is shown:



Step 5: Rule development

When the "getCostsWithTax" rule is included in the Java IDE project, two classes are obtained:

Name	Description
getCostsWithTax_1	Includes the rule logic
getCostsWithTax_1_Test	Includes the values to test from the IDE

The following code is defined in the getCostsWithTax_1 class:

```
public class getCostsWithTax_1 extends
 getCostsWithTax_1_abstract{
protected voidrun() throws ja-
va.lang.Exception{
 // The sum of the amounts is calcula-
ted
totalAmount = 0.0;
for (Double amount: AmountList
 totalAmount += amount;
   }
// 21% VAT on the amount entered is
calculated
 taxCalculationService myService =
new taxCalculationServi-
ce(getApiClient());
```

```
 taxCalculation myRule =new taxCalcula-
tion();
 myRule.setInputAmount(totalAmount);
 myRule.setTaxRate(new Double(21.0));
 myRule = myService.execute(myRule);
// Gets the amount of VAT applied
amountOfTaxApplied = myRu-
le.getTaxValue() ;

    }
}
```

Setup of the Local Environment

The steps necessary to setup the local environment to work with Java are described below. After completing these steps, the rule can be included in the Java IDE.

> *For projects created in a version earlier than 7.7, it is recommended to build a new project with the Maven project structure.*

Step 1: Install the Java JDK

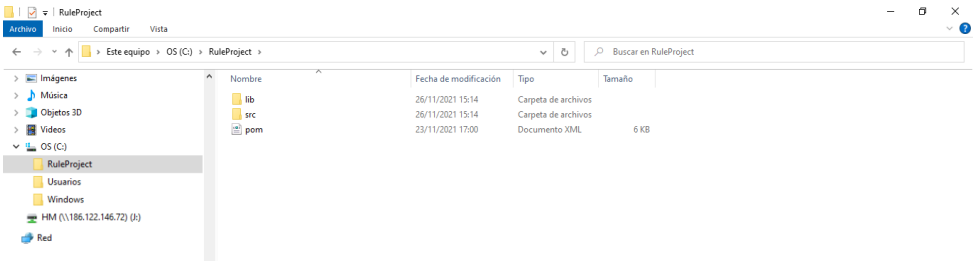
Download and install the Java JDK (Java Development Kit).

Step 2: Install an IDE of preference to work with Java

Download and install a Java IDE.

Step 3: Download the Maven Project

From the top toolbar in the save submenu of the rule modeler select the option "Download SDK". Downloading the Maven project, a file with a zip extension is obtained, once it is unzipped into a folder, it is subsequently imported into the project defined in the IDE.



Step 4: Create the project in a Java IDE

In order to create a project in a Java IDE it is necessary to:

- Import from the IDE the Maven project previously downloaded in the folder in Step 3. Folders will automatically be associated with the project in the IDE used.

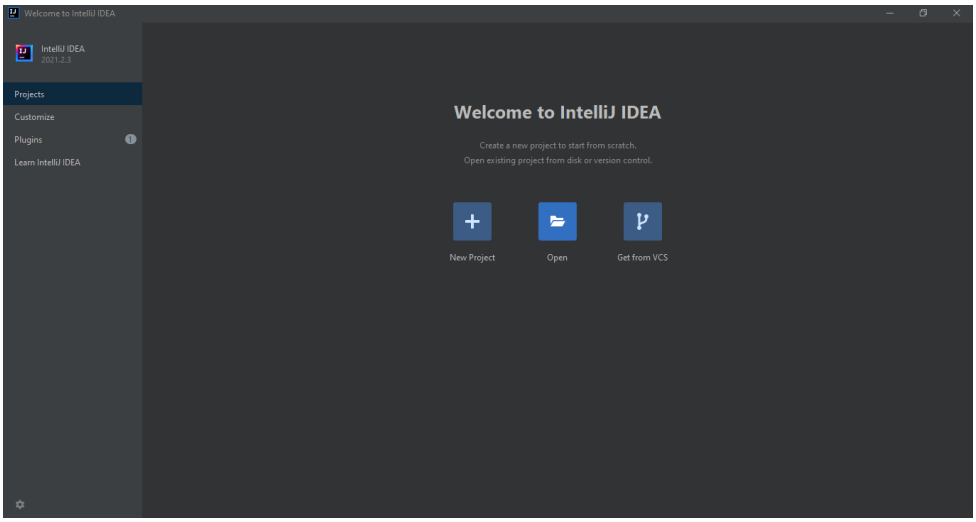
Guide to create the project in the IntelliJ IDE

Guide to create a project in Eclipse IDE

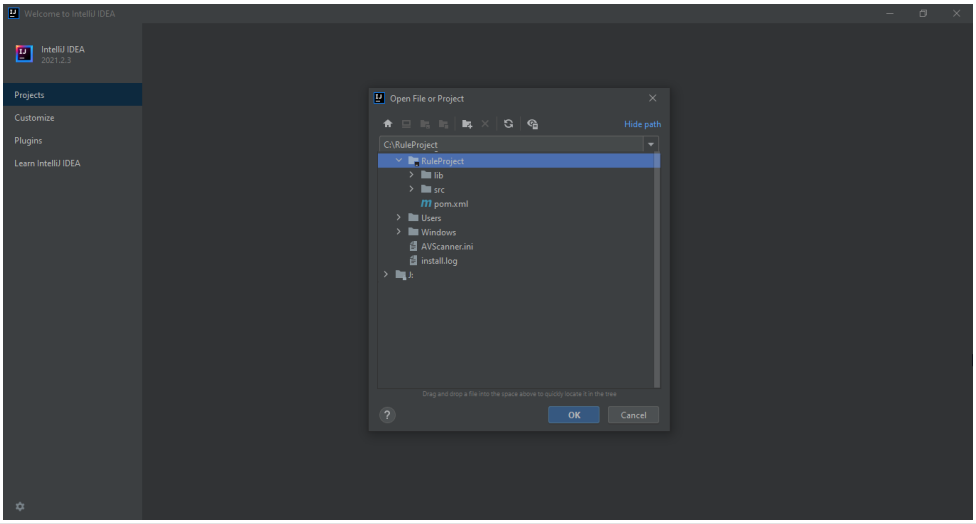
Guide to create the project in the IntelliJ IDE

The steps required to create the project in the Intellij IDE are described below.

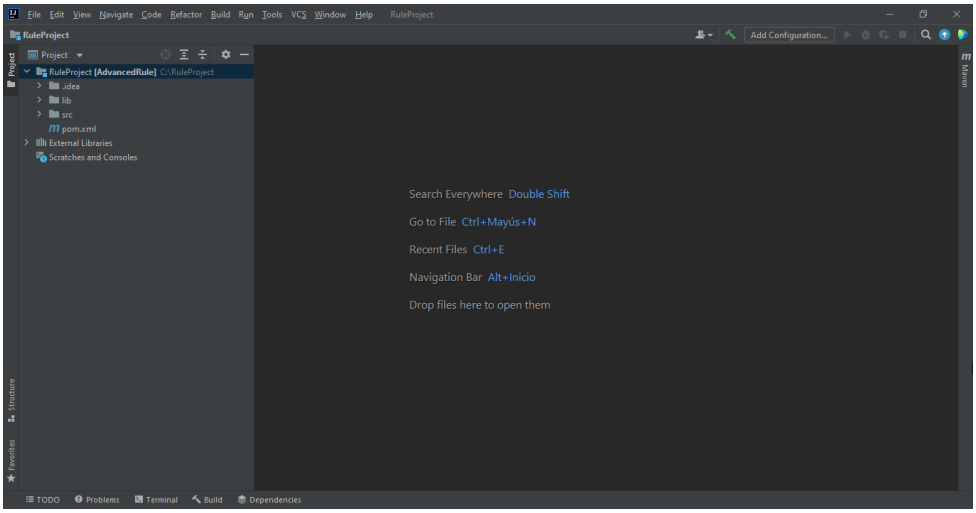
1. Press the button corresponding to open.



2. Select the folder in which the Maven project downloaded from **Deyel** was unzipped. Press the button corresponding to accept.



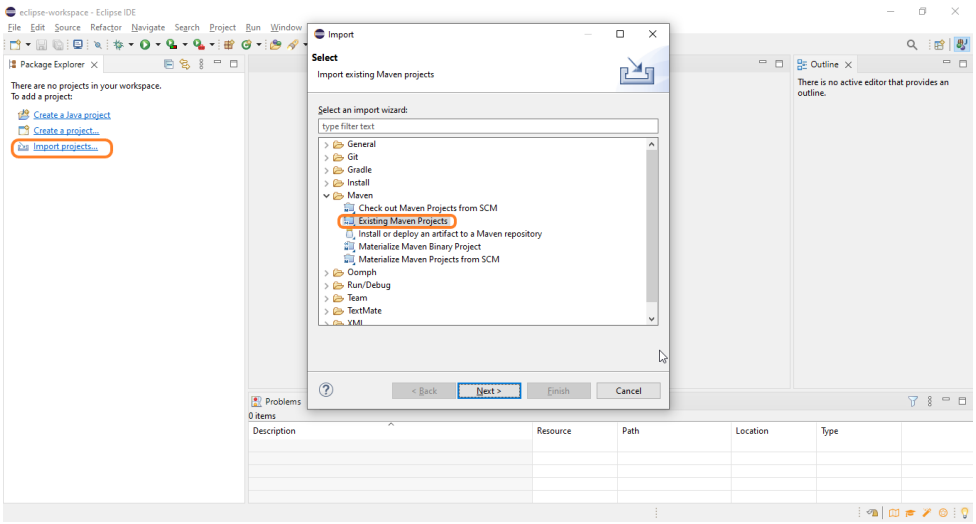
3. End of installation, the project has already been imported.



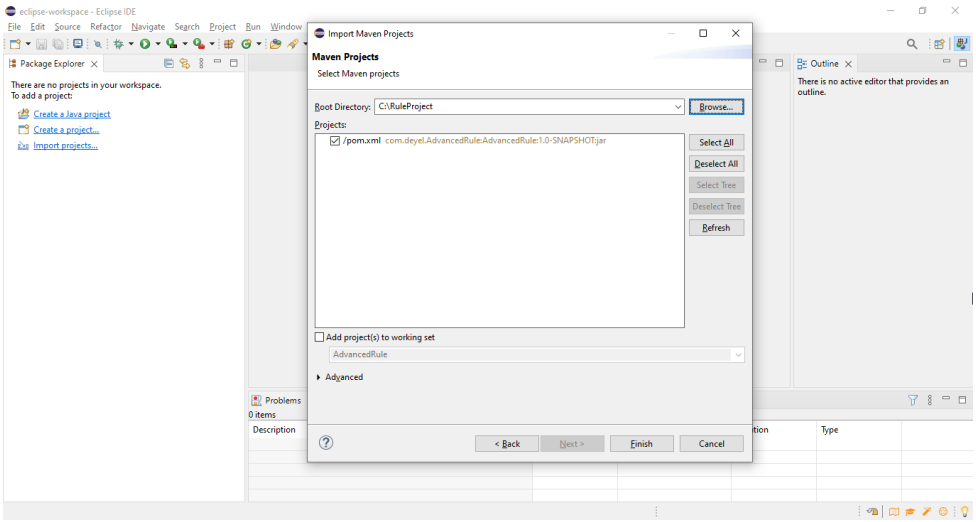
Guide to create a project in Eclipse IDE

The steps required to create a project in Eclipse IDE are described below.

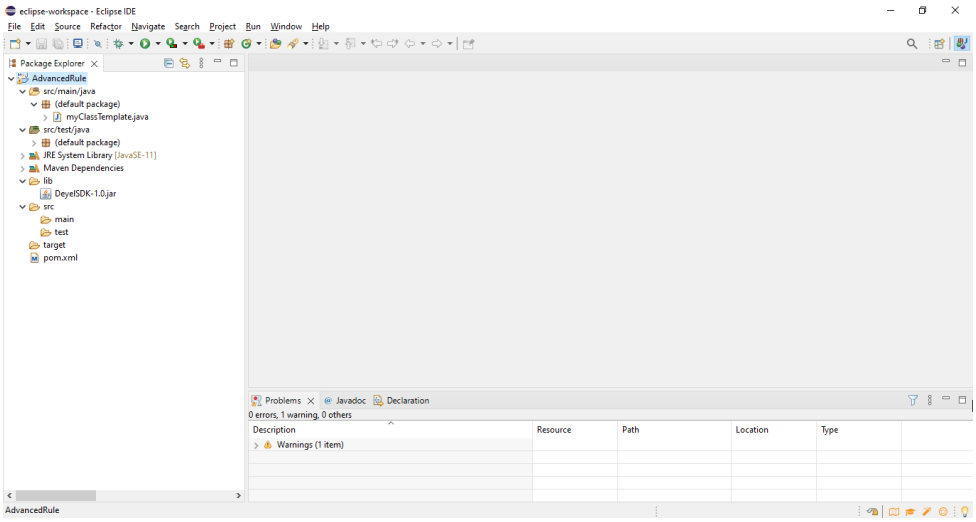
1. Select "Import Projects". Choose the option "Existing Maven Projects'' and press the button corresponding to next.



2. Select the folder in which the Maven project downloaded from **Deyel** was unzipped. Press the button corresponding to end.



3. End of installation, the project has already been imported.



3.6.5.2. Adapters

Adapters are defined in **Deyel** and they allow integration with different applications and platforms through the use of advanced business rules. Its purpose is to exchange data and share behaviors.

An adapter encapsulates the complexity of communicating with an external component. It allows different advanced rules to use it to invoke the operations that such component exposes.

When an adapter is defined, the following is configured:

- The attributes that identify the platform with which you want to interact.
- The way to establish communication.
- The operations the software component exposes.

The definition of adapters is done using the modeling facilities property.

Types

Default

These adapters are defined with **Deyel** and they are available in the modeler's grid. There can only be one of each type.

- Twitter
- Mercado Libre
- Tienda Nube
- Deyel SDK
- Docusign
- AwsTextract
- GoogleDrive
- OneDrive

For each of the above adapters, with the exception of AwsTextract and Deyel SDK, an application must be created on each of the platforms, which is the one that establishes the relation between the platform and the environment of **Deyel**. When creating this application, each platform provides the necessary values to set the connection credentials of the corresponding adapter.

The Deyel SDK adapter, for development and execution of business rules, allows the rules that use it to make use of the Deyel SDK facilities.

Standard

These adapters allow access to web services from external providers, in their different communication architectures:

- Rest API
- SOAP

The rules used by these adapters make invocations to specific endpoints, made up of the service URL and information about the operation to be performed. These calls can use different types of authentication, which depend on the security settings implemented by the provider.

Database

Adapters that allow to define access to relational databases through the Java JDBC protocol.

Deyel provides custom adapters for the most popular database engines on the market, facilitating configuration and optimizing interaction with such engines.

The adapters available are:

- IBM/DB2
- Informix
- MySQL
- Microsoft SQL Server
- ORACLE

The rules used by these adapters can execute SQL statements to get and write data to and from any database.

Application

These adapters allow to define access to market applications that expose services to interact with them, integrating operation between **Deyel** and such applications. Thus, the rules that use these adapters can do it safely and encapsulated.

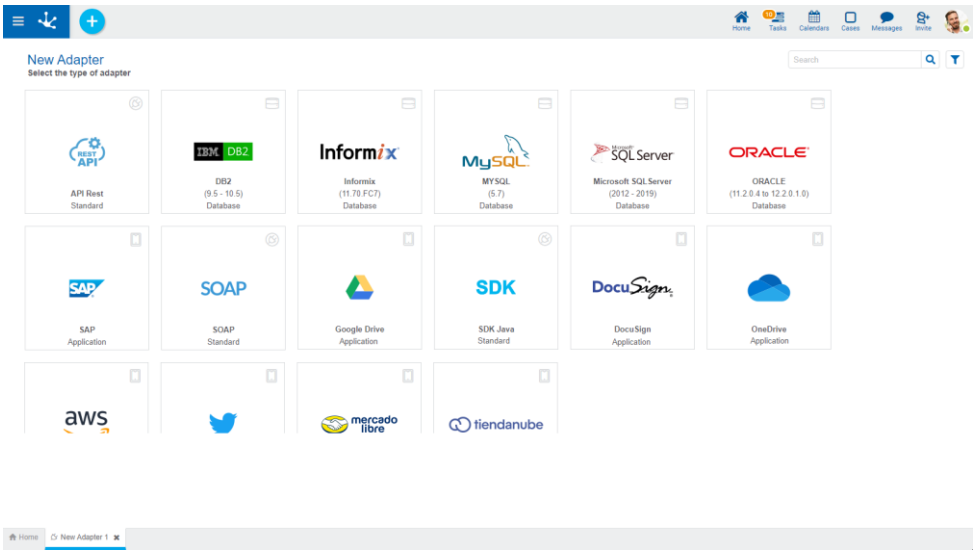
The adapters available are:

- SAP

3.6.5.2.1. Modeling Facilities

New Adapter

To define a new adapter, the modeling user must select its type from a gallery.



In the gallery, for each adapter, the type it belongs to is displayed, and its version is displayed only if applicable. Once the type is selected, go to the properties panel of each one to define its own characteristics.

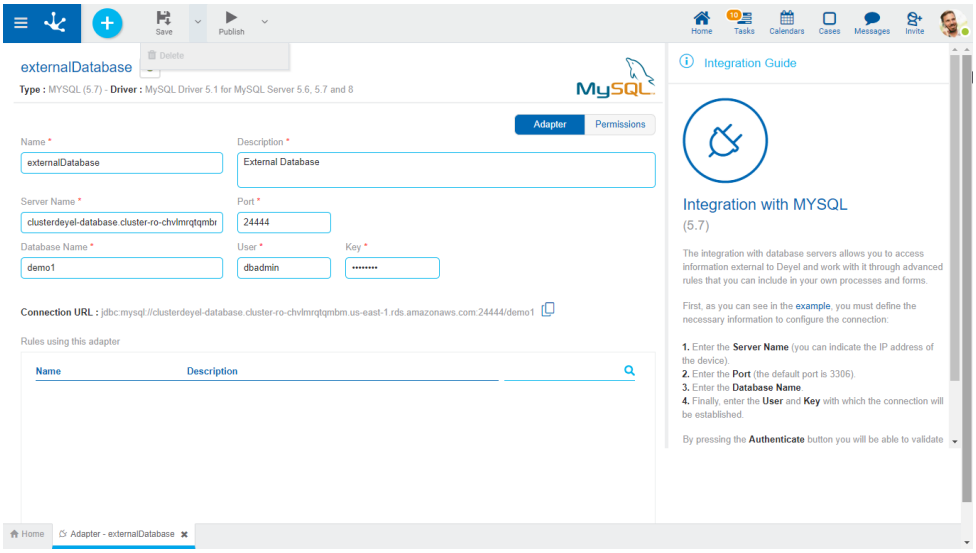
Workspace Sections:

- Top Toolbar
- Adapter Definition Area

3.6.5.2.1.1. Top Toolbar

This toolbar contains icons and submenus from which operations on the adapter can be performed.

The bar is displayed on the toolbar of **Deyel**.



Save

This icon allows to save the object in the repository of **Deyel**, leaving its state as "Draft" or "Modified". It is verified that certain conditions are met.

Main Conditions

- The name must be unique.
- All the fields indicated as required must be complete.

Publish

By means of this icon the object goes to "Published" state, after verifying a set of additional conditions to those of the "Save" operation.
As for integration adapters with services from other platforms, authentication is performed. On the other hand, if it is a database integration adapter, the connection is tested.

Main Conditions

- Integration adapters with services from other platforms require valid URLs, authentication methods and other attributes of the connection such as user, token or keys.
- Database integration adapters require valid specified values for the server name, database, schemas, users, and user password used to establish the connection.

Delete

Allows to delete the object only if it is in "Draft"state, closing the tab in which it is located and deleting it from the modeler's grid. This operation is not available for predefined adapters.

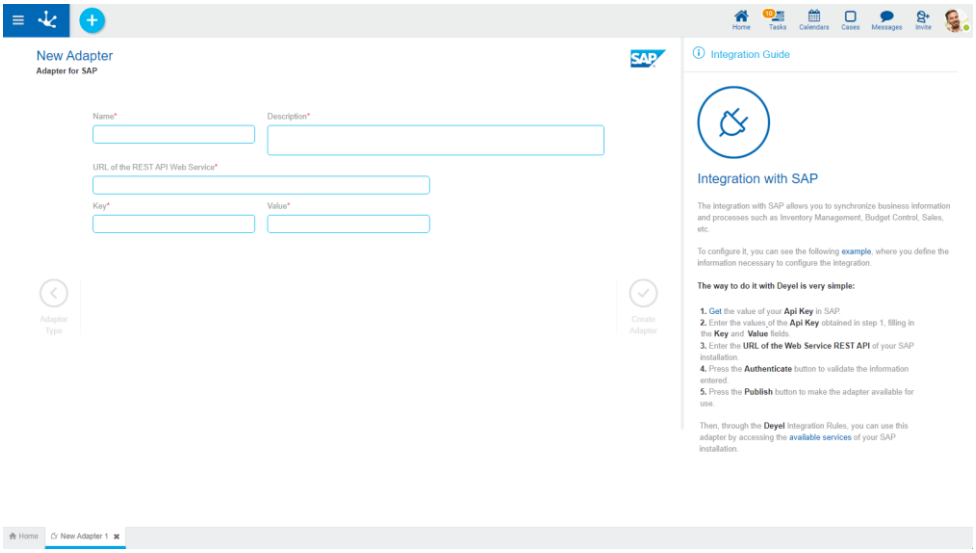
Delete Publication

This icon allows to remove the adapter from use by returning it to the state"Draft".

3.6.5.2.2. Definition Area

The definition area consists of a panel with the adapter properties and an integration guide that assists and guides the modeler. Depending on each adapter, it may contain examples of use or references to external sites that facilitate configuration.

The adapter properties can be entered both at the time of their creation and when modifying an existing one. Each adapter allows configuration of specific properties, although they share some, for example the name that identifies them.



> *An asterisk "*" on the label indicates that the property is required.*

Properties

In addition to these shared properties, each type of adapter has specific properties.

Name

Name of the adapter that uniquely identifies it. It is mandatory. Identifier that is displayed in the modeler's grid and selecting the adapter when creating an advanced rule.

Description

Text that defines the adapter describing its functionality. It is mandatory. This description is displayed along with its name by selecting the adapter when creating an advanced rule.

Predefined Adapters

- Twitter
- Mercado Libre
- Tiendanube
- Deyel SDK
- DocuSign
- AwsTextract

- GoogleDrive
- OneDrive

Standard Adapters

- Rest API
- Soap

Database Adapters

- IBM/DB2
- Informix
- MySQL
- Microsoft SQL Server
- Oracle

Application Adapters

- SAP

3.6.5.2.2.1. Twitter

Deyel allows integration with the Twitter platform through this adapter. In it, the connection credentials are registered and the interaction of the network users of **Deyel** with Twitter can be centrally managed.

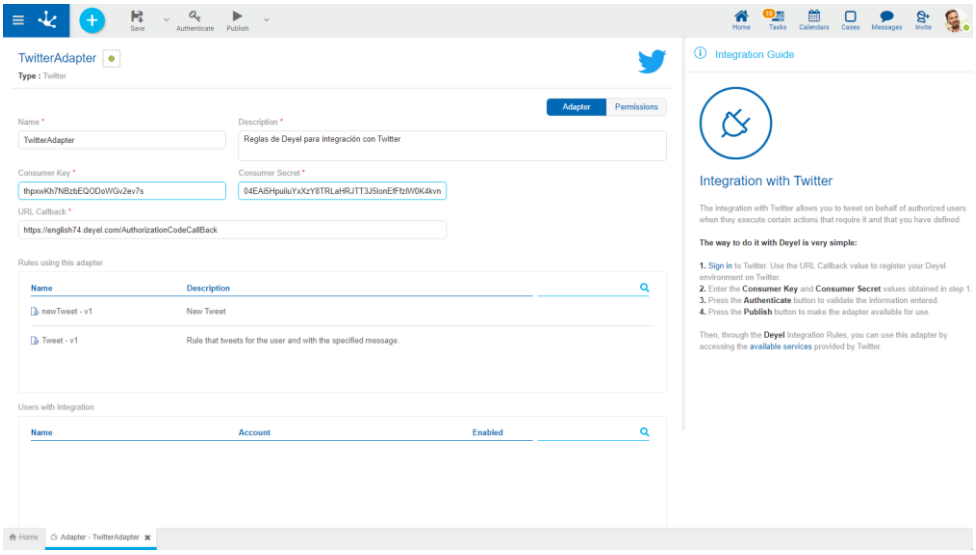
From their profile, each user can enter their credentials to access their Twitter account so that **Deyel** uses them when connecting on its behalf, for example, to publish a post.

Steps for Integration with Twitter

- Register an application on Twitter that identifies the instance of **Deyel**.
- Model the properties of the Twitter adapter detailed below.
- Users must authorize **Deyel** to use the Twitter account from their profile.

Properties

Apart from properties shared by adapters those specific to Twitter are added.



An asterisk "" on the label indicates that the property is required.*

Consumer Key
Identification key of the service that Twitter issues when creating an application on its platform.

Consumer Secret
Secret key associated to the property Consumer Key, are used together when requesting access to the resources provided by the platform.

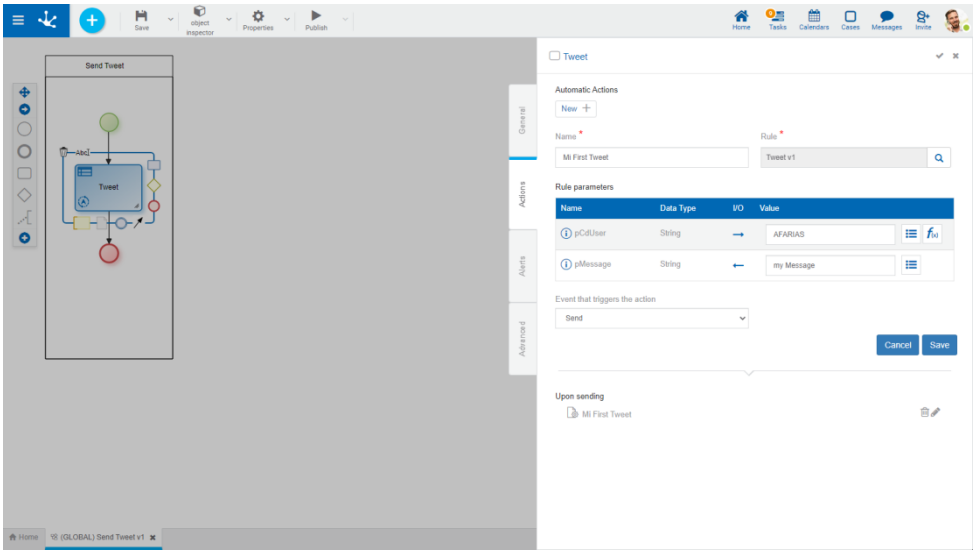
URL Callback
This property must be defined when creating the application on Twitter and corresponds to the URL that handles the return of requests for services that are made from the **Deyel** environment to the Twitter platform.
The value for this property must be composed with the name of the environmentDeyel https://.deyel.com/AuthorizationCodeCallBack.

Rules using this adapter
In this grid, the advanced rules used by the adapter are displayed and their definition can be shown from each line.

Users with integration
This grid shows users who have configured the integration with Twitter in their profile and the state of each one, which can be enabled or disabled. The state can be modified from each line.

Example of Use

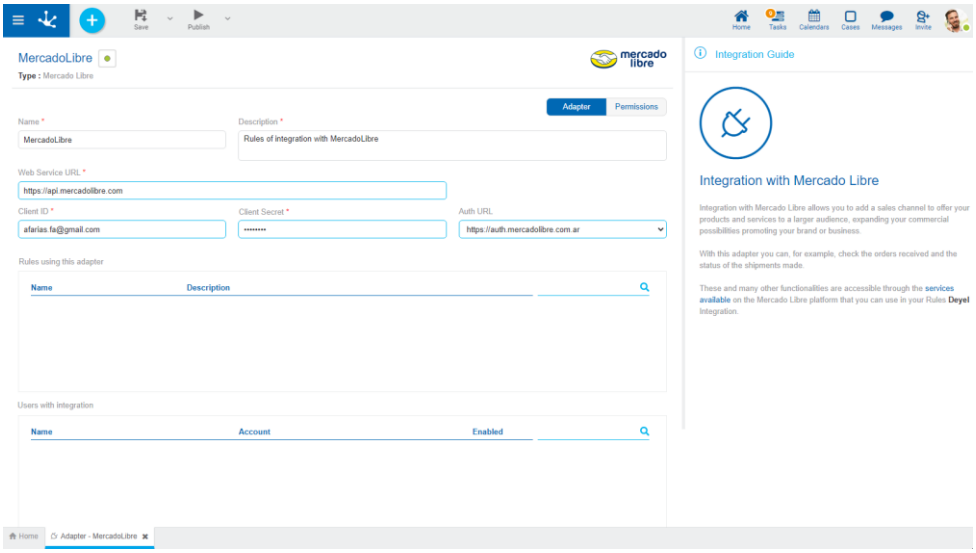
In a content publishing management process, an activity can publish a tweet automatically.



It must be a rule type activity and as an automatic action it invokes a "Tweet v1 - Rule" rule that publishes a tweet to the user with the specified message. The user that publishes the tweet and the message are reported as parameters.

3.6.5.2.2.2. Mercado Libre

Apart from properties shared by adapters those specific to Mercado Libre are added.



> *An asterisk "*" on the label indicates that the property is required.*

Web Service URL

Corresponds to the URL that attends requests for services that are made from the **Deyel** environment to the Mercado Libre platform.

Client ID

Identification of the service that Mercado Libre issues when creating an application on its platform.

Client Secret
Secret key associated to the Client ID property, they are used together when requesting access to the resources provided by the platform.

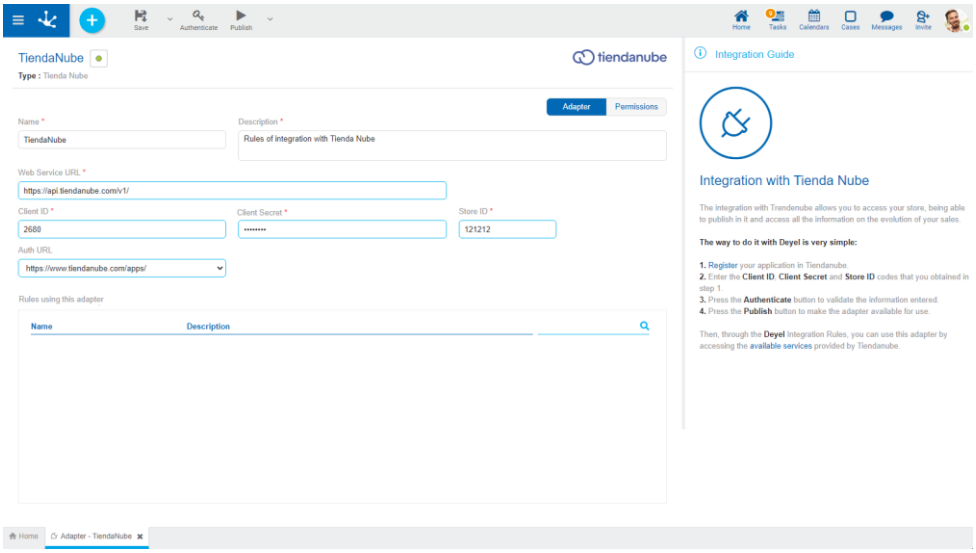
Auth URL
Corresponds to the Mercado Libre URL against which the authentication must be performed.

Rules using this adapter
In this grid, the advanced rules used by the adapter are displayed and their definition can be shown from each line.

Users with integration
In this grid you can see the users who have the Mercado Libre integration configured and the state of each one, which can be enabled or disabled. The state can be modified from each line.

3.6.5.2.2.3. Tiendanube

Apart from properties shared by adapters those specific to Tiendanube are added.



> *An asterisk "*" on the label indicates that the property is required.*

Web Service URL
Corresponds to the URL that handles the return of requests for services that are made from the **Deyel** environment to the Tiendanube platform.

Client ID
Identification of the service that Tiendanube issues when creating an application on its platform.

Client Secret
Secret key associated to the Client ID property, they are used together when requesting access to the resources provided by the platform.

Store ID

It is the unique identification of the store in Tiendanube

Auth URL

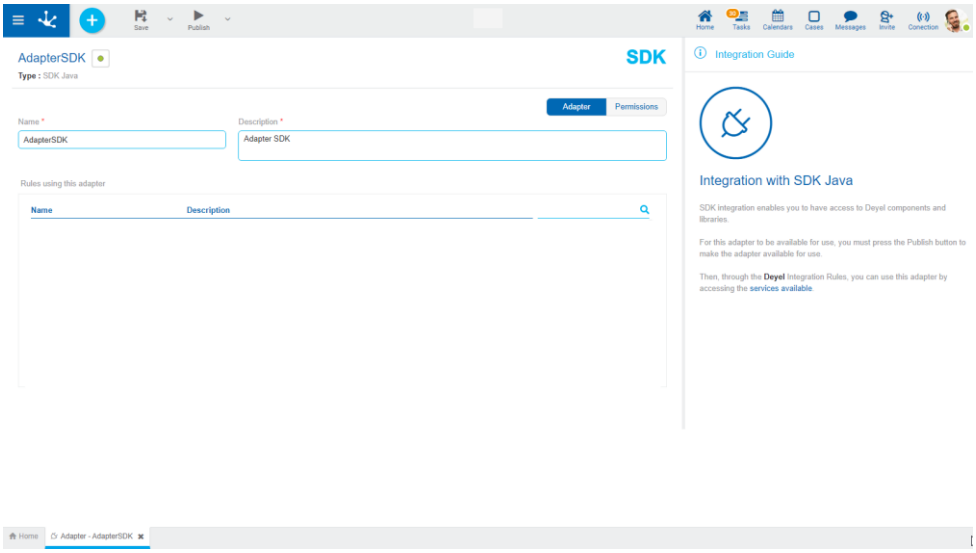
It corresponds to the URL of Tiendanube against which the authentication must be performed.

Rules using this adapter

In this grid, the advanced rules used by the adapter are displayed and their definition can be shown from each line.

3.6.5.2.2.4. Deyel SDK

Apart from properties shared by adapters the grid of rules used by the adapter is added.



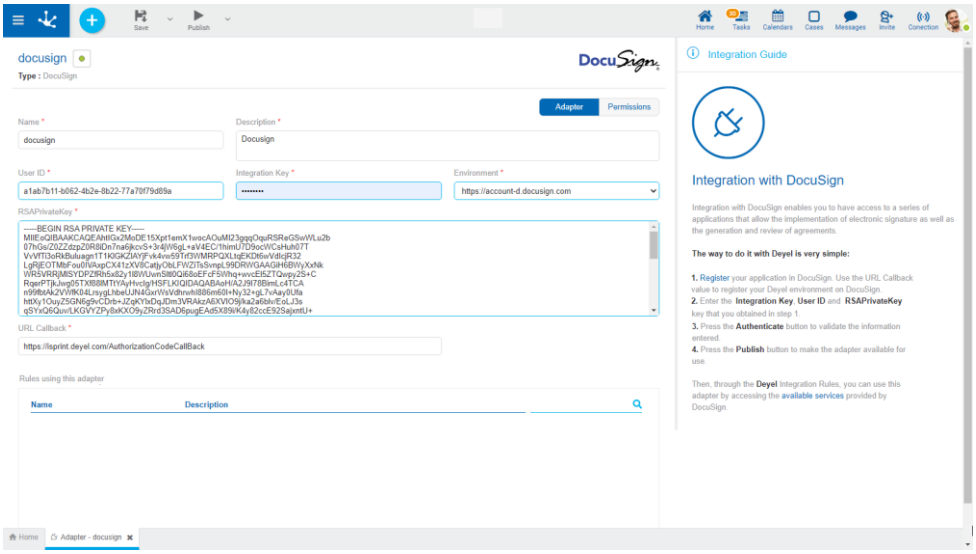
> *An asterisk "*" on the label indicates that the property is required.*

Rules using this adapter

In this grid, the advanced rules used by the adapter are displayed and their definition can be shown from each line.

3.6.5.2.2.5. DocuSign

Apart from properties shared by adapters those specific to DocuSign are added.



An asterisk "" on the label indicates that the property is required.*

User ID

Identification of the user account registered in DocuSign.

Integration Key

Identification key of the service that DocuSign issues when creating an application on its platform.

Environment

Corresponds to the DocuSign URL against which APIRest operations are performed, it can be the production environment or the development environment.

RSAPrivateKey

Secret key associated to the property Integration Key, are used together when requesting access to the resources provided by the platform.

URL Callback

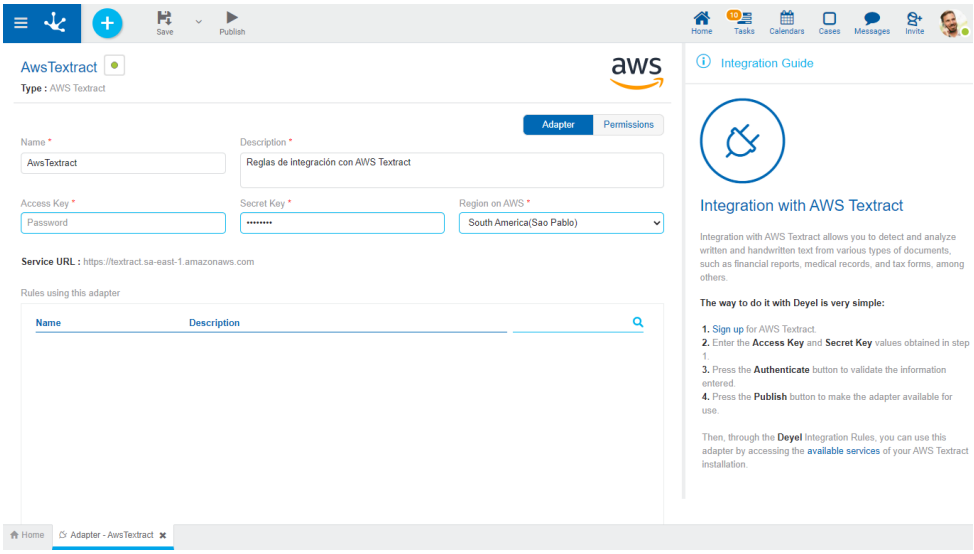
This property must be defined when creating the application on DocuSign and corresponds to the URL that handles the return of requests for services that are made from the **Deyel** environment to the DocuSign platform.
The value for this property must have the name of the environmentDeyel https://.deyel.com/AuthorizationCodeCallBack.

Rules using this adapter

In this grid, the advanced rules used by the adapter are displayed and their definition can be shown from each line.

3.6.5.2.2.6. AwsTextract

Apart from properties shared by adapters those specific to AwsTextract are added.



An asterisk "" on the label indicates that the property is required.*

Access Key
Identification key of the service that AwsTextract issues when creating an application on its platform.

Secret Key
Secret key associated to the property Access Key, are used together when requesting access to the resources provided by the platform.

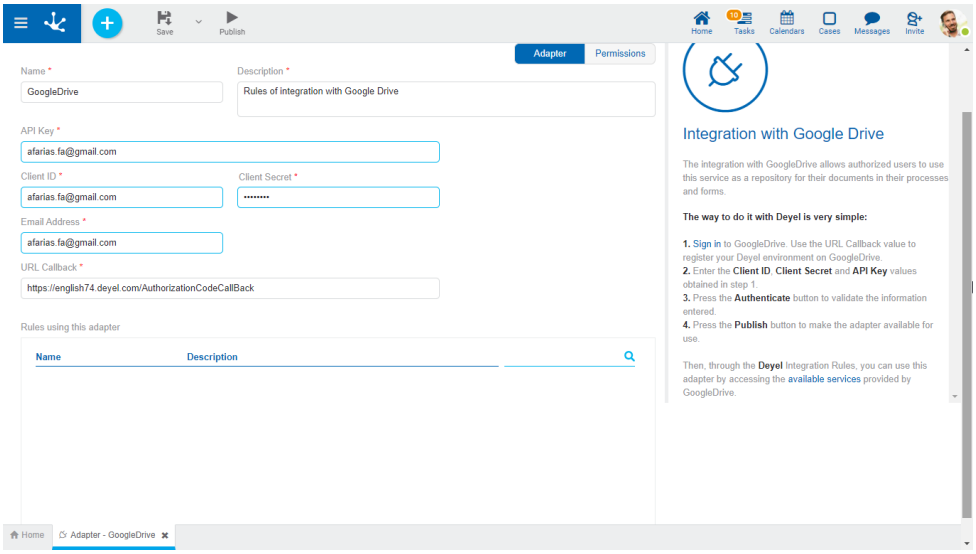
Region on AWS
Indicates the location of the services to be used. The region determines the transaction processing quotas, based on the documentation.

Service URL
This property is filled in automatically when selecting the Region on AWS property.

Rules using this adapter
In this grid, the advanced rules used by the adapter are displayed and their definition can be shown from each line.

3.6.5.2.2.7. GoogleDrive

Apart from properties shared by adapters those specific to GoogleDrive are added.



> *An asterisk "*" on the label indicates that the property is required.*

API Key
Access key to the service that GoogleDrive issues when creating an application on its platform.

Client ID
Identification of the service that GoogleDrive issues when creating an application on its platform.

Client Secret
Secret key associated to the Client ID property, they are used together when requesting access to the resources provided by the platform.

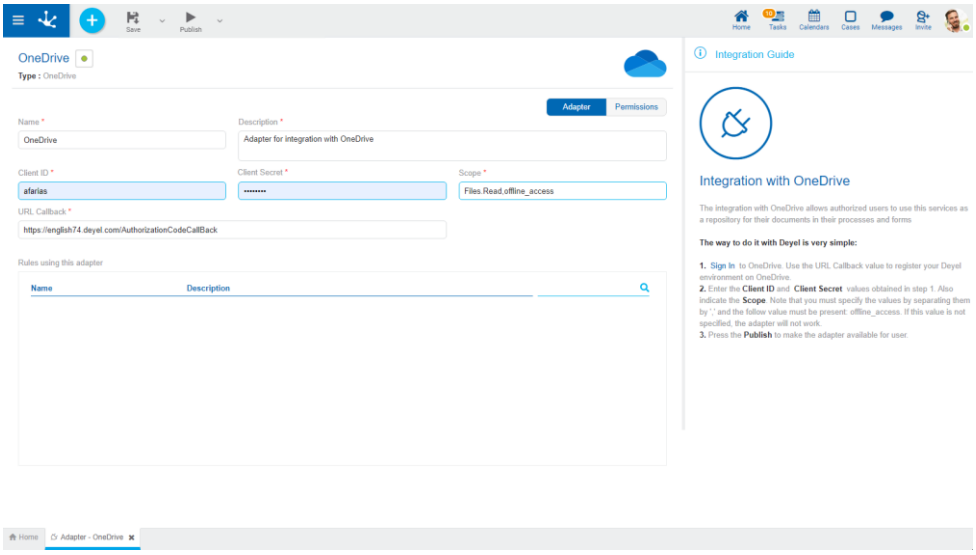
Email Address
Associated account and owner of the GoogleDrive space.

URL Callback
This property must be defined when creating the application on OneDrive and corresponds to the URL that handles the return of requests for services that are made from the **Deyel** environment to the GoogleDrive platform.
The value for this property must be composed with the name of the environment Deyel https://<miambiente>.deyel.com/AuthorizationCodeCallBack.

Rules using this adapter
In this grid, the advanced rules used by the adapter are displayed and their definition can be shown from each line.

3.6.5.2.2.8. OneDrive

Apart from properties shared by adapters those specific to OneDrive are added.



An asterisk "" on the label indicates that the property is required.*

Client ID

Identification of the service that OneDrive issues when creating an application on its platform.

Client Secret

Secret key associated to the Client ID property, they are used together when requesting access to the resources provided by the platform.

Scope

Determines what type of access is granted to the application when the user logs in, based on documentation.

URL Callback

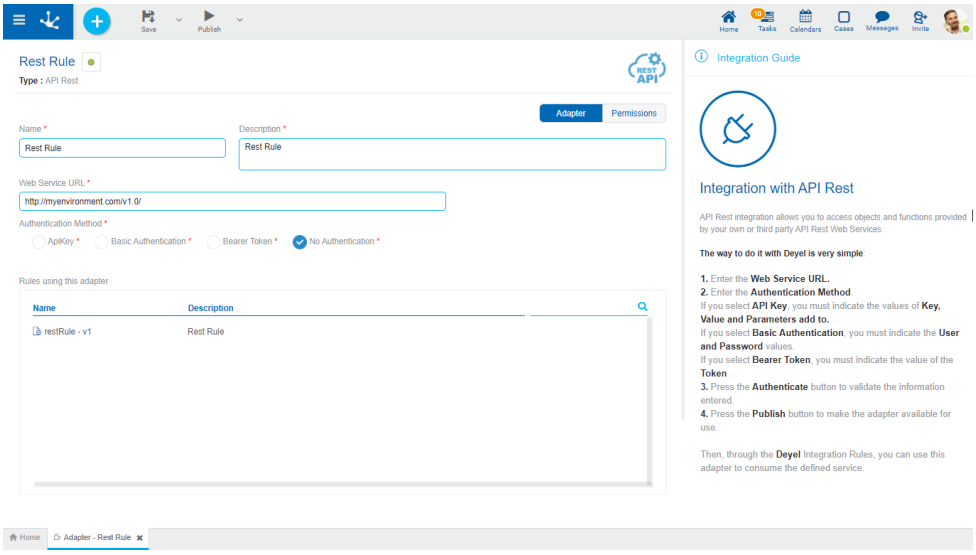
This property must be defined when creating the application on OneDrive and corresponds to the URL that handles the return of requests for services that are made from the **Deyel** environment to the GoogleDrive platform.
The value for this property must be composed with the name of the environmentDeyel https://.deyel.com/AuthorizationCodeCallBack.

Rules using this adapter

In this grid, the advanced rules used by the adapter are displayed and their definition can be shown from each line.

3.6.5.2.2.9. API Rest

Apart from properties shared by adapters those specific to API Rest are added.



An asterisk "" on the label indicates that the property is required.*

Web Service URL

Address of the Rest service with which it is being integrated.

Authentication Method

Identifies the way **Deyel** must authenticate with the service depending on the security scheme it implements.

The possible schemes are:

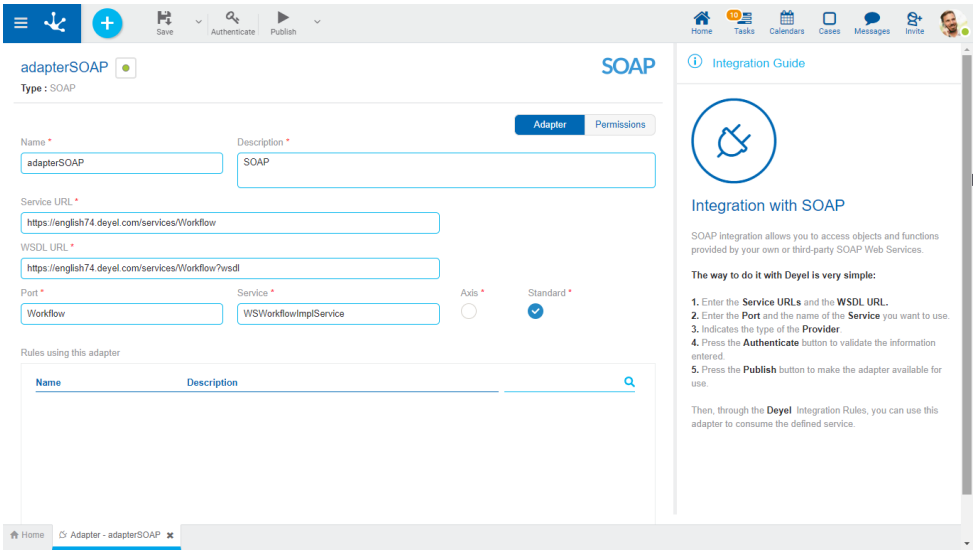
- ApiKey: in this case the service platform generates a combination of unique values called Key and Value. In addition to these values, the value for the property Parameters add to must be indicated, where it is specified in which way the parameters should be sent to the service, it can be done in the Header or as Query Params.
- Basic Authentication: it is the simplest authentication scheme. It must specify the User and Password, provided by the participant responsible of the service.
- Bearer Token: it is a scheme similar to ApiKey but in this case it must specify the Token, which is generated by the service.
- No Authentication: it does not require any access validation.

Rules using this adapter

In this grid, the advanced rules used by the adapter are displayed and their definition can be shown from each line.

3.6.5.2.2.10. Soap

Apart from properties shared by adapters those specific to Soap are added.



An asterisk "" on the label indicates that the property is required.*

Service URL
Address of the SOAP service with which you want to integrate.

WSDL URL
Address where the definition of the SOAP service classes and methods is obtained.

Port
Port number used by the SOAP service.

Service
It is the name of the service class.

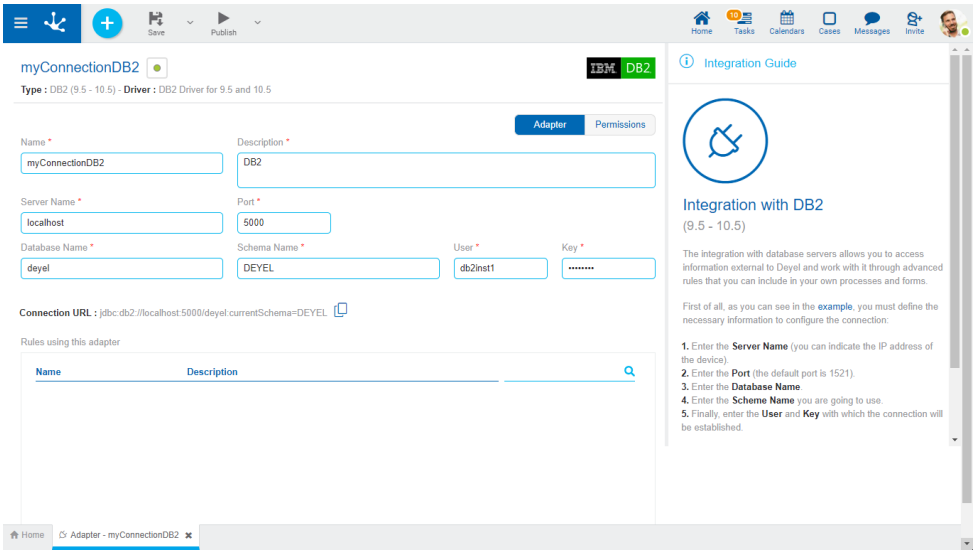
Authentication Method
Identifies how **Deyel** will authenticate with the service. Possible values are:

- Axis
- Standard

Rules using this adapter
In this grid, the advanced rules used by the adapter are displayed and their definition can be shown from each line.

3.6.5.2.2.11. IBM/DB2

Apart from properties shared by adapters those specific to IBM/DB2 are added.



An asterisk "" on the label indicates that the property is required.*

Server Name
Name or IP address of the server where the database engine is installed.

Port
Port number used by the database engine.

Database Name
Corresponds to the identification of the database with which you want to integrate.

Schema Name
Name of the schema within the database with which you want to integrate.

User
Identification of the database user that has permissions on the schema.

Key
Password of the database user making the connection.

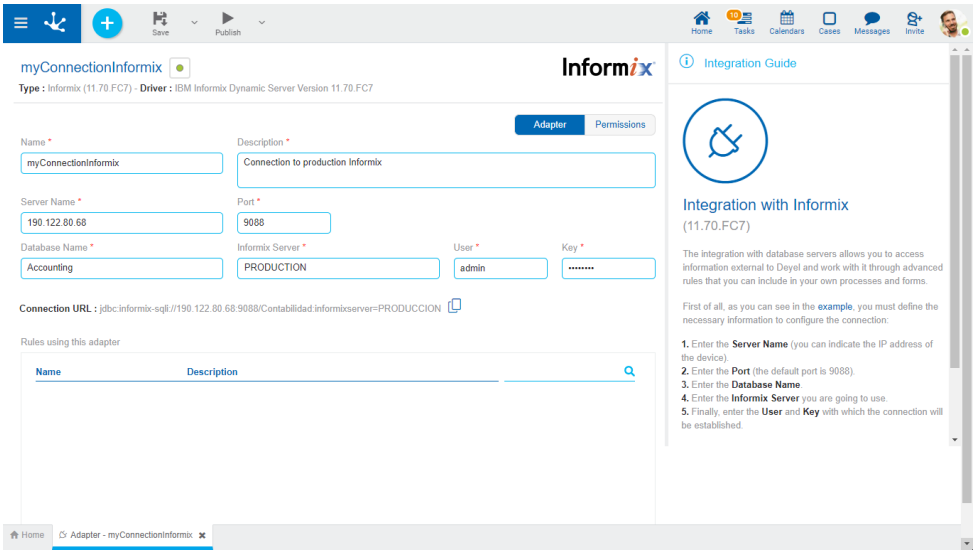
Connection URL
Shows the connection string for the adapter.

Allows to copy the connection URL.

Rules using this adapter
In this grid, the advanced rules used by the adapter are displayed and its definition can be shown from each line.

3.6.5.2.2.12. Informix

Apart from properties shared by adapters those specific to Informix are added.



An asterisk "" on the label indicates that the property is required.*

Server Name
Name or IP address of the server where the database engine is installed.

Port
Port number used by the database engine.

Database Name
Corresponds to the identification of the database with which you want to integrate.

Informix Server
Name of the server that manages the database with which you want to integrate.

User
Identification of the database user that has permissions on the schema.

Key
Password of the database user making the connection.

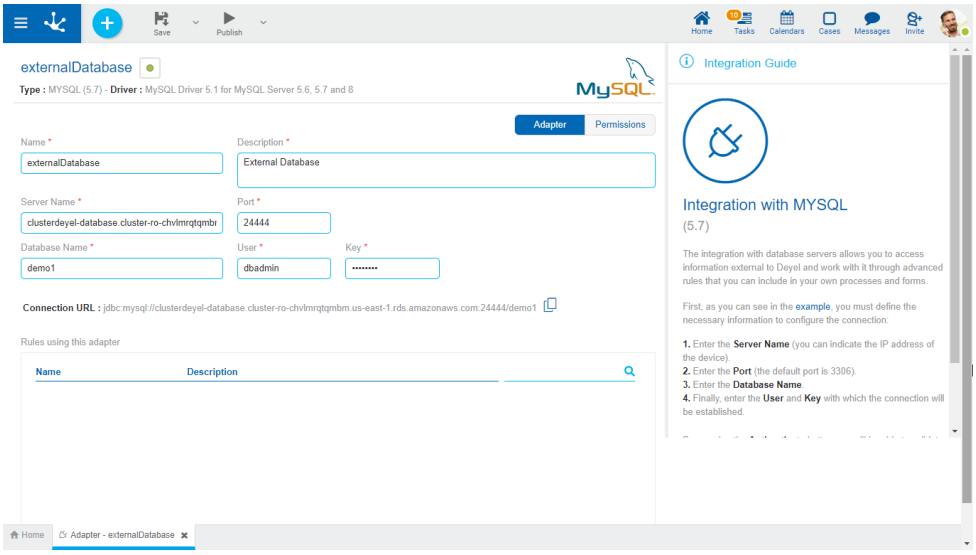
Connection URL
Shows the connection string for the adapter.

Allows to copy the connection URL.

Rules using this adapter
In this grid, the advanced rules used by the adapter are displayed and their definition can be shown from each line.

3.6.5.2.2.13. MySQL

Apart from properties shared by adapters those specific to MySQL are added.



An asterisk "" on the label indicates that the property is required.*

Server Name
Name or IP address of the server where the database engine is installed.

Port
Port number used by the database engine.

Database Name
Corresponds to the identification of the database with which you want to integrate.

User
Identification of the database user that has permissions on the schema.

Key
Password of the database user making the connection.

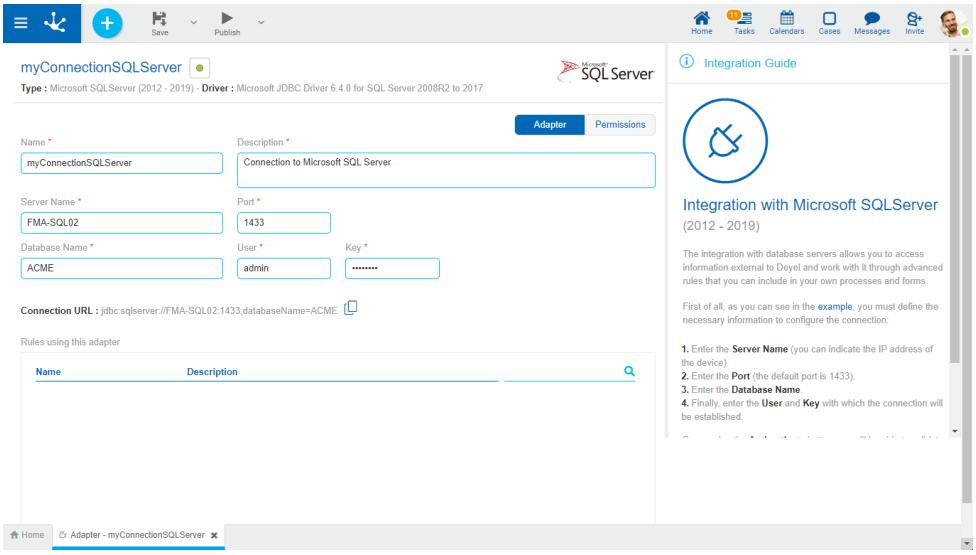
Connection URL
Shows the connection string for the adapter.

Allows to copy the connection URL.

Rules using this adapter
In this grid, the advanced rules used by the adapter are displayed and their definition can be shown from each line.

3.6.5.2.2.14. Microsoft SQL Server

Apart from properties shared by adapters those specific to Microsoft SQL Server are added.



An asterisk "" on the label indicates that the property is required.*

Server Name
Name or IP address of the server where the database engine is installed.

Port
Port number used by the database engine.

Database Name
Corresponds to the identification of the database with which you want to integrate.

User
Identification of the database user that has permissions on the schema.

Key
Password of the database user making the connection.

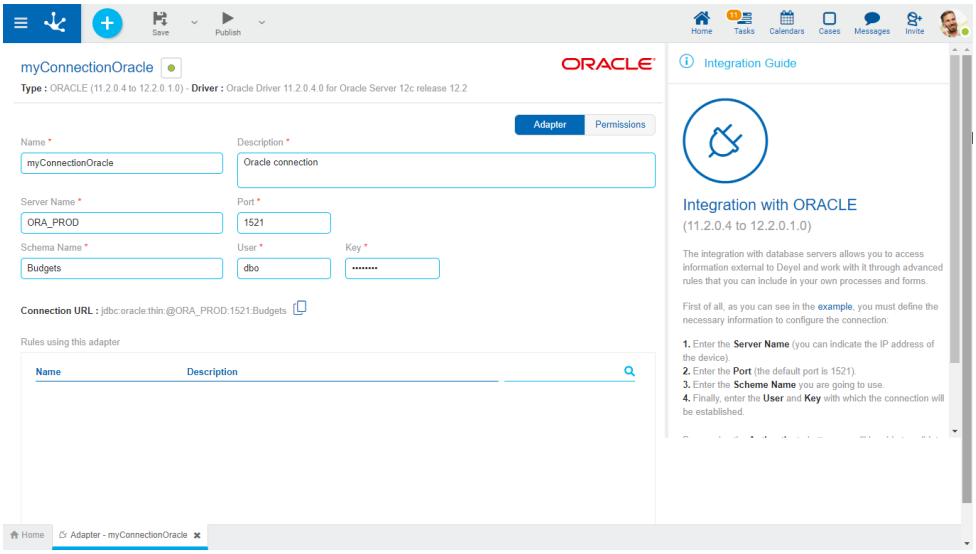
Connection URL
Shows the connection string for the adapter.

Allows to copy the connection URL.

Rules using this adapter
In this grid, the advanced rules used by the adapter are displayed and their definition can be shown from each line.

3.6.5.2.2.15. Oracle

Apart from properties shared by adapters those specific to Oracle are added.



An asterisk "" on the label indicates that the property is required.*

Server Name
Name or IP address of the server where the database engine is installed.

Port
Port number used by the database engine.

Schema Name
Name of the schema within the database with which you want to integrate.

User
Identification of the database user that has permissions on the schema.

Key
Password of the database user making the connection.

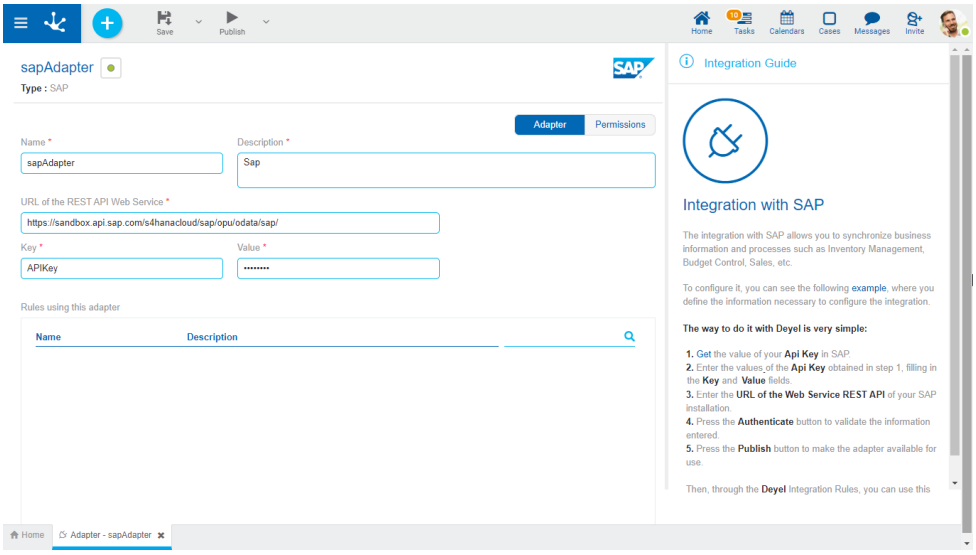
Connection URL
Shows the connection string for the adapter.

Allows to copy the connection URL.

Rules using this adapter
In this grid, the advanced rules used by the adapter are displayed and their definition can be shown from each line.

3.6.5.2.2.16. SAP

Apart from properties shared by adapters those specific to SAP are added.



An asterisk "" on the label indicates that the property is required.*

URL of the REST API Web Service

Service URL of the SAP environment with which it is integrating.

Key

Identifier that serves as a means of authentication to use of the service, provided by the SAP environment.

Value

Key associated to the property Key, also provided by the SAP environment.

Rules using this adapter

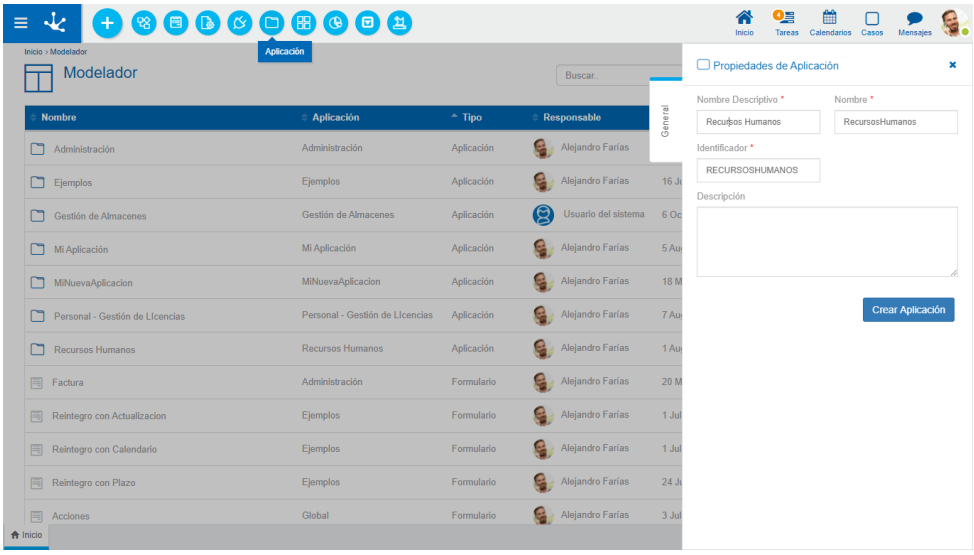
In this grid, the advanced rules used by the adapter are displayed and their definition can be shown from each line.

3.6.6. Modelado de Aplicaciones

Desde la grilla del modelador **Deyel** se puede crear una nueva aplicación o realizar la administración de las existentes.

Nueva Aplicación

Al crear una aplicación o modificarla se abre el panel de propiedades de la misma.



Un asterisco "" en la etiqueta indica que la propiedad es obligatoria.*

Propiedades

Nombre
Se usa para referirse a la aplicación dentro de reglas o como parámetro. Soporta multi-idioma.
Ejemplos: Contratos, Autoatencion.

Nombre Descriptivo
Se usa en la interfaz cuando se la referencia, en la grilla del modelador y en el árbol de objetos. Soporta multi-idioma.
Ejemplos: Sistema de Contratos, Autoatención de Empleados.

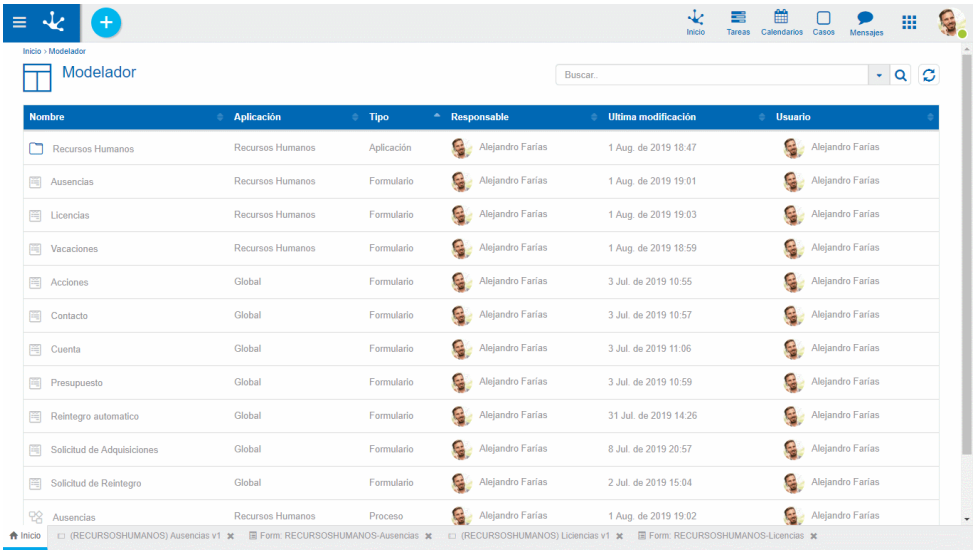
Descripción
Texto que define a la aplicación. Soporta multi-idioma.

Identificador
Se usa en el código de programación. No es multi-idioma. Al crear una nueva aplicación se genera automáticamente un identificador sugerido y el mismo puede modificarse, mientras que al modificar una aplicación, el identificador ya no puede cambiarse.
Ejemplos: CONTRACTS, SELFSERVICE.

Grilla

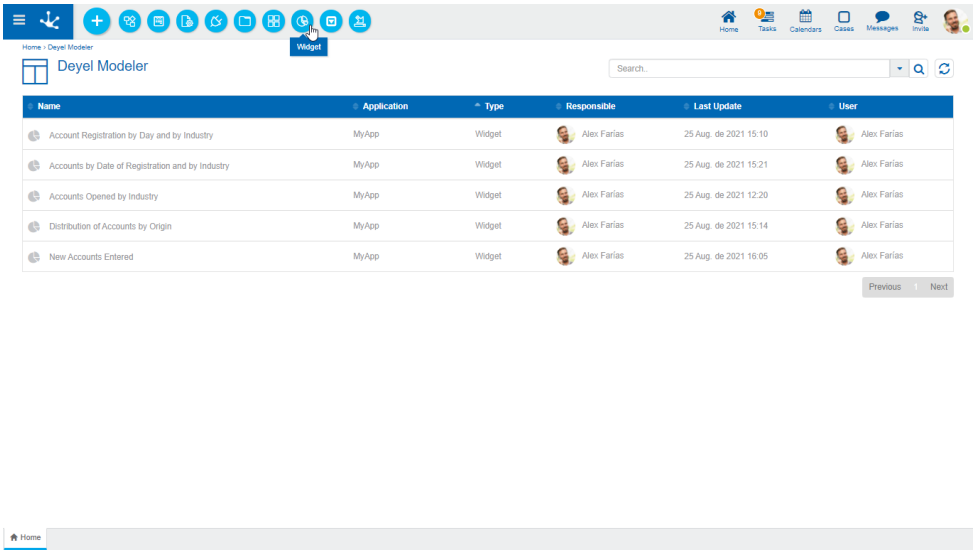
Se pueden agrupar diferentes objetos modelados bajo una aplicación, mediante la definición de la propiedad Aplicación en los respectivos paneles de propiedades formularios, procesos y reglas.

En la grilla del modelador**Deyel**, si se presiona el botón “Abrir” que se habilita al pasar el cursor sobre la línea de la aplicación seleccionada, se puede visualizar el conjunto de objetos asociados a dicha aplicación.



3.6.7. Widgets Modeling

The widgets modeling can be used to design and implement in an easy and intuitive way, the required widgets to display the relevant information for the user, as management issues, tasks and calendars.



The general characteristics of the widgets modeling and the elements that compose it are described in the topics:

- Modeling Facilities
- Widget Properties

A new widget can be created from the icon corresponding to thecontext menu.

- From the icon
- From the menu displayed with the option "Widget"

An existing widget is modified by selecting it from the grid with the "Open" button.

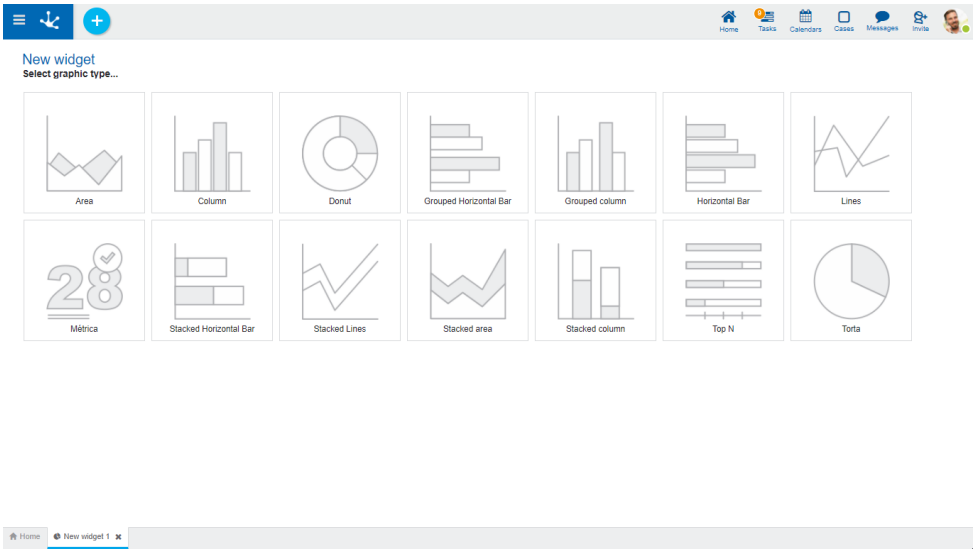
3.6.7.1. Modeling Facilities

New Widget

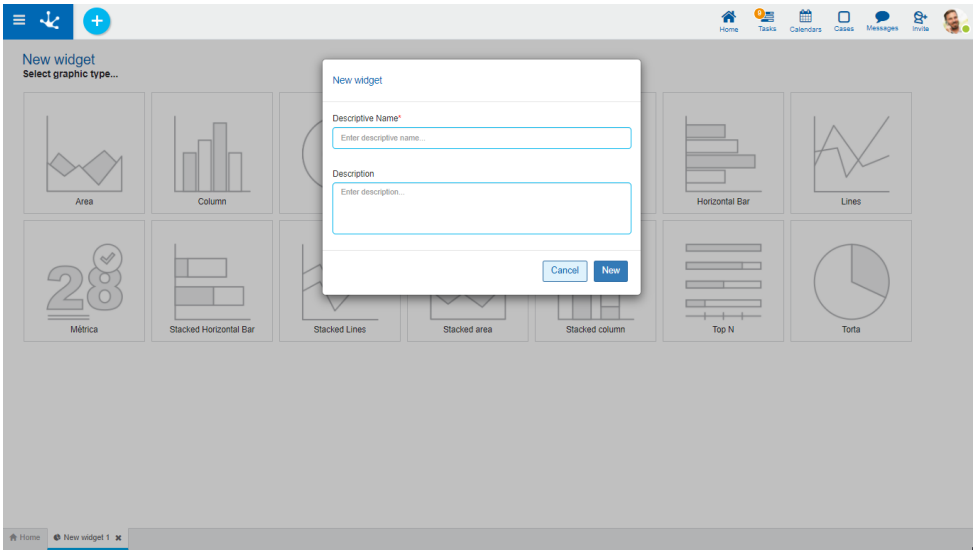
The TI modeler user can design a new widget, which after being published, is available to be used on a dashboard.

Steps for Creating a New Widget

Step 1
Select the graphic type.



Step 2
Assign the name, optionally write the description and press the "Create" button.

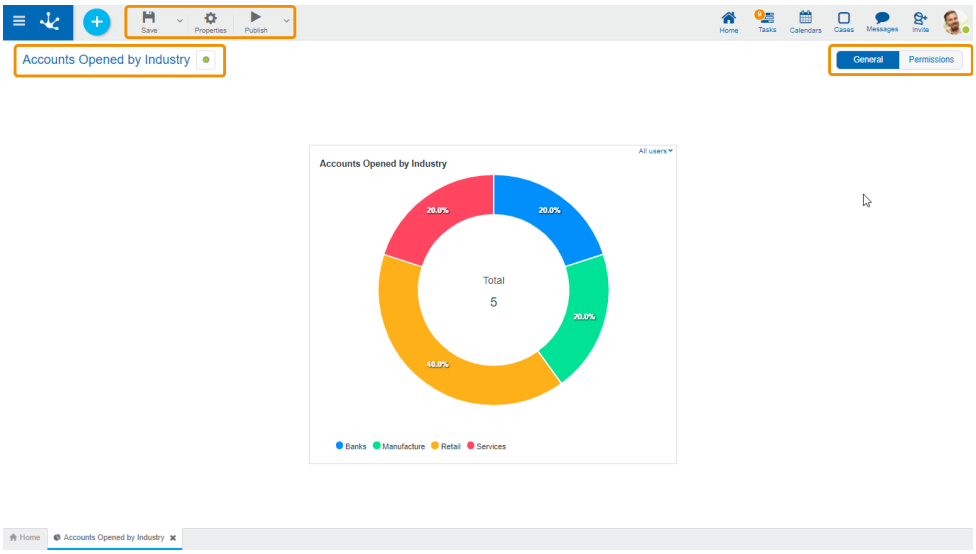


Step 3

Complete the widget using the sections of the workspace.

Workspace Sections

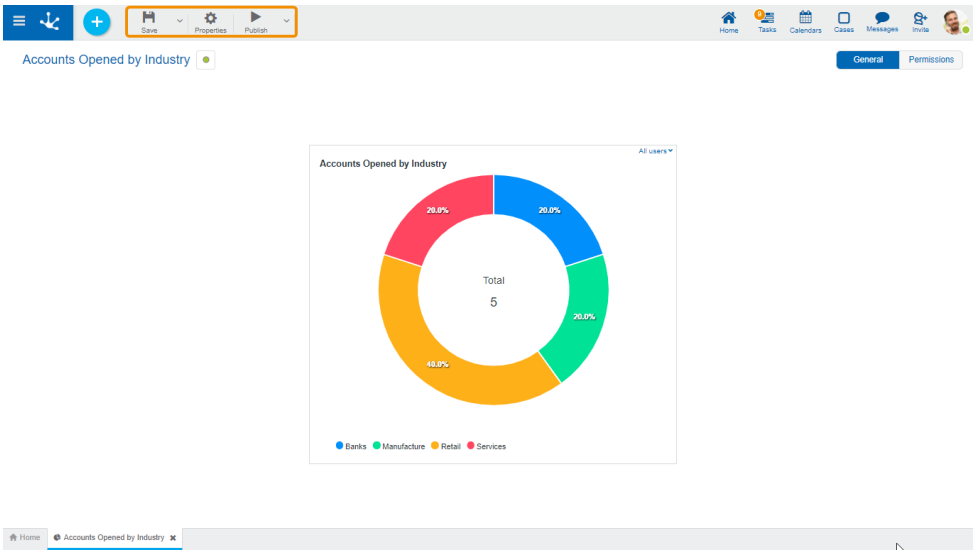
- Top Toolbar
- Widget Information
  - Name
  - State
- Design Options



3.6.7.1.1. Top Toolbar

This toolbar contains icons and submenus from which you can perform operations on the object. Depending on its state, some options may be disabled.

The bar is displayed on the toolbar of **Deyel**.



Save

This icon allows to save the object in the repository of **Deyel**, leaving its state as "Draft" or "Modified". If certain conditions are met, the modeler user receives a message indicating that the operation was performed correctly, otherwise they receive an explanatory message.

Main Conditions

- The application of the object must exist.
- The name must be unique in the application.
- The object must not be locked by another user.

To the right of the icon the option to open the submenu.

Properties

Open the widget properties panel.

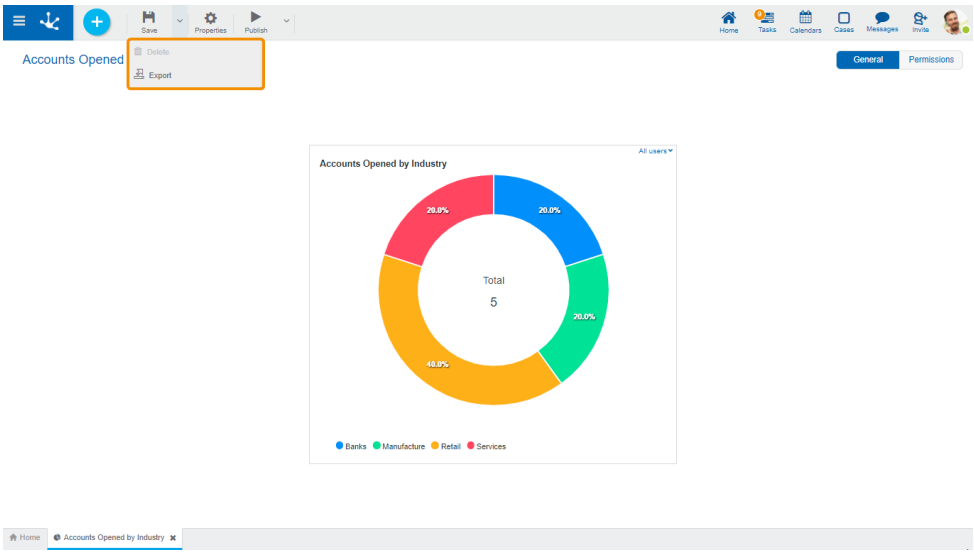
Publish

By means of this icon the object goes to "Published" state, after verifying a set of additional conditions to those of the "Save" operation. The modeling user receives the corresponding message, indicating the result of the operation.

To the right of the icon the option to open the submenu.

3.6.7.1.1.1. Save Submenu

This submenu is opened by pressing the icon immediately to the right of the icon corresponding to "Save".

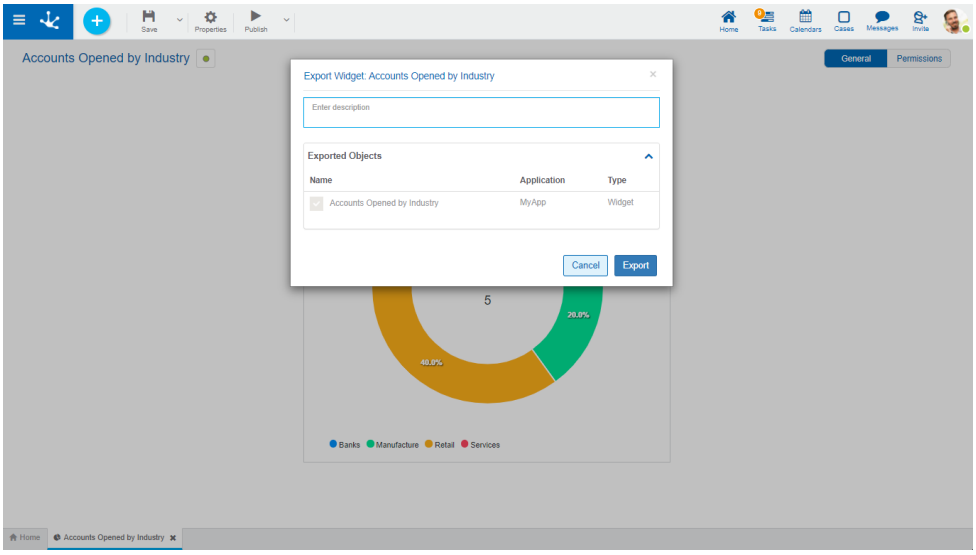


Delete

Allows to delete the object only if it is in "Draft"state, closing the tab in which it is located and deleting it from the modeler's grid.

Export

This icon opens a window for the user to select and confirm the export of the object.



Description

In this property a text explaining the reason for the operation can be entered.
This text can be modified upon import and is displayed in the description column of the export record.

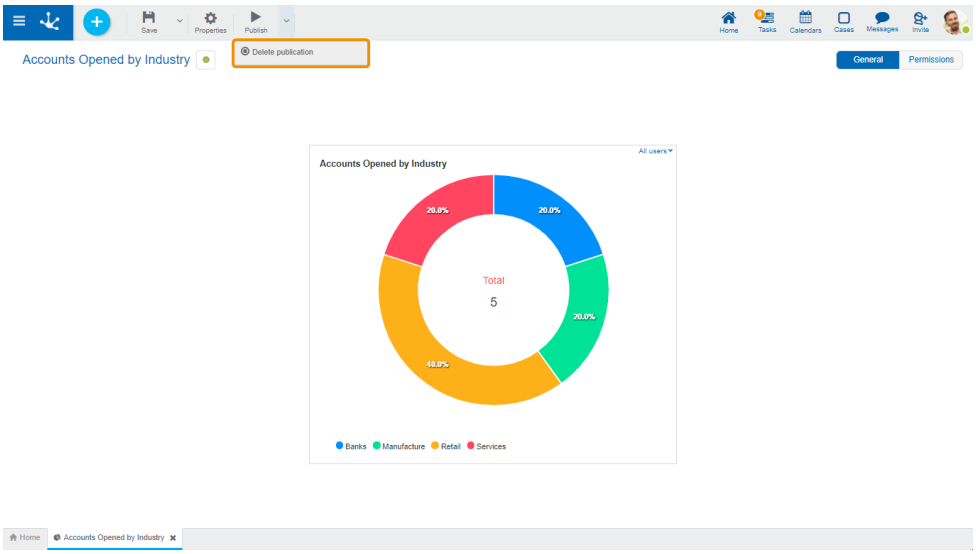
Exported objects

By expanding the container, the name of the object being exported is displayed.

You must press the “Cancel” button to leave the export without effect or the “Export” button to finish the operation.

3.6.7.1.1.2. Submenu Publish

This submenu can be opened by pressing the icon immediately to the right of the icon corresponding to "Publish".

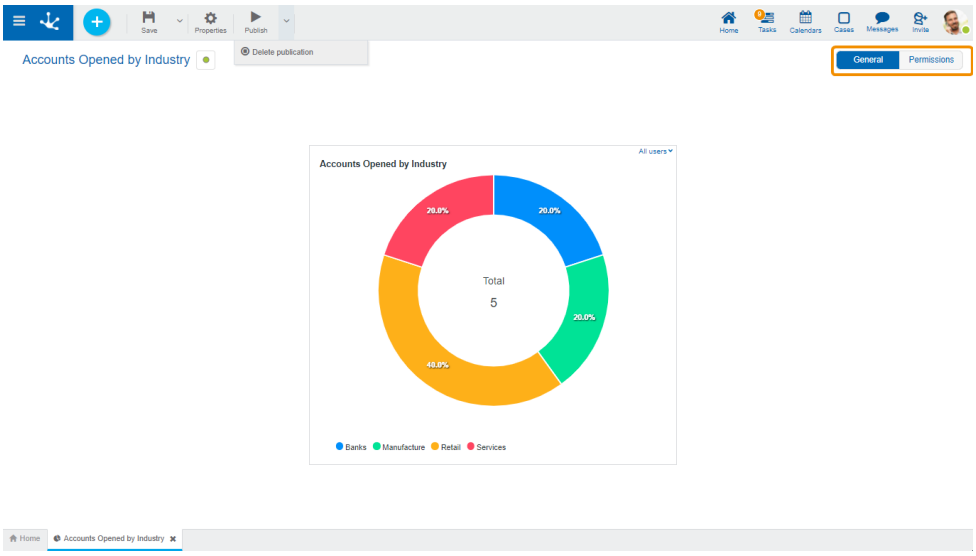


Delete Publication

Allows to leave the widget unavailable for use by returning it to the "Draft" state.

3.6.7.1.2. Design Options

These options are presented on the upper right margin.



- General
  From this design option the widget is visualized and when hovering over it the data values that compose it can be seen.

- Permissions

3.6.7.1.2.1. Permissions

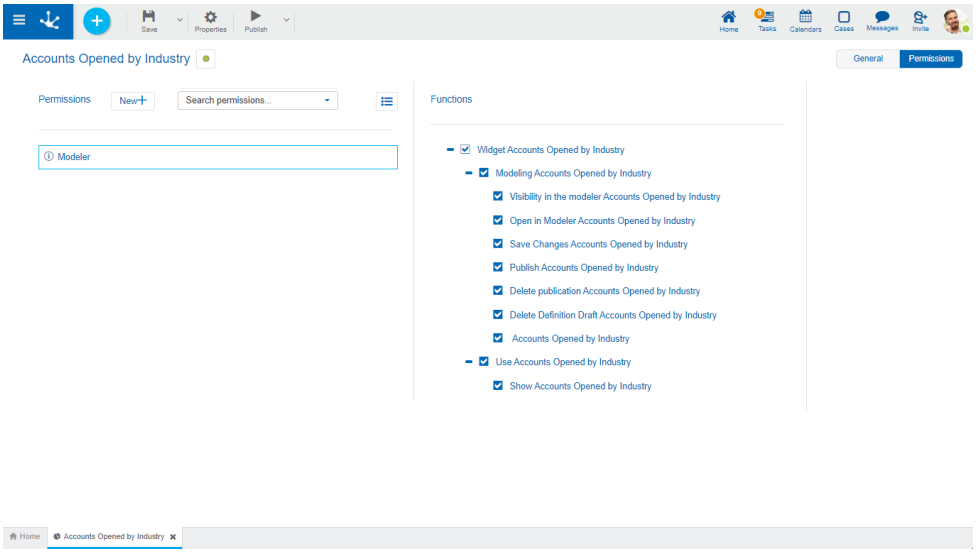
Allows to assign the security functions for use and modeling of the object to the existing permissions or to new permissions that the user creates, without having to go to the permission settings option from the menu.

Sections

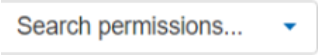
- Permissions: Permissions to which object functions are assigned.
- Security Functions: Represents the total set of security functions, modeling and use of the object. Those that are marked are the ones included in the selected permission.

By default, all security functions for a new object are assigned to the permission Modeler.

Users who are assigned the permissions have access to the functions included in it.



Opens a panel to create a new permission and once created, the security functions included in it must be selected .



Allows to select a permission from a list and enables the input of characters to filter the values in the list.

Opens the wizard to select a permission and once chosen, the necessary object security functions must be checked.

To unrelate a permission from the object, hover the cursor over the permission and press the icon ✖. If there are functions selected for that permission, they must be unchecked in order to delete it.

Security Functions to Model the Widget

- Visibility in the modeler: Allows to display the object in the Deyel modeler.
- Open in modeler: Allows to show the object from the Deyel modeler.
- Save changes: Enables the operation of saving modifications made to the object.
- Publish: Enables the operation of publishing the object leaving its state as "Published".
- Delete publication: Enables the operation of deleting the object publication leaving its state as "Draft".
- Delete draft definition: Enables the operation of deleting the object.
- Export widget: Enables the operation to export the object from the save submenu of the upper toolbar.

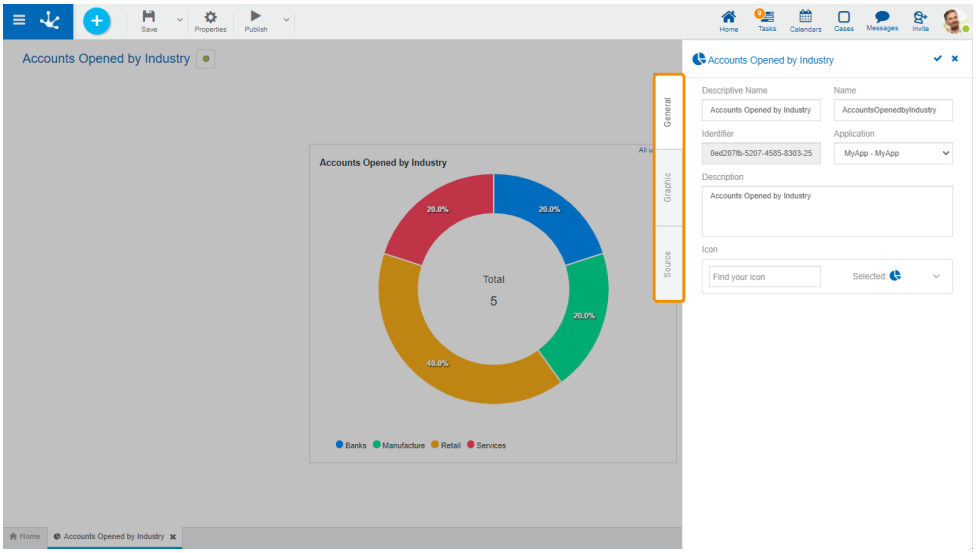
Security Functions to Use the Widget

Show: Enables the operation of showing the object.

3.6.7.2. Widget Properties

The properties of the widgets can be entered both at the time of their creation, and when modifying an existing one.

Entering the widget properties panel is done using the icon which is in the top toolbar.

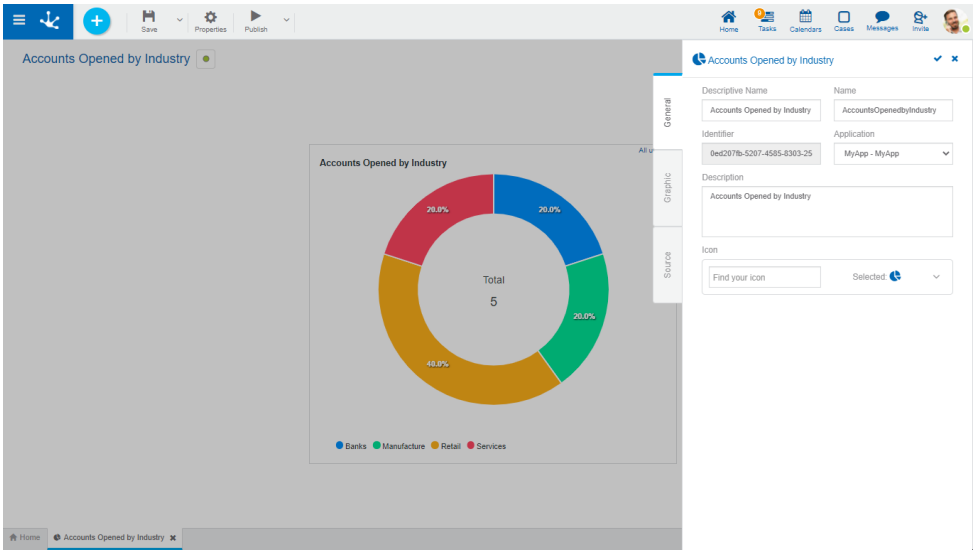


Tabs

- General
- Graphic
- Source

3.6.7.2.1. General

The properties panel is displayed on the right side of the widget modeler, where the first tab corresponds to general information.



Properties

Descriptive Name
It is the name used by users to reference the widget, for example in the modeler's grid and in the dashboards. It is required.

Name
It is internally used to reference the widget. It does not allow blanks or special characters. It is unique and required.

Identifier
Uniquely identifies the widget. It is automatically generated.

Application
It is the name of the application to which the widget belongs. It is not required, just as it happens with the rest of the modeled objects.

Description
Text that describes the topic the widget allows to analyze, when incorporated into a dashboard.

Icon
Allows to incorporate an icon that visually identifies the widget when the operation is made Add widget, from the use of the dashboard.
The list of available icons is displayed to select one of them. The list is reduced as the name is written on the text "Find your icon".

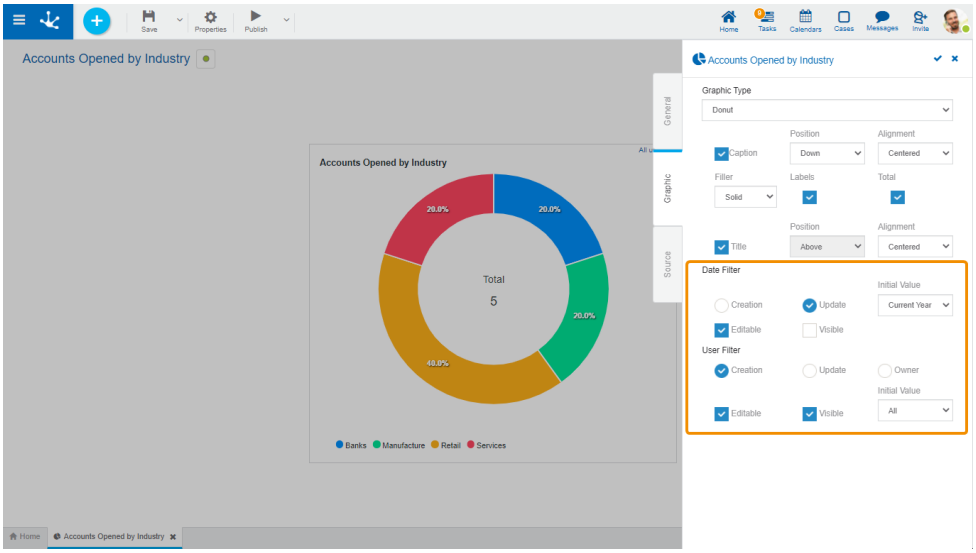
Actions

The icon ✔ is used to confirm the modifications made in the properties panel.

The icon ✖ is used to close the properties panel, if it was not previously saved, changes are discarded.

3.6.7.2.2. Graphic

The second tab of the side panel corresponds to the charts information. Allows to do the management of the properties of each chart type.



Properties

In addition to these shared properties, each type of widget has specific properties, which are detailed for each chart.

- Metrics
- Areas
- Lines
- Bars and Columns
- Pies and Donuts

Date Filter

Creation

Indicates that the date filter is applied on the creation date.

Update

Indicates that the date filter is applied on the modification date.

Editable

Indicates if the user can modify the filter conditions when using the widget from the dashboard, only if they have permission to do it.

If the user makes modifications and saves the dashboard, the new filter conditions established are maintained.

Visible

Indicates whether the filter is visualized or not when using the widget from the dashboard.

Initial Value

Indicates the value displayed from the dashboard by default.

Possible Values:

- Today
- From
- Last 7 days
- Current month

- Current year
- Last month
- Last year

User Filter

Creation

Indicates that the user filter is applied on the creation user.

Update

Indicates that the user filter is applied on the modification user.

Owner

Indicates that the user filter is applied on the instance owner.

Editable

Indicates if the user can modify the filter conditions when using the widget from the dashboard, only if they have permission to do it.

If the user makes modifications and saves the dashboard, the new filter conditions established are maintained.

Visible

Indicates whether the filter is visualized or not when using the widget from the dashboard.

Initial Value

Indicates the value displayed from the dashboard by default.

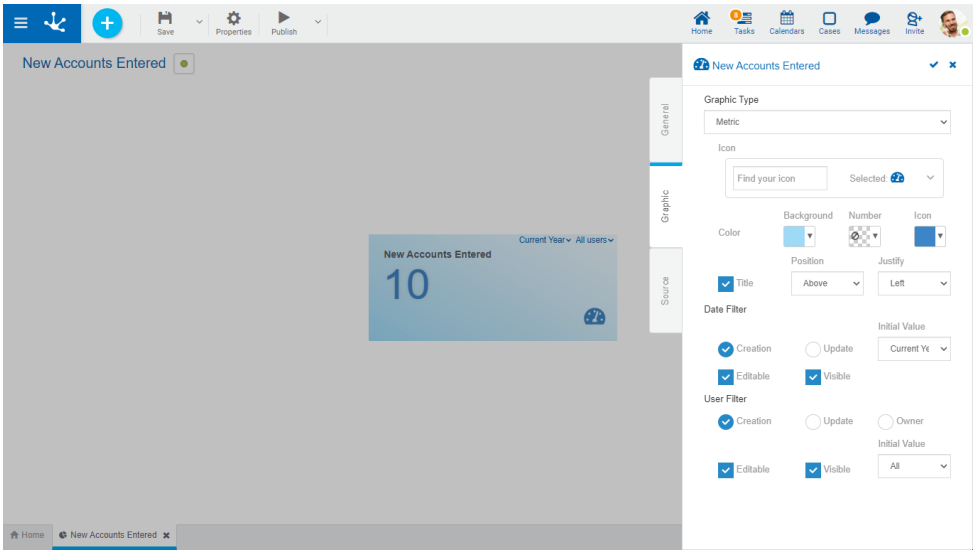
Possible Values:

- Current user
- All
- My Team

> *My team is made up by all of the users that belong to the organizational unit for which the logged in user is responsible and all users of a role for which the logged in user is responsible.*

3.6.7.2.2.1. Metrics

To theproperties shardby all the widgets, those specific to metrics are added.



Properties

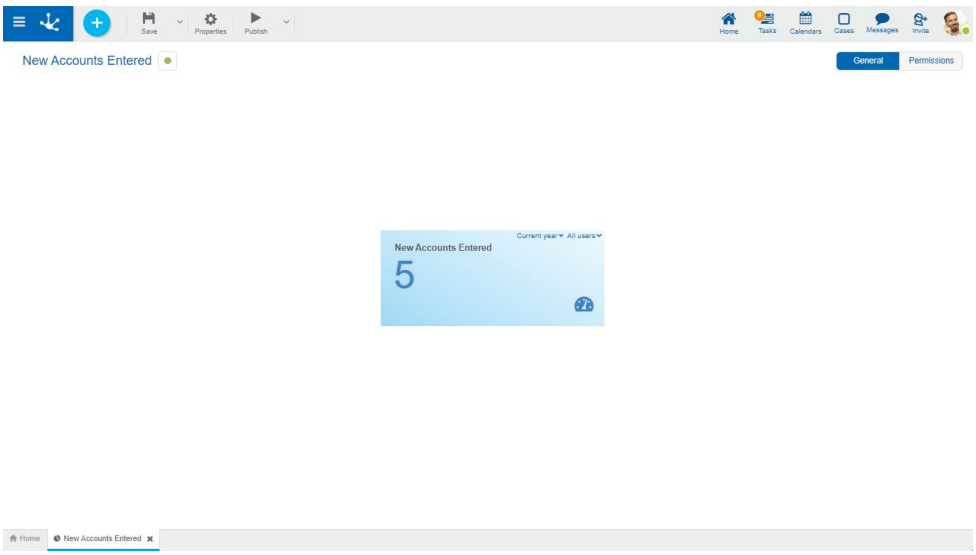
Icon
Allows to incorporate an image displayed as a background, inside the widget. It is incorporated in the same way as the icon of the "General" tab.

Color
Background
Allows to modify the background color of the widget, selecting it from the color palette.
Number
Allows to modify the value color of the widget, selecting it from the color palette.
Icon
Allows to modify the icon color, selecting it from the color palette.

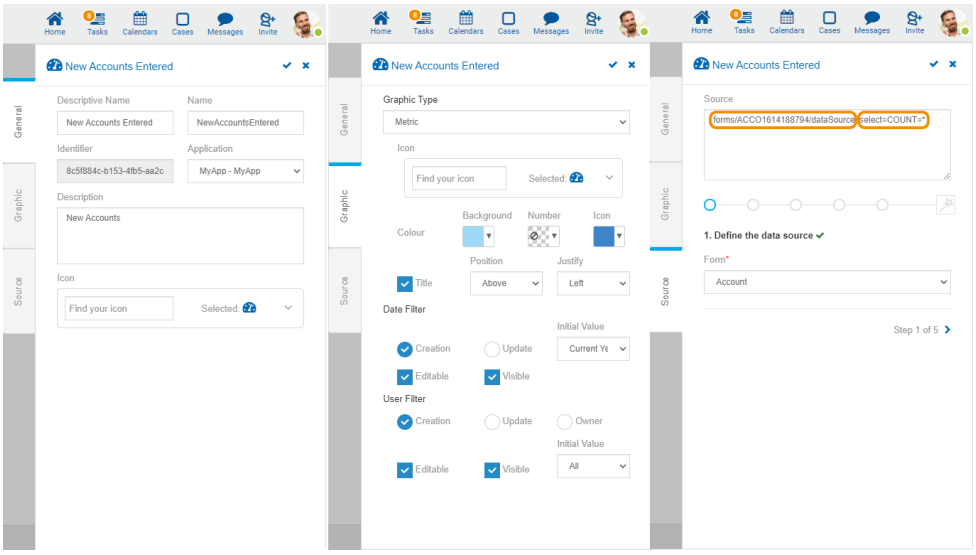
Title
If this property is selected, when using the widget on a dashboard, the descriptive name is displayed. In this case it must also indicate:
Position
Allows to locate the title in the upper or lower sector of the widget area.
Justify
Allows to position the title to the right, center, or left, within the widget area.

Example of Use

This metric chart shows the number of accounts entered during a selected period and for the users indicated in the filter.



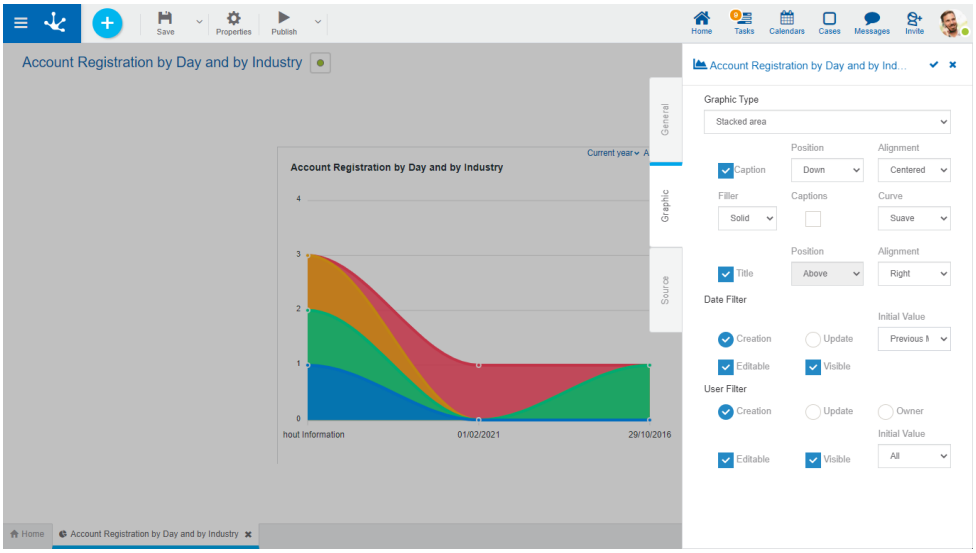
The source code to model this chart is detailed.



In the "Source" tab, each orange box represents the code for a particular step.

3.6.7.2.2.2. Areas

Widgets of area type, comprehend the area and stacked area charts. To theproperties sharedby all the widgets, those specific to areas are added.



Properties

Legend

Indicates the incorporation of the data series description.
If this property is selected, it must also indicate:

Position
Select above or below.
Justify
If above or below position is selected, it allows to indicate right, center or left alignment.

Filler
Allows to highlight the area of data series in a gradient or solid way.

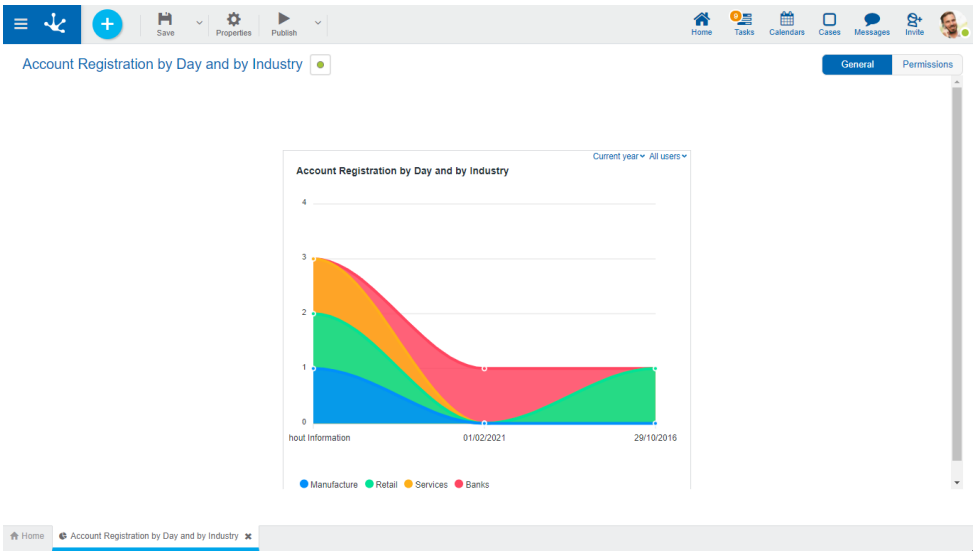
Labels
Allows to indicate whether the labels corresponding to the data series are displayed.

Curve
Indicates whether data lines are displayed as smooth, straight, or stepwise curve.

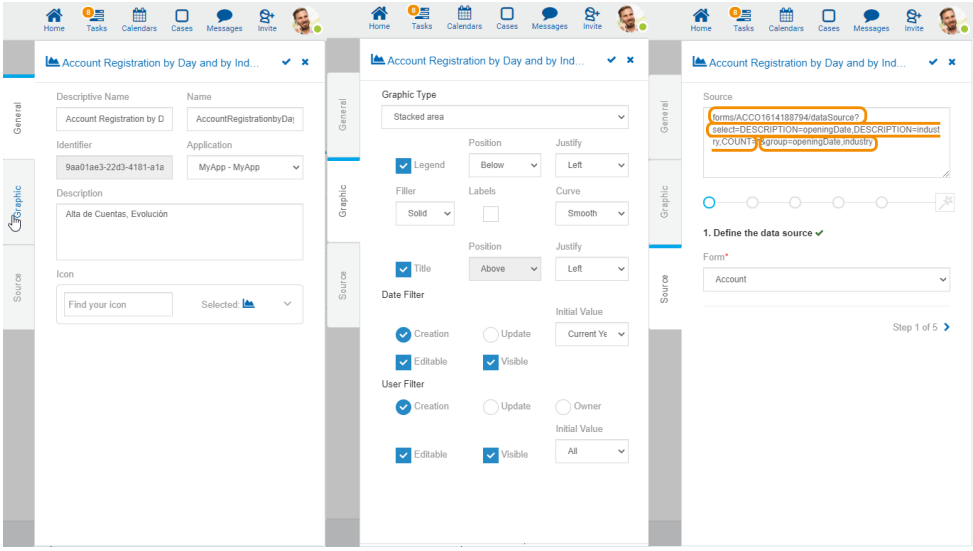
Title
If this property is selected, when using the widget on a dashboard, the descriptive name is displayed. In this case it must also indicate:
Position
Allows to locate the title in the upper or lower sector of the widget area.
Justify
Allows to position the title to the right, center, or left, within the widget area.

Example of Use

In this stacked area chart the number of accounts entered per day and by industry is displayed, for a selected period and for the users indicated in the filter.
On the horizontal axis are represented the registration dates and on the vertical axis the sum of the registrations made.
The percentage of each industry is given by the colored surface, and the grouping of all surfaces represents the total number of discharges in the day.



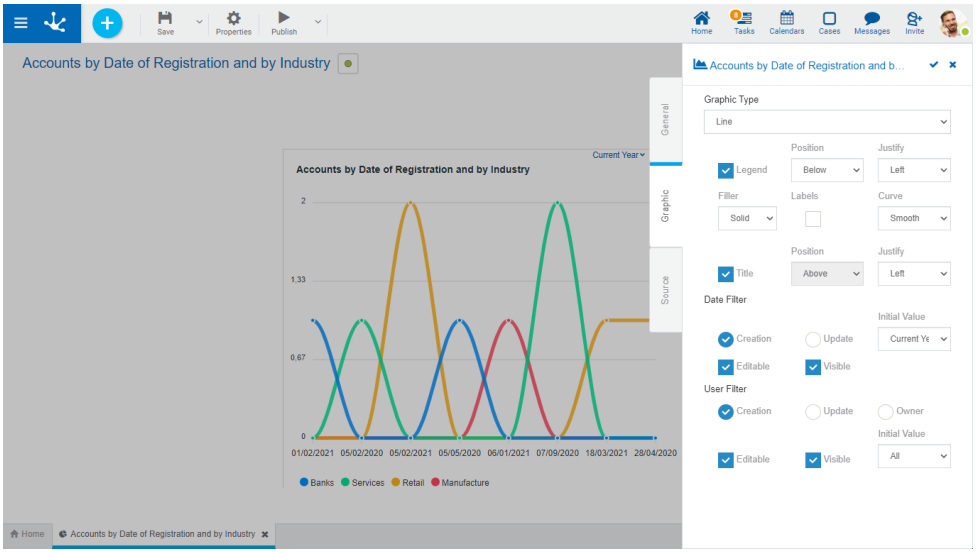
The source code to model this chart is detailed.



In the "Source" tab, each orange box represents the code for a particular step.

3.6.7.2.2.3. Lines

Widgets of line type comprehend line and stacked line charts. To theproperties sharedby all the widgets, those specific to lines are added.



Properties

Legend
Indicates the incorporation of the data series description.
If this property is selected, it must also indicate:

Position
Select above or below.

Justify
If above or below position is selected, it allows to indicate right, center or left alignment.

Filler
Allows to highlight the area of data series in a gradient or solid way.

Labels
Allows to indicate whether the labels corresponding to the data series are displayed.

Curve
Indicates whether data lines are displayed as smooth, straight, or stepwise curve.

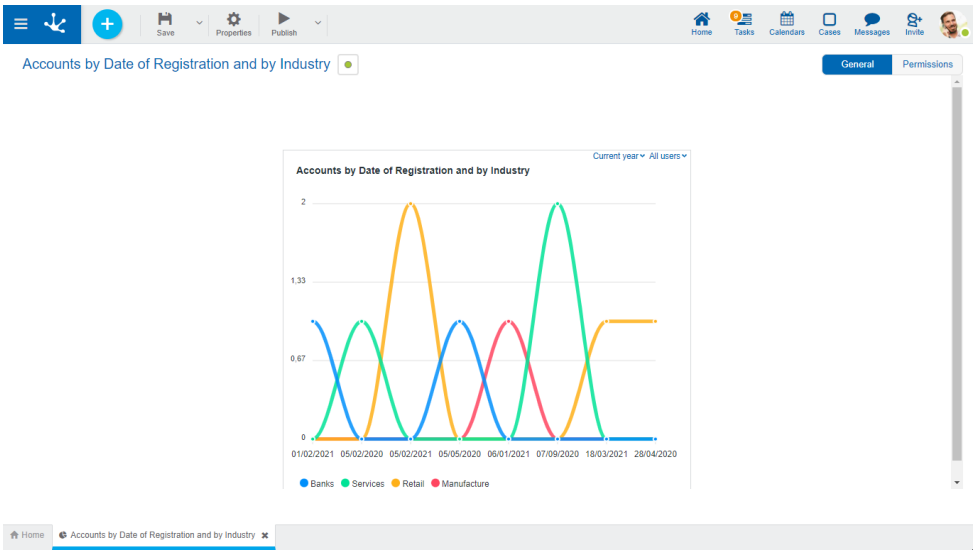
Title
If this property is selected, when using the widget on a dashboard, the descriptive name is displayed.
In this case it must also indicate:

Position
Allows to locate the title in the upper or lower sector of the widget area.

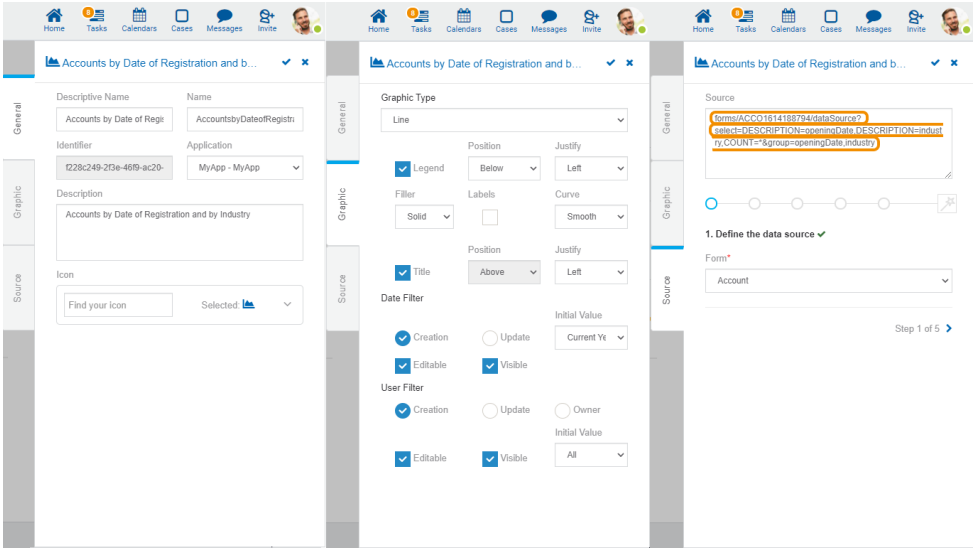
Justify
Allows to position the title to the right, center, or left, within the widget area.

Example of Use

In this line chart the number of CRM accounts entered per day and by industry is displayed, for a selected period and for the users indicated in the filter.
On the horizontal axis are represented the registration dates and on the vertical axis the sum of the registrations made.
The height of the line for each industry indicates the total number of discharges per day.



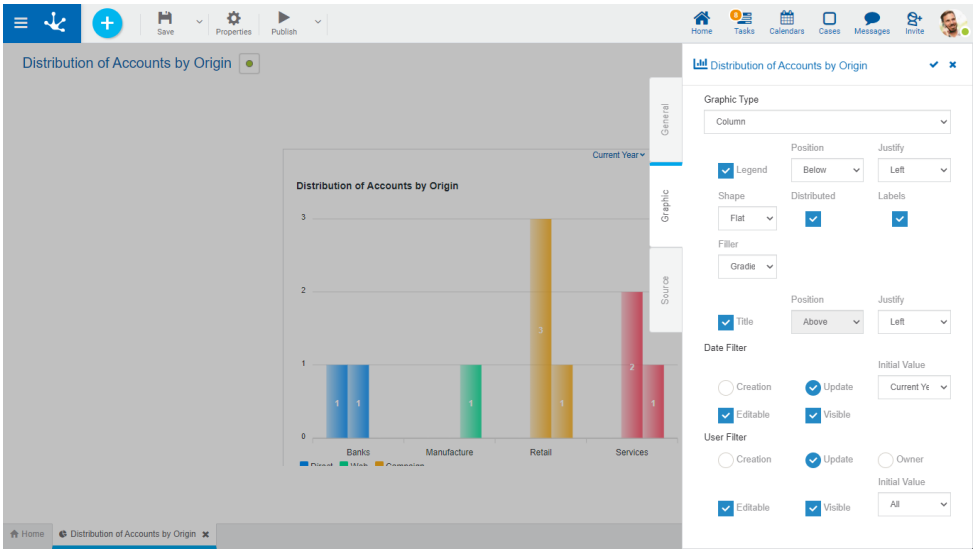
The source code to model this chart is detailed.



In the "Source" tab, each orange box represents the code for a particular step.

3.6.7.2.2.4. Bars and Columns

The bar and column widgets, comprehend the charts of type, bar, grouped bar and stacked bar, column, grouped column and stacked column and N Top. To theproperties shredby all the widgets, those specific to bars and columns are added.



Properties

Legend
Indicates the incorporation of the data series description.
If this property is selected, it must also indicate:

Position
Select above or below.

Justify
If above or below position is selected, it allows to indicate right, center or left alignment.

Shape
It allows to select whether the figure has a flat or curved edge.

Distributed
Indicates whether the series are grouped or interleaved.

Labels
Allows to indicate whether the labels corresponding to the data series are displayed.

Filler
Allows to highlight the area of data series in a gradient or solid way.

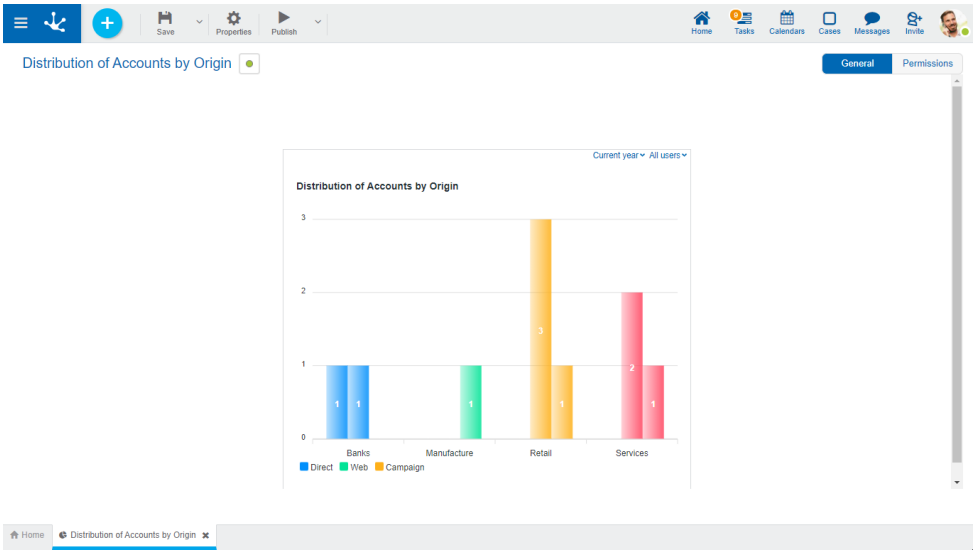
Title
If this property is selected, when using the widget on a dashboard, the descriptive name is displayed. In this case it must also indicate:

Position
Allows to locate the title in the upper or lower sector of the widget area.

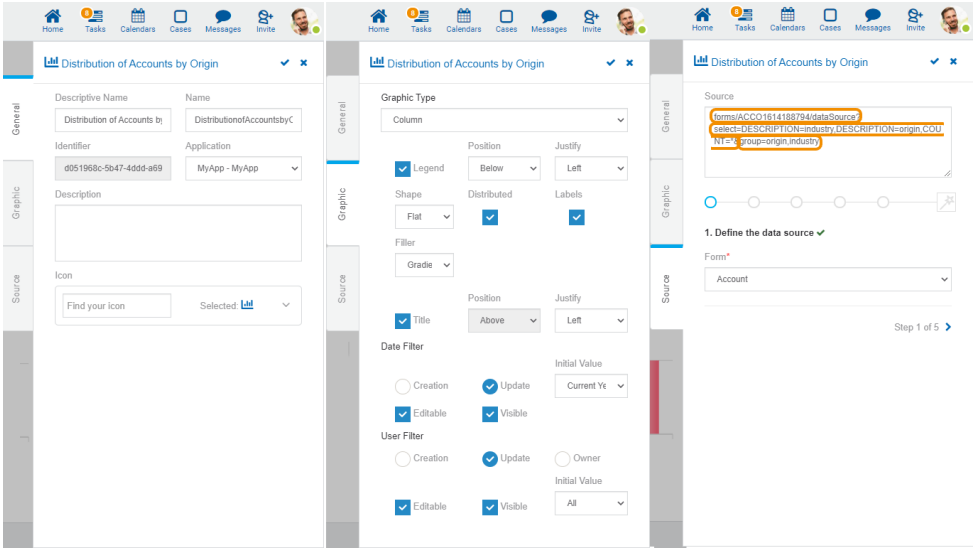
Justify
Allows to position the title to the right, center, or left, within the widget area.

Example of Use

In this column or bar chart the series type of industry and origin are displayed at the same time, so as to make a comparative analysis of the total sum of CRM accounts.



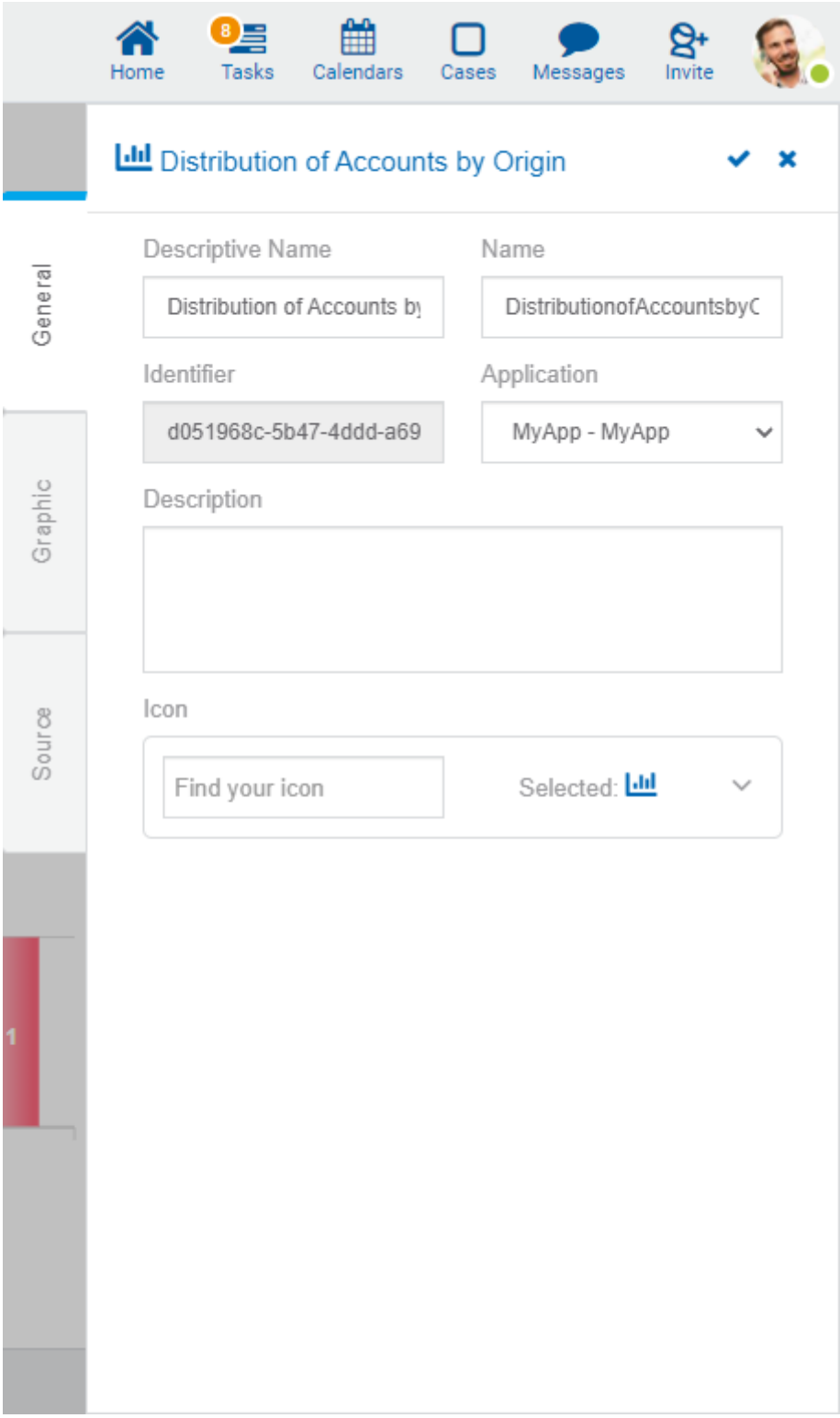
The source code to model this chart is detailed.



In the "Source" tab, each orange box represents the code for a particular step.

3.6.7.2.2.5. Pies and Donuts

To theproperties sharedby all the widgets, those specific to pies and donuts are added.



Properties

Legend
Indicates the entry of data series description.
If this property is selected, also indicate:
Position
Select above, below, left, right.

Justify
If above or below position is selected, it allows to justify right, center or left.
If right or left position is selected, the legend is displayed top right or left as indicated.

Filler
Allows to highlight the area of data series in a gradient or solid way.

Labels
Allows to indicate whether the labels corresponding to the data series are displayed.

Total
Indicates whether or not the total of all series is displayed, available for donuts.

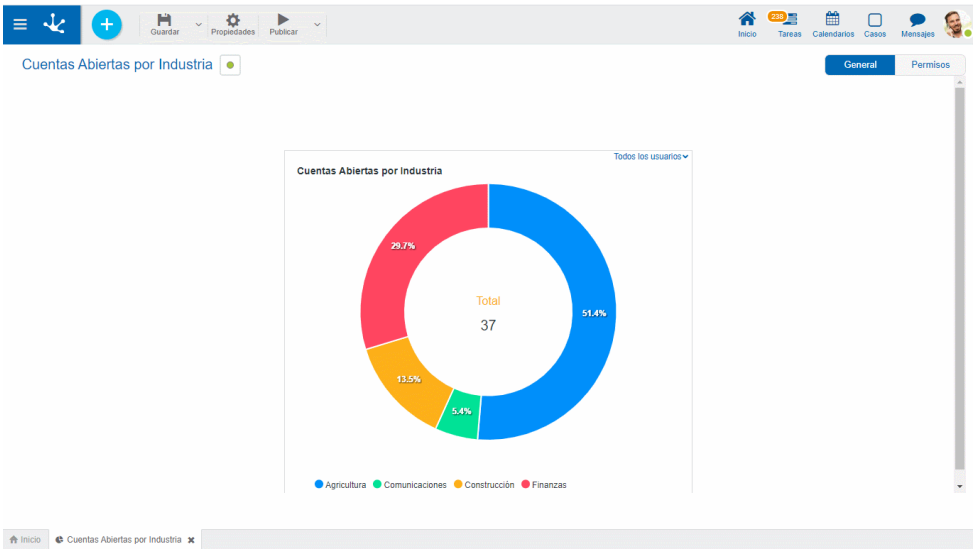
Title
If this property is selected, when using the widget on a dashboard, the descriptive name is displayed.
In this case it must also indicate:

Position
Allows to locate the title in the upper or lower sector of the widget area.

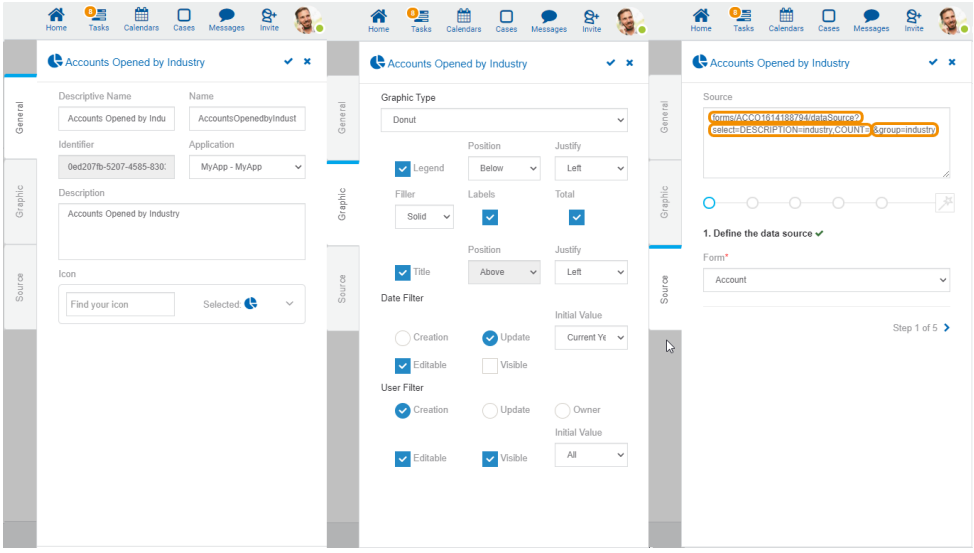
Justify
Allows to position the title to the right, center, or left, within the widget area.

Example of Use

In this donut chart, the percentage of CRM accounts is displayed, grouped by type of industry.



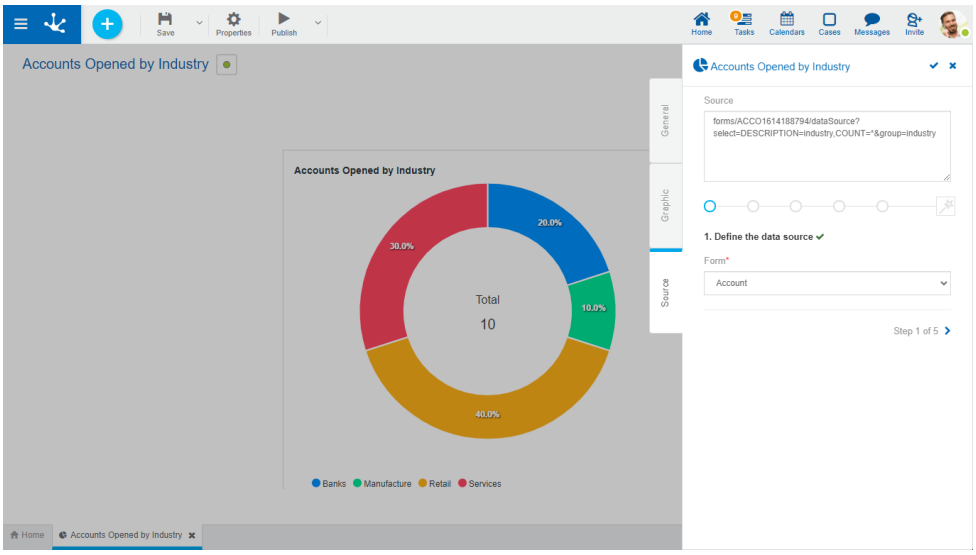
The source code to model this chart is detailed.



In the "Source" tab, each orange box represents the code for a particular step.

3.6.7.2.3. Source

The third tab of the side panel corresponds to the Rest API for obtaining the data of the selected graphic and the wizard for its automatic generation, following a series of steps.



Properties

Source

Corresponds to the URI on the Rest API to obtain the data displayed on the selected graphic. It can be codified or automatically generated through the wizard steps.

Wizard for Widgets Modeling

Allows to automatically generate the URI of the Rest API in the area of the source following the sequence of steps represented by the lines of circles.

Each step is identified by a number and accompanied by an informative paragraph, represented by the "i" icon.

The required steps are represented by the red circle and once they are completed they are displayed in light blue, indicating that the part corresponding to the source code has been completed. The circles corresponding to the steps not yet completed are displayed in grey.

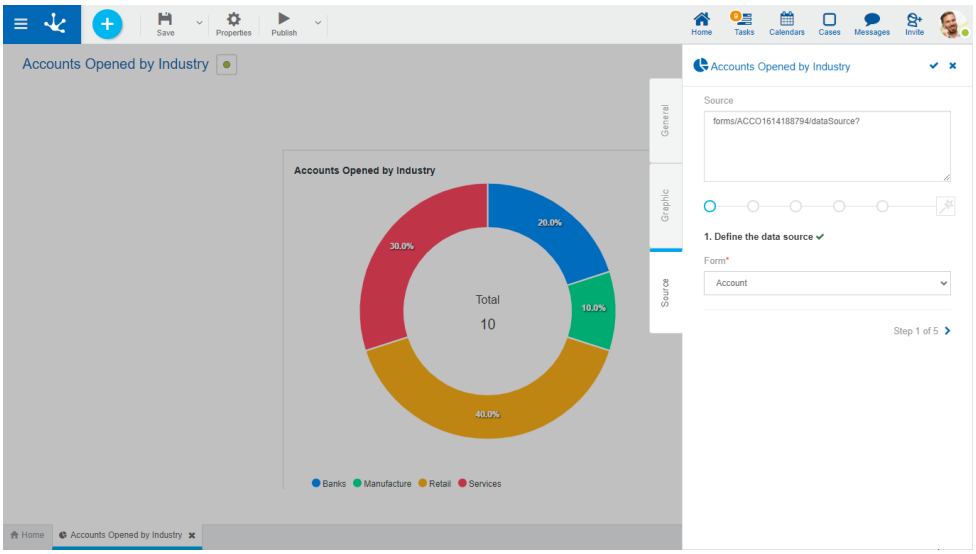
You must respect the pre-established order and advance by selecting the next circle or by pressing the paging icon, located at the bottom of the wizard. You can navigate between the completed steps by pressing the circles or by using the paging icon.

Wizard Steps for Widgets Modeling

Step 1: Define the data source

This step is required.

Form
Allows to select the entity from which the data to be displayed on the graphic will be taken. Once it is selected there is a check mark to the right of the step description, indicating that it was completed successfully.
When finishing the step, the generated code is displayed on the source.

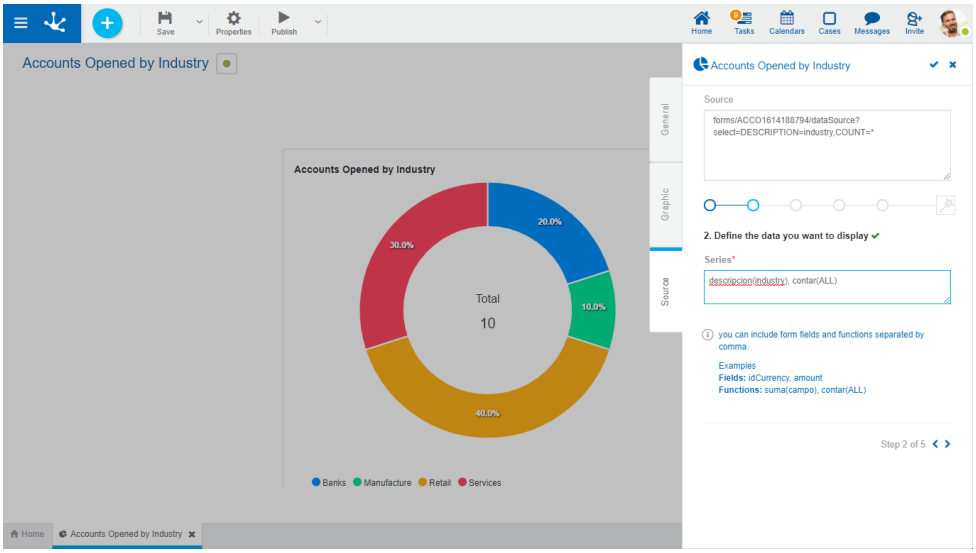


Step 2: Define the data you want to display.

This step is required.

Series
It must be indicated which elements are going to be displayed on the graphic, being able to use the help of the wizard. Once the series are entered, there is a check mark to the right of the step descrip-

tion, indicating that it was completed successfully. When completing the series, the added code is displayed on the source.

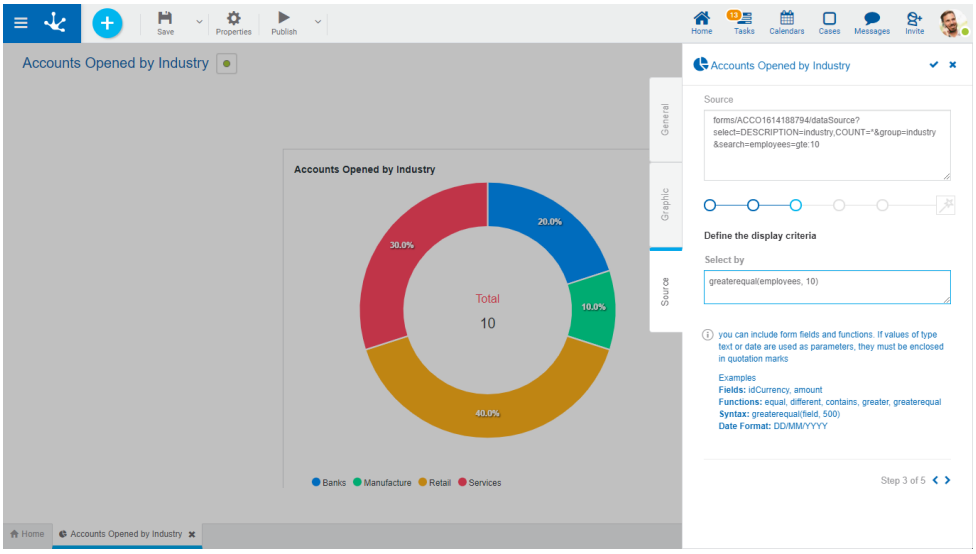


Step 3: Define display criteria.

This step is optional.

Select by

Allows to indicate with the help of the wizard the elements that must meet a condition for the data to be displayed on the graphic.
When completing this step, the added code is displayed on the source.

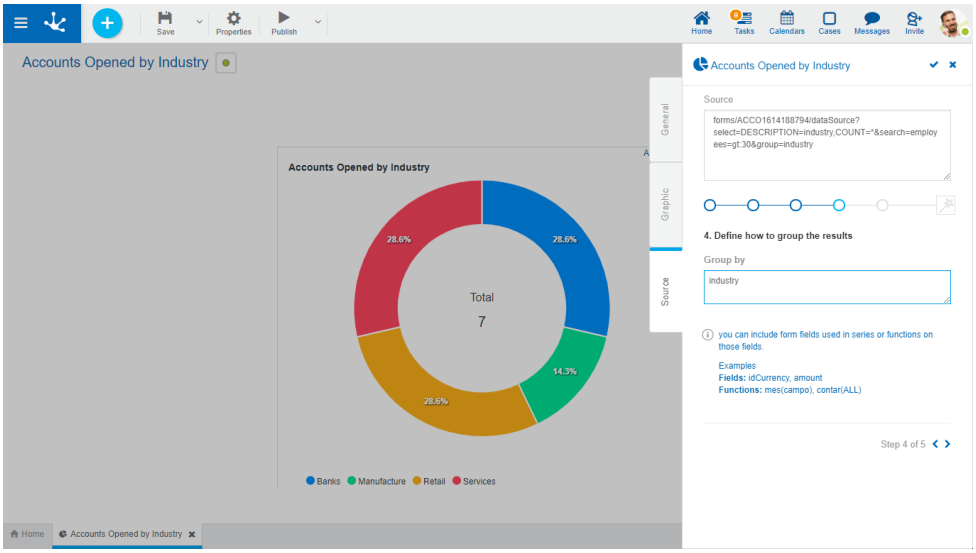


Step 4: Define how to group the results.

This step is optional.

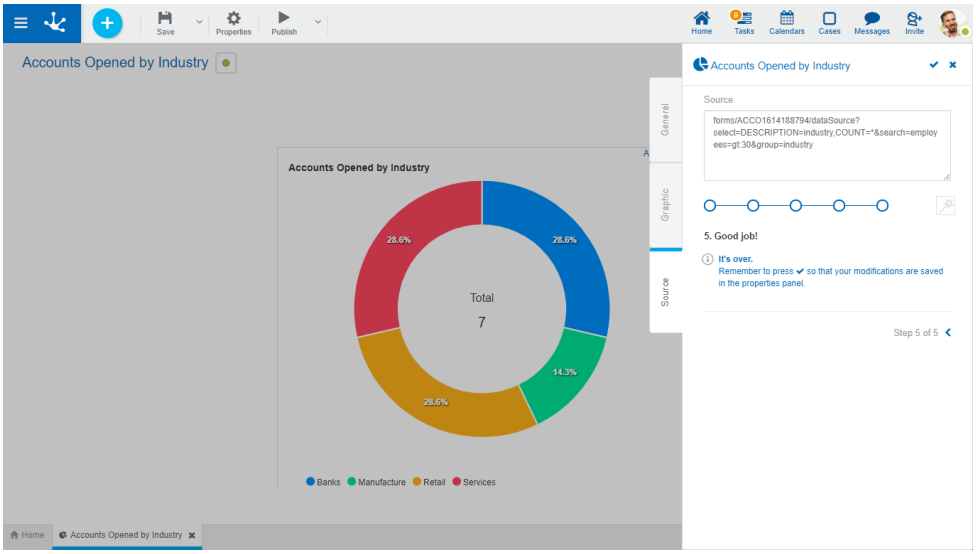
Group by

Allows to indicate with the help of the wizard the elements of the series, by which the results will be grouped.
When completing this step, the added code is displayed on the source.



Step 5: Good job.

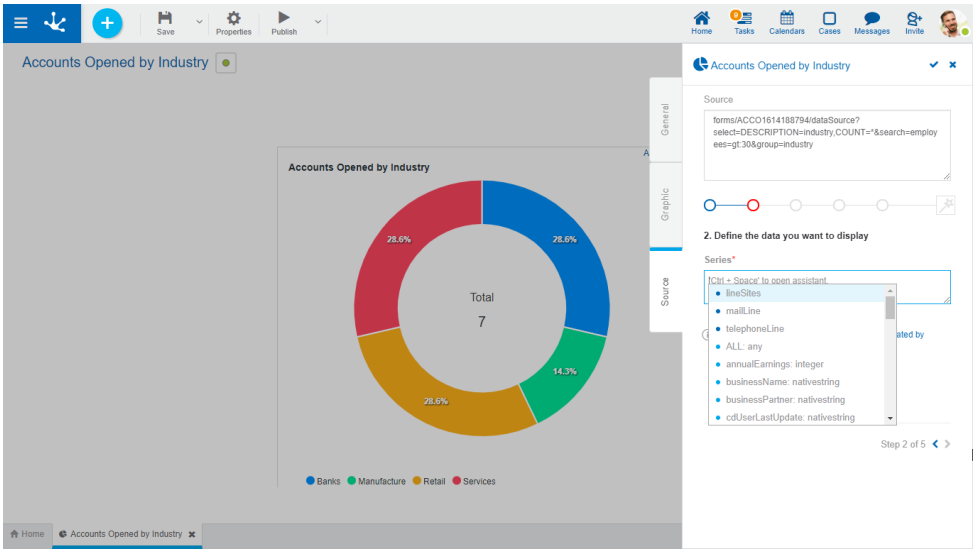
Informs that all the steps have been completed, and reminds that changes made on the indicator properties panel must be saved.



3.6.7.2.3.1. Wizard

In the editing area of each of the steps you can use a wizard for the automatic generation of the source.

Such wizard is activated pressing the "Ctrl + Space" keys, using the functionality of autocompleting, that is, as text is entered in the wizard, only are shown the elements which names start with the text the user writes.
In case of not using autocomplete the complete list of elements available for the step you are working on opens.



3.6.7.2.3.2. Elements Used in the Wizard

The steps can be composed by different types of elements, separated by a ".".

- Form field
  Form fields are used identifying them by the property Name. In the wizard they are displayed with ● intheir left.

- Iterative containers
  The property Name of the container is shown. Whenever an iterative container is used, then the field separated by a "." must be indicated. In the wizard they are displayed with ● in their left.
  Example: Items.Number

- Functions
  In the wizard they are displayed with ● in their left.

In every case it is indicated on the right, the data type they represent and in the case of the functions, the data type they receive as a parameter.

In case of a syntax error, a message is displayed indicating the problem.

Wizard Functions

These functions are displayed in the wizard, for each of them it is indicated a brief description of its functionality and in which steps of the wizard it is suggested.

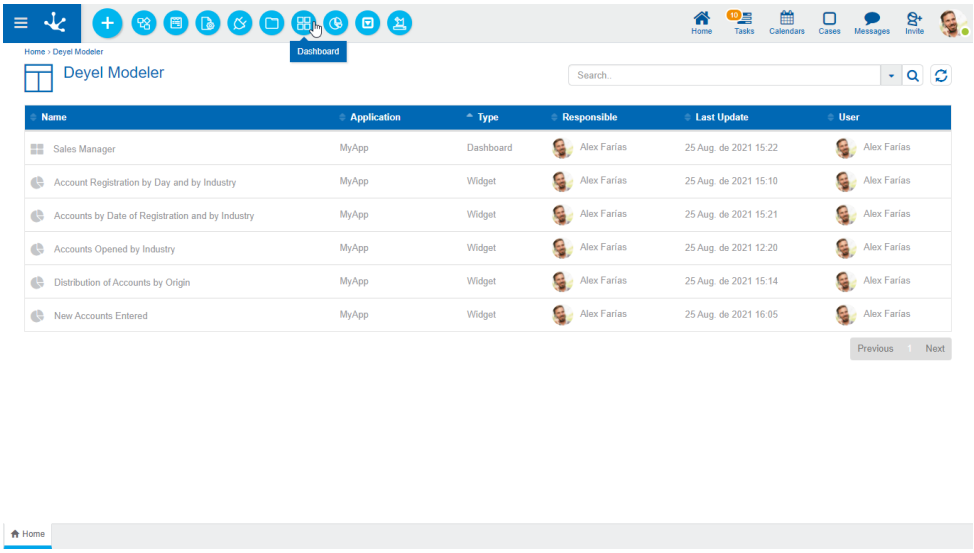
Function	Description	Wizard (Step)	Return Type
count(field)	Counts the occurrences of a certain field.	Series and group	Numeric
descrip-tion(field)	Returns the field description.	Series	Text
max(field)	Returns the maximum value.	Series and group	Field type used
month(field)	Month of the year. It is used for date type fields.	Series	Numeric
min(field)	Returns the minimum value.	Series and group	Field type used
average(field)	Returns the average value.	Series and group	Numeric
sum(field)	Sums the values entered from the field.	Series and group	Numeric Deci-mal
equal(field, field or value)	Compares by equality the first field with the other field or value.	Display criteria	Boolean
diffe-rent(field, field or value)	Compares by inequality the first field with the other field or value.	Display criteria	Boolean
contains(field, value)	Checks if the field contains a determined value.	Display criteria	Boolean
starts-With(field, value)	Checks if the field starts with a determined value.	Display criteria	Boolean
doesNotStar-tWith(field, value)	Checks if the field does NOT start with a de-termined value.	Display criteria	Boolean
greater(field, field or value)	Checks if the first field is greater than the second field or value entered.	Display criteria	Boolean
greaterE-qual(field, field or value)	Checks if the first field is greater than or equal to the second field or value entered.	Display criteria	Boolean

Function	Description	Wizard (Step)	Return Type
lower(field, field or value)	Checks if the first field is lower than the second field or value entered.	Display criteria	Boolean
lowerEqual(field, field or value)	Checks if the first field is lower than or equal to the second field or value entered.	Display criteria	Boolean
included(field, value1, value2, ...)	Checks if the field value exists in the value list separated by a comma.	Display criteria	Boolean
notIncluded(field, value1, value2, ...)	Checks if the field value does not exists in the value list separated by a comma.	Display criteria	Boolean

3.6.8. Dashboards Modeling

The dashboard modeler is a tool that allows to model in an easy and intuitive way the applications dashboards and to publish them. The indicators included in the dashboards are modeled with the indicator modeler.

From the portal, each business user can design its own dashboard, with the personalized information and their use preference, or they can use the dashboards created by the IT modelers.



The general characteristics of the dashboards modeling and the elements that compose it are described in the topics:

- Modeling Facilities
- Dashboard Properties

A new dashboard can be created from the icon corresponding to thecontext menu.

- From the icon displayed when hovering over the context menu.
- From the menu displayed with the option “Dashboard”.

An existing dashboard is modified by selecting it from the grid with the "Open" button.

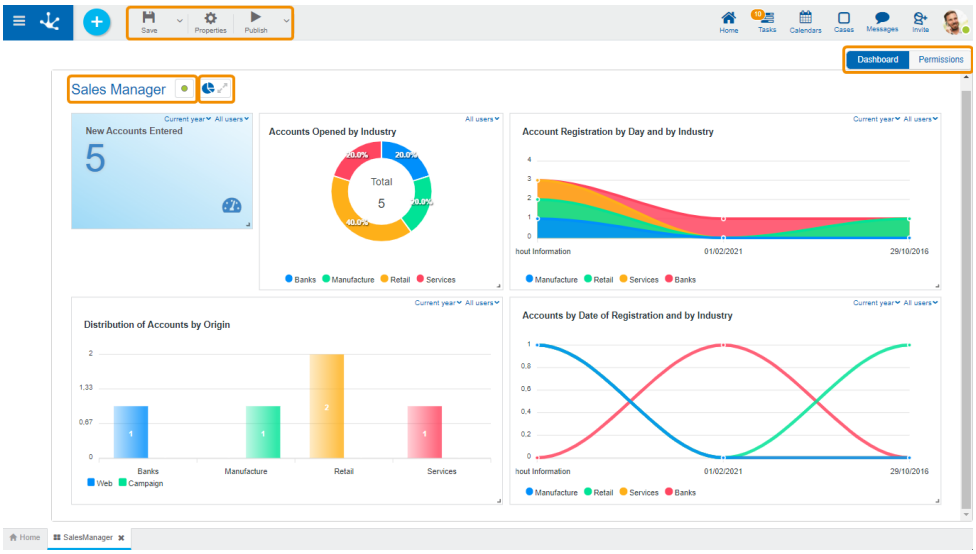
3.6.8.1. Modeling Facilities

New Dashboard

The IT modeler user can define application dashboards. After being published it is available for users to have permission in an application or, in case of being more than one application dashboard, to have the specific permission.

Workspace Sections

- Top Toolbar
- Design Options
- Data Area
- Icon to Add Indicators
- Dashboard Body



Data Area

Contains the dashboard name and the state of the dashboard.

Icon to Add Indicators

Allows to incorporate indicators to the dashboard.

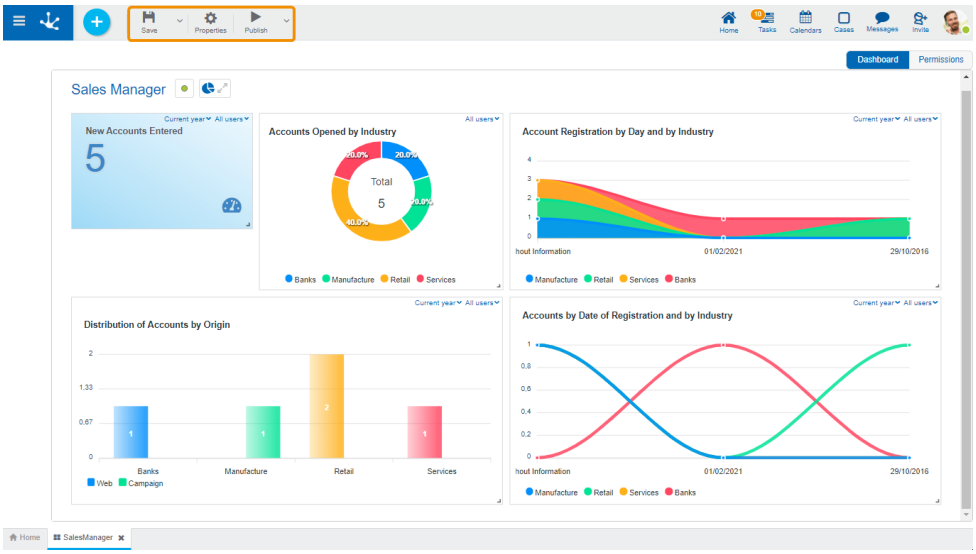
Body

In this section the published indicators that make up the dashboard are located. Its maintenance is done from the modeling option "Dashboard".

3.6.8.1.1. Top Toolbar

This toolbar contains icons and submenus from which you can perform operations on the object. Depending on its state, some options may be disabled.

The bar is displayed on the toolbar of **Deyel**.



Save

This icon allows to save the object in the repository of **Deyel**, leaving its state as "Draft" or "Modified". If certain conditions are met, the modeler user receives a message indicating that the operation was performed correctly, otherwise they receive an explanatory message.

Main Conditions

- The application of the object must exist.
- The name must be unique in the application.
- The object must not be locked by another user.
- The object's permissions must exist.

To the right of the icon the option to open the submenu.

Properties

Open the dashboard properties panel.

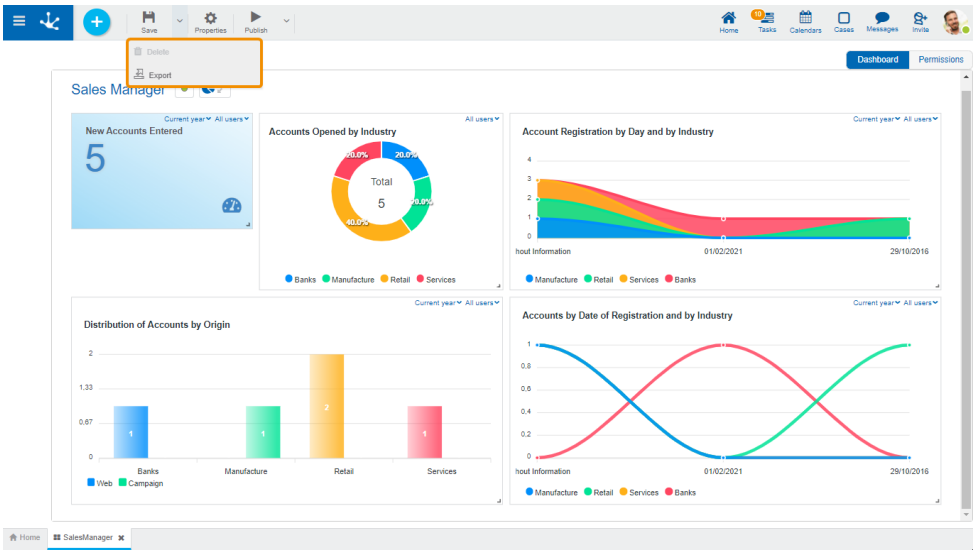
Publish

By means of this icon the object goes to "Published" state, after verifying a set of additional conditions to those of the "Save" operation. The modeling user receives the corresponding message, indicating the result of the operation.

To the right of the icon the option to open the submenu submenu.

3.6.8.1.1.1. Save Submenu

This submenu is opened by pressing the icon immediately to the right of the icon corresponding to "Save".

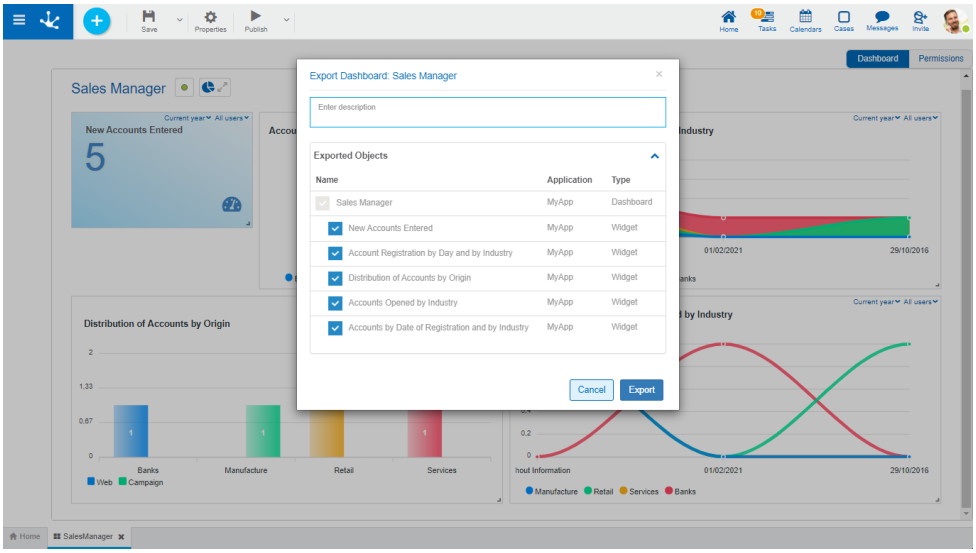


Delete

Allows to delete the object only if it is in "Draft"state, closing the tab in which it is located and deleting it from the modeler's grid.

Export

This icon opens a window for the user to select and confirm the export of the object and the related objects included in it.



Description

In this property a text explaining the reason for the operation can be entered.
This text can be modified upon import and is displayed in the description column of the export record.

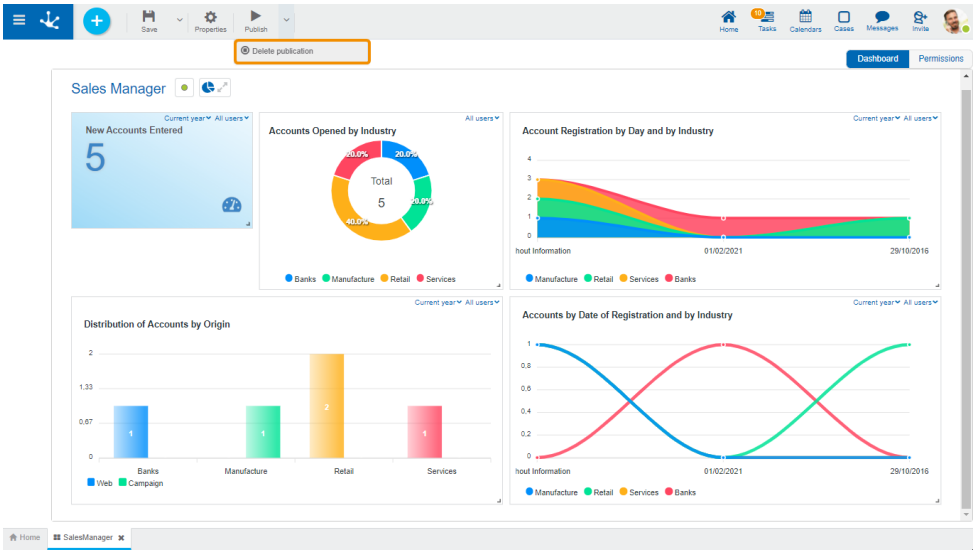
Exported objects

By expanding the container, the objects related to the dashboard being exported are shown. Objects not meant to be exported are unchecked.

You must press the “Cancel” button to leave the export without effect or the “Export” button to finish the operation.

3.6.8.1.1.2. Publish Submenu

This submenu can be opened by pressing the icon immediately to the right of the icon corresponding to "Publish".

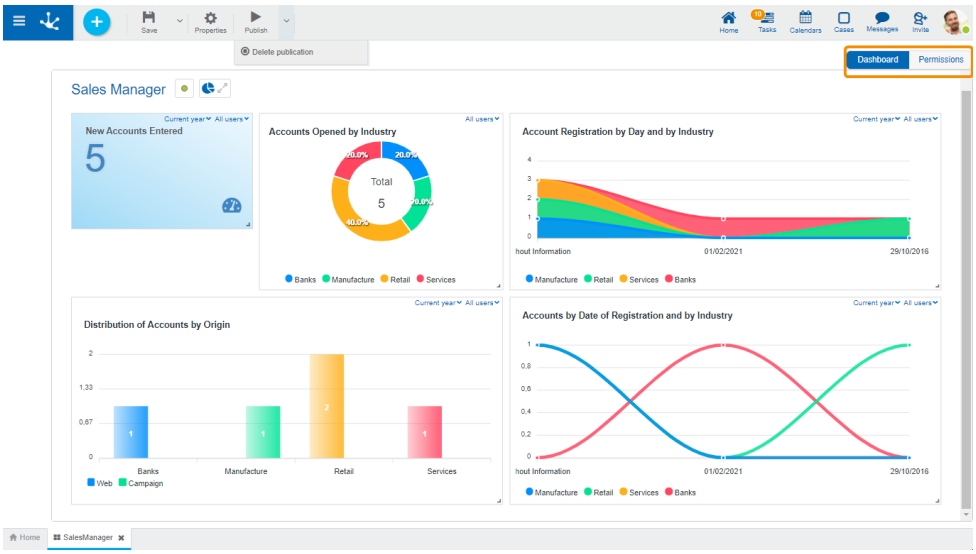


Delete Publication

Allows to leave the dashboard unavailable for use by returning it to the "Draft"state.

3.6.8.1.2. Design Options

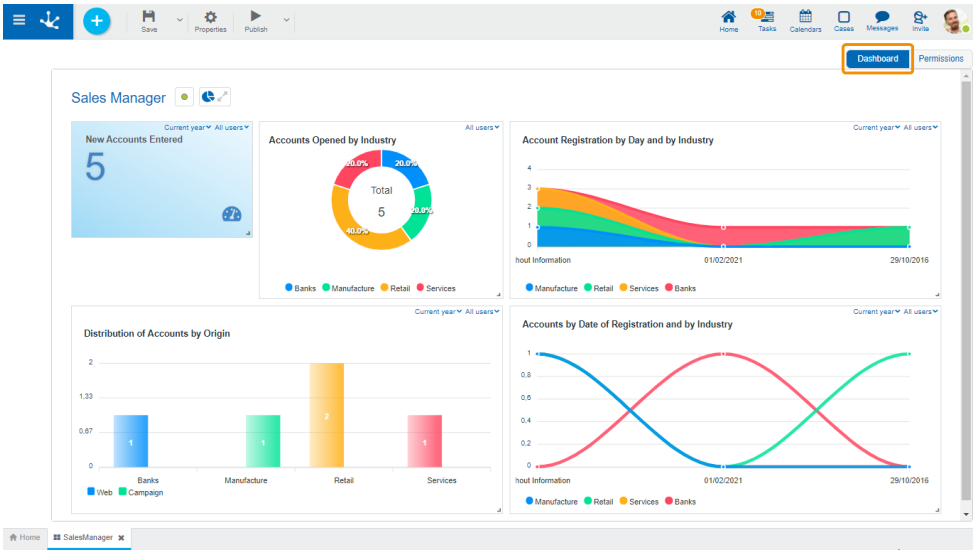
These options are presented on the upper right margin.



- Dashboard
- Permissions

3.6.8.1.2.1. Dashboard

In this section the maintenance of the dashboard widgets is done.



Operations

Add Indicators

Displays the gallery of published indicators on which the user has permission. For each indicator its name is visualized as well as its icon.
The “Search” option, in the gallery allows to filter indicators by the entered text.

Closes the indicators gallery.

Modify Indicators

The modifications the user applies on the indicators of a dashboard, have impact only in the visualization of such dashboard.

- Modify the indicators size. in order to secure the suitable visualization, each indicator is assigned a minimum size to respect.
- Modify the location of the dashboards in the dashboard area.
- Change user filters.
- Change date filters.
- Delete dashboard indicator.
- Add indicator.

> *Operation modify size and modify location are done with the function "Drag and Drop".*

3.6.8.1.2.2. Permissions

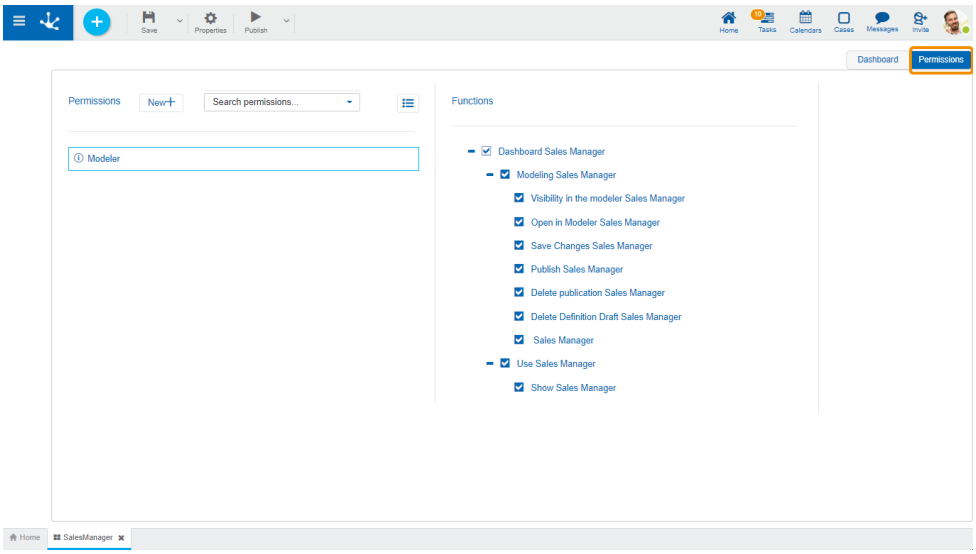
Allows to assign the security functions for use and modeling of the object to the existing permissions or to new permissions that the user creates, without having to go to the permission settings option from the menu.

Sections

- Permissions: Permissions to which object functions are assigned.
- Security Functions: Represents the total set of security functions, modeling and use of the object. Those that are marked are the ones included in the selected permission.

By default, all security functions for a new object are assigned to the permission Modeler.

Users who are assigned the permissions have access to the functions included in it.



New+ Opens a panel to create a new permission and once created, the security functions included in it must be selected .

Search permissions... Allows to select a permission from a list and enables the input of characters to filter the values in the list.

Opens the wizard to select a permission and once chosen, the necessary object security functions must be checked.

To unrelate a permission from the object, hover the cursor over the permission and press the icon ✖. If there are functions selected for that permission, they must be unchecked in order to delete it.

Security Functions to Model the Dashboard

- Visibility in the modeler: Allows to display the object in the Deyel modeler.
- Open in modeler: Allows to show the object from the Deyel modeler.
- Save changes: Enables the operation of saving modifications made to the object.
- Publish: Enables the operation of publishing the object leaving its state as "Published".
- Delete publication: Enables the operation of deleting the object publication leaving its state as "Draft".
- Delete draft definition: Enables the operation of deleting the object.
- Export: Enables the operation to export the object from the save submenu of the top toolbar.

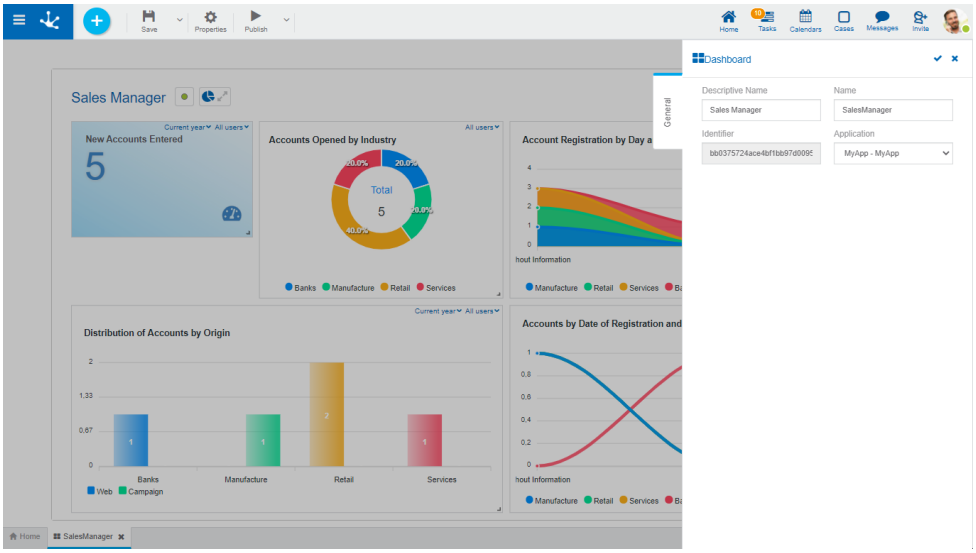
Security Functions to Use the Dashboard

- Show: Allows to show the object.

3.6.8.2. Dashboard Properties

The properties of the dashboards can be entered both at the time of their creation and when modifying an existing one.

Entering the dashboard properties panel is done using the icon which is in the top toolbar.



General Tab

Properties

Descriptive Name

It is the name visualized from the dashboard.

Name
It is the dashboard´s name. It does not allow blanks or special characters. It is unique and required.

Identifier
It is the internal dashboard´s name. It is automatically generated.

Application
It is the name of the application to which the dashboard belongs. It is not required, just as it happens with the rest of the modeled objects.

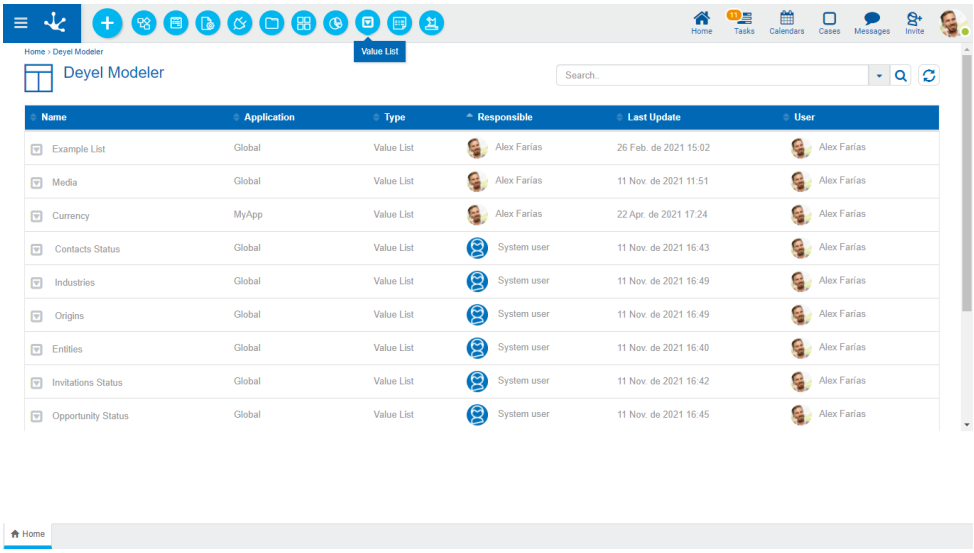
Actions

The icon ✔ is used to confirm the modifications made in the properties panel.

The icon ✖ is used to close the properties panel, if it was not previously saved, changes are discarded.

3.6.9. Value Lists Modeling

The value list modeler allows to define a set of possible values for a field, grouped under some criteria. These values can be associated to form fields.



The value list modeler allows to administer each list, respecting their progress through the different states of a modeling object.

- Save
- Publish
- Delete
- Export
- Delete Publication

The general characteristics of the list of values modeling environment are described in the topics:

- Modeling Facilities
- Value List Properties

A new value list can be created from the:

- Context menu .

- The icon .
- The menu displayed with the “Value List” option.

- The form modeler, in the “Relation” tab of a field selecting the value list option of a form or of an agile form.

- The form modeler, through the control Value List.

- The agile modeler, through the control Value List.

An already existing value list is modified by selecting it from the grid with the "Open" button.

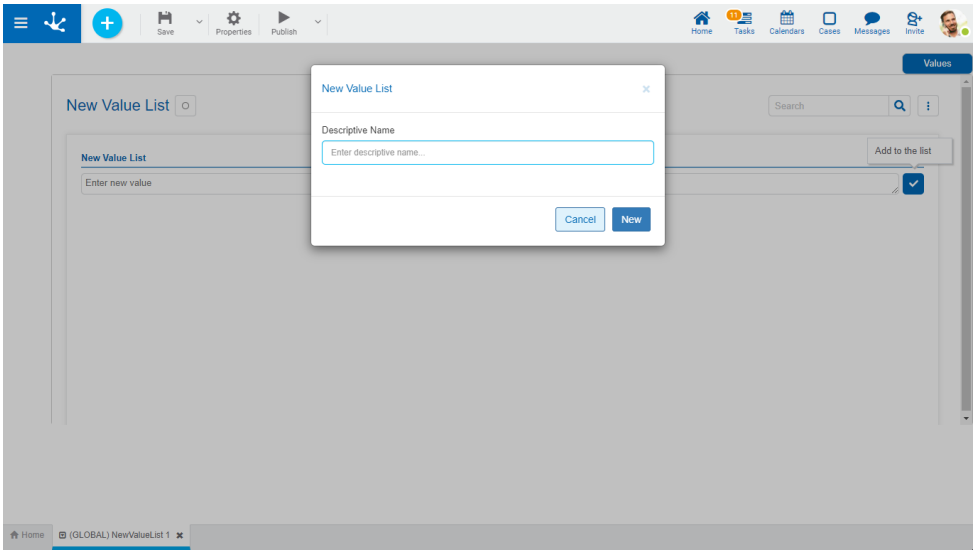
3.6.9.1. Modeling Facilities

New Value List

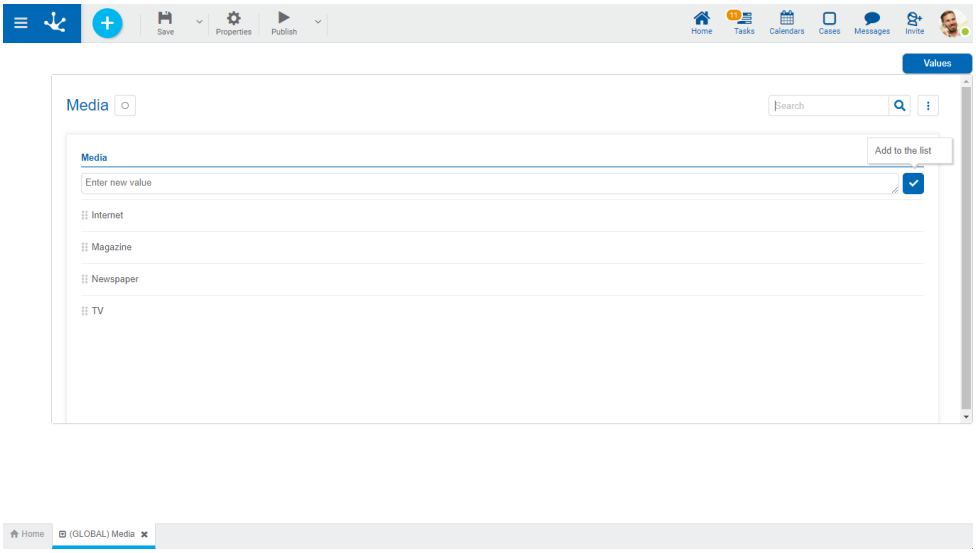
The IT modeler user can design a new value list, which is available to be used from any form after being published.

Steps for Creating a New Value List

Step 1: Assign the name and press the "Create" button.

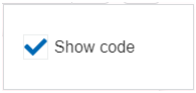


Step 2: Enter a new value and press the accent mark icon. Repeat this step until all the desired values are entered.



Workspace Sections

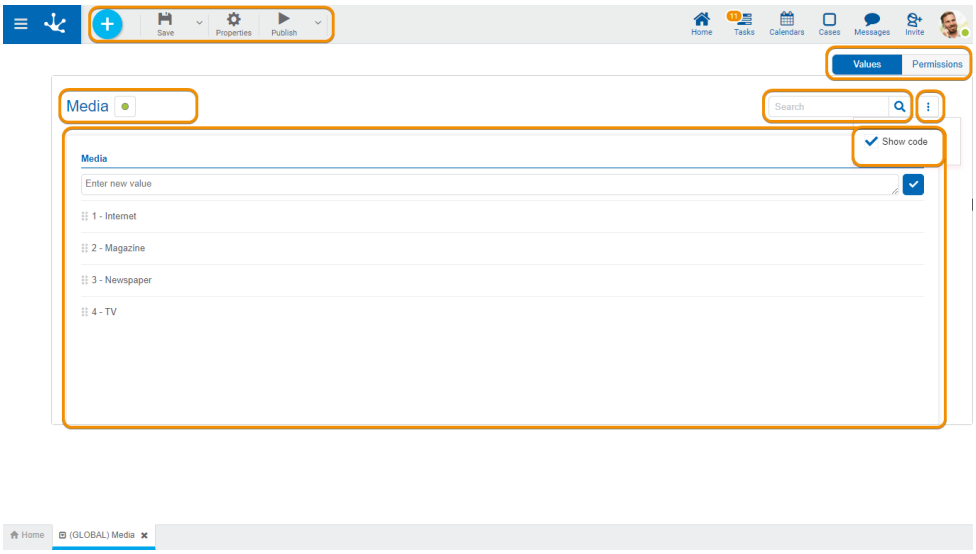
- Top Toolbar
- Value List Information
  - Name
  - State
- Allows to filter list values based on the entered characters. If a list is too long, it helps users to easily display the desired values.

- Enables an icon 

which allows to add the internal code to the list values.

- Design Options

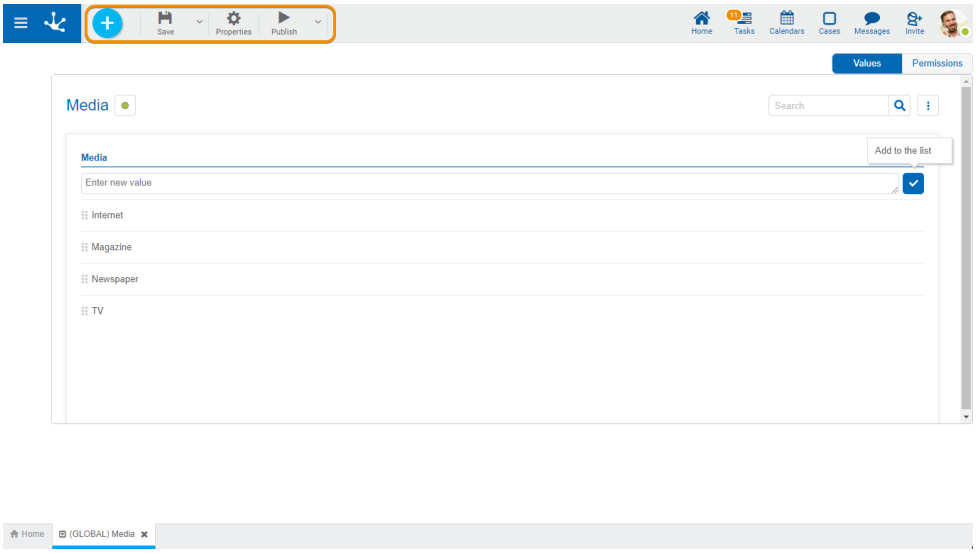
- Modeling Area



3.6.9.1.1. Top Toolbar

This toolbar contains icons and submenus from which you can perform operations on the object. Depending on its state, some options may be disabled.

The bar is displayed on the toolbar of **Deyel**.



Save

This icon allows to save the object in the repository of **Deyel**, leaving its state as "Draft" or "Modified". If certain conditions are met, the modeler user receives a message indicating that the operation was performed correctly, otherwise they receive an explanatory message.

Main Conditions

- The application of the object must exist.
- The name must be unique in the application.
- The object's permissions must exist if you select the property Models Security in the “General” tab of the properties panel.

To the right of the icon the option to open the submenu.

Properties

Open the value list properties.

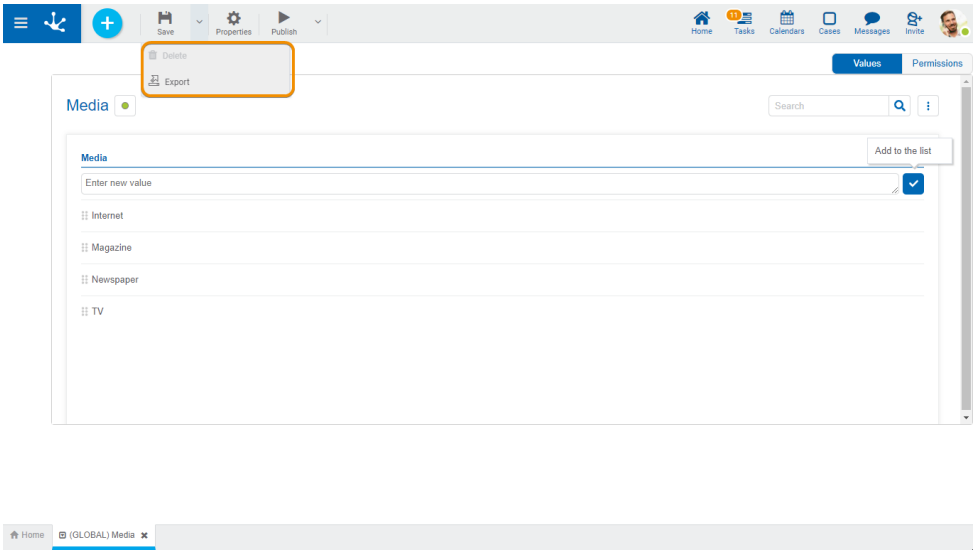
Publish

By means of this icon the object goes to "Published" state, after verifying a set of additional conditions to those of the "Save" operation. The modeling user receives the corresponding message, indicating the result of the operation.

To the right of the icon the option to open the submenu.

3.6.9.1.1.1. Save Submenu

This submenu is opened by pressing the icon immediately to the right of the icon corresponding to "Save".

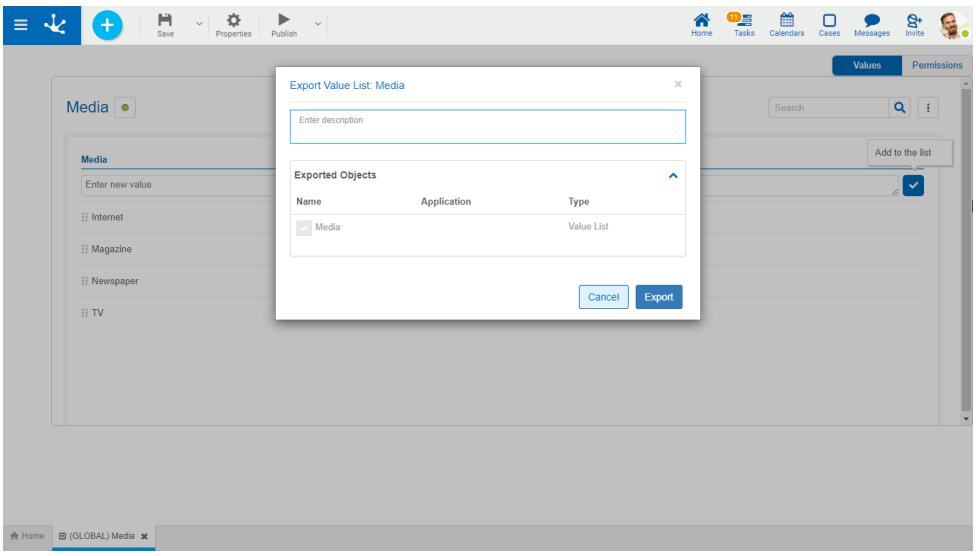


Delete

Allows to delete the object only if it is in "Draft"state, closing the tab in which it is located and deleting it from the modeler's grid.

Export

This icon opens a window for the user to select and confirm the export of the object.



Description

In this property a text explaining the reason for the operation can be entered.
This text can be modified upon import and is displayed in the description column of the export record.

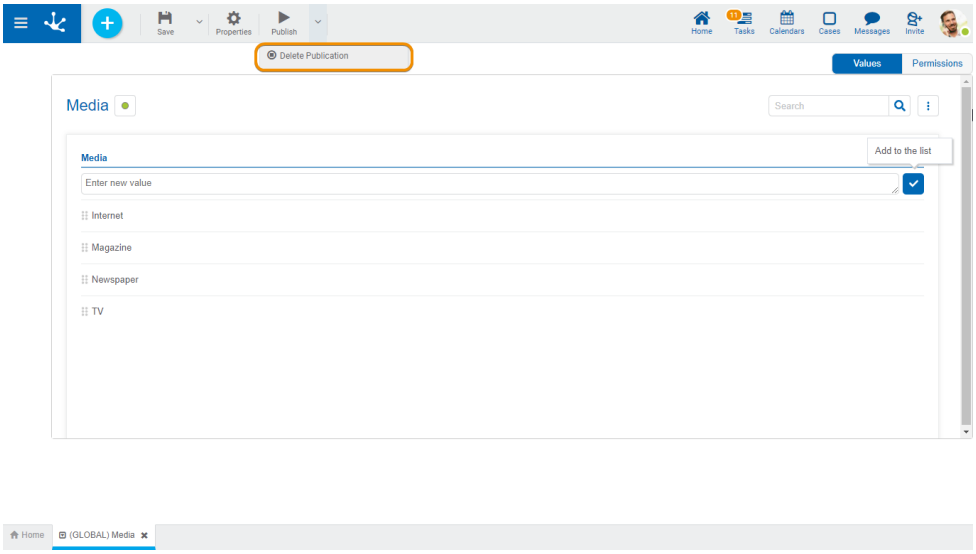
Exported Objects

By expanding the container, the object being exported is displayed.

You must press the “Cancel” button to leave the export without effect or the “Export” button to finish the operation.

3.6.9.1.1.2. Publish Submenu

This submenu can be opened by pressing the icon immediately to the right of the icon corresponding to "Publish".

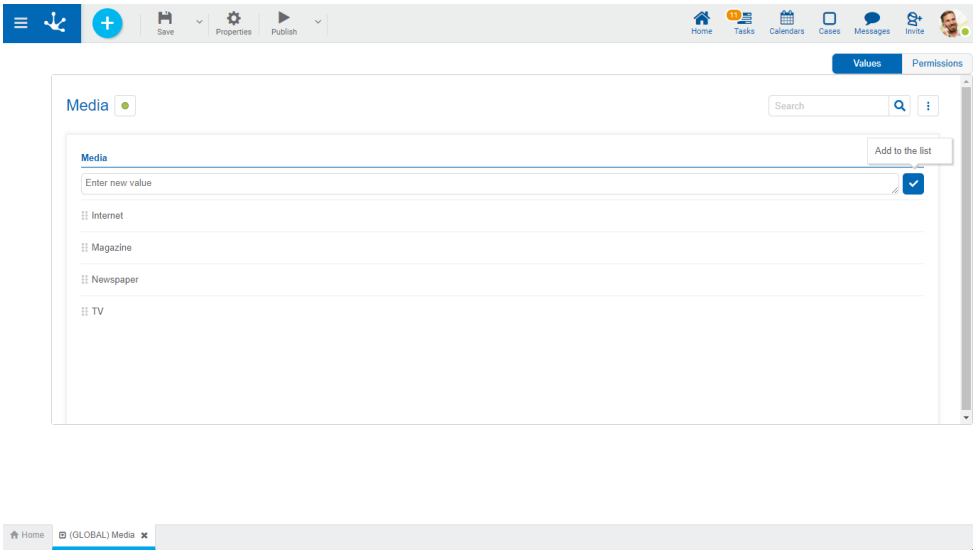


Delete Publication

Allows to leave the value list unavailable for use by returning it to the "Draft"state.

3.6.9.1.2. Modeling Area

The modeling are is the space where you can enter, modify or delete values from the lists.



Operations on Values

- Allows adding each entered value to the list of values.
- Double click: Allows to modify a value in the list.
- Move: Allows to change the position of a value within the list by dragging the value with the mouse.

Hovering the cursor over each of the values entered, a set of icons is displayed and this allows to perform different operations.

- Allows to delete a value from the list of values. Once deleted, it is displayed in gray and crossed out.
- Allows to restore a previously deleted value.

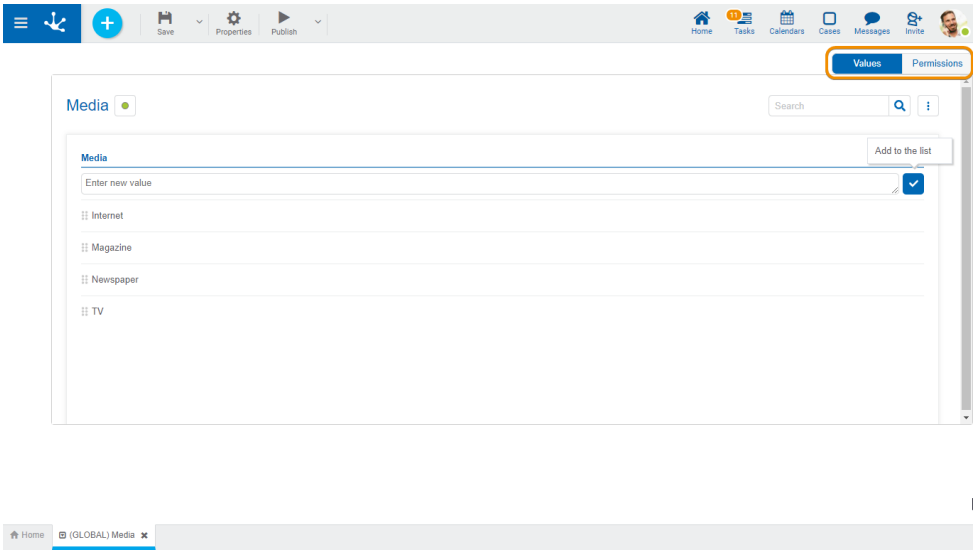
Display the Selected Line

- Hides the icons that are displayed.
- Shows hidden icons.

3.6.9.1.3. Design Options

When creating or modifying a value list, it allows to enter the values that are going to be displayed in it.

These options are presented on the upper right margin.



- Values

From this design option the value list is displayed and new values can be added in it in the modeling area.

- Permissions

This option is displayed if the property Models Security is selected in the properties panel of the value list.

3.6.9.1.3.1. Permissions

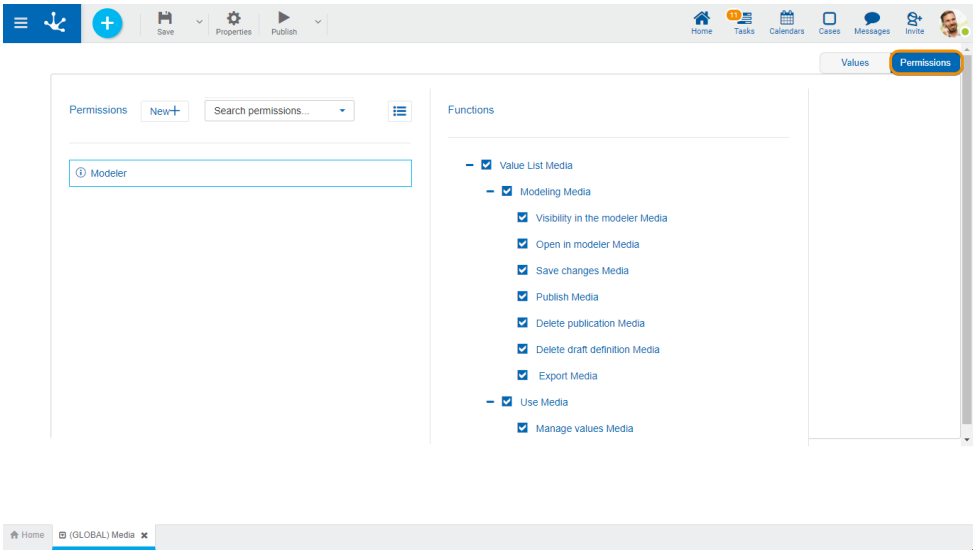
Allows to assign the security functions for use and modeling of the object to the existing permissions or to new permissions that the user creates, without having to go to the permission settings option from the menu.

Sections

- Permissions: Permissions to which object functions are assigned.
- Security Functions: Represents the total set of security functions, modeling and use of the object. Those that are marked are the ones included in the selected permission.

By default, all security functions for a new object are assigned to the permission Modeler.

Users who are assigned the permissions have access to the functions included in it.



New+ Opens a panel to create a new permission and once created, the security functions included in it must be selected .

Search permissions... Allows to select a permission from a list and enables the input of characters to filter the values in the list.

Opens the wizard to select a permission and once chosen, the necessary object security functions must be checked.

To unrelate a permission from the object, hover the cursor over the permission and press the icon ✖. If there are functions selected for that permission, they must be unchecked in order to delete it.

Security Functions to Model the Value List

- Visibility in the modeler: Allows to display the object in the Deyel modeler.
- Open in modeler: Allows to show the object from the Deyel modeler.
- Save changes: Enables the operation of saving modifications made to the object.
- Publish: Enables the operation of publishing the object leaving its state as "Published".
- Delete publication: Enables the operation of deleting the object publication leaving its state as "Draft".
- Delete draft definition: Enables the operation of deleting the object.
- Export: Enables the operation to export the value list from the save submenu of the upper toolbar.

Security Functions to Use the Value List

Security functions to use the value list are available if the value list is in “Published” state.

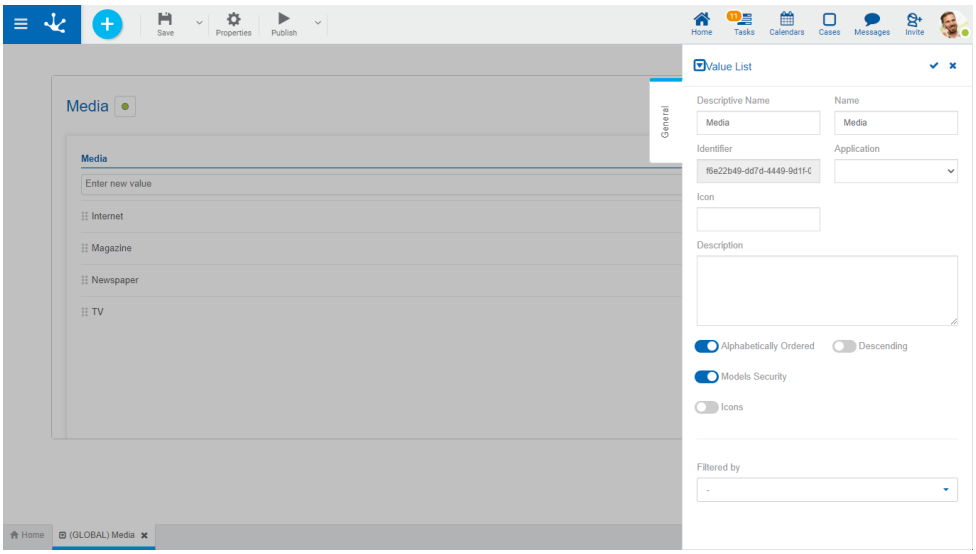
- Manage values: Enables the operation to enter, modify or delete values in the list.

> *The design option "Permissions" is only displayed if the property Models Security has been selected.*

3.6.9.2. Value List Properties

The properties of the value list can be entered both at the time of their creation and when modifying an existing one.

Entering the value list properties panel is done using the icon which is in the Top Toolbar



General Tab

Properties

Descriptive Name
It is the name used by the user to reference the value list.

Name
It is the name used internally by the user to reference the value list. It does not allow blanks or special characters. It is unique and required.

Identifier
It is the name which identifies unequivocally the value list. It is automatically generated.

Application
It is the name of the application to which the value list belongs. It is not required, just as it happens with the rest of the modeled objects.

Description
Text which defines the value list and its respective content.

Alphabetically Sorted

Checking this property indicates that the values in the value list are sorted alphabetically, and allows to select if the order will be descending.

Descending

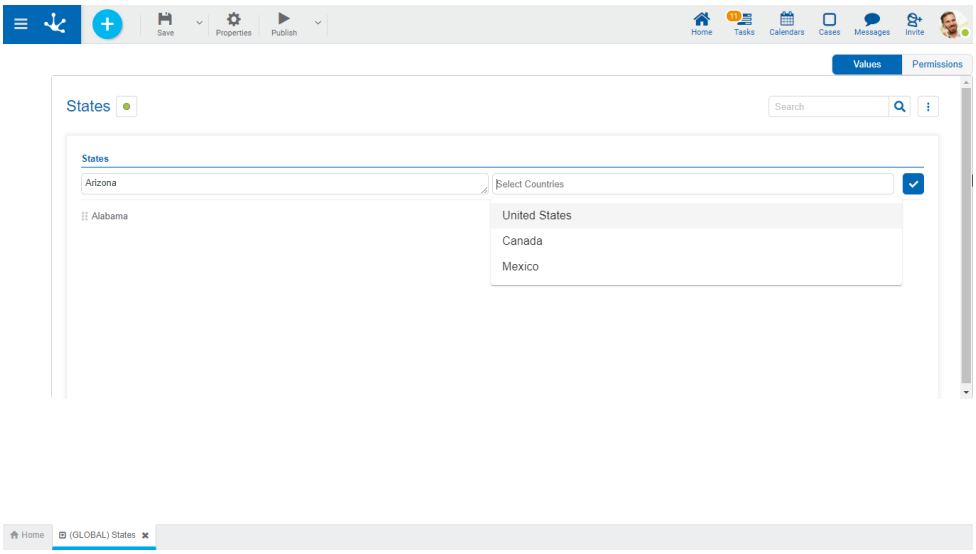
Indicates that the order of the values in the value list is descending. This property can be selected if it was previously indicated that the list is sorted alphabetically.

Models Security

Allows to add security to the value list, and indicates the creation of the security functions to model and the security functions to use, to be assigned in the design option "Permissions" or in the permission configuration.

Filtered by

Allows to select the value list to be related to the list being modeled. When a value is entered in the modeled list its corresponding is indicated in the list selected in this property.
Example: If the modeled list is States and the list selected in the property Filtered by is Countries, when entering a province you must indicate which country it corresponds to.



Actions

The icon ✔ is used to confirm the modifications made in the properties panel.

The icon ✖ is used to close the properties panel, if it was not previously saved, changes are discarded.

3.6.10. Exportación e Importación

El usuario modelador puede exportar sus objetos con diferentes finalidades, para realizar el pasaje de objetos entre ambientes, importando en el ambiente destino o para realizar las copias de seguridad de los mismos.

Al exportar un objeto se genera un archivo comprimido que contiene la definición del objeto y la de sus objetos relacionados seleccionados.

La exportación e importación tienen consideraciones diferentes según tengan los objetos estado o no.

- Con estado: al ser importados en el ambiente destino, no tienen impacto hasta tanto no sean publicados.
- Sin estado: al ser importados en el ambiente destino, tienen impacto inmediato.

Antes de realizar el pasaje de objetos de un ambiente a otro es recomendable tener configurado correctamente el ambiente destino en lo que respecta a unidades organizacionales, usuarios, roles y adaptadores.

Pasos para el Pasaje de los Objetos entre Ambientes:

Paso 1: Importación de los objetos

- Primero se importan las aplicaciones y los permisos
- El orden en que se importan los demás objetos es indistinto

Paso 2: Publicación de los objetos

- Se recomienda publicar primero los objetos que no tienen relación con otros objetos. Primero reglas, lista de valores, indicadores siguiendo por formularios, tableros y procesos.

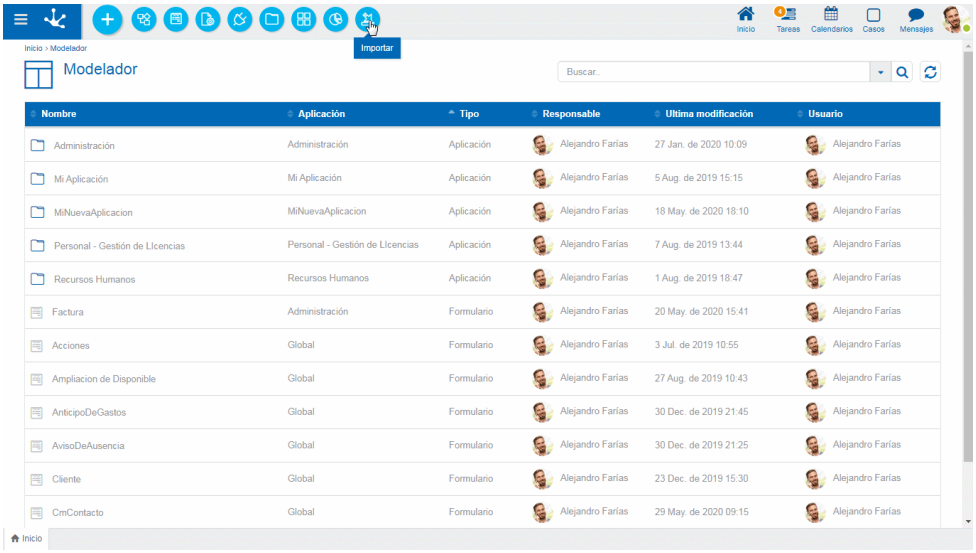
Exportación

La exportación de los objetos se realiza desde el ícono de la barra de herramientas superior de cada modelador de objeto, con excepción de aplicación, rol y permiso, que se exportan desde la grilla del modelador.

Importación

La importación de todos los objetos se realiza desde el menú contextual

- Desde el ícono .
- Desde el menú desplegado con la opción “Importar”.



Panel de Importación

Al iniciarse la importación, se abre un panel con información del objeto que se está importando.

Descripción

Descripción ingresada al momento de exportar el objeto.
Este texto puede modificarse al importar y se visualiza en la columna descripción del registro de importaciones.

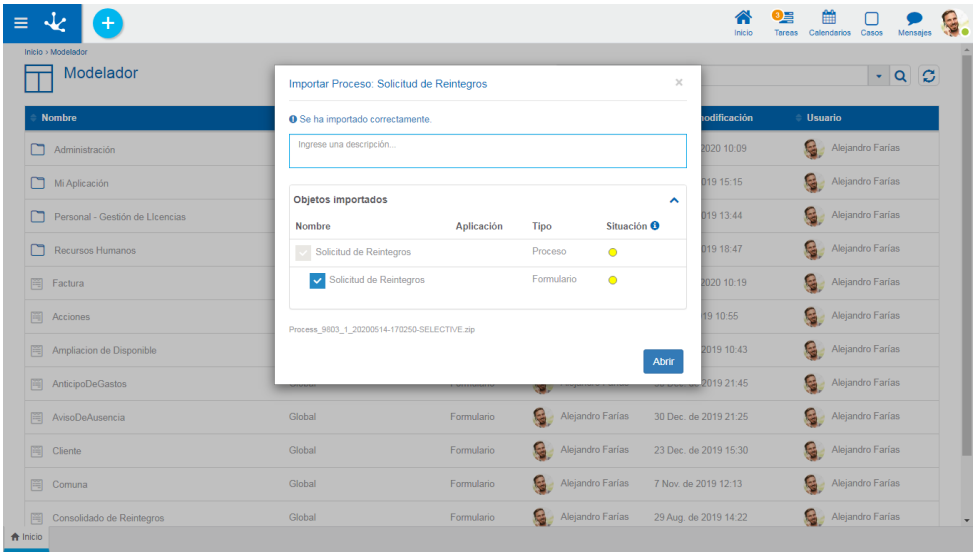
Objetos Importados

Desplegando este contenedor se listan los objetos relacionados ordenados jerárquicamente, que fueron incluidos al exportar el objeto principal. Pueden quitarse las marcas de selección a los objetos que no se desean importar.

Situación antes de realizar la importación

- Para el caso de objetos con estado, indica el estado en que se encuentra el objeto en el ambiente en que se está importando:
  - : Modificado
  - : Publicado
  - : Borrador
- Para el caso de objetos sin estado, indica si el objeto existe o no en el ambiente en que se está importando:
  - : Existe
  - : No existe

Al presionar el botón "Importar", los objetos marcados quedan importados en el ambiente destino y el panel presenta información del resultado de la operación.
Si la importación finalizó sin problemas se visualiza el mensaje correspondiente, caso contrario se visualiza un mensaje explicativo en color rojo.



Al presionar el ícono de la columna Situación se visualiza la explicación de su contenido.

Si los objetos marcados no cumplen alguna de las condiciones para importar, la importación queda sin efecto en su totalidad, es decir, se importan todos los objetos o ninguno.

> *En caso de importar objetos exportados en versiones anteriores a 7.0, se muestran únicamente los objetos relacionados permitidos.*

Situación después de realizar la importación

- Para el caso de objetos con estado:
  - Si existían en el ambiente destino con estado publicado o modificado, luego de la importación quedan con estado modificado
  - Si existían en el ambiente destino con estado borrador o no existían, luego de la importación quedan con estado borrador
- Para el caso de objetos sin estado:
  - Si existían en el ambiente destino , la importación los actualiza directamente
  - Si no existían previamente , la importación los crea

Condiciones para Importar

Cuando se importan objetos se verifican las mismas condiciones que al realizar la operación de guardar de cada uno de ellos.

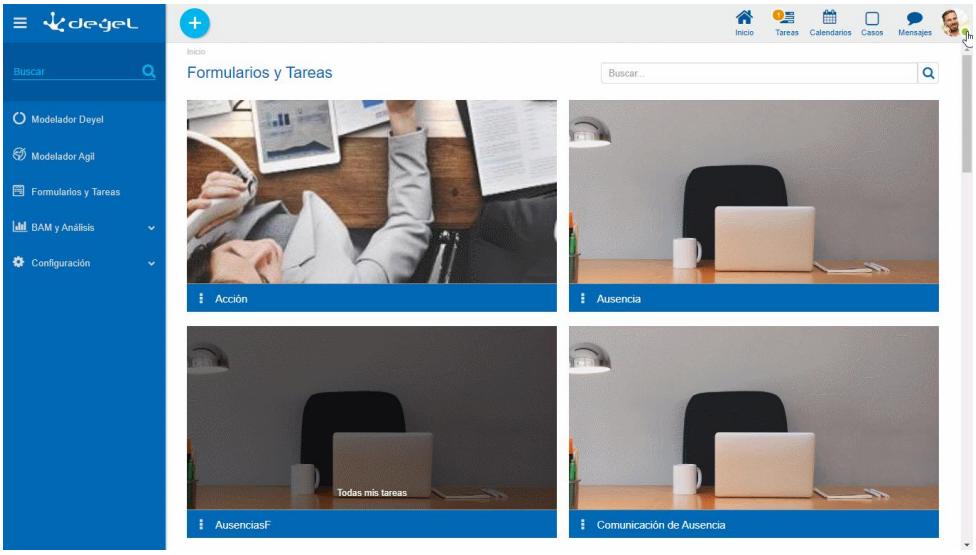
Adicionalmente, se verifican las siguientes condiciones según el objeto:

- Procesos
  Si el proceso tiene casos iniciados, no puede importarse si se han eliminado actividades o compuertas.
- Reglas
  El adaptador debe existir.
- Roles
  La aplicación del rol, los actores y el usuario coordinador deben existir.

- Permisos.
  La aplicación del permiso debe existir en el ambiente destino.
  Pueden ocurrir que algunas de las funciones de seguridad contenidas en el permiso no existan en el ambiente destino. En estos casos se emite un aviso en el archivo de log y se permite realizar el import.
  El permiso importado queda sin referenciar a estas funciones inexistentes.

3.6.10.1. Registro de Exportaciones

Deyel permite visualizar un registro de las exportaciones ejecutadas, para que puedan ser consultadas por los usuarios modeladores de TI.
Para ello, en el ambiente debe estar seleccionada la propiedad para habilitar el registro de objetos exportados.
Se puede acceder a dichos registros utilizando la búsqueda rápida del menú, ingresando "Registro de Exportaciones".



Datos Generales

Número
Es un número consecutivo que identifica unívocamente al registro.

Identificador
Es una identificación del registro que resulta de la concatenación de los siguientes datos:
- Fecha y hora de ejecución del proceso de exportación.
- Nombre del servidor.

Fecha y Hora
En que se realizó la exportación del objeto.

Nombre del Servidor
URL de la aplicación donde se ejecutó la exportación.

Usuario

Código del usuario que solicitó ejecutar la exportación.

Archivo
Archivo generado en la exportación.

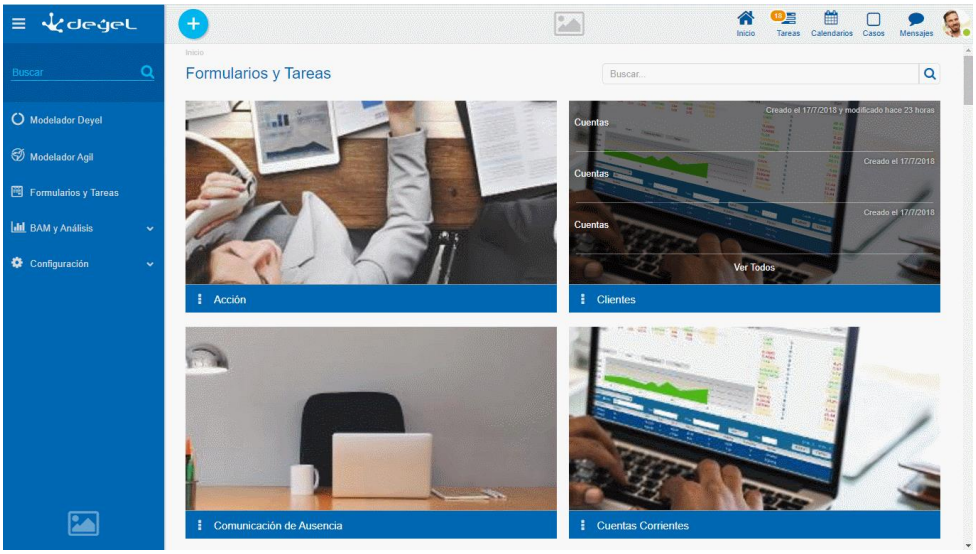
Descripción
Si el usuario ingresó observaciones al realizar la exportación del objeto, se visualizan en este campo.

Objeto Raíz y Objetos Incluidos

El objeto raíz es el objeto principal de la exportación. Mientras que todos los objetos incluidos son aquellos relacionados al objeto principal, que se indicaron como dependencias al exportarlo.

3.6.10.2. Registro de Importaciones

Deyel permite visualizar un registro de las importaciones ejecutadas, para que puedan ser consultadas por los usuarios modeladores de TI.
Para ello, en el ambiente debe estar seleccionada la propiedad para habilitar el registro de objetos importados.
Se puede acceder a dichos registros utilizando la búsqueda rápida del menú, ingresando "Registro de Importaciones".



Datos Generales

Número
Es un número consecutivo que identifica unívocamente al registro.

Identificador
Es una identificación del registro que resulta de la concatenación de los siguientes datos:
- Fecha y hora de ejecución del proceso de importación.
- Nombre del servidor.

Fecha y Hora
En que se realizó la importación del objeto.

Nombre del Servidor
URL de la aplicación donde se ejecutó la importación.

Usuario
Código del usuario que solicitó ejecutar la importación.

Archivo
Archivo utilizado en la importación.

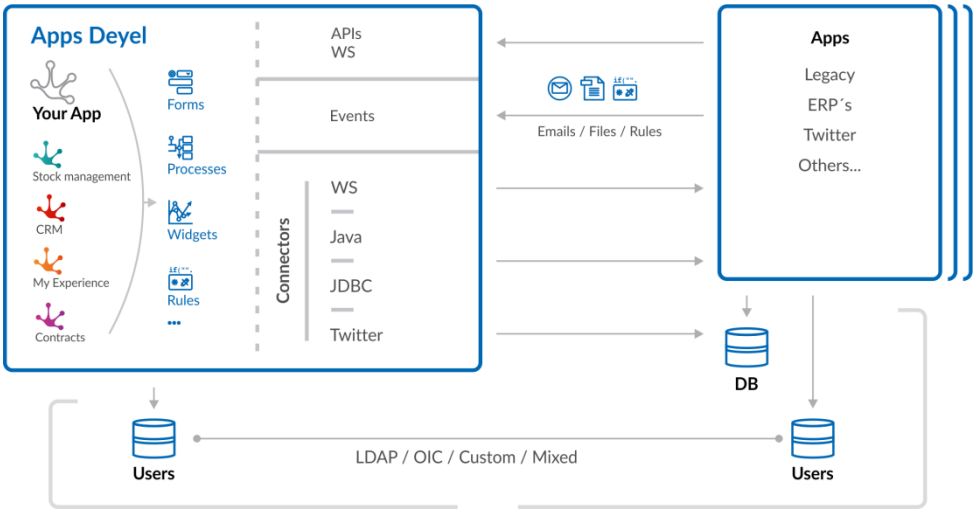
Descripción
Si el usuario ingresó observaciones al realizar la importación, se visualizan en este campo.

3.7. Integration

Applications developed with **Deyel** can interact with other applications, by using different tools and services of the platform.

On the one hand, **Deyel** allows to integrate its applications with external applications and existing data sources, that is, Legacy systems, databases, Twitter or other external applications, making use of adapters based on web services, Java and JDBC.

It also has a web services API based on the Richardson maturity model design guides, level 3, which allows any external application to **Deyel** to interact with applications developed with **Deyel**, or with solutions, such as CRM, Contracts or Stock Management, among others. It also allows the use of emails, rules, files and other events.



Applications developed with **Deyel** have their own user directory and if necessary they can be integrated through LDAP, OIC, in a personalized and mixed way, that is, allowing groups of users with different authentications.

Integration Facilities

Applications can be integrated using different platform tools and services.

Integration with External Applications

Applications developed with **Deyel** can be integrated with other applications by different means, either consulting or updating their information.

- Rules based on Java, JDBC and web services adapters
  This integration uses advanced rules and adapters which are defined using the different modelers of **Deyel**.

- Rules based on application adapters that require user authorization
  . Integration with Twitter is an example of this type of integration through an adapter where the connection credentials are recorded. This adapter allows to centrally administrate the interaction of **Deyel** users with Twitter.

Integrating with External Applications

Other applications can be integrated with applications developed with **Deyel**, mainly using Deyel Rest API.There are also other means that can be very useful, depending on the characteristics of the applications to be integrated.

- Use of web services API
  Through the use of Deyel Rest API other applications can interact with applications developed with **Deyel**, having access to forms and cases through web services.

- Other media
  Integration can be done by using emails, rules and files.

Authentication Schemas

When applications are integrated, it may be required for them to be the same users or to be synchronized among them. The authentication procedure ensures that any use of **Deyel** and the use of its solutions are performed by "someone" or "something" that can be identified as a user, within the user registry.

Deyel uses different authentication schemas to respond to this request.

- Native
  Deyel verifies that the user exists in its registry and checks the password, which is stored encrypted in its database.

- LDAP
  When an organization uses directory services to register its users, **Deyel** may be configured to delegate the authentication process to LDAP.

- Federated

When an organization uses Google accounts for its users, the authentication process can be delegated to Google.

- Mixed
It allows an organization to use different types of authentication simultaneously.

Using the Integration

Integration rules can be used from the different objects of the applications developed with **Deyel**.

- Rules in forms and processes
Advanced adapter-based rules can be used using embedded rules.

- Automatic actions of process activities
Advanced adapter-based rules can be used in the modeling of automatic actions of process activities.

- Scheduled tasks and events
Advanced adapter-based rules can be used in the modeling of events and scheduled tasks.

3.7.1. Deyel API Rest

By using Deyel Rest API, other applications can interact with applications developed with **Deyel** and use the information associated with their objects.

The Rest API of **Deyel** uses JWT to get the authentication token. The token granted by **Deyel** allows third-party applications to use all the methods of the implemented endpoints.

Endpoints

- **/forms**
Allows administration of **Deyel** forms, being able to carry out the following operations: creation, show, modification, deletion and search.

- **/cases**
Allows the creation of cases and the execution of activities.

- **/rules**
Allows to obtain parameters and execute rules.

Methods

The Rest API of **Deyel** is composed of a set of methods that allow access to the objects that compose the applications of **Deyel**.

To test these examples, the json of forms or cases must be downloaded depending on what is to be tested.

Examples

To facilitate the use the Rest API of **Deyel** examples that are contained in collections are defined and must be imported into Postman.

To execute examples, the following steps must be followed beforehand:

- Define the User
- Import Collections
- Get the Token

Each collection has a list of operations that can be executed to access the Rest API of **Deyel**.

- Forms Collection
- Cases Collection
- Rules Collection

3.7.1.1. Define User

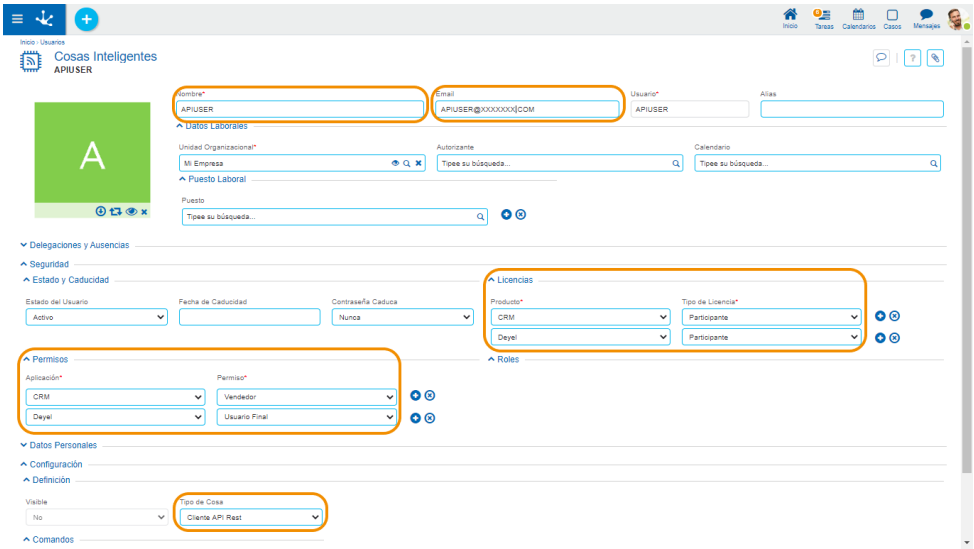
To access the objects used in the examples, a smart thing type user must be defined, configuring its properties to indicate name, user, type and permissions for each application.

User Properties

The smart thing type user properties must be completed with the suggested values.

Name	APIUSER
User	APIUSER
E-mail	Email address where the user receives the password.
Type	Rest API Client
Product Type of License	CRM Participant
Product Type of License	Deyel Participant
Application	CRM

Permission	Seller
Application Permission	Deyel End User



> *Permission for the CRM application is included so that the Account form (CRM_ACCOUNT) in the form samples can be accessed.*

3.7.1.2. Import Collections

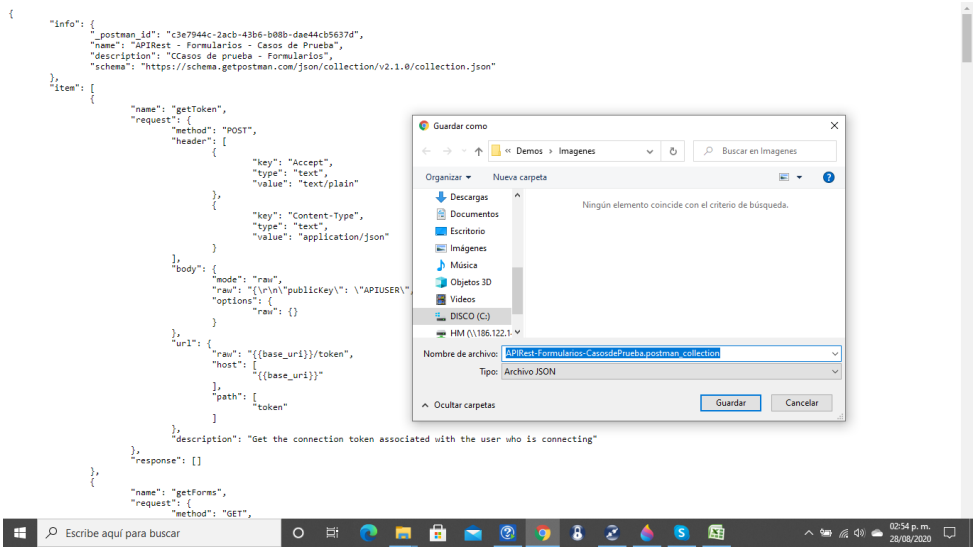
To start using Deyel Rest API from Postman, the collection corresponding to the selected example must be imported.

- Examples for Forms

The example is implemented on a project in Postman and can be opened from here.

```
{
	"info": {
		"_postman_id": "c3e7944c-2acb-43b6-b08b-dae44cb5637d",
		"name": "APIRest - Formularios - Casos de Prueba",
		"description": "CCasos de prueba - Formularios",
		"schema": "https://schema.getpostman.com/json/collection/v2.1.0/collection.json"
	},
	"item": [
		{
			"name": "getToken",
			"request": {
				"method": "POST",
				"header": [
					{
						"key": "Accept",
						"type": "text",
						"value": "text/plain"
					},
					{
						"key": "Content-Type",
						"type": "text",
						"value": "application/json"
					}
				],
				"body": {
					"mode": "raw",
					"raw": "{\r\n\"publicKey\": \"APIUSER\",\r\n\"privateKey\" : \"APIUSER\"\r\n}\r\n\r\n",
					"options": {
						"raw": {}
					}
				},
				"url": {
					"raw": "{{base_uri}}/token",
					"host": [
						"{{base_uri}}"
					],
					"path": [
						"token"
					]
				},
				"description": "Get the connection token associated with the user who is connecting"
			},
			"response": []
		},
		{
			"name": "getForms",
			"request": {
				"method": "GET",
				"header": [
```


The example must be saved with a json extension and imported into Postman.

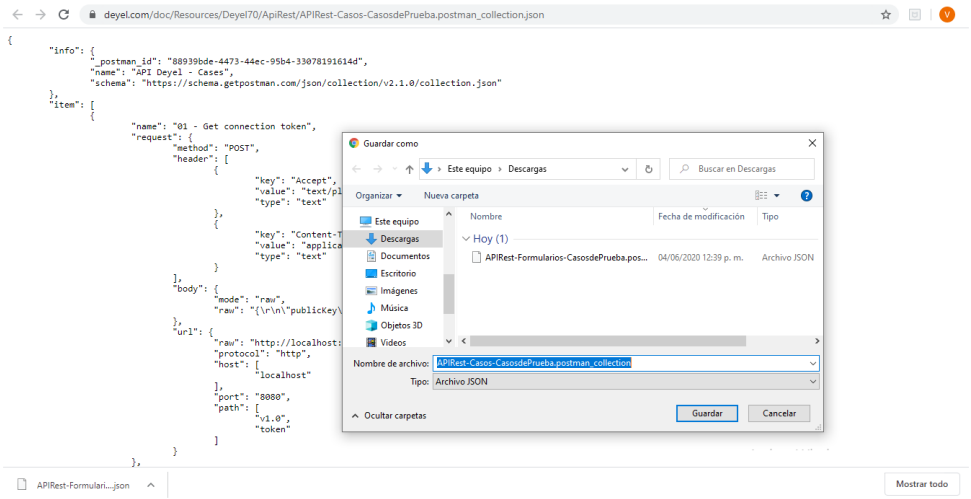


- [Examples for Cases]

The example is implemented on a project in Postman and can be [opened from here].

```
{
        "info": {
                "_postman_id": "91356180-6a92-4253-b313-93b17e29e319",
                "name": "API Deyel - Cases",
                "schema": "https://schema.getpostman.com/json/collection/v2.1.0/collection.json"
        },
        "item": [
                {
                        "name": "getToken",
                        "request": {
                                "method": "POST",
                                "header": [
                                        {
                                                "key": "Accept",
                                                "value": "text/plain",
                                                "type": "text"
                                        },
                                        {
                                                "key": "Content-Type",
                                                "value": "application/json",
                                                "type": "text"
                                        }
                                ],
                                "body": {
                                        "mode": "raw",
                                        "raw": "{\r\n\"publicKey\": \"APIUSER\",\r\n\"privateKey\" : \"APIUSER\"\r\n}\r\n\r\n"
                                },
                                "url": {
                                        "raw": "{{base_uri}}/token",
                                        "host": [
                                                "{{base_uri}}"
                                        ],
                                        "path": [
                                                "token"
                                        ]
                                }
                        },
                        "response": []
                },
                {
                        "name": "startCase",
                        "request": {
                                "method": "POST",
                                "header": [
                                        {
                                                "key": "Authorization",
                                                "type": "text",
                                                "value": "Bearer
eyJraWQiOiJrMSIsImFsZyI6IlJTMjU2In0.eyJpc3NiOiJEZXllbCIsImV4cCI6MTU5MTIwNDcxNiwiY2RPcmdVbml0IjoiMDAwMDAwMDAwMyIsInBlcm1pc3Npb25zIjoie1wiZHRWYWxpZGl0eVwiOlwiSnVuIDIsIDIwMjAgMjoxODozNiBQTV
```


The example must be saved with a json extension and imported into Postman.

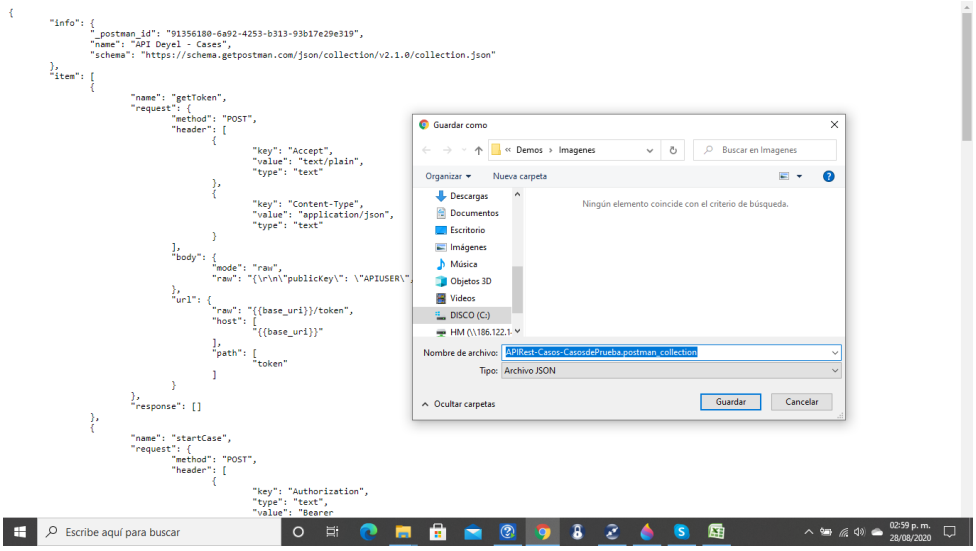


- [Examples for Rules]

The example is implemented on a project in Postman and can be [opened from here].

```
{
        "info": {
                "_postman_id": "95aed289-601b-4b4a-b87a-c901a9f5df47",
                "name": "APIRest - Reglas ",
                "schema": "https://schema.getpostman.com/json/collection/v2.1.0/collection.json"
        },
        "item": [
                {
                        "name": "1- Get Token",
                        "request": {
                                "method": "POST",
                                "header": [
                                        {
                                                "key": "Accept",
                                                "type": "text",
                                                "value": "text/plain"
                                        },
                                        {
                                                "key": "Content-Type",
                                                "type": "text",
                                                "value": "application/json"
                                        }
                                ],
                                "body": {
                                        "mode": "raw",
                                        "raw": "{\r\n\"publicKey\": \"APIUSER\",\r\n\"privateKey\" : \"APIUSER\"\r\n}\r\n\r\n"
                                },
                                "url": {
                                        "raw": "{{base_uri}}/token",
                                        "host": [
                                                "{{base_uri}}"
                                        ],
                                        "path": [
                                                "token"
                                        ]
                                }
                        },
                        "response": []
                },
                {
                        "name": "2- Get Parameters Rule",
                        "request": {
                                "method": "PUT",
                                "header": [
                                        {
                                                "key": "Authorization",
                                                "type": "text",
                                                "value": "{{token}}"
                                        }
                                ],
```


The example must be saved with a json extension and imported into Postman.



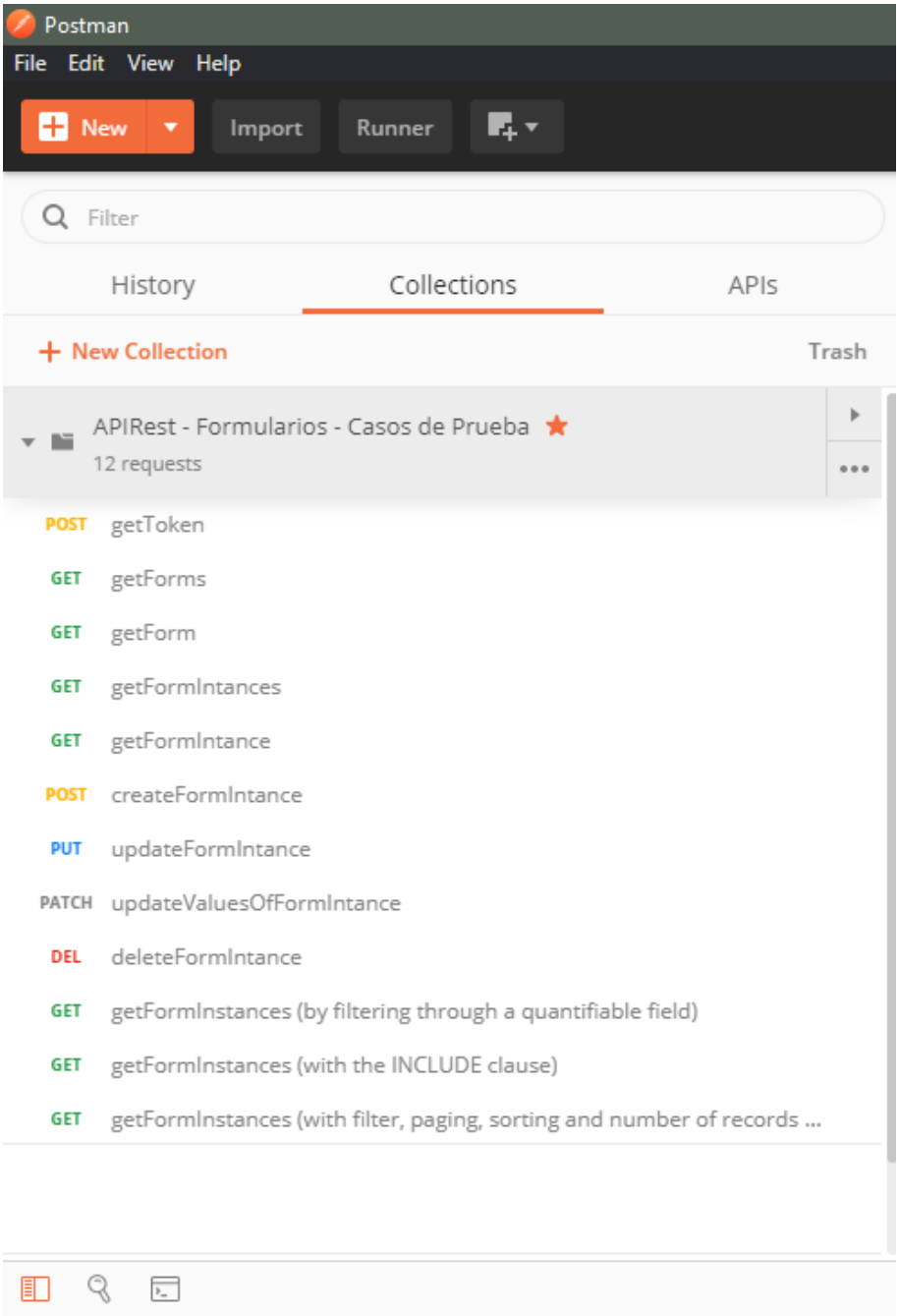
3.7.1.3. Obtain Token

To use Deyel Rest API from Postman, once the selected collection is imported, an authentication token must be obtained from **Deyel**.This token allows third-party applications to use the implemented methods.

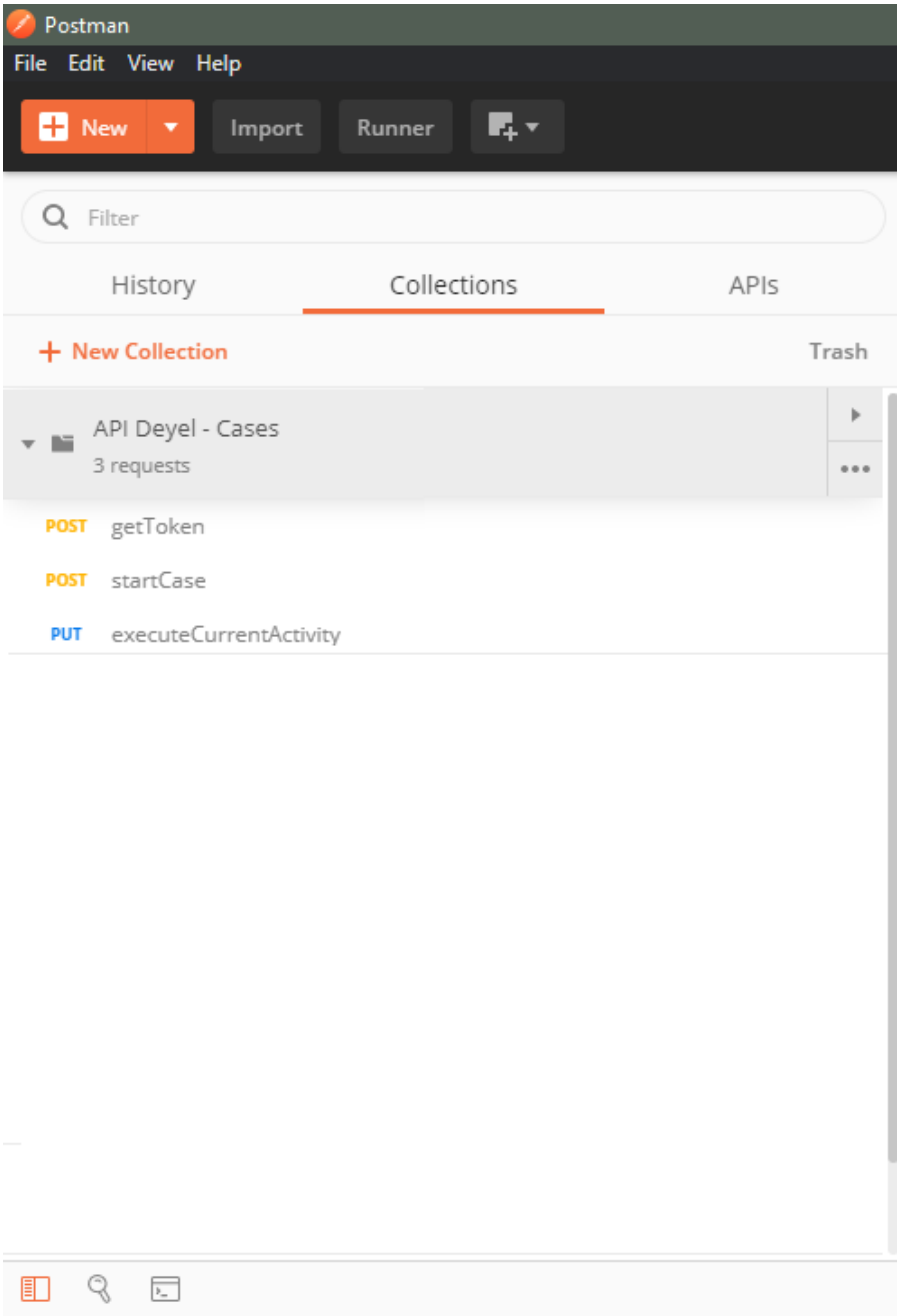
Step 1

Configure the uri used for testing, from the collection edition.

- For Forms Examples

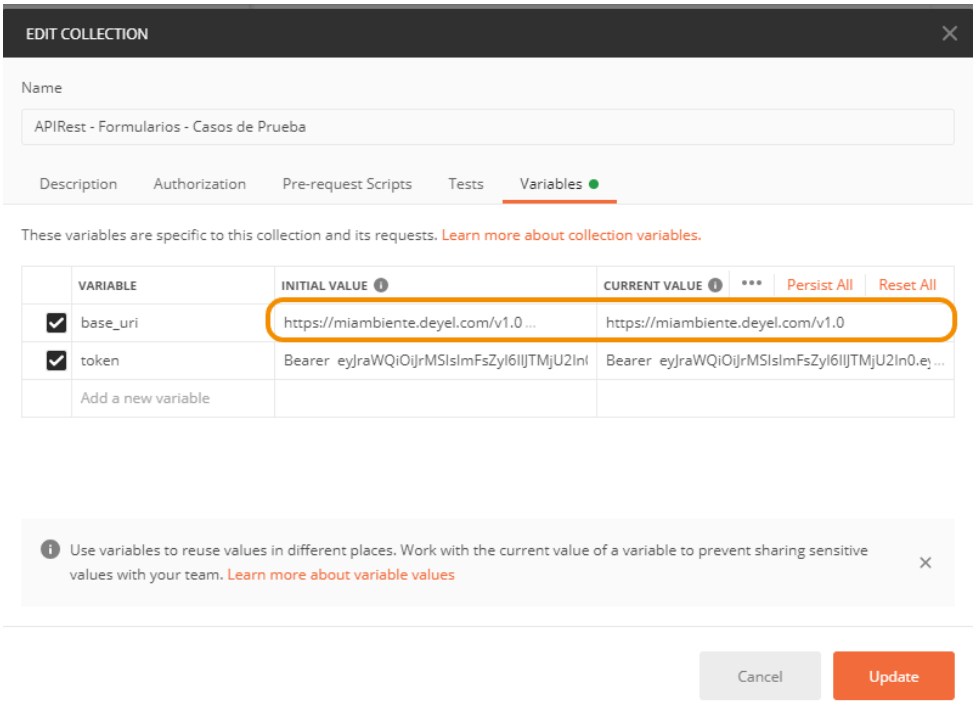


- For Cases Examples



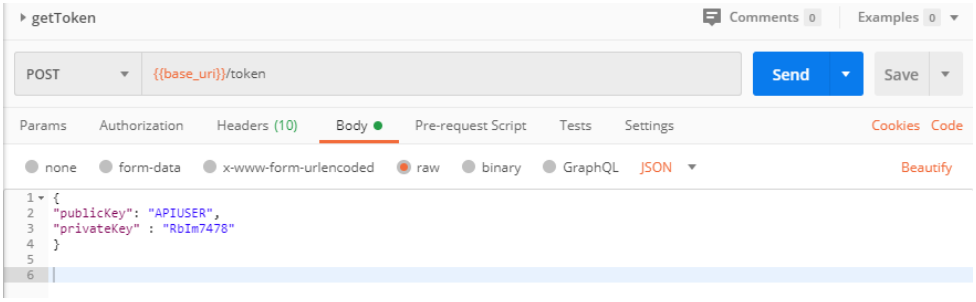
- For Rules Examples

The value of the "base_uri" variable must be modified, replacing <miambiente> by the name of the corresponding environment and press the "Update" button.



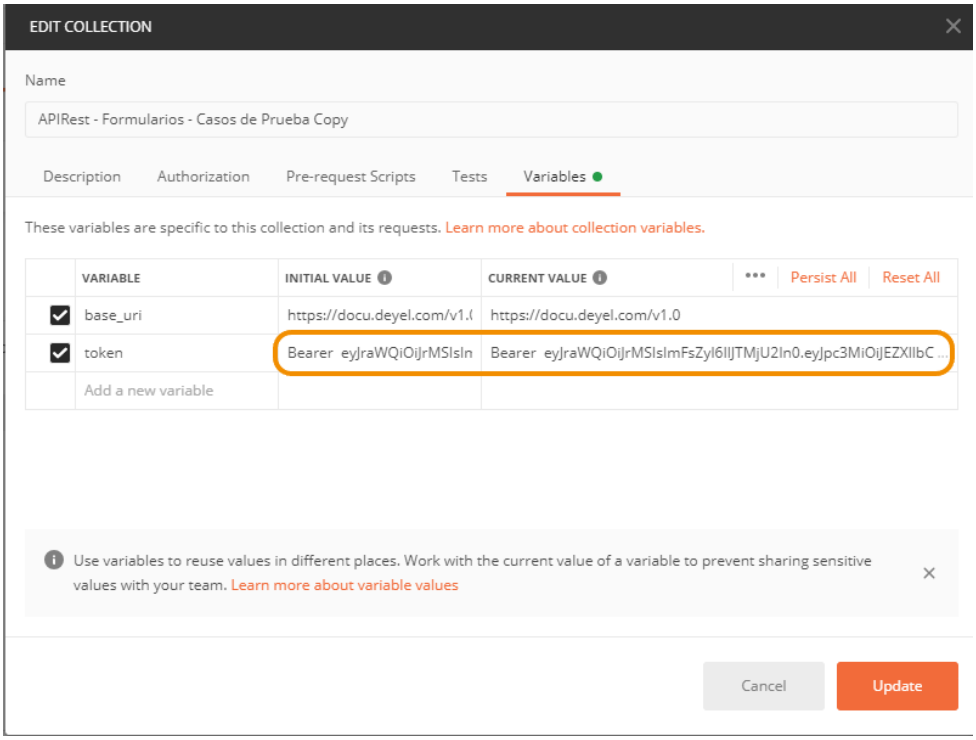
Step 2

From the getToken method, in the "Body" tab, you must indicate the user defined as thing type and the password received by mail. Press "Send" button.



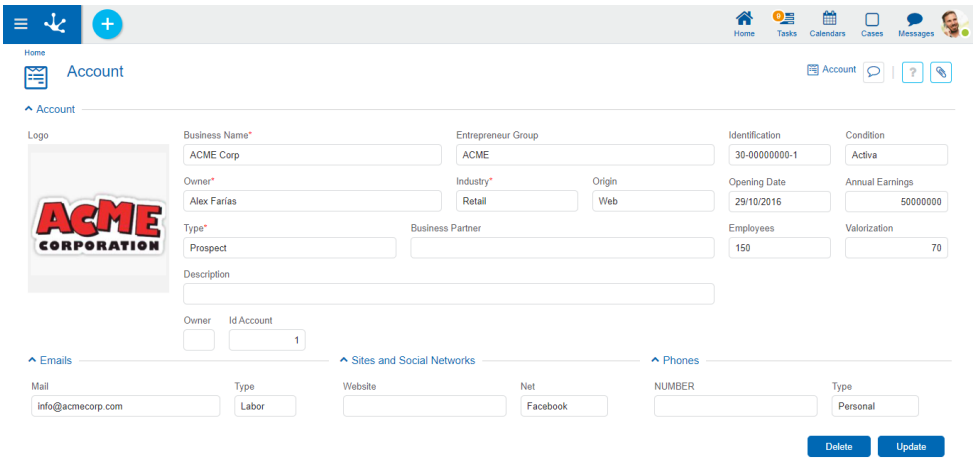
Step 3

Copy from the getToken method response, the "Authorization" parameter value of the "Header" tab, in the "token" variable of the collection edition. Press "Update" button.



3.7.1.4. Form Examples

These examples are based on the (CRM_ACCOUNT) Account form of the CRM solution, available in all versions of **Deyel**.



Forms Collection

Examples are implemented from Postman to execute each of the operations described in the following table. Previously, you must import the Form collection.

Operation	Description
getToken	Obtains token.
getForms	Returns a list with the CRM application forms.
getForm	Returns the information from the Account form definition (CRM_ACCOUNT).
getFormInstances	Returns a list with all the Account form instances (CRM_ACCOUNT).
getFormInstance	Returns a specific instance from the previous item list.
createFormInstance	Creates a new Account form instance (CRM_ACCOUNT). Values must be sent for all form fields. The ones that want to be omitted must be sent empty ("field_name": null).
updateFormInstance	Modifies the fields values in an Account form instance (CRM_ACCOUNT). Values must be sent for all form fields. The ones that want to be omitted must be sent empty ("field_name": null).
updateValuesofFormIns-tance	Modifies only the dsDescription field value in an Account form instance (CRM_ACCOUNT).
deleteFormInstance	Deletes an Account form instance (CRM_ACCOUNT).
getFormInstance (by filte-ring through a quantifiable field)	Returns a list with all the Account form instances (CRM_ACCOUNT), filtering through quantifiable fields.
getFormInstance (with the INCLUDE clause)	Returns a list with all the Account form instances (CRM_ACCOUNT), using inclusion clauses.
getFormInstance (with filter, paging, sorting and number of records per page)	Returns a list with all the Account form instances (CRM_ACCOUNT) with filter, paging and sorting options.

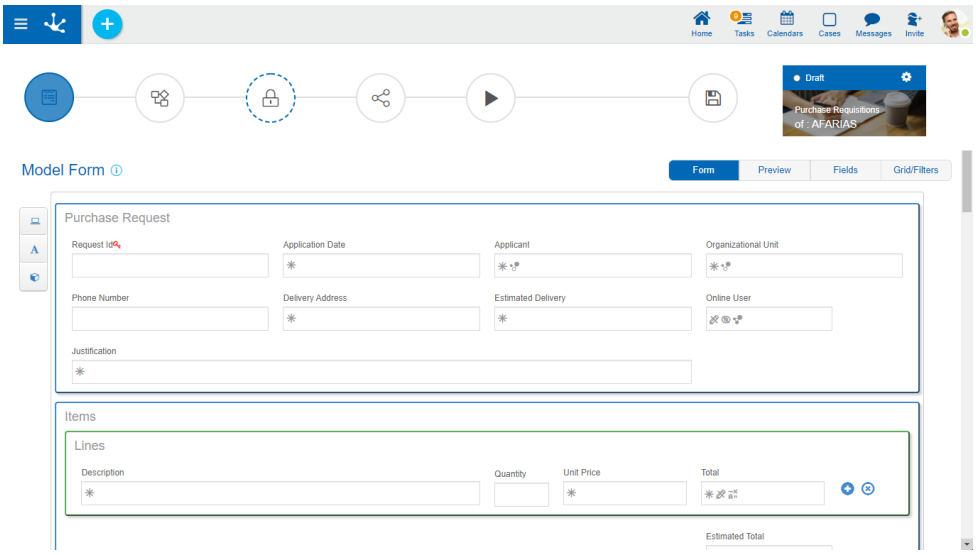
3.7.1.5. Cases Examples

These examples show how to create a case of a process, which is modeled using an agile forms tem-plate and allows you to execute an activity of the same.

Agile Form Modeling for Examples

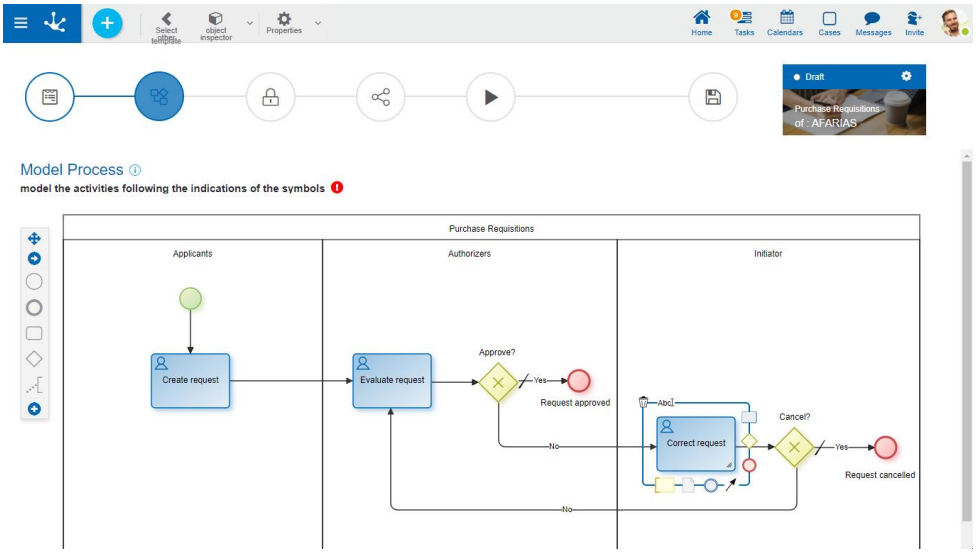
Step 1

Model the "Rest API Test" agile form based on the "Purchase Requisitions" forms template.



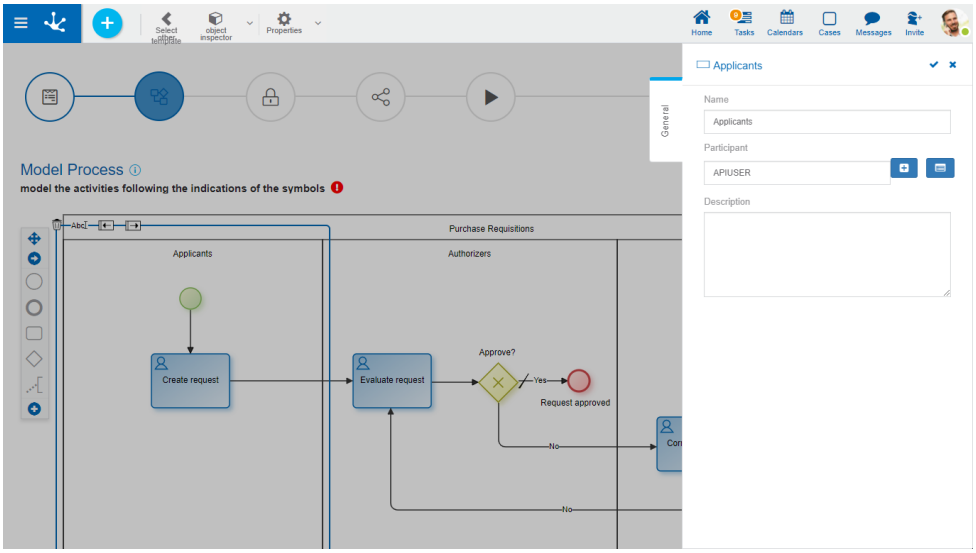
Step 2

Model the process related to the agile form "Rest API Test" based on the "Approval 1 Level" process template.



Step 3

Select the previously created user as "Applicants" lane participant.



Cases Collection

Examples are implemented from Postman to execute each of the operations described in the following table. Previously, you must import the Cases collection

Operation	Description
getToken	Obtains token.
startCase	Creates a case for the created process. The process code and the version of the modeled process properties must be shown. The required fields of the first process activity called "Create request" must be shown and they must always be sent. Editable fields do not need to be sent.
executeCurrentActivity	Executes the “Correct request” activity by sending the “Telephone” and “Observations” fields and the first occurrence of the “lines.quantity” field. The "Correct" button of the activity is executed. In the uri, case number and activity number must be entered {{base_uri}}/cases/<NRO_CASO>/activities/<NRO_ACTIVIDAD>/execute

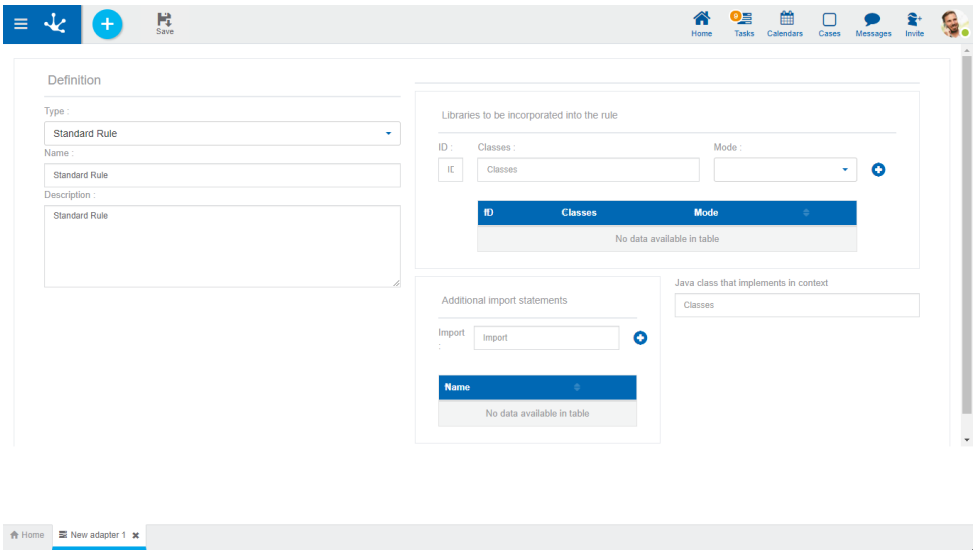
3.7.1.6. Rules Examples

This example shows how to create a rule, show parameters, and execute it.

Rules Modeling for Examples

Step 1

Register an adapter of Standard Rule type.

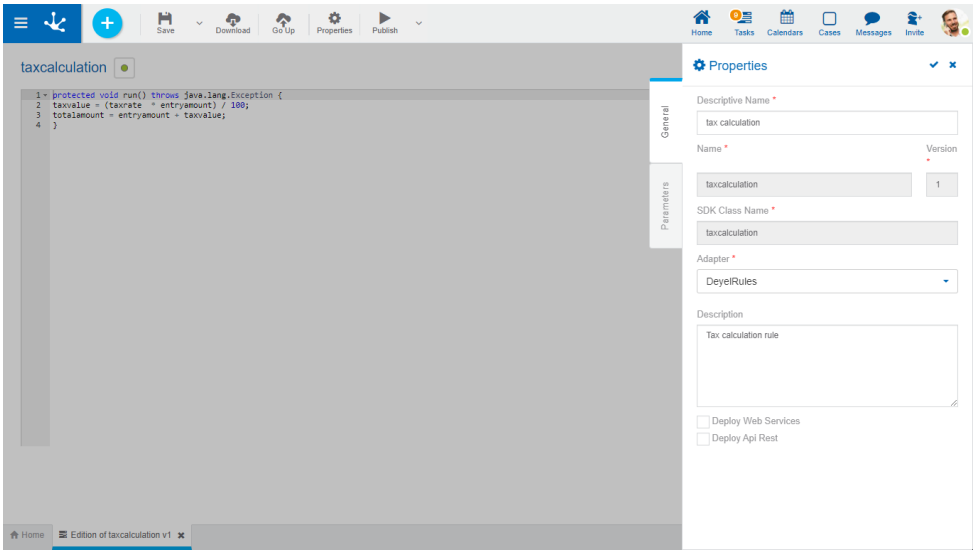


Step 2

Model a “VATcalculation” rule with the suggested values.

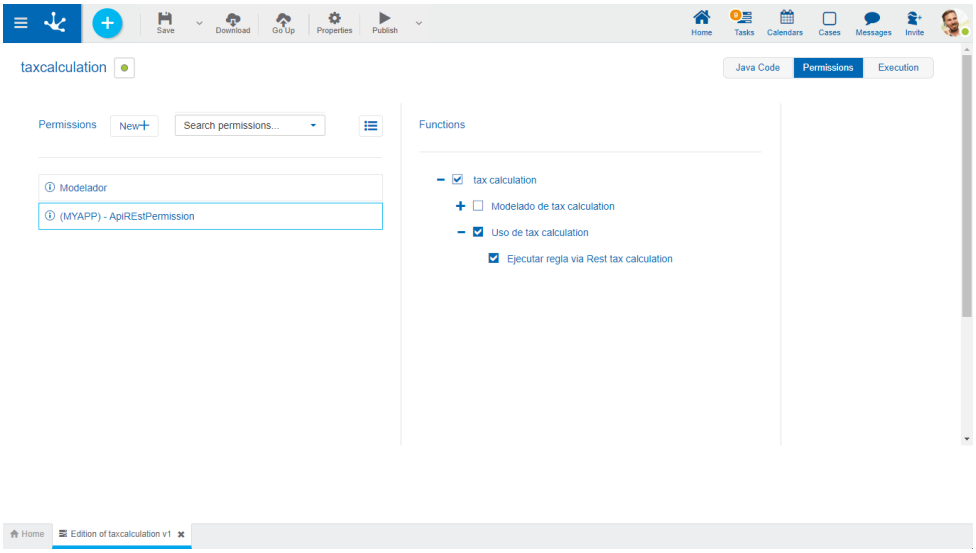
Descriptive Name	VATcalculation
Adapter	Standard Rule
JAVA code	VATvalue = (VATrate * Entryamount) / 100 ; Totalamount = Inputamount + VATvalue ;
Input Parameters	VATrate Inputamount
Output Parameters	VATvalue Totalamount

In the properties panel, on the "General" tab, select the Deploy Rest API property. This option allows the rule to be consumed as Rest API externally.



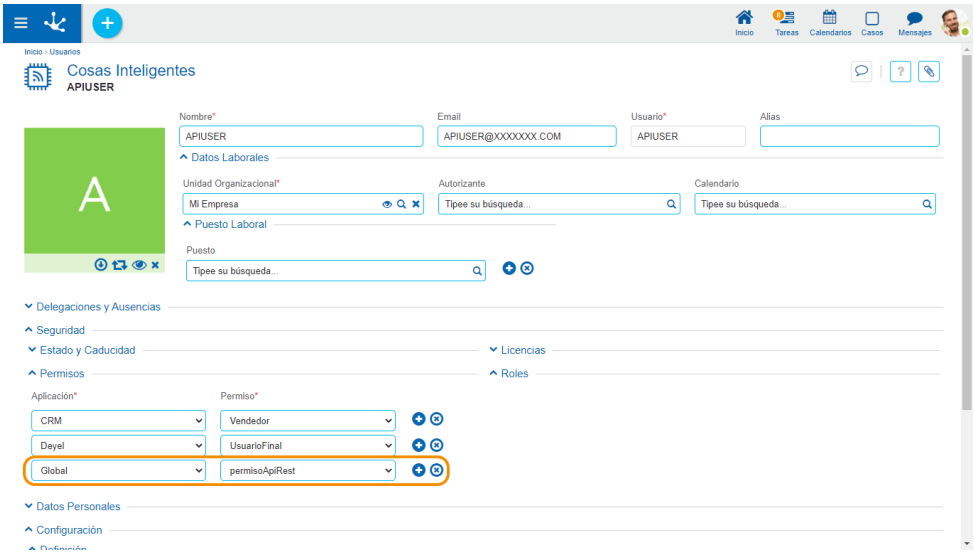
Step 3

In the "Permissions" option, add the security function "Execute rule via Rest + RuleName" to a new permission, for example "ApiRestPermission".



Step 4

Assign to the previously created user the generated permission.



Rules Collection

Examples are implemented from Postman to execute each of the operations described in the following table. Previously, you must import the Rules collection

Operation	Description
getToken	Obtains token.
getParameters	Returns the input and output parameters of the rule.
execute	Executes the rule "VATcalculation" sending the fields "Inputamount" and "VATrate". Returns "VATvalue" and "Totalamount".

3.8. Enterprise Social Network - Tedis

Tedis es una red social empresarial orientada a mejorar la comunicación y colaboración dentro de la organización. Integrada a **Deyel** brinda un gran potencial a la hora de relacionar equipos de trabajo, procesos y casos con una mensajería ágil, rápida y muy intuitiva.

Con **Tedis**, en **Deyel** se dispone de:

- Chats
- Comentarios asociados a objetos
- Ejecución de Comandos

Por medio de los **chats**, los usuarios pueden intercambiar mensajes libremente, recibiendo notificaciones instantáneas y agilizando de esta forma el intercambio de información.

Asociar comentarios a objetospermite que los usuarios registren comentarios en un chat grupal, que se encuentra asociado a los distintos objetos de **Deyel**:

- Asociar comentarios a **casos** permite que los usuarios registren comentarios sobre un caso particular, quedando esta información como parte del caso y disponible para otros usuarios interesados. La participación de múltiples usuarios que incorporan y comparten sus comentarios constituye una poderosa herramienta que promueve la colaboración y hace más eficientes los procesos. Este mecanismo permite por ejemplo, realizar consultas a otros usuarios sin necesidad de entregarles el caso para que intervengan formalmente.

- Los comentarios relacionados a la definición de **procesos** habilitan un canal de comunicación directo entre los participantes del caso y los responsables de dichos procesos, con el objetivo de mejorar su definición.

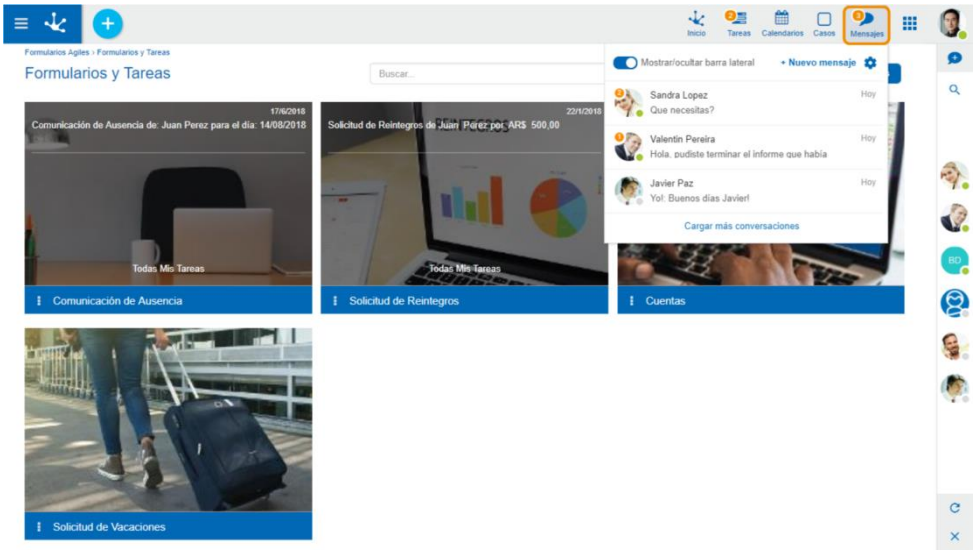
- Relacionar comentarios a **formularios** permite que los usuarios puedan interactuar, colaborar y discutir sobre instancias de formularios. Además, se puede consultar las conversaciones de los casos asociados a formularios, con el fin de mantener un seguimiento de los formularios, informarse de las interacciones de los usuarios y también aportar asesoramiento u opiniones de su uso.

  A través de suscripciones, cada usuario puede optar por recibir las novedades sobre los comentarios relacionados al objeto de interés. Esto permite hacer un seguimiento ágil de nuevos comentarios sin tener que consultar puntualmente un caso o un formulario.

Mediante el uso de **comandos**, es posible iniciar la ejecución de un proceso desde una ventana de chat, con la posibilidad de ejecutarlo íntegramente desde una ventana de chat utilizando **mensajes accionables**. Para ello, solo basta con definir un proceso con cuyo evento de inicio sea por comando, para luego asociar el comando del proceso a uno o varios bots del sistema. Los mensajes accionables permiten interactuar con el proceso mostrando los campos del formulario asociado en un mensaje instantáneo, con el fin de que se puedan ejecutar las actividades de un proceso desde una ventana de chat.

3.8.1. Chats

El acceso al chat se encuentra en la parte superior del Portal de Usuarios, haciendo clic sobre el ícono "Mensajes". El número que se encuentra sobre este ícono indica la cantidad total de mensajes sin leer del usuario.



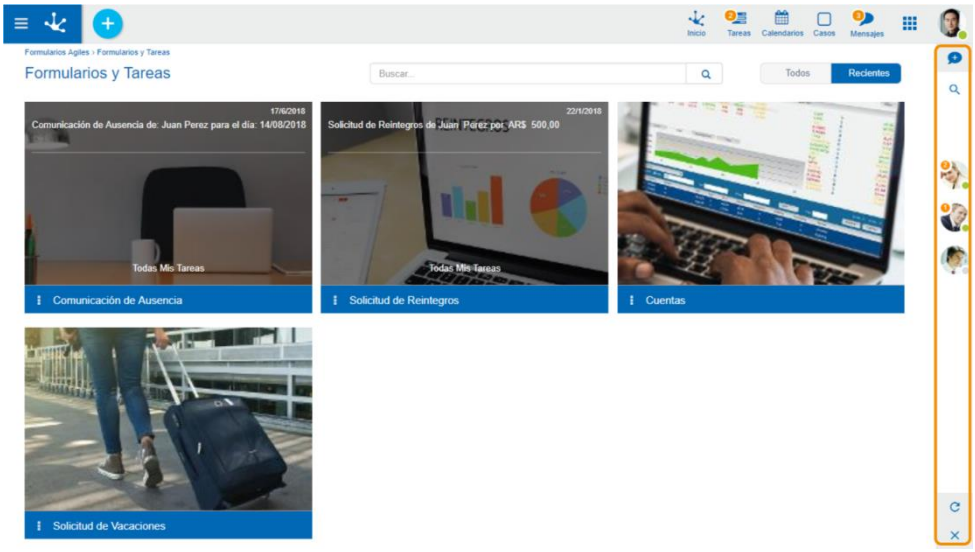
Al hacer clic sobre el ícono de mensajes, se puede observar:

Preferencias de usuario

Permite configurar las notificaciones de las conversaciones. Esto se explica en detalle en el tópico [Preferencias de usuario](#).

Mostrar/ocultar barra lateral de chats

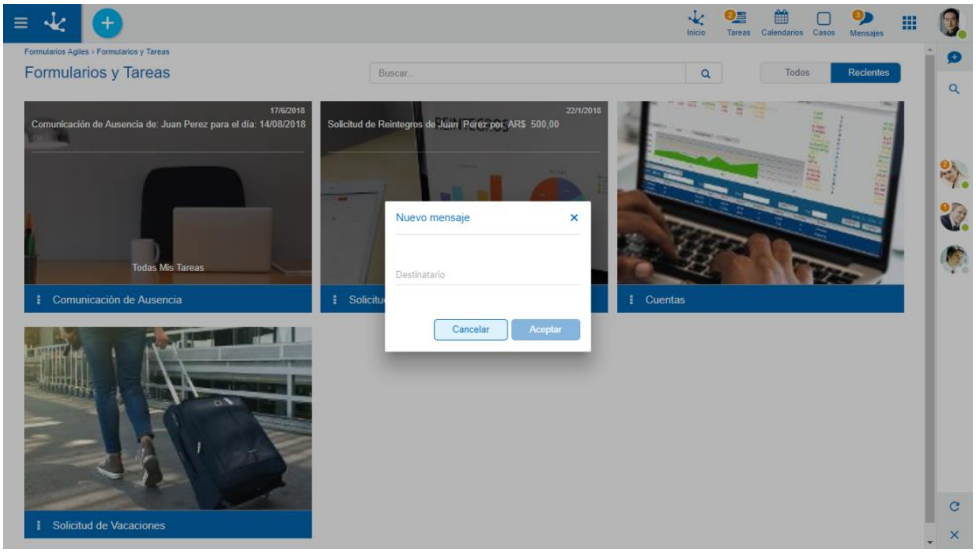
Dependiendo de si se encuentra activada esta opción se mostrará u ocultará la barra lateral de chats, la cual contiene la lista completa de usuarios y chats.
Por defecto esta barra se encuentra activa, mostrando en una primera instancia el listado de chats en caso de que existiesen, o el listado de usuarios en caso contrario.



Presionando el ícono de búsqueda se expande la barra lateral mostrando información detallada.En el tópico [Barra de Chats](#) se explica en detalle sus características y funciones.

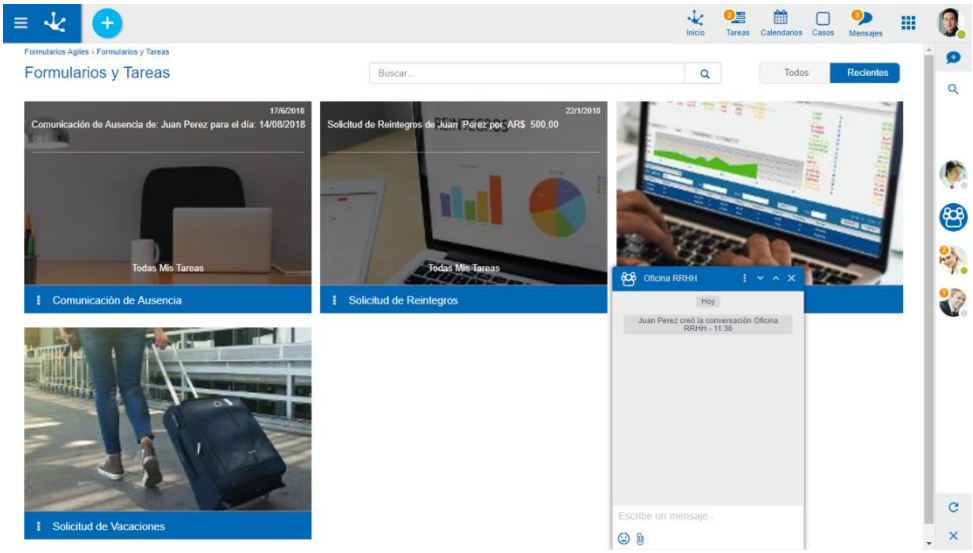
Nuevo Mensaje

Abre una ventana de envío de un nuevo mensaje.



Se debe identificar el destinatario del mensaje. Ingresando su nombre o apellido un asistente ayudará en la selección desde la lista de usuarios.

Se puede agregar la cantidad de destinatarios que se desee, generando así un nuevo chat grupal. Una vez ingresado más de un destinatario, se habilita el campo título del nuevo chat grupal, el cual es obligatorio.

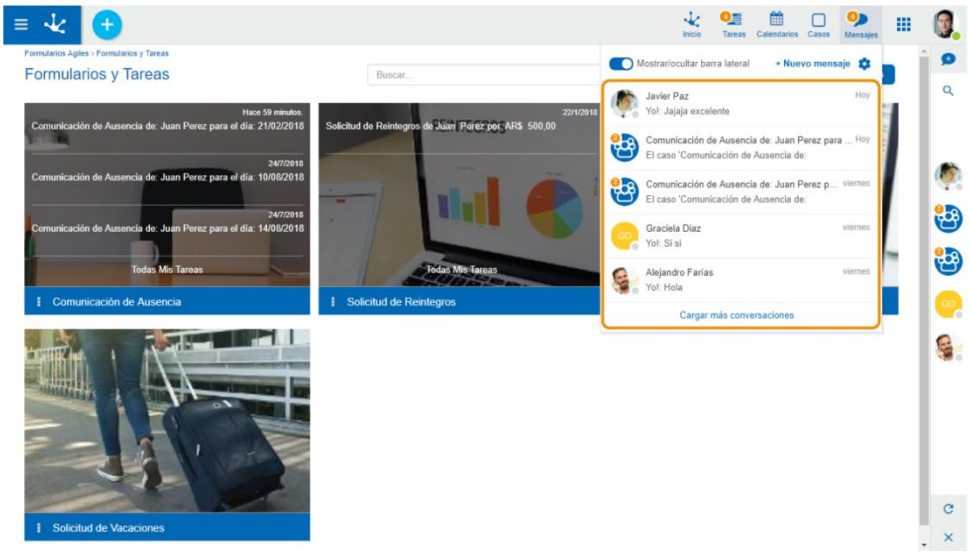


Presionando el botón "Aceptar" se crea el grupo y se abre el chat, permitiendo el envío de mensajes.

En la sección Ventana de Chat se explica en detalle toda su funcionalidad.

Lista de chats

Muestra una lista con las últimas conversaciones del usuario. Se muestran en primer lugar las más recientes.



El botón "Cargar más conversaciones" permite incorporar conversaciones más antiguas a la lista.

Permiso de Usuario invitado

Deyel presenta entre sus permisos, uno llamado “Usuario invitado en Tedis”, el cual limita las acciones que puede realizar un usuario en la red social.
En caso de que un usuario tenga este permiso en su perfil, no podrá realizar las siguientes acciones:

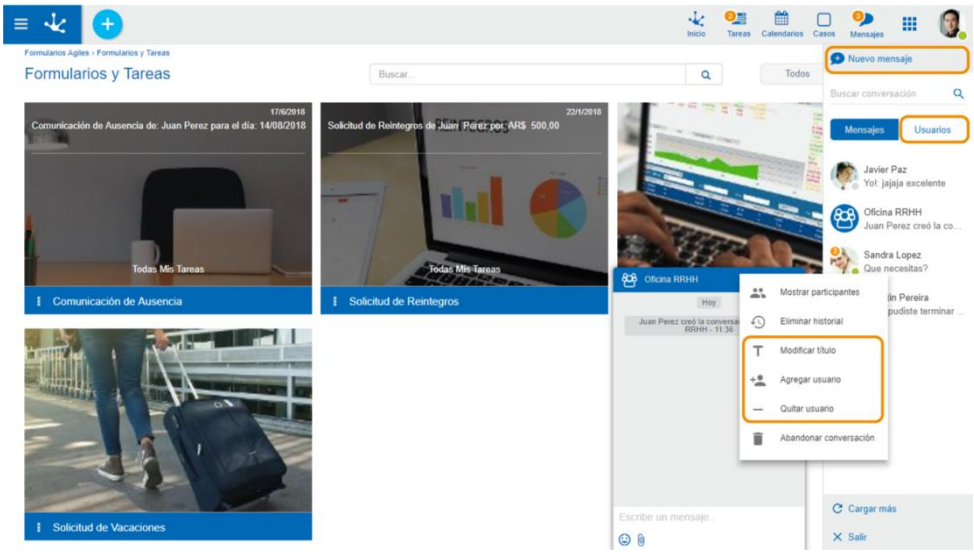
- **Ver lista de usuarios**
  La pestaña de la Barra de chats que muestra la lista de usuarios del sistema junto a su estado de conexión no estará disponible para este tipo de usuarios.

- **Crear nuevos chats**
  El ícono para enviar un nuevo mensaje no estará disponible, con lo cual el usuario solo podrá enviar mensajes a chats que ya existan en su lista de conversaciones. Para el caso de los chats privados, otro usuario deberá enviarle un mensaje, mientras que para los chats grupales deberá ser parte del grupo.

- **Agregar y eliminar usuarios en chats grupales**

- **Modificar el título de un chat grupal**

En la siguiente imagen se muestran resaltadas las funciones no disponibles:



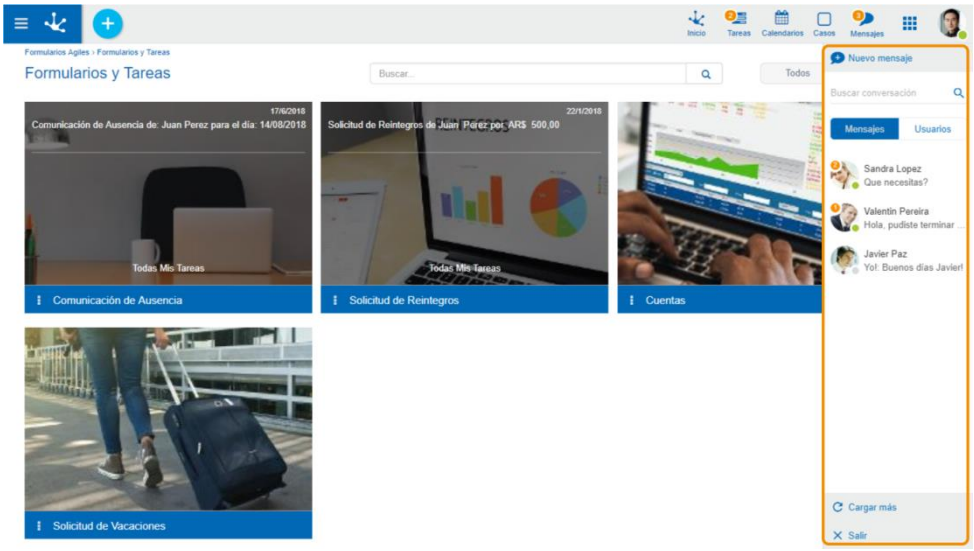
En resumen, el usuario con este permiso estará limitado a poder responder a mensajes de chats en los que tenga mensajes previos.

> *En caso de que se elimine el historial del chat, este no podrá ser accedido hasta que se reciba un nuevo mensaje.*

3.8.1.1. Barra de Chats

Muestra la lista de usuarios y la lista de chats en que el usuario participa.

Para expandir la barra de chats, se debe hacer clic en el ícono de búsqueda .



> *La barra volverá a su tamaño original cuando el mouse no se encuentre situado dentro de ella.*

Nuevo Mensaje

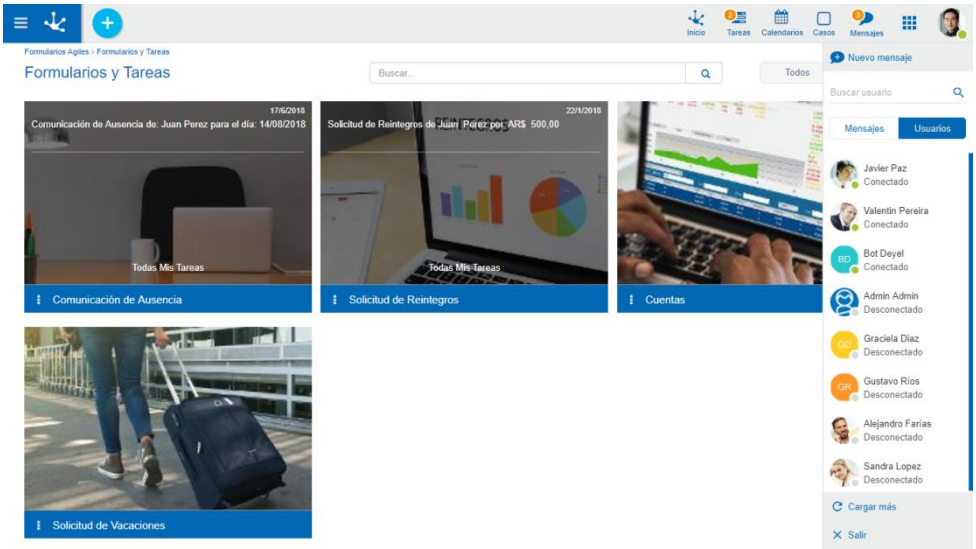
Permite en envío de un nuevo mensaje. Para más información sobre su uso, ver Nuevo mensaje.

Lista de Chats

En la pestaña de "Mensajes" se encuentran todos los chats del usuario ordenados cronológicamente, iniciando con los más recientes.

Lista de Usuarios

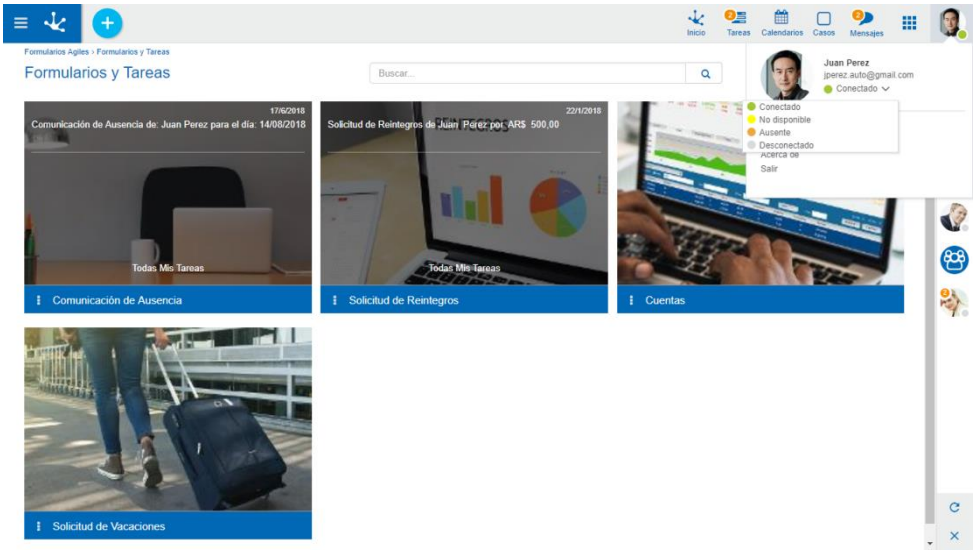
En la pestaña de “Usuarios” se pueden ver todos los usuarios de **Deyel**. Se visualizan primero aquellos que no estén desconectados, y luego todos los desconectados, ambos ordenados alfabéticamente.



En la lista de usuarios y en los chats privados, se muestra el estado de conexión junto al nombre del usuario. Los posibles estados de conexión son:

- Conectado
- No disponible
- Ausente
- Desconectado

El estado de conexión puede ser modificado desde el perfil de usuario que se encuentra en el margen superior derecho.



Buscar Usuarios / Chats

Para el caso de la lista de usuarios, permite la búsqueda de usuarios por nombre y apellido, mientras que para la solapa de mensajes, la búsqueda se realiza por el contenido de un mensaje, o el nombre de un chat grupal.
La lista de usuarios o chats se filtra a medida que se ingresa texto en este campo.

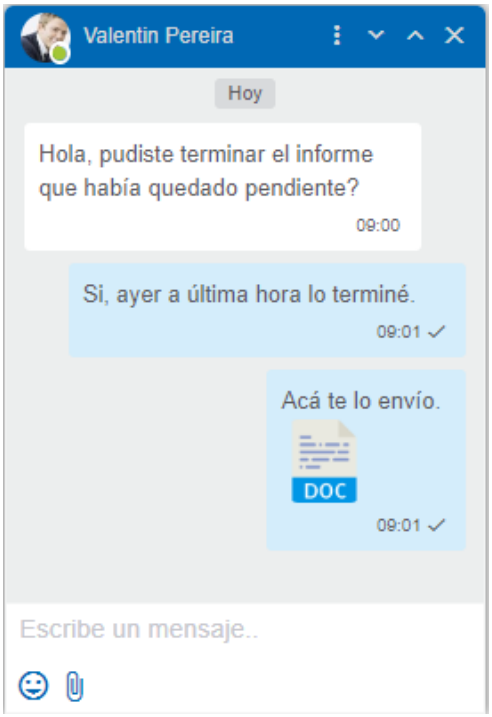
Cargar más chats

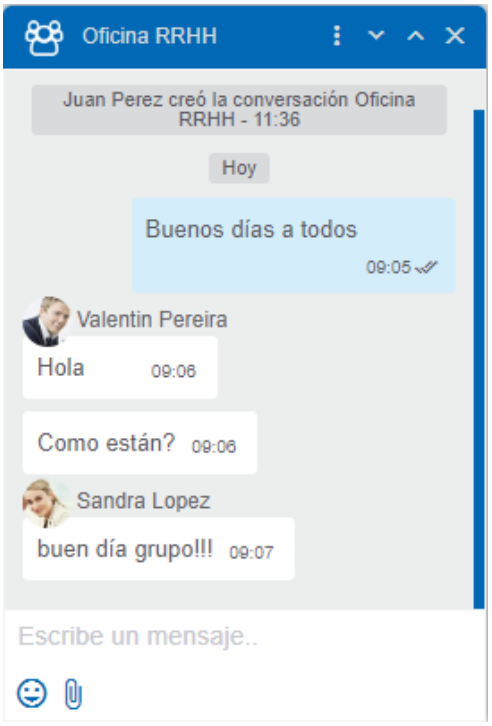
Presionando sobre este ícono se incrementa la lista de chats, mostrando conversaciones más antiguas debajo de las demás.

3.8.1.2. Ventana de Chat

Tiene como objetivo el intercambio de mensajes en tiempo real entre dos o más usuarios. Dependiendo de si el chat es privado o grupal, la funcionalidad varía.

Chat Privado **Chat Grupal**





Como se puede observar en la imagen, los chats grupales tienen el nombre del grupo junto a su imagen.

Estados de los mensajes

Cada mensaje enviado tiene un estado a su derecha que indica si ya fué enviado o leído.

Los estados posibles son:

- Enviando: el mensaje aún no ha sido enviado.
- Enviado: el mensaje ha sido enviado y fue recibido por el destinatario.
- Leído: el mensaje ha sido leído por el destinatario.

Para el caso de los chats grupales, el estado solo aparecerá una vez que todos los integrantes del grupo hayan leido el mensaje. Manteniendo el mouse sobre el estado del mensaje, se podrá obtener información sobre quienes ya han visto el mensaje.

Menú

Se accede presionando el ícono que se encuentra en la parte superior de la ventana del chat. Funciones:

Eliminar historial: elimina todos los mensajes de la conversación para ese usuario. Una vez que se elimina el historial de un chat grupal, este no podrá ser accedido hasta se reciba un nuevo mensaje.

Agregar usuario: para los chats privados, esta opción permite agregar usuarios a la conversación, creando de esta forma un nuevo chat grupal. La ventana para agregar usuarios es similar a la que se utiliza para crear un Nuevo Mensaje.
Para los chats grupales, simplemente agrega usuarios al chat. Cualquier participante del grupo puede añadir nuevos participantes.

Los chats grupales además pueden:

Mostrar participantes: muestra el listado de usuarios pertenecientes al chat, junto con su estado de conexión. Al hacer click sobre la imagen de un usuario, se abre la ventana de chat de dicho usuario.

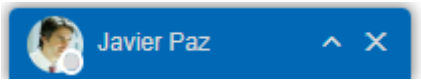
Modificar título: abre una ventana para la modificación del título de la conversación.

Quitar usuario: permite eliminar uno o más participantes de un chat grupal. Ingresando su nombre o apellido un asistente ayudará en la selección desde la lista de usuarios que forma parte del chat.
Sólo el creador del chat puede eliminar participantes.

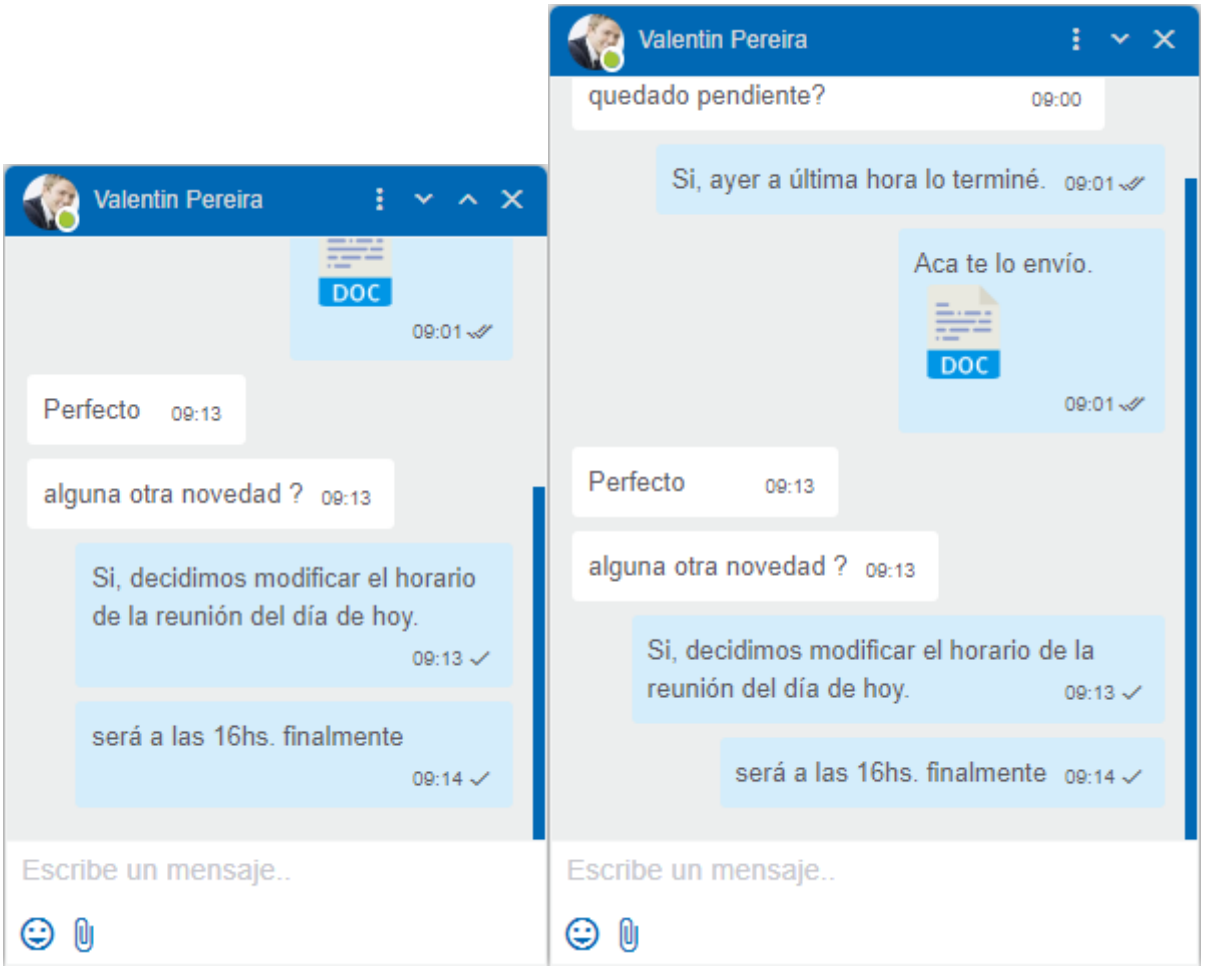
Abandonar conversación: permite al usuario desvincularse de un chat grupal. Esto implica que ya no tendrá acceso al chat ni a sus mensajes.

Achicar y Expandir ventana

Minimiza la ventana de chat, mostrando únicamente la información de la barra superior del chat.



Para el caso de las ventanas minimizadas, presionar este ícono maximizará la ventana, visualizando nuevamente los mensajes.
En caso de que se requiera que la ventana de chat sea aún más grande, se podrá utilizar este mismo ícono para incrementar su tamaño, pudiendo volver a su tamaño original presionando el ícono de achicar ventana.

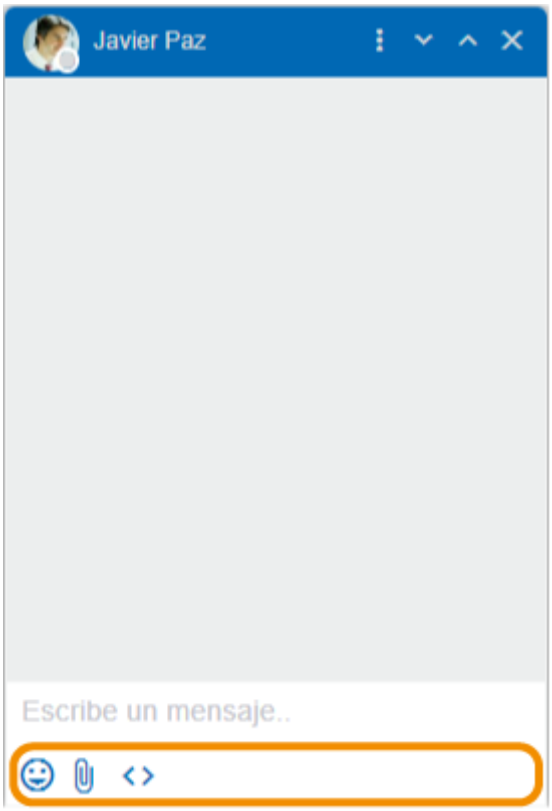


Cerrar ventana

Permite cerrar la ventana de chat, eliminándola del listado de conversaciones abiertas.

Herramientas asociadas a mensajes

Las herramientas que se describen a continuación están asociadas al envío de mensajes dentro de una conversación:

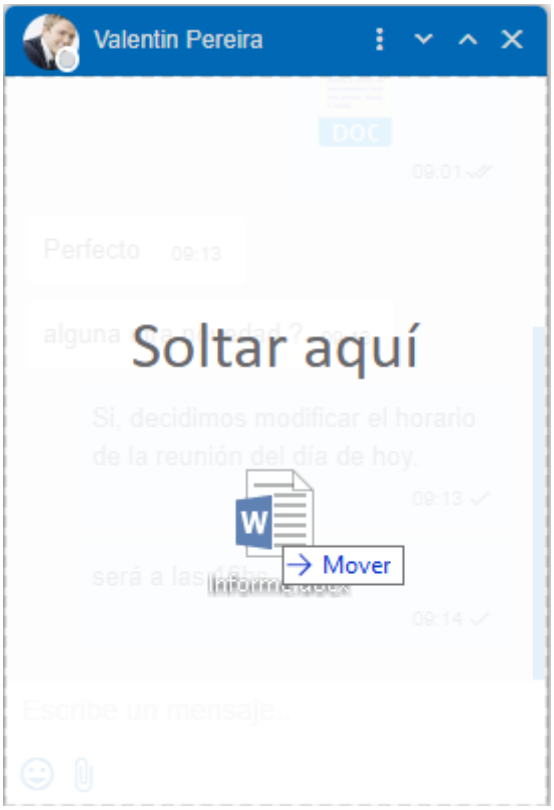


Emojis

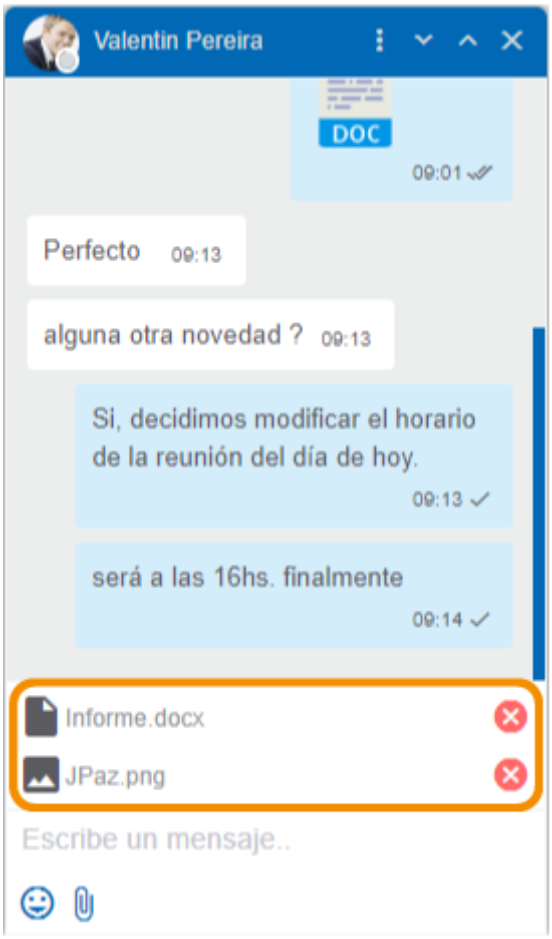
Permite la inclusión de emojis en los mensajes del chat, ya sea haciendo clic sobre el ícono de emojis y seleccionándolos, o bien escribiendo la clave del mismo.

Archivos adjuntos

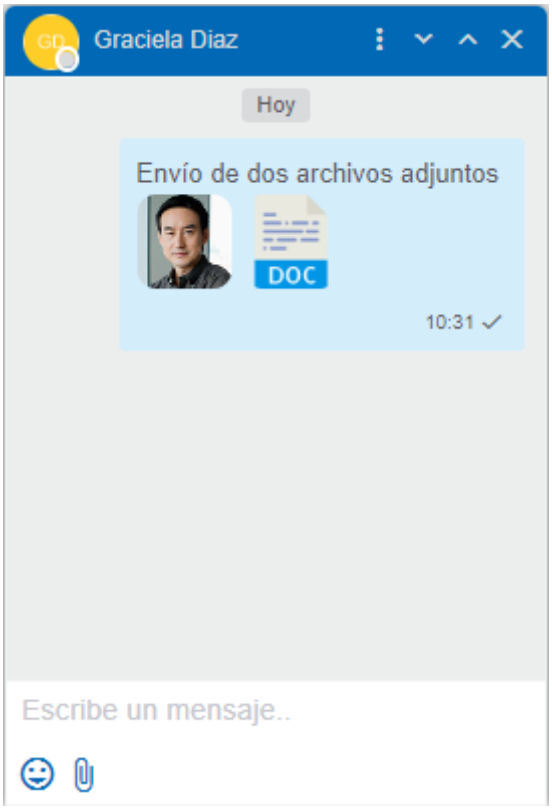
Permite asociar uno o más archivos adjuntos. Para ello se debe presionar el ícono de adjuntos, para luego seleccionar los archivos a enviar. Otra alternativa consiste en arrastrar los archivos hacia la ventana de chat, siempre y cuando esta se encuentre expandida, y soltarlos sobre la ventana.



Luego de adjuntar el archivo, se muestra el listado de archivos que ya se encuentran adjuntos en el mensaje que se desea enviar, permitiendo adjuntar un máximo de 6 archivos por mensaje.
Entre los archivos no permitidos por cuestiones de seguridad se encuentran los ejecutables (archivos con extensión .exe), y los que contengan extensión msi.



El mensaje con adjuntos podrá ser enviado con o sin texto, presionando la tecla Enter. Haciendo clic sobre el ícono que se encuentra a la derecha de cada adjunto permite que se elimine de la lista de adjuntos a enviar.



Para el caso de las imágenes, se muestra una pre-visualización una vez que fueron enviados, mientras que para el resto de los archivos, se mostrará un ícono que identifique su tipo.
Haciendo clic sobre los adjuntos, comenzará su descarga.

Si por algún error no fuese posible adjuntar algún archivo, se mostrará un ícono a la derecha del archivo para volver a intentarlo.

> *El tamaño máximo permitido de cada archivo adjunto dependerá del valor de la variable Tamaño máximo en MB del archivo al que se le puede hacer upload de la configuración de **Deyel**, como así también del valor máximo de almacenamiento de la base de datos utilizada.*

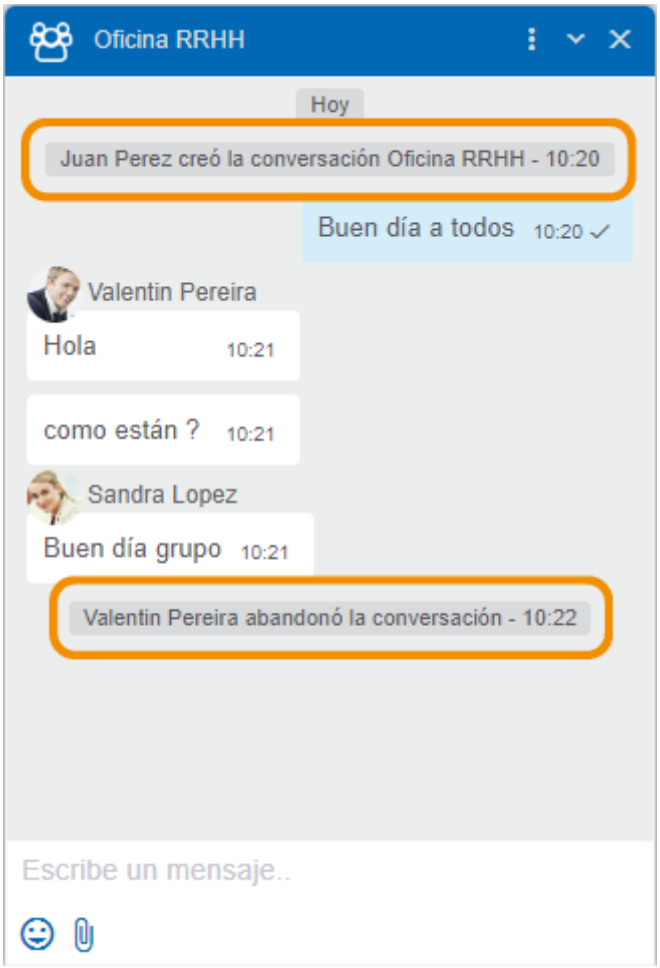
Comandos

Permite la ejecución de comandos previamente definidos. Este ícono puede o no estar presente dependiendo de varios factores. Esto se explicará en detalle en el tópico Bots y Comandos.

3.8.1.2.1. Eventos

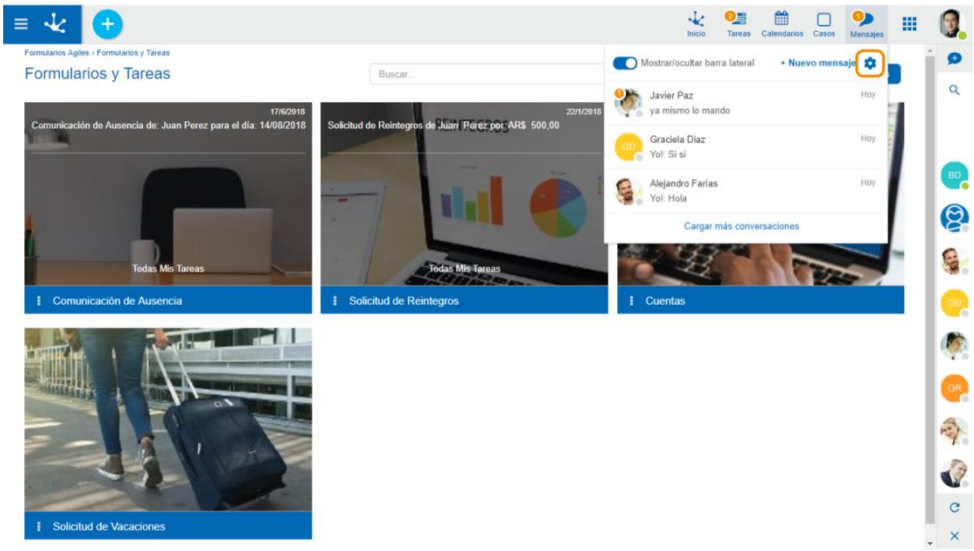
Los eventos son un tipo especial de mensajes que se generan ante la ocurrencia de un suceso.

Estos eventos son generados de forma automática por el sistema cuando ocurre algo que debe ser informado a los integrantes de un chat, como por ejemplo el evento de que un usuario ha abandonado una conversación.

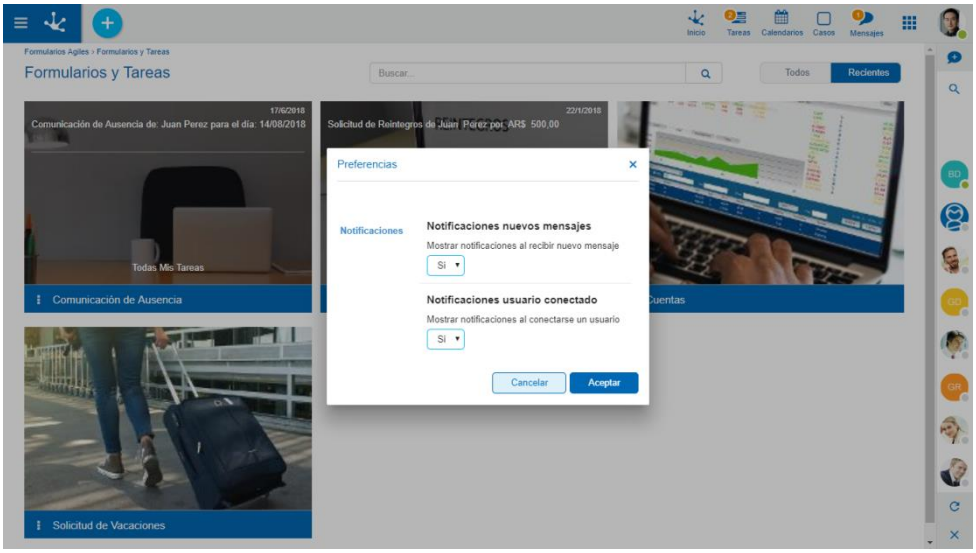


3.8.1.3. Preferencias de usuario

Deyel ofrece la posibilidad de habilitar o deshabilitar las distintas notificaciones de una forma muy sencilla. Para acceder a las preferencias de notificaciones, se debe hacer clic en las opciones del chat en la parte superior del Portal de Usuarios y luego, hacer clic en el icono de preferencias .



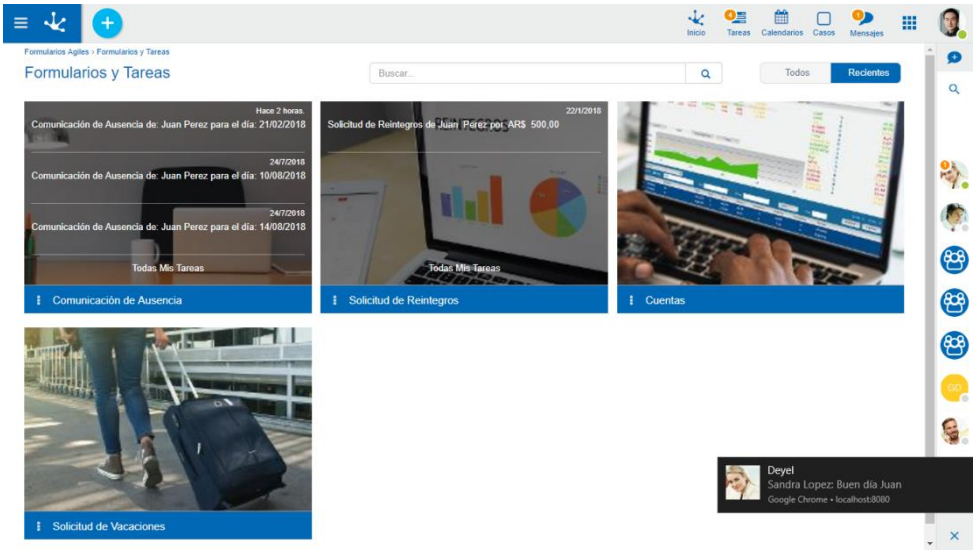
Dentro de esta ventana, **Deyel** permite que el usuario pueda configurar las notificaciones del chat a su gusto.



Notificaciones nuevos mensajes

Permite al usuario habilitar/deshabilitar las notificaciones al recibir un nuevo mensaje. Los valores posibles son: "Si" (habilitado), "No" (deshabilitado). El valor predeterminado es "Si".

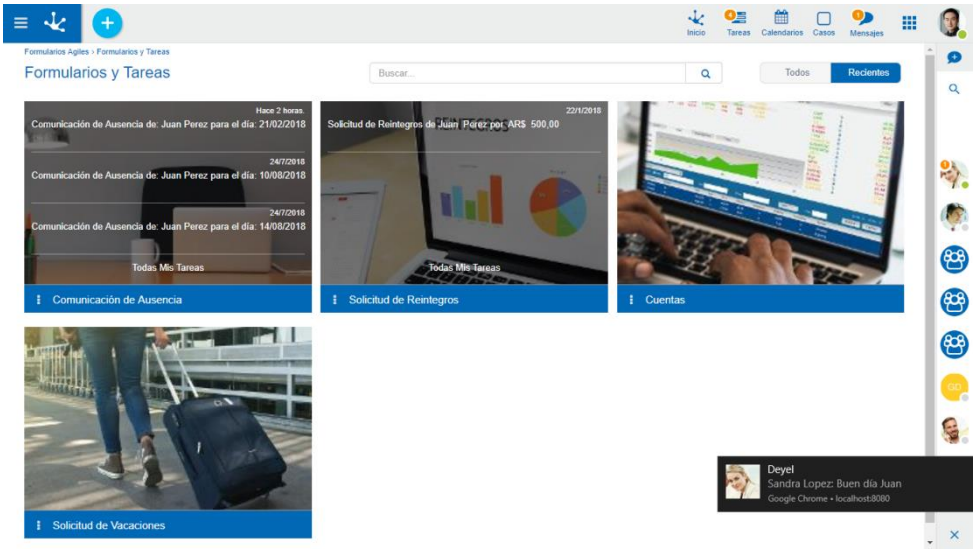
Si el usuario habilita este tipo de notificaciones, se muestra en la parte inferior derecha de la pantalla una notificación cada vez que el usuario reciba un nuevo mensaje.



Notificaciones usuario conectado

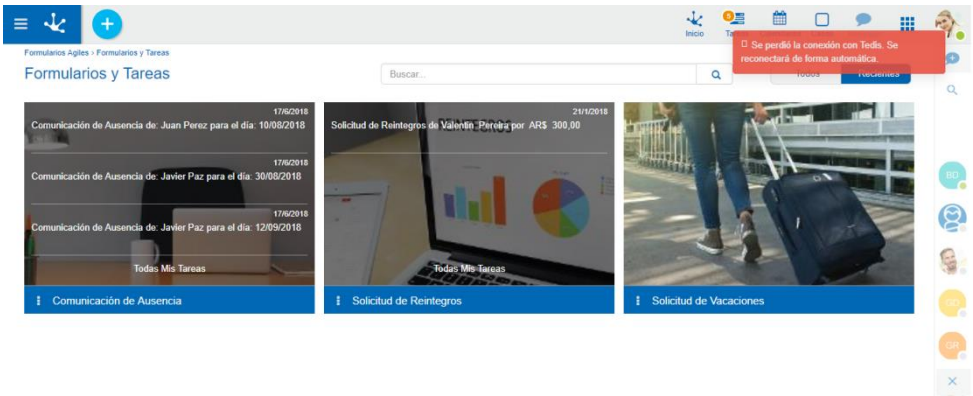
Permite al usuario habilitar/deshabilitar las notificaciones al conectarse un usuario. Los valores posibles son: "Si" (habilitado), "No" (deshabilitado). El valor predeterminado es "Si".

Si el usuario habilita este tipo de notificaciones, se muestra en la parte inferior derecha de la pantalla una notificación cada vez que un usuario se conecta al sistema.



3.8.1.4. Pérdida de conexión

Para los casos en los que el usuario pierda la conexión con **Deyel**, ya sea por falta de internet o cualquier otro tipo de inconveniente, la red social se pondrá en modo "desconectado". Se deshabilitarán todas las funciones de la mensajería, informando del problema por medio de una alerta.



Una vez que se recupere la conexión, todas las funciones de la red social volverán a estar disponibles, sin necesidad de tener que recargar el navegador.

Para los casos en que se detecte algún tipo de error en el funcionamiento de la mensajería que no estén relacionados con la conexión a **Deyel**, se informará con una alerta, indicándole al usuario que reintente la operación más tarde.

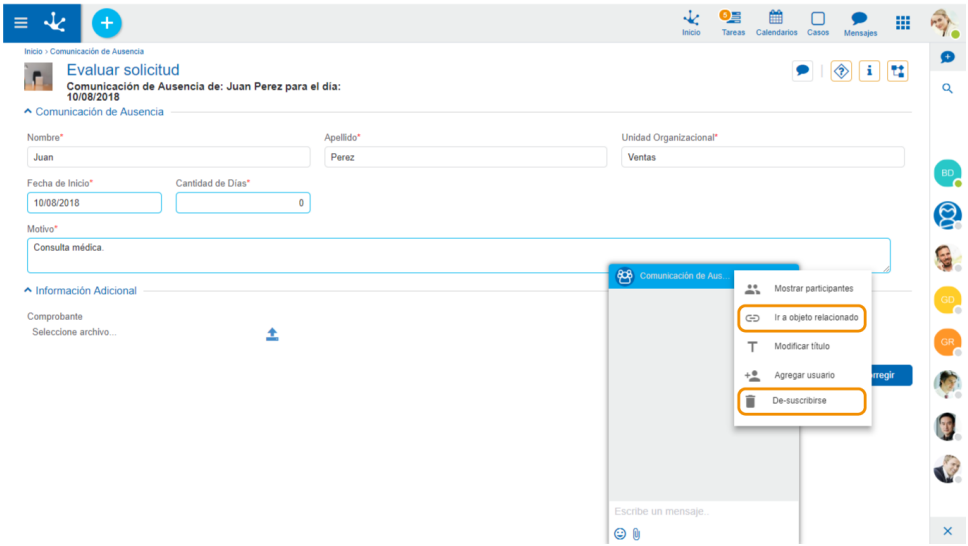
3.8.2. Comentarios asociados a Objetos

Dentro de la ejecución de los casos, y de las instancias de formularios, es común que la persona que se encarga de este trabajo deba realizar una consulta con sus pares para poder completar la tarea.

Esa comunicación e información adicional para la toma de decisiones, normalmente existe por fuera del sistema (mediante llamados telefónicos, emails, etc).

Deyel propone un chat exclusivo para estos objetos, es decir para los casos de los procesos, y para las instancias de formularios, en el cual los usuarios pueden registrar comentarios sobre estos objetos.

Estas conversaciones, tendrán un comportamiento diferente a las conversaciones grupales mencionadas anteriormente.



Las opciones "Mostrar participantes", "Modificar título" y "Agregar usuario" funcionan de la misma forma que en los Chats grupales.

- Ir a objeto relacionado
  Esta opción permite al usuario ir directamente a la consulta del objeto relacionado a la conversación.

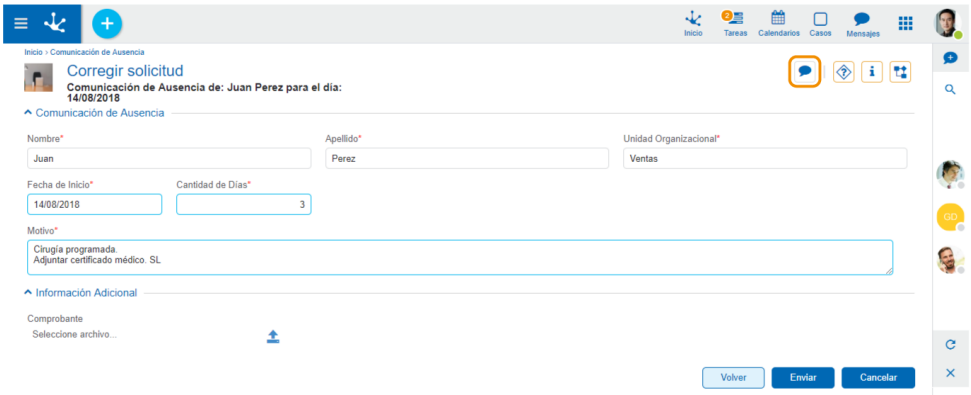
- Suscribirse / De-suscribirse
  Cualquier usuario del sistema que pueda consultar un objeto, podrá participar de los comentarios del mismo. Aquellos usuarios que deseen estar al tanto sobre cualquier novedad que surja en los comentarios, pueden utilizar la función de **suscripción**.
  Un usuario suscripto a los comentarios es notificado cada vez que se ingrese un nuevo mensaje en este chat, ya que es un participante del chat asociado al objeto. Además, puede agregar al chat a otros usuarios para que también participen y sean notificados sobre nuevos comentarios.
  Un usuario suscripto podrá también abandonar el chat si ya no desea recibir notificaciones sobre el caso.

Comentarios asociados a Casos

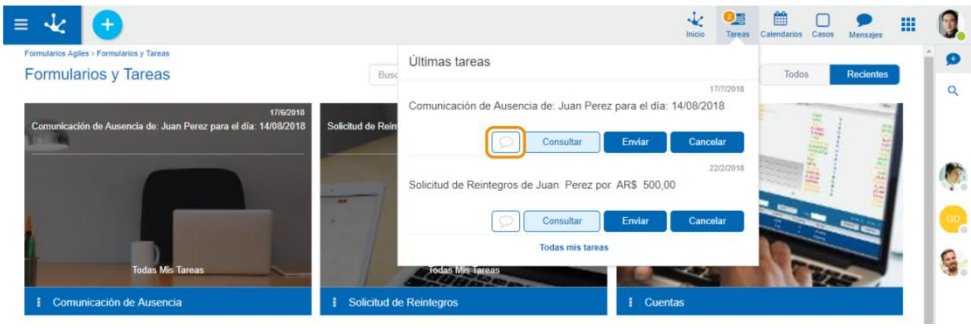
Mantener una conversación propia del caso simplifica el modelado del proceso, ya que innumerables situaciones podrán resolverse mediante la interacción de los usuarios más que con el modelado del proceso, además de nutrir al caso con información utilizada para la toma de decisiones.

Los comentarios asociados a un caso pueden ser accedidos desde:

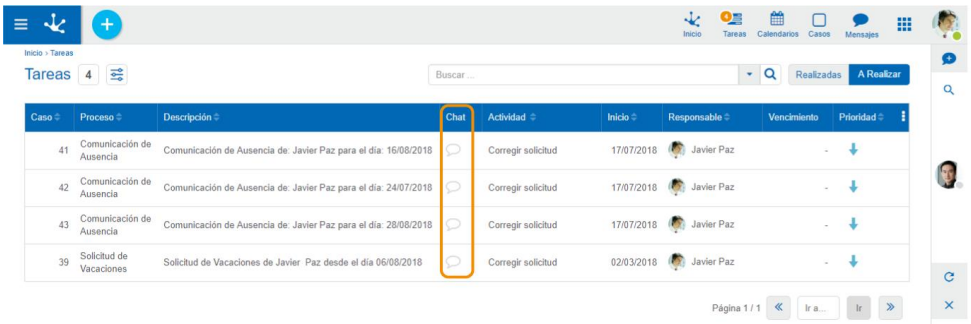
- La consulta puntual del caso.



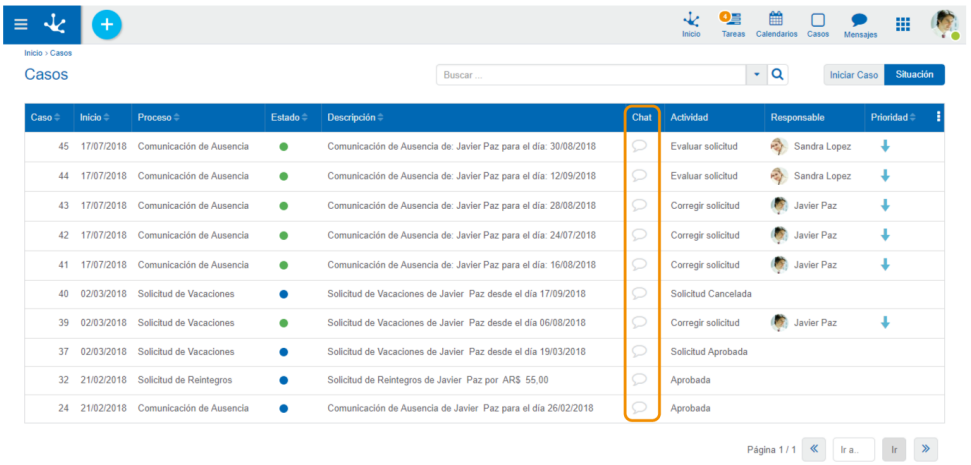
- Desde la lista de últimas tareas.



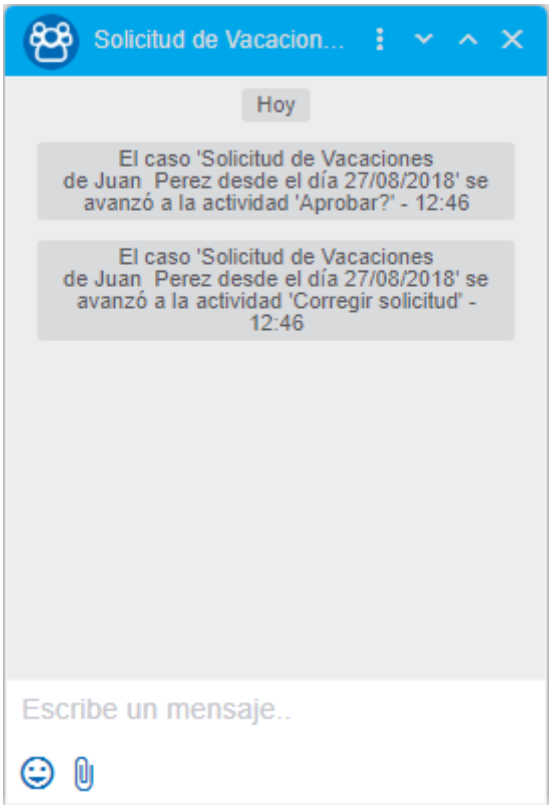
- Desde la lista de tareas.



- Desde la lista de casos



Cuando el caso se va moviendo dentro de las actividades, se genera un evento informando la actividad a la cual se movió el caso, como se muestra a continuación:

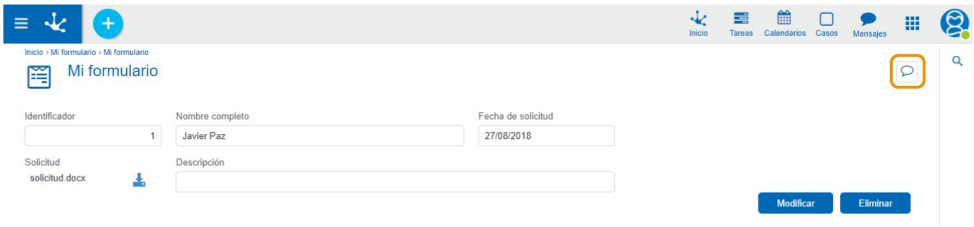


Comentarios asociados a Formularios

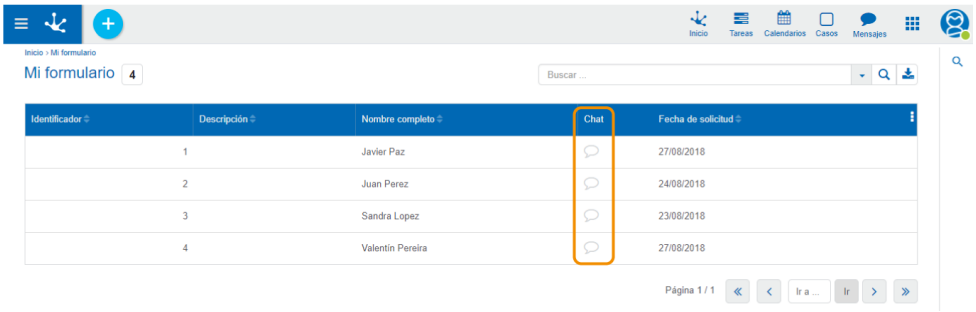
Deyel permite crear conversaciones de instancias de formularios, para que los usuarios interactúen y colaboren sobre temas específicos de un formulario.
Cualquier usuario que tenga acceso, puede suscribirse a la conversación, y ver u opinar sobre el mismo.

Los comentarios asociados a una instancia de formulario pueden ser accedidos desde:

- La consulta puntual de la instancia de un formulario.



- Desde la lista de instancias de un formulario.



Aquellos usuarios que deseen estar al tanto sobre cualquier novedad que surja en los comentarios de un formulario, pueden **suscribirse**. Un usuario suscripto a los comentarios del formulario es notificado cada vez que se ingrese un nuevo comentario en este chat. Además, puede suscribir al chat a otros usuarios para que también participen y sean notificados sobre nuevos comentarios.
Un usuario suscripto podrá también abandonar el chat si ya no desea recibir notificaciones sobre el formulario.

La opción "**Ir a objeto relacionado**", mostrará la consulta de la instancia del formulario.

3.8.3. Bots y Comandos

Un comando es una instrucción específica definida para un Usuario, o una Cosa de tipo “ChatBot”, y se encarga de iniciar la ejecución de un proceso de negocios específico.
Un ChatBot, o Bot, es un tipo especial de usuario que se utiliza para la ejecución de comandos por medio de una ventana de chat.
Para más información sobre la definición de “Cosa” y su configuración y relación con los usuarios y comandos, ver Administración de usuarios.

El uso de comandos le permite al usuario iniciar la ejecución de un proceso desde una ventana de chat

Por ejemplo, supongamos que se necesita informar a todos los usuarios el horario en que se realizará el mantenimiento del sistema. Para ello, se cuenta con un comando ya disponible en en Bot de **Deyel**, que permite enviar un mensaje a todos los usuarios del sistema por medio de la mensajería instantánea.

El Bot de **Deyel** tiene entre sus comandos por defecto, uno que permite este tipo de notificaciones masivas. Dicho esto, lo único que tiene que hacer el usuario que requiera enviar el mensaje, es abrir un chat con el Bot, hacer clic sobre el ícono de comandos, y seleccionar el comando “Comunicar novedad”.

Como se puede apreciar en la imagen, al presionar el botón de comandos, se despliega una ventana con los comandos disponibles que pueden ser ejecutados sobre el Bot.

Al hacer clic sobre el comando se envía el mensaje, que es procesado por el Bot, para luego mostrar un mensaje accionable.

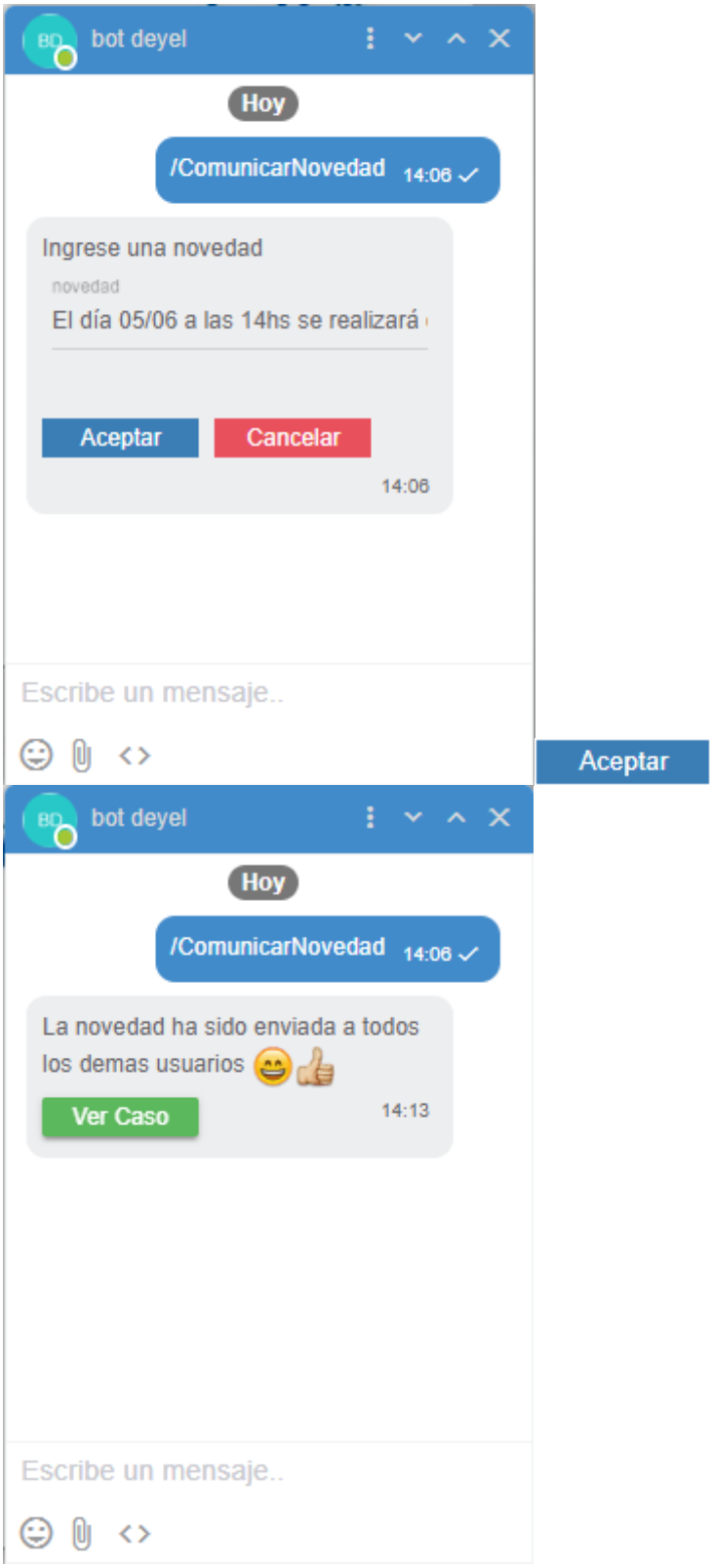
3.8.3.1. Mensajes accionables

Un mensaje accionable es un mensaje instantáneo de un chat, que además de texto, tiene campos de un formulario sobre los cuales se puede ingresar información, permitiendo ejecutar actividades desde el chat.

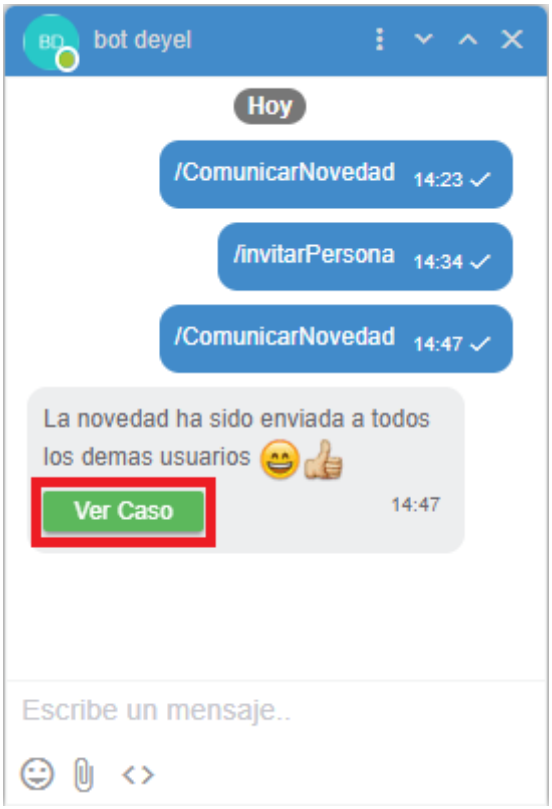
Siguiendo con el ejemplo del tópico anterior, el proceso asociado al comando "Comunicar novedad", tiene modelado en su primera actividad la ejecución de un formulario. Además, se definió un mensaje accionable en su primera actividad, con el campo del formulario que se utiliza para ingresar la novedad a enviar.

> *Los campos del mensaje accionable tendrán las validaciones especificadas tanto en el modelado del formulario, como de la actividad del proceso en la que se encuentre.*

El botón Aceptar es el botón por defecto de la ejecución de casos, mientras que el botón cancelar solo está disponible en la primera actividad del proceso, en caso de que se quiera cancelar la acción iniciada por el comando y eliminar el mensaje accionable. Presionando aceptar, el proceso se moverá de actividad, cuya función es enviar la novedad a todos los usuarios.



En caso de que las compuertas lógicas de un proceso definan botones, se mostrarán estos y no los botones por defecto.
Existe un botón extra que se encuentra presente en todos los mensajes accionables, a excepción del primero, que permite ir a la "Consulta del caso" que se está ejecutando.



3.9. BAM and Process Analysis

Phase 7: BAM

Deyel has a set of predefined reports that allow analyzing the operation of processes and their tasks, both historically (Analysis) and in real time (BAM).

Through these reports, it is possible to observe and understand the operation of processes and detect possible improvements.

Predefined reports summarize the behavior of processes and their tasks. They allow identifying what was done on time, behind schedule, deviations against maximum and expected durations, trends, participants performance, bottlenecks, etc.

The behavioral information is displayed in the form of a grid and through charts, always having the reference of the expected trend curves to contrast against reality. In this way, behavior can be graphically monitored. In all cases, the information can be drilled down, to reach the process level, task, executor, date range, case, etc.

Although the reports are predefined, multiple views of the information can be generated, when analyzing it by applying search criteria, such as time, priorities, participants, initiators, cases states and tasks, among others.

In the case of analysis in real time (real time), in addition to identifying the cases and tasks that are found on time and behind schedule, those that are found at riskare identified, thus providing tools to act on what is actually happening in the organization.

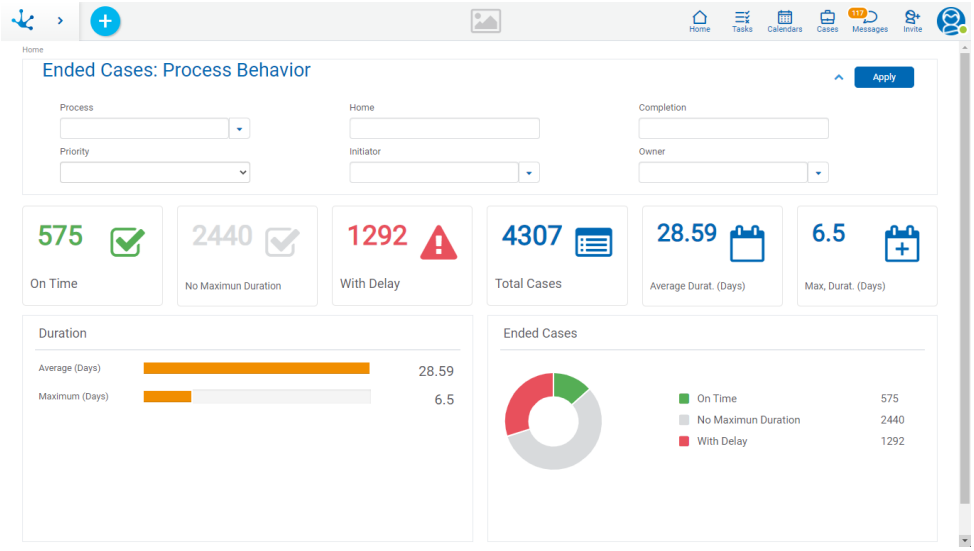
In addition to the predefined reports, user-defined reports can be incorporated based on existing or supplement them with specific characteristics of the organization.
Reports are accessed through the different menu options that are displayed by pressing "BAM and Analysis".

3.9.1. Process Analysis - Process Behavior

Phase 7: BAM > Process analysis

In this report, the behavior of processes is quickly displayed, indicating the cases that were completed on time, those behind schedule and the real average duration compared to the maximum duration.

The information can be displayed by applying the search filters shown in the upper section of the report.

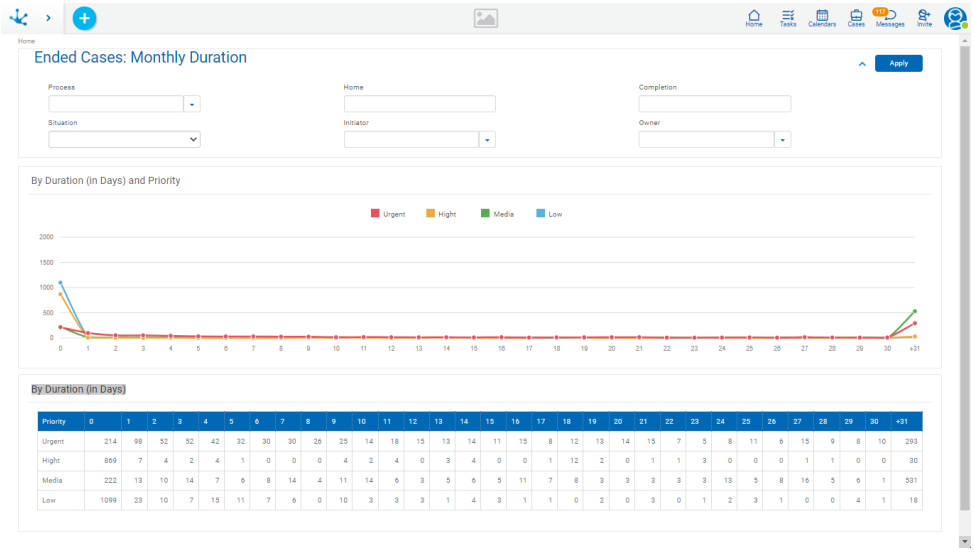


3.9.2. Process Analysis - Monthly Duration

Phase 7: BAM > Process analysis

The report on distribution of cases by duration must show a chart similar to a Gaussian bell, where the center of the bell is the average duration, provided that the behavior of the processes is adequate.
To analyze behavior in detail, drill down on the values of the grid in order to reach the cases that make up each value.

The information can be displayed by applying the search filters shown in the upper section of the report.



Priority	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	+31
Urgent	214	98	52	52	42	32	30	30	26	25	14	18	15	13	14	11	15	8	12	13	14	15	7	5	8	11	6	15	9	8	10	293
Hight	869	7	4	2	4	1	0	0	0	4	2	4	0	3	4	0	0	1	12	2	0	1	1	3	0	0	0	1	1	0	0	30
Media	222	13	10	14	7	6	8	14	4	11	14	6	3	5	6	5	11	7	8	3	3	3	3	3	13	5	8	16	5	6	1	531
Low	1099	23	10	7	15	11	7	6	0	10	3	3	3	1	4	3	1	1	0	2	0	3	0	1	2	3	1	0	0	4	1	18

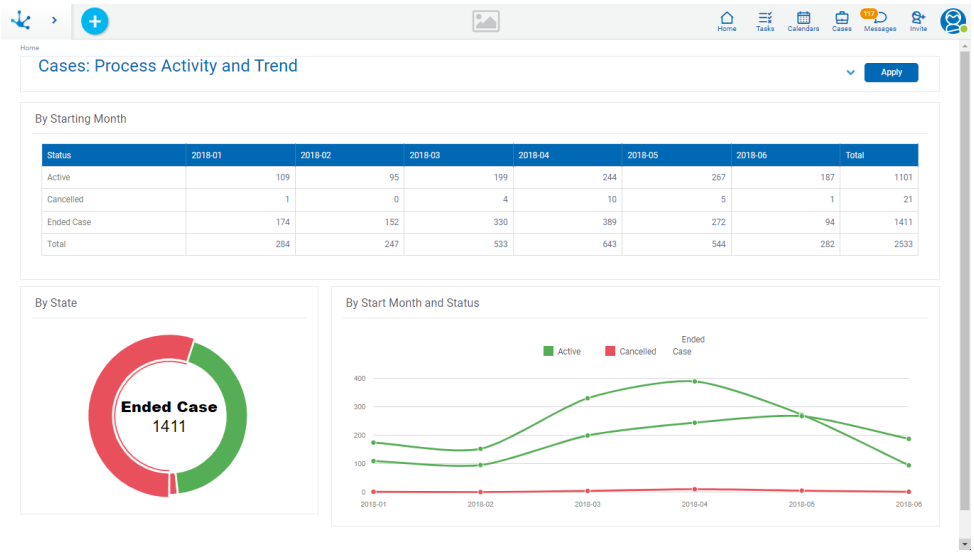
3.9.3. Process Analysis - Process Activity

Phase 7: BAM > Process analysis

In this report, the activity carried out by the processes is displayed, showing the values of cases initiated by periods, how many were completed, how many remain in execution, and the cases suspended and canceled.

If the information on process activity for the last months is analyzed, it should be observed that as months go by, the cases in execution tend to increase, and the cases already completed tend to decrease. In other words, during the first months, a greater number of completed cases should be observed, while during the last months, there are more cases in execution. Any anomaly, that is to say that the chart shows a different trend, can be investigated by drilling down on the values.

The information can be displayed by applying the search filters shown in the upper section of the report.

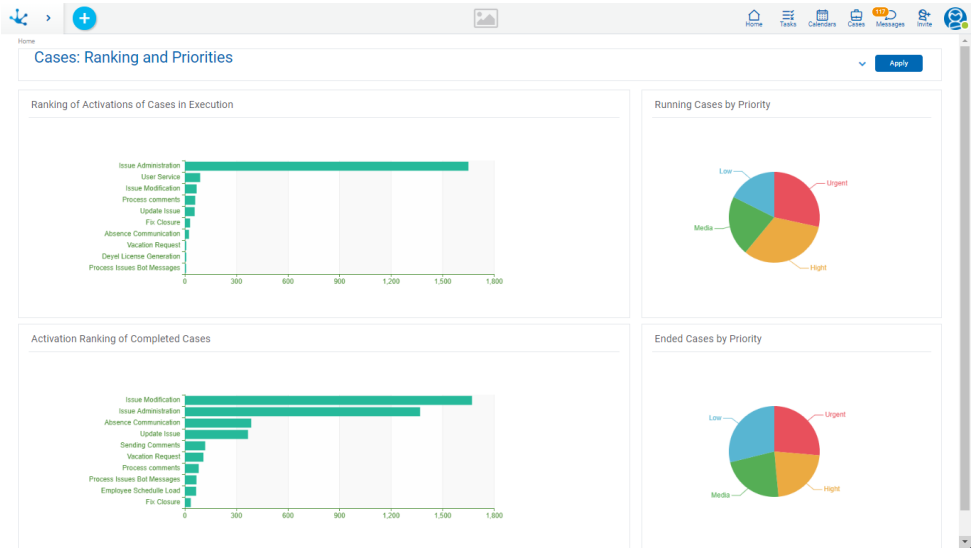


3.9.4. Process Analysis - Activation Ranking

Phase 7: BAM > Process analysis

This report shows a ranking of processes, through the number of cases initiated for each one, also considering their execution priority.

The information can be displayed by applying the search filters shown in the upper section of the report.

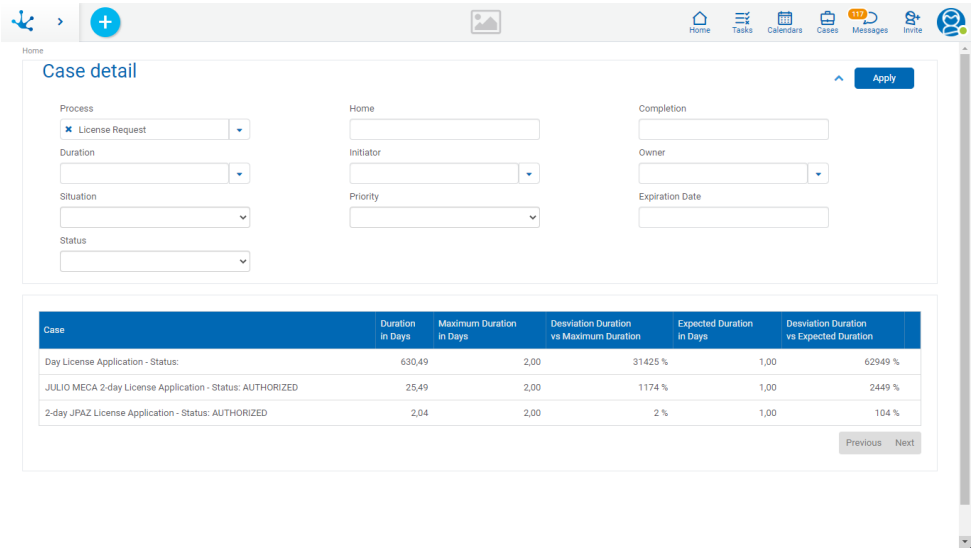


3.9.5. Process Analysis - Case Details

Phase 7: BAM > Process analysis

In this report, cases with their actual, maximum and expected duration can be displayed, together with comparisons of duration with maximum duration and duration with expected duration.

The information can be displayed by applying the search filters shown in the upper section of the report.

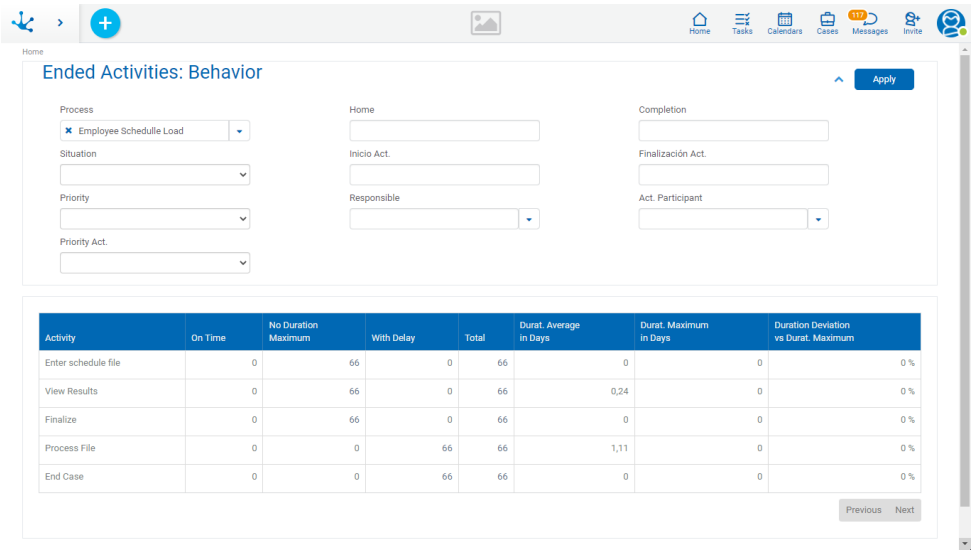


3.9.6. Activity Analysis - Activity Behavior

Phase 7: Activity BAM - Activity analysis

In this report, the behavior of a process completed activities can be analyzed (including or not all its versions). The amount of activities on time and behind schedule is detailed for each of them. Additionally, the average duration, the maximum duration and the deviation between both of them is mentioned.

The information can be displayed by applying the search filters shown in the upper section of the report.

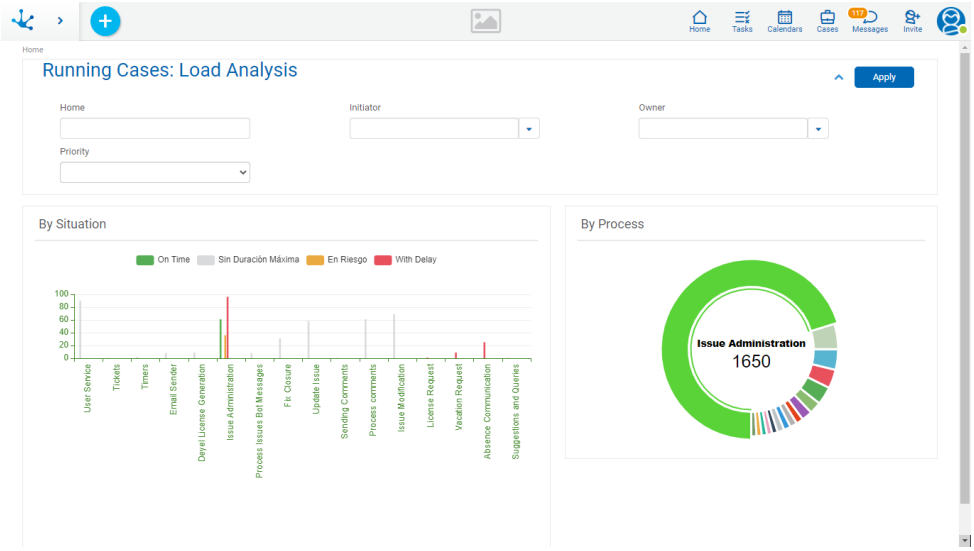


3.9.7. Process BAM - Load Analysis

Phase 7: BAM > Process monitoring

In this report, the current workload can be analyzed, considering the distribution of cases by process. For each process, it also shows how many cases are on time, how many at risk and how many are behind schedule.

The information can be displayed by applying the search filters shown in the upper section of the report.



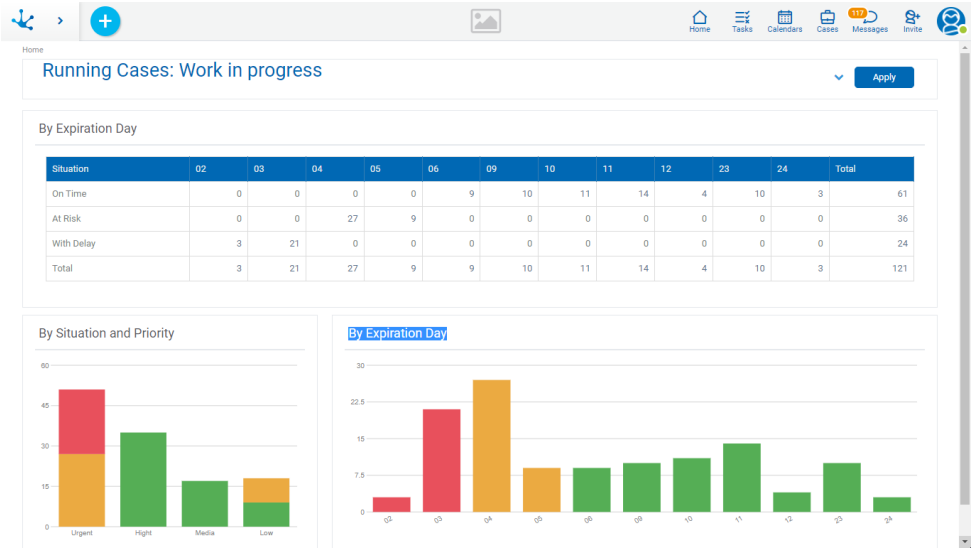
3.9.8. Process BAM - Work in Progress

Phase 7 > Process monitoring

This report shows the status of the cases that are in execution, together with their distribution according to their expected due date in periods of time.
Regarding the status of the cases, it is graphically displayed how many are on time, how many at risk and how many are behind schedule.

Regarding the distribution of cases according to due date, the chart indicates how many are scheduled to expire in the time period indicated, and it may or may not count those that are behind schedule. If those behind schedule are counted, they are shown in the first period considered in the chart.

The information can be displayed by applying the search filters shown in the upper section of the report.



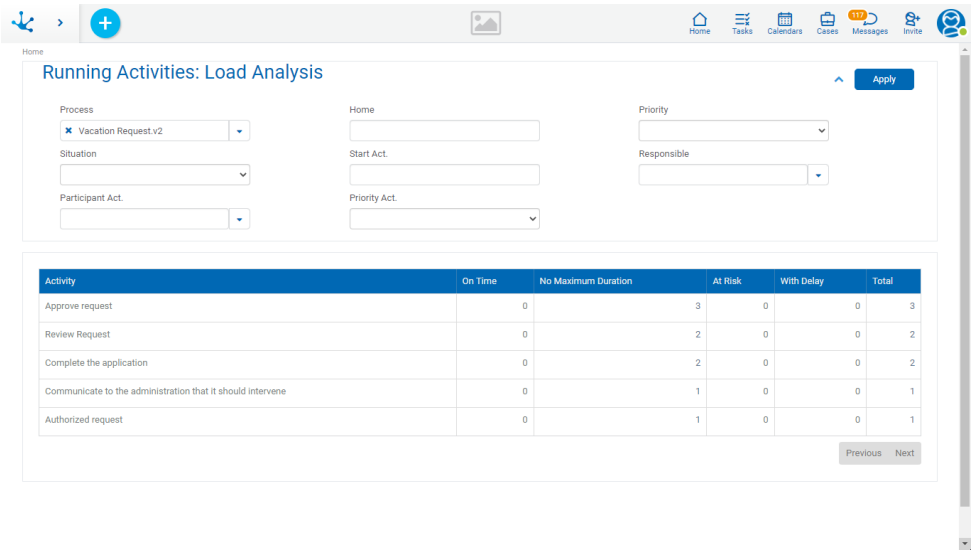
Situation	02	03	04	05	06	09	10	11	12	23	24	Total
On Time	0	0	0	0	9	10	11	14	4	10	3	61
At Risk	0	0	27	9	0	0	0	0	0	0	0	36
With Delay	3	21	0	0	0	0	0	0	0	0	0	24
Total	3	21	27	9	9	10	11	14	4	10	3	121

3.9.9. Activity BAM - Load Analysis

Phase 7 > Activity monitoring

In this report, the behavior of activities in progress of a process can be analyzed (including or not all its versions). The amount on time, at risk and behind schedule is detailed for each activity.

The information can be displayed by applying the search filters shown in the upper section of the report.



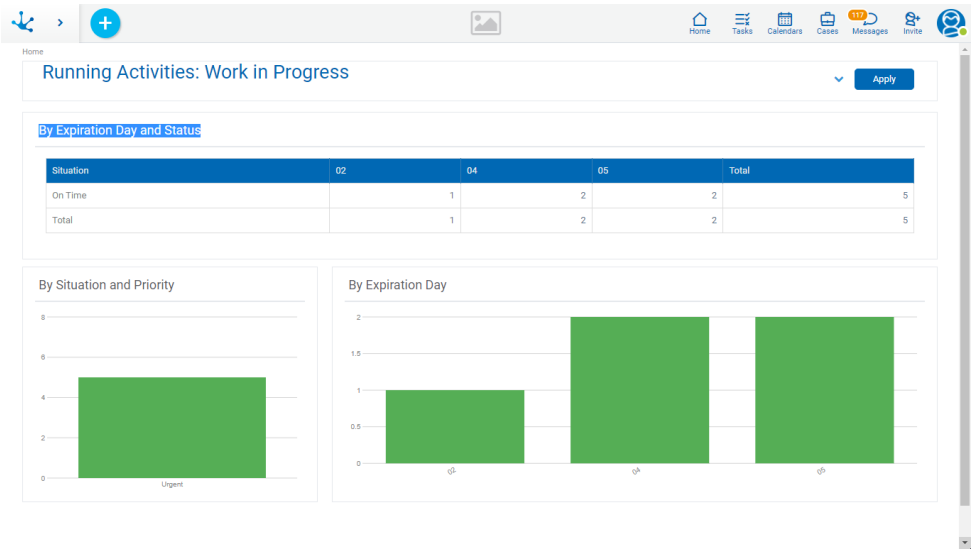
3.9.10. Activity BAM - Work in progress

Phase 7 > Activity monitoring

This report shows the status of the activities that are in execution, together with their distribution according to their expected due date in periods of time.
Regarding the status of the activities, it is graphically displayed how many are on time, how many at risk and how many are behind schedule.

Regarding the distribution of activities according to due date, the chart indicates how many are scheduled to expire in the time period indicated, and it may or may not count those that are behind schedule. If those behind schedule are counted, they are shown in the first period considered in the chart.

The information can be displayed by applying the search filters shown in the upper section of the report.



3.10. Configuration

Phase 8: Configuration

Configuration of **Deyel** is realized through the different menu options which are displayed by pressing "Configuration".

Organization
Allows to manage the organizational structure, defining the hierarchical relations between the different areas.

Security
Allows to define and administer the security scheme of **Deyel** through users, job positions, roles, agents, permissions and security functions.

Localization
Allows to define aspects that depend on the geographic location or preferences of **Deyel** users, such as language, time and date formats, decimal symbols and holiday calendars, among other topics.

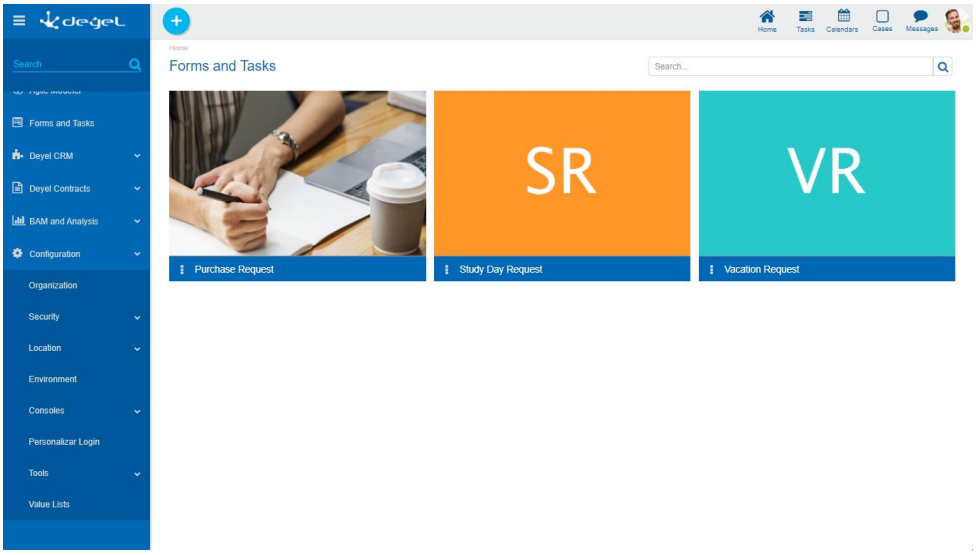
Environment
Allows to define the preference settings for the use of **Deyel** through properties, that adapt the product operation to the specific requirements of the company.

Consoles
Allows to monitor the execution of scheduled tasks, to do alerts procedure, to access monitor cache and to consult event and log consoles, among other functionalities.

Tools
Options to import or export data, show historical records of imports and exports, download log files, etc.

List of Values
Allows to visualize the names on the lists of values of different applications and manage its values.

Parameterization Entities
Allows to define and manage **Deyel** auxiliary tables and business process tables.



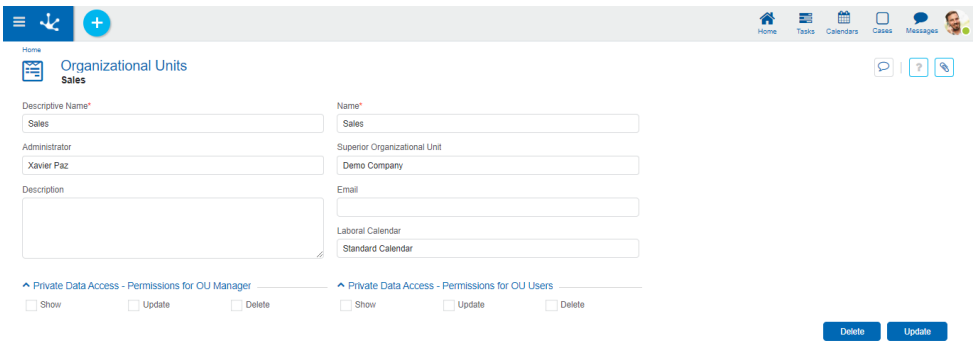
3.10.1. Organization

Phase 8: Configuration > Organization

The organizational structure of **Deyel** defines the areas or the offices from the company and the relation of hierarchy between them

When modeling a process, the organizational units can be defined as owner, initiator and participant.

When defining the users the organizational unit in which each of them work is indicated.



An asterisk "" on the label indicates that the property is required.*

Properties

Code
Identification code of the organizational unit. This property is visible if the environment property Type of encoding of organizational units takes values “Manual” or “Semiautomatic”, while it is not visible if the value is “Automatic”.

Descriptive Name
Complete name of the organizational unit. This text is the one visualized in the organizational structure and as a grid column.

Name
It is an abbreviated or reduced name. It is used when referring to the unit in an error message or when there is the need to abbreviate the descriptive name. No blank spaces admitted. This name must be unique.

Description
Property where the use of the organizational unit can be detailed so as to extend what is expressed in the Descriptive Name property.

Administrator
Optionally an user that performs as coordinator or head of the unit can be indicated.

Superior Organizational Units
Allows to identify the superior organizational unit inside the hierarchical structure. Not allowed to select the “Organizational Structure” unit as a superior unit.

Work Calendar
Calendar code that is associated to the organizational unit, it is selected from a list. If a calendar is not defined for the organizational unit, **Deyel** uses the one of the closest higher-level unit with a defined calendar. The root unit of all the hierarchy has defined a calendar by default.

Email
Email associated to the organizational unit.

Private Data Access - Permissions for Organizational Unit Manager
For the administrator of the unit, the access permissions to the private data of their work teams are established.

Show
Can the administrator of the unit see the private data of their subordinates? (Yes/No)

Modify
Can the administrator of the unit modify the private data of their subordinates? (Yes/No)

Delete
Can the administrator of the unit delete the private data of their subordinates? (Yes/No)

Private Data Access - Permissions for Users of the Unit
For a user that is part of the unit which establishes the access permissions to private data of their colleagues.

Show
Can a user of the unit see the private data of their colleagues? (Yes/No)

Modify
Can a user of the unit modify the private data of their colleagues? (Yes/No)

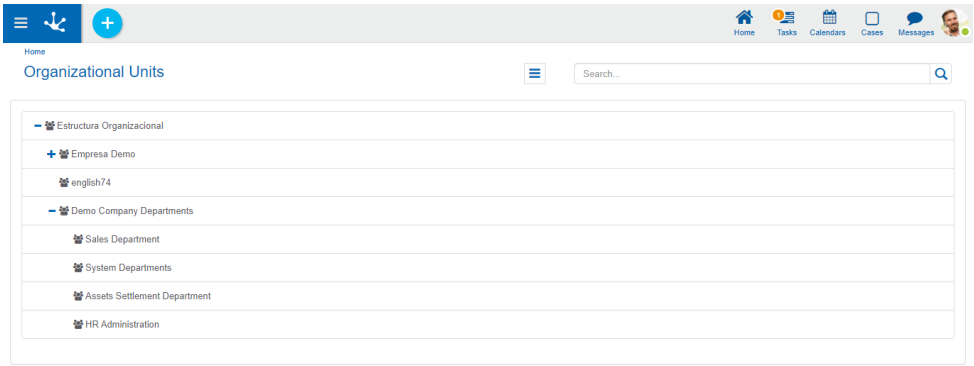
Delete
Can a user of the unit delete the private data of their colleagues? (Yes/No)

3.10.1.1. Organizational Structure

To do the creation and administration of the organizational units, the organizational structure can be accessed from **Deyel** menu.

- From the option "Configuration", selecting "Organization".
- From the search facility.

Organizational units can be visualized as a hierarchical structure or as an object grid.



Hierarchical View

The organizational units are visualized in the form of a hierarchical structure where each unit can be identified by Descriptive Name property. The location within the structure depends on the definitions realized in Superior Organizational Units property of each unit.

The icon allows to change the visualization to grid mode.

Grid

The grid of organizational units allows to visualize them with the standard presentation of **Deyel** grid, with facilities of:

- Order
- Paging
- Search Bar and Filters
- Data Download

The icon allows to change the visualization to hierarchical view mode.

The number displayed on the right of the grid title, indicates the amount of organizational units included in the grid.

The following properties of the organizational units are visualized as grid columns:

- Descriptive Name
- Name
- Administrator
- Superior Organizational Units

Operations

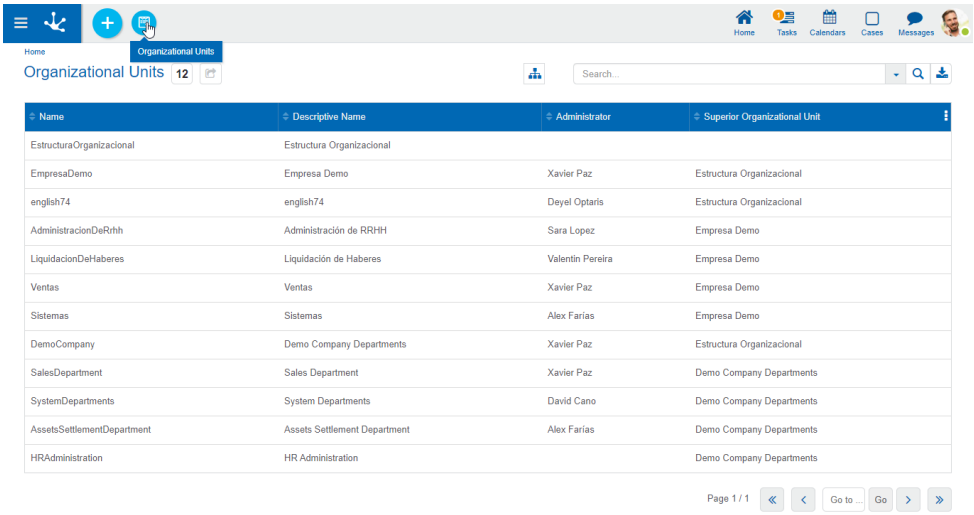
Both in hierarchical or grid view you can do operations on each unit, depending on the security permissions that the connected user has.

Hovering the mouse on each of the lines, buttons corresponding to the operations are visualized.

- Show
- Modify
- Delete

The exception is the operation "Create", that may be done from the context menu in different ways.

- Hover the mouse over the icon and select the icon on its right. Organizational Units.
- Click on the icon and select the option Organizational Units on the vertical panel displayed.



3.10.1.2. Operations

The buttons to do operations on the organizational units are enabled hovering the mouse over a unit on the organizational structure, whether it is visualized as hierarchical or as a grid, with exception of the creation operation, that is initiated from the icon corresponding to context menu.

> *The organizational units "Organizational Structure" and "External Units" cannot be deleted or modified, but units with lower levels in the hierarchical view can.*
> *The organizational unit "My Company" can be modified but not deleted.*
> *The organizational unit "Demo Company" can only be modified or deleted by a user that has "Account Administrator" permission.*

Create

Opens properties panel of the new organizational unit with all of them as editable. The creation is done by pressing the button "Accept" and the user receives a message indicating that the data has been saved.

Code

It is required whenType of encoding of organizational units property is configured as "Manual". When the value of this property is "Semiautomatic", it is optional and if the value is "Automatic" it is not visible.

Superior Organizational Unit

The unit "Organizational Structure" is not allowed to be selected as a superior unit.

Show

Opens properties panel of the selected organizational unit. The properties are visualized as not editable and depending on the security permissions of the user, the buttons available to operate on the organizational unit are enabled.

Buttons

- Modify
- Delete

Modify

Opens properties panel of the selected organizational unit, with those properties that can be modified, as editable. The modification is done by pressing the button "Accept" and the user receives a message indicating that the data has been saved.

Superior Organizational Unit

A unit with lower level in the hierarchical view is not allowed to be selected as a superior unit.

Administrator

When the unit has an administrator defined and the unit has been modeled as owner or participant of a process in use, this property can be modified entering a new administrator but not leaving it without any value.

Delete

Open the selected organizational unit and its properties are visualized as not editable. The deletion is done by pressing the button "Accept" and the user receives a message indicating that the data has been deleted.

> *There must not exist any user that belongs to the unit.*
> *The unit must not be actor in any role.*
> *The unit must not be owner or participant of any process that is "in use".*

3.10.2. Security

Phase 8: Configuration > Security

Deyel uses a security scheme based on in two concepts:

- **Authentication** of the users, that allows to control who uses the product.
- **Authorization** for the specific functions that each user can use.

3.10.2.1. Authentication

Deyel keeps a register of all the users that are enabled to use the product, collecting their identifying data (user code, alias, email address) their personal and labor data, their state, etc.

The procedure of authentication ensures that all use of **Deyel** and its solutions are made by "someone" or "something" that can be identified as a user, inside the user register.

Also, to ensure that "someone" or "something" is who they say to be, they must inform their user code or alias and their password.

Deyel uses different authentication methods but in each context only one of them should be used.

The mechanism of authentication to be used is defined in the configuration of the **Deyel** execution environment.

Native Authentication
This is the predetermined mechanism of authentication. **Deyel** verifies that the user exists in its register and verifies the password, which is stored encrypted in its data base.

LDAP Authentication
When an organization uses directory services to register its users, **Deyel** can be configured to delegate to LDAP the authentication process.

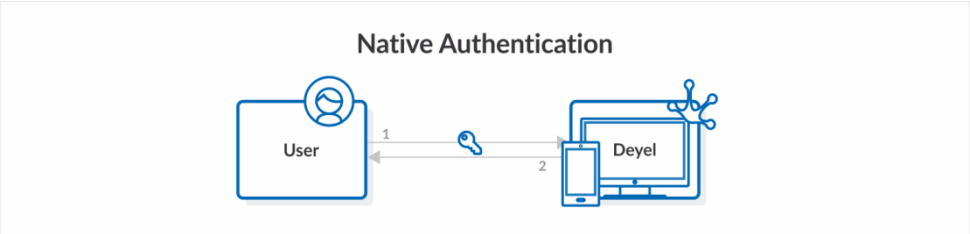
Federated Authentication
Deyel implements this type of authentication integrating with Google. When an organization uses Google accounts for its users, the authentication process can be delegated to Google.

Personalized Authentication
This kind of authentication delegates the authentication of the users to an advanced rule.

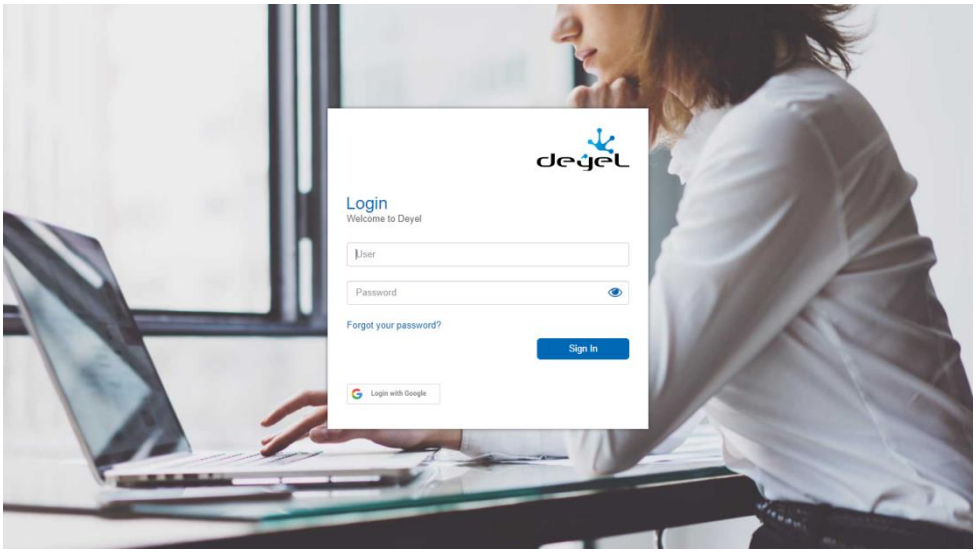
Mixed Authentication
Allows an organization to use simultaneously different types of authentication.

3.10.2.1.1. Native Authentication

This is the predetermined mechanism of authentication.



The user enters the user code or alias and the password on the user portal access page.

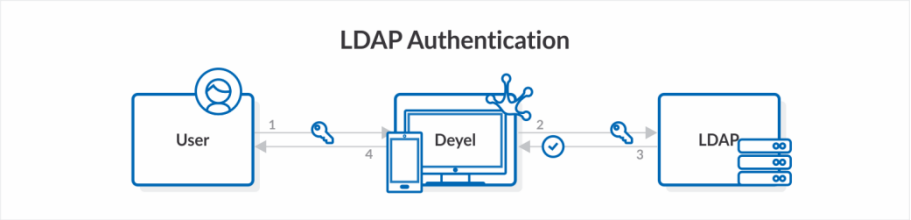


Deyel verifies that a user exists in its register, with the user code or alias informed and verifies the password. Access keys persist in **Deyel** database encrypted with SHA3.

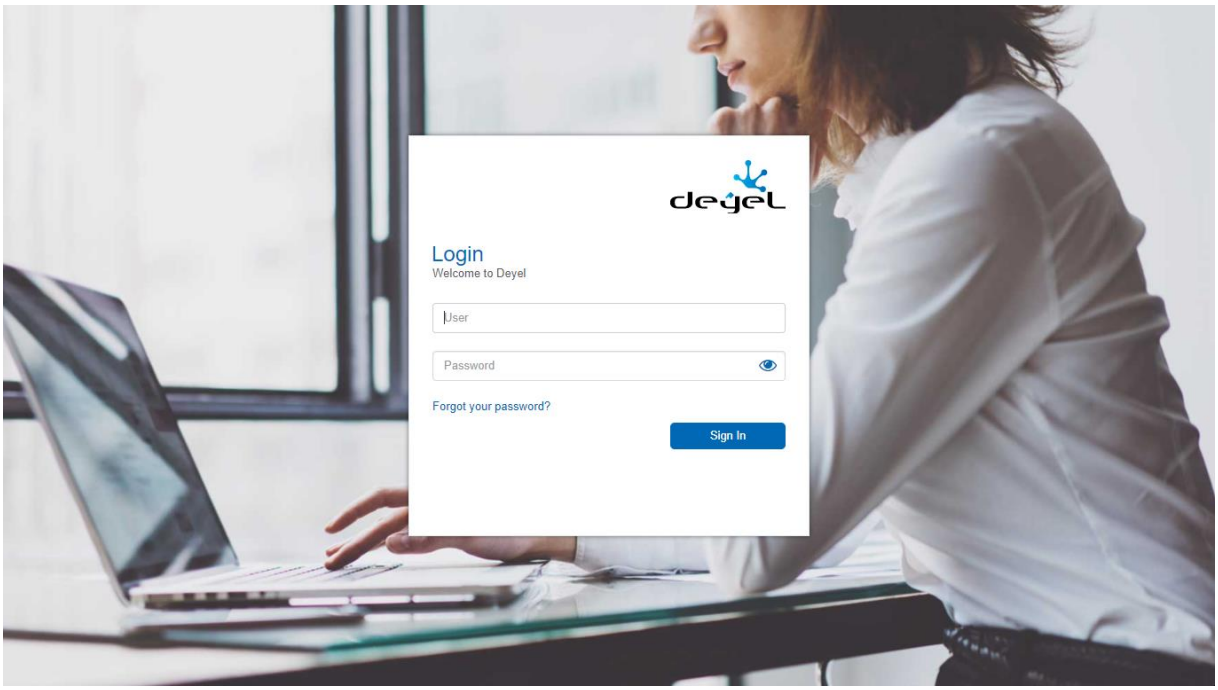
If the user or the password are incorrect, the access to the product is denied.

3.10.2.1.2. LDAP Authentication

When an organization uses directory services to register its users, **Deyel** can be configured to delegate the authentication process to LDAP.



The user enters the user code or alias and the password on the user portal access page.



Deyel verifies that a user exists in its register, with the user code or alias entered and delegates the authentication to LDAP. If the user is registered in **Deyel** and LDAP reports a correct authentication, then the access is allowed.

> *IMPORTANT*
>
> *Before activating LDAP authentication it must be ensured that a registered user exists in Deyel and can authenticate correctly in LDAP.*
> *If none of the users meet these conditions, it is not possible to enter the portal.*
>
> *Deyel verifies that the administator that is configurating the environment can authenticate correctly in LDAP.*
> *When LDAP is used as one of the Mixed authentication methods, this last verification is not done.*

> *When Deyel can not establish communication with LPAD server, an administrator is allowed to enter the portal by using Native authentication.*
> *In these cases, the user does not have all administration options available, but can only access to the environment configuration, to reconfigure the mechanism of authentication.*
>
> *With this mechanism of authentication the option "Forgot your Password" is not available.*
> *If the user does not remember their password, they must observe the procedures that the organization determines to solve the problem.*

In the configuration of the execution environment of **Deyel**, different aspects of the integration with LDAP can be configured.

LDAP - Server Connection
Configuration of access to LDAP server. All properties are required to activate LDAP authentication.

LDAP - User Search
Configuration of users search in LDAP Directory. Establishes the search subtree, LDAP attributes which are considered search keys and additional user selection filters.

LDAP - Attribute Synchronization
Configuration of **Deyel** user properties that are synchronized with LDAP attributes

Attribute Synchronization

Deyel allows certain properties of the users can be recovered from attributes of LDAP directory, avoiding that these properties can be modified in **Deyel**.

When **Deyel** connects correctly and determines that the user exists in LDAP, it is marked as a "Synchronyzed User". This indicates that some of their attributes have been obtained from LDAP and can not be modified in **Deyel**. In the same way, **Deyel** automatically removes the "Synchronized User" mark when it determines that the user does not exist in LDAP. This way its attributes can be modified in **Deyel**.

There are different moments in which these attributes synchronization is done.

Login

When the user authenticates correctly against LDAP, attributes are synchronized. If **Deyel** detects an error that does not enable to do this synchronization, a register of this is left in the logs console and log in to the portal is not allowed.

User Creation

When a user is created, by informing the user code or the alias, the existence in LDAP is verified. If the user exists in LDAP, attributes are obtained and remain protected. They can not be modified in **Deyel.** If the values recovered from LDAP are incorrect or the user does not exist in LDAP, creating the user in **Deyel** is not allowed.

User Modification

When a user is modified, **Deyel** syncs again the attributes before showing the information on the screen. If **Deyel** detects an error that does not enable to do this synchronization, a register of this is left in the logs console and the synchronization is not done. Then, there is access to the user information, but the synchronized properties can not be modified from **Deyel**.

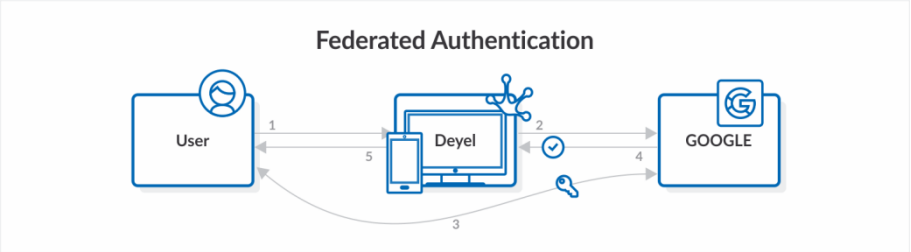
Query and Deletion of Users

When these operations are executed, an attributes synchronization is also done before showing the user information on the screen. If **Deyel** detects an error that does not enable to do this synchronization, a register of this is left in the logs console and the synchronization is not done.

3.10.2.1.3. Federated Authentication

An organization can use the services of an identity provider (IDP) to do the authentication of its users. In this way it grants a single access mechanism to the applications, avoiding users having to remember multiple login credentials.

Deyel implements this type of authentication integrating with Google. When an organization uses Google accounts for its users, the authentication process can be delegated to Google.

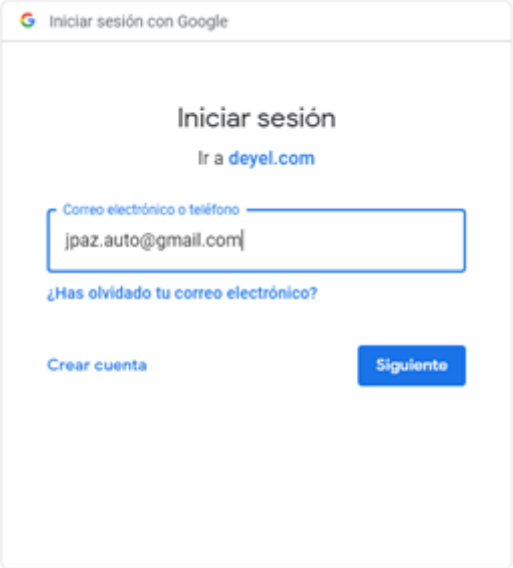


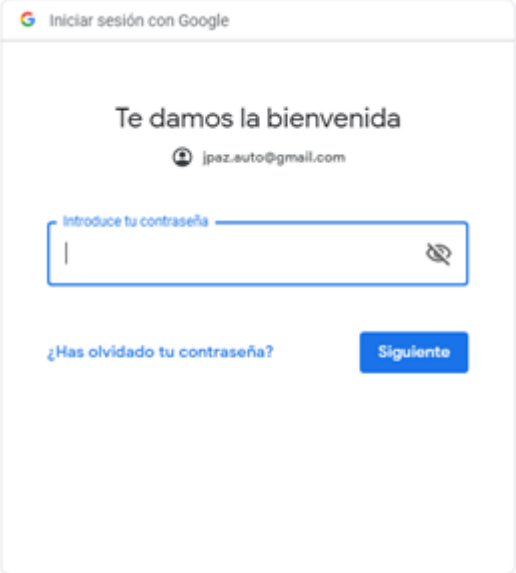
In the access page to the user portal there is a button that allows to redirect users to Google login in order to enter their credentials.

When users enter their account, Google informs **Deyel** that they have been successfully authenticated and the access code is never informed to **Deyel**.

Deyel only verifies that a user exists in its register, whose email account is the Google account entered. Users can use the product only when they are correctly authenticated to Google and they are defined in **Deyel**.

By pressing the button "Enter with Google" the users go to the login panel where they enter their credentials.

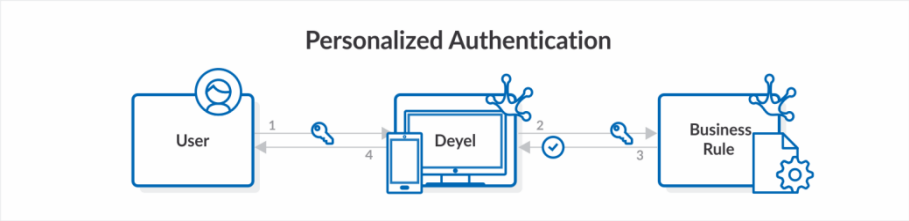




When users enter their account, Google informs **Deyel** that they have been successfully authenticated. **Deyel** verifies that the users are defined and their emails are the Google accounts they entered.

3.10.2.1.4. Personalized Authentication

This kind of authentication uses an advanced rule to authenticate users.



The user informs their user code or their alias and password, in the access page to the user portal. **Deyel** invokes this advanced rule, passing the data entered by the user. The advanced rule implements the authentication logic and returns if it is correct or not.

When **Deyel** receives that the authentication is correct, it is verified that the user exists in **Deyel** and allows entry.Passwords do not persist in **Deyel**.

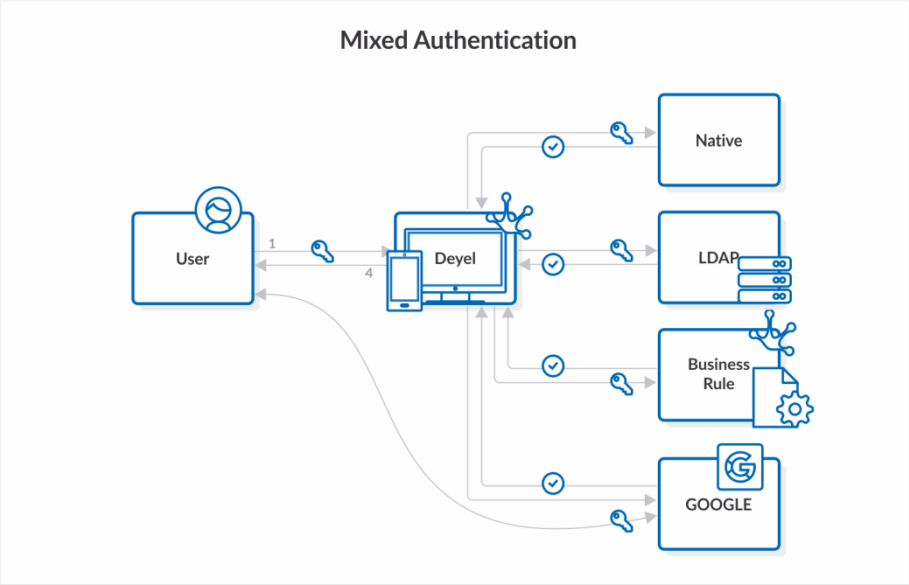
3.10.2.1.5. Mixed Authentication

The mixed authentication allows to combine multiple schemes of authentication, in a predetermined order.

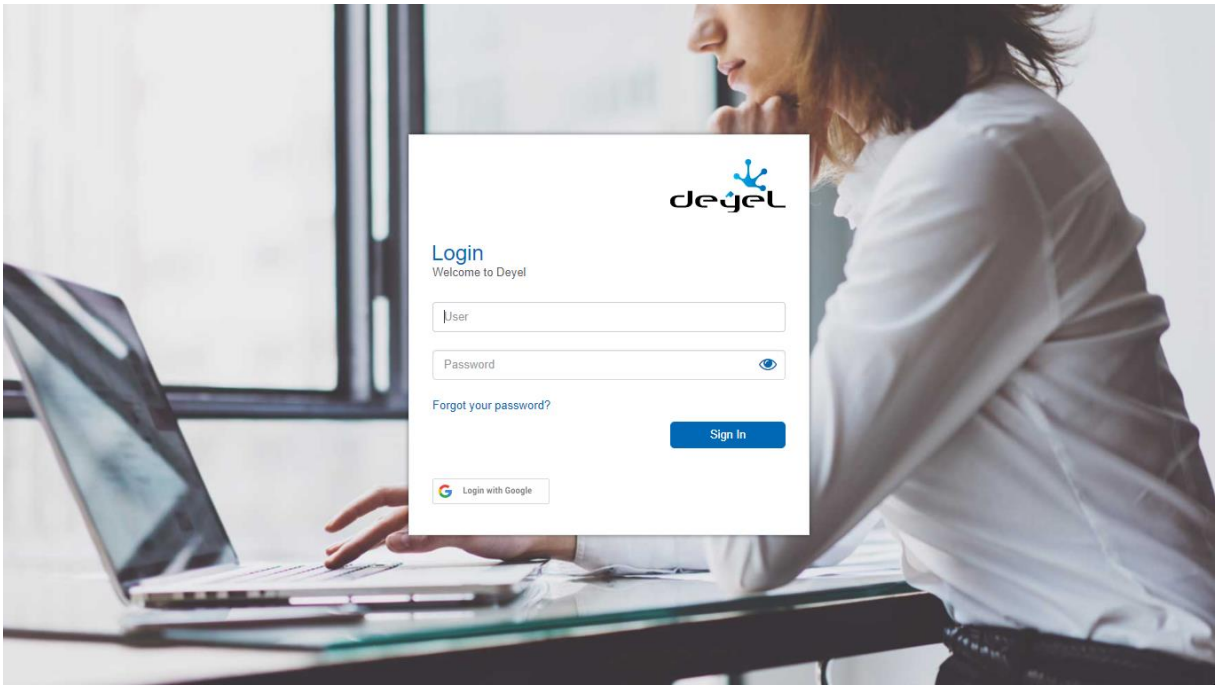
Let´s consider for example an organization that delegates in LDAP the authentication of its internal users, but needs to use addtionally the native authentication and the federated authentication (Google), to authenticate other groups of external users, which are not registered in the directory services.

This situation is solved by using the mixed authentication, indicating in the environment configuration which are the supported authentication methods and in what order should they be applied:

- First Method of Mixed Authentication = LDAP
- Second Method of Mixed Authentication = Native
- Third Method of Mixed Authentication = Google



The access page to the portal allows to use the different alternatives.



If the user presses the button "Enter with Google", they are redirected to the Google login. The user can use the product only when they are correctly authenticated in Google and registered in **Deyel**.

If the user informs their user code or their alias and password, then **Deyel** tries to authenticate them first in LDAP and then by Native authentication.

When a mechanism authenticates correctly, **Deyel** allows the login of the user if it exists.

If a mechanism fails, it continues with the next one and if every configurated mechanism fails, the user receives an authentication error and the product can not be used.

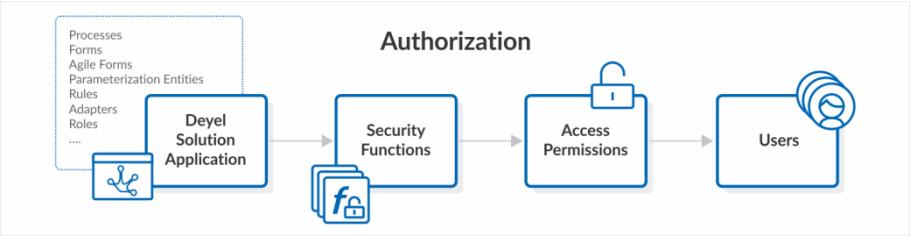
3.10.2.2. Authorization

The concept of authorization implies to verify which specific security functions each user can execute, preventing unauthorized actions to be carried out.

In the users profile they have defined the access permits and a user can only execute a security function, only if it is contained within their access permits.

Modelers use **Deyel** to develop the applications the organization needs and each application, is modeled as a set of processes, business entities based on forms, rules, adapters, and roles, among other objects related to each other, with the objective of solving a determined requirement.

The design methodologies focused on the user, they contemplate the definition of the characters, to represent user groups that use the application similarly. Modelers can identify the characters in each application and determine which security functions they need to use. From this the access permits assigned to the users are defined.

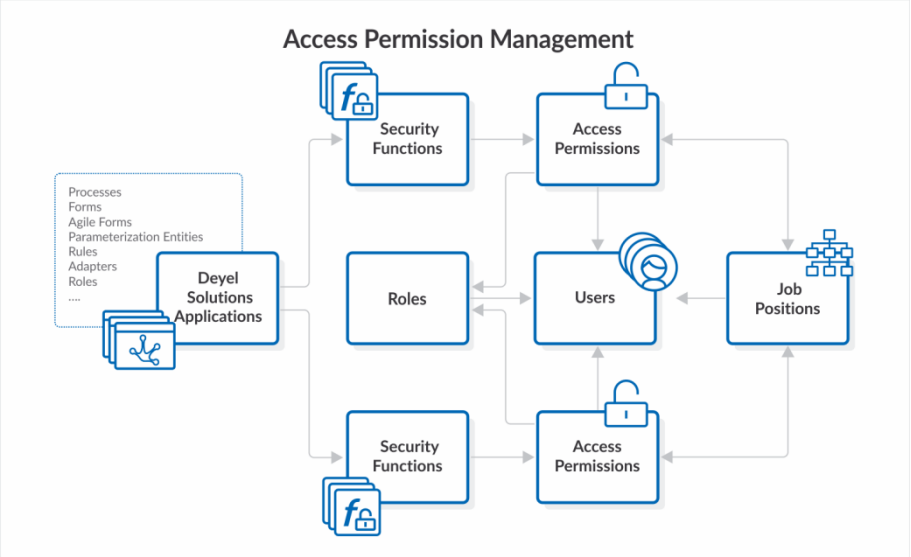


Access Permits Management

An access permits is an entity that groups security functions. Access permits of each user need to adjust dynamically. The appearance of new applications, new features or changes on the user work assignments, imply that the access permits it has assigned must be redefined.

The security administrator can add or remove access permits to each of the users individually.

In organizations with great number of users or applications, this task can be complex. The concepts of roles and job positions allow to simplify it, assigning permits to multiple users and not individually.



Predefined Access Permits

Deyel and its solutions have a set of predefined access permits, designed so that the different characters can use the feature they require.

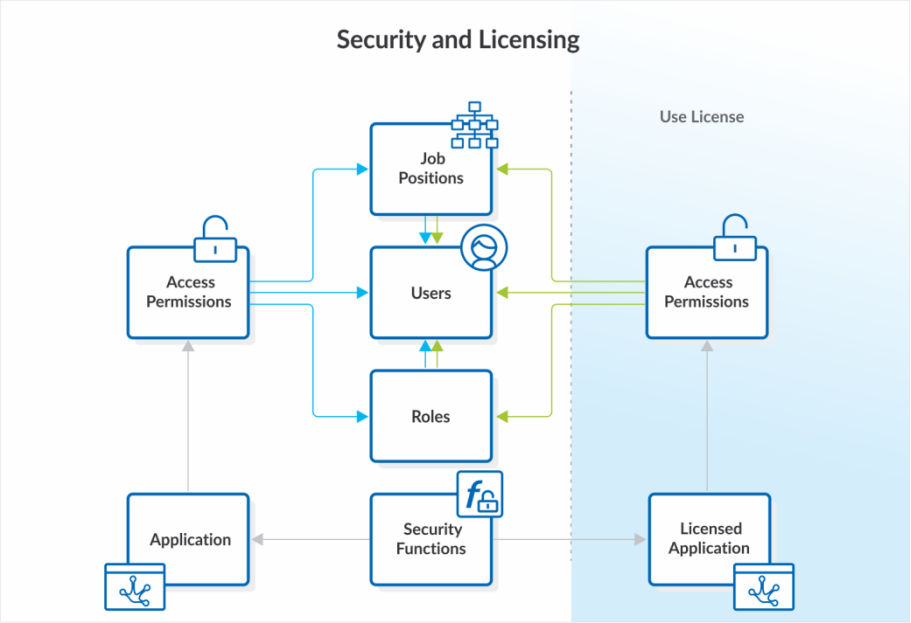
For example, **Deyel** has defined in its initial base different combinations of predefined access permits, for end users, for business modelers, for IT modelers and for security administrators, among others. In the same way, **CRM** has access permits designed for sellers and sales managers.

The predefined access permits cannot be modified, in case of needing to a copy of them can be made.

3.10.2.3. Security and Licensing

Phase 8: Configuration > Licensing

The licensing scheme uses different concepts and affects the operation of the security scheme.



Licensed Application

A licensed application is the one that requires a license in use and those that the user has are visualized through icons in the upper right part of the panel. About.

License of Use

A license of use is generated by the maker and distributed in an XML archive digitally signed. Each environment **Deyel** must import the corresponding license of use.

A license of use determines:

- The name of the company for whom the license is defined.
- Its identification.
- The platform or modality (cloud, on premise).
- The license type (sucription, perpetual).
- The environment (development, test, production).
- The validity period, with its issuance and renewal dates.
- The licensed applications and for each of them the amount of each type of user.

User License

Each of the licensed applications defines different types of user licenses. The license of use defines for each licensed application, the maximum quantity of user licenses of each type, that can be assigned in the environment.

In the case of **Deyel** the users can be of the following types:

- Participant

A participant enters or uses **Deyel** to participate in the defined processes, as well as using forms, calendars, chats, etc. either initiating a case or realizing some subsequent activity inside that same case.

- Agile Modeler

An agile modeler is a business user that uses the platform **Deyel** to define their web forms from predefined templates or realizing its complete design. They can adequate them to the business needs, by adding a process or defining access permits and its visibility, to finally publish it for its use.

- Deyel Modeler

A Deyel modeler is an IT user that uses the platform **Deyel** to design and build their object models for the development of applications of low code. Uses tools of graphical process modeling and forms to build, document and implement the process and business entities of the organization. They also define the roles and agents, they can implement more complex business logic using assistants what allows them to incorporate Java components.
They can implement the integration with other applications or basis of external data, behavior or more complex validations in the forms, using the advanced edition of the scripting. Uses tools to export and import their models and information of the environment.

In the case of solutions as **CRM** and **Contracts**, users can only be Participant type, that is to say that users can use the solutions according to the access permits they have assigned.

Example:

- Enable the use of **Deyel**, considering up to 100 users of Participant type, up to 50 of Agile Modeler type and up to 10 of Deyel Modelers type.
- Enable the CRM use considering up to 100 participating users.

Access Permits - Required User License

The access permits of a licensed application, they establish what type of user license is required. In the case of **Deyel**, there exist predefined permits that require a license of a Deyel Modeler type, others require a license of an Agile Modeler and others only require a license of a Participant.

Assignment of User Licenses and Access Permissions

On the profile of each user, it is indicated:

- The licensed applications that can be used and the type of user license that they are assigned to use.
- The assigned access permits.

When access permits of a licensed application are assigned, it is controlled that:

- The licensed application can be used by the user.

- The type of license of the user is compatible with the required one by the access permit.

To authorize the execution of a security function, it is verified if it is in any of the access permits assigned to the user.

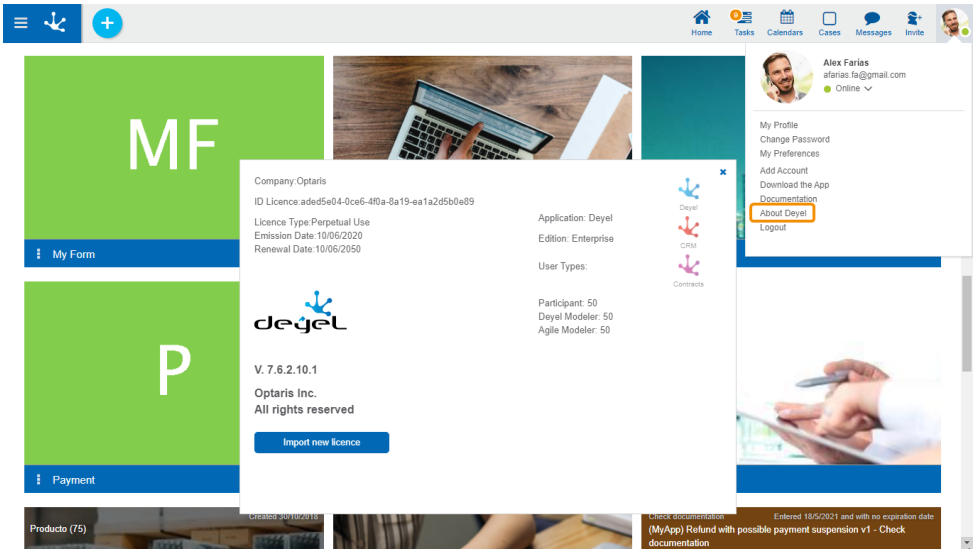
> *The corresponding permits to a licensed application, are ignored when:*
> - *The license of use does not enable the use of such licensed application.*
> - *The user does not have the required license type by the security permit.*

3.10.2.3.1. Showing Use License

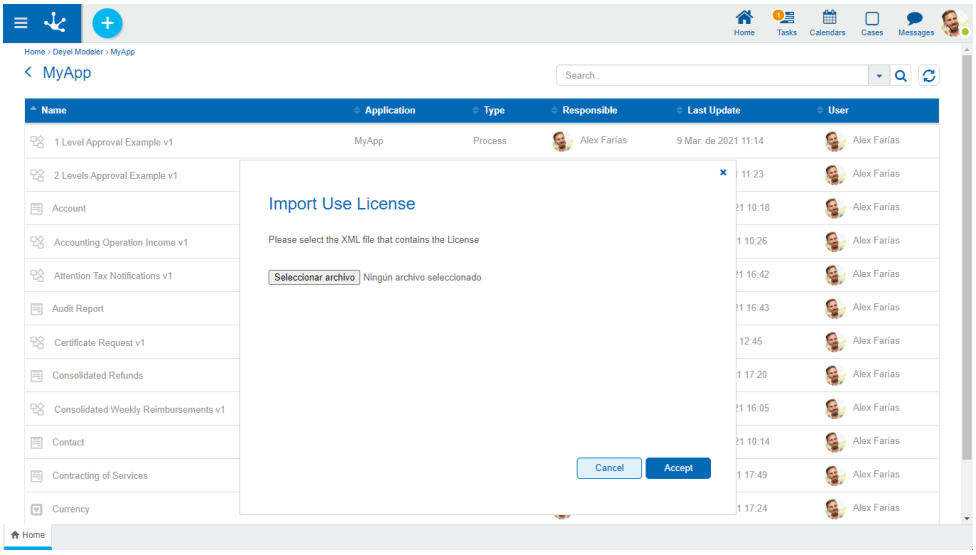
The information contained in the license in current use can be consulted from the functionality "About" on the top toolbar of the user portal.

About Deyel

When opening this panel the information about license in current use can be seen.



The button "Import New License" is only visualized if the user is authorized to perform this operation. By pressing it, a window is displayed which allows to upgrade the license of use of the environment.



The XML archive must be selected that contains the license in use and press “Accept”.

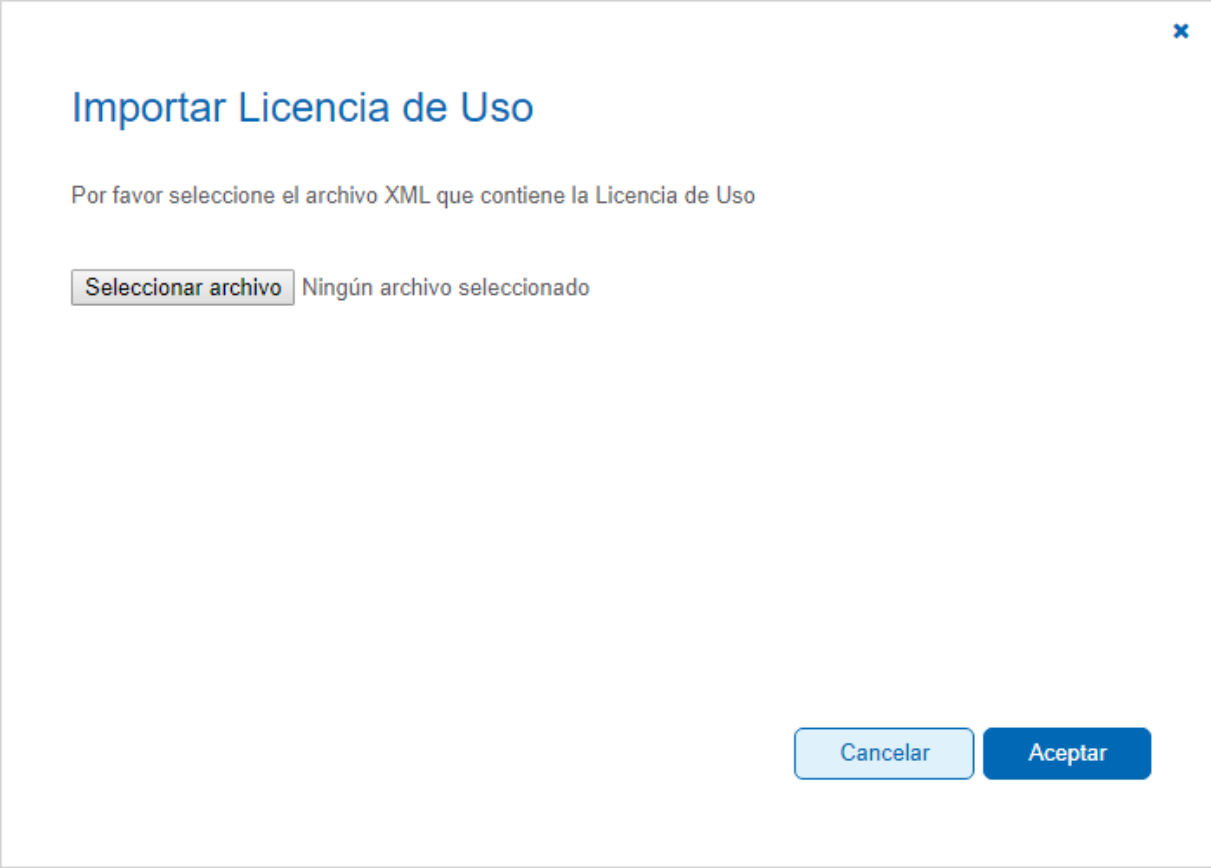
Deyel verifies the digital signature of the XML archive to ensure its integrity and verifies that the new license has the correct validity.

3.10.2.3.2. Licensing Controls

Deyel controls the environment licensing and emits alerts when users enter it.

Non-existent License

If the use license is not detected, a panel is opened to initially load the license.



Invalid License

The use license exists but the validity of the digital signature is not verified, then the license is invalid and the user is notified when entering the portal.

User Level Licensing

It is verified that the logged in user has user licenses granted and are compatible with the use license. If the user has not user licenses assigned, they receive a message informing they of the situation.

Number of Licensed Users

For every licensed application it is controlled that the number of user licenses assigned do not exceed what is defined in the use license. This control is realized for each type of user license.

Expired Licenses

The use license has a specified renewal date.It is considered that the license expires the day after such date.

3.10.2.4. Users

Every person using **Deyel** or any of its applications based on this platform are called users. As such, they have a code that uniquely identifies them, a password which allows them to authenticate and initiate the work session and a user profile, that collects their personal information, working information, their preferences, additional data, etc.

There exists a type of user that does not represent a person, called "Intelligent Thing". It is about devices, virtual or physical, that have the capacity of communicating with other devices, reacting to events and executing specific functions. Intelligent things can participate of business processes, as initiators of them or as responsible of executing activities, so as to optimize the execution of such processes. For example, the execution of daily tasks or calculation tasks can be delegated to them.

Chatbots

Among the users of type "Intelligent Thing" there exists a particular type that is the chatbot.

A chatbot can communicate with the other users using Tedis, interpreting specific messages. Each chatbot can define a process to model the way in which it responds to each of the messages received. The answer can be simple, for example responding with a predefined phrase, or it can be more complex and shoot the execution of a business process.

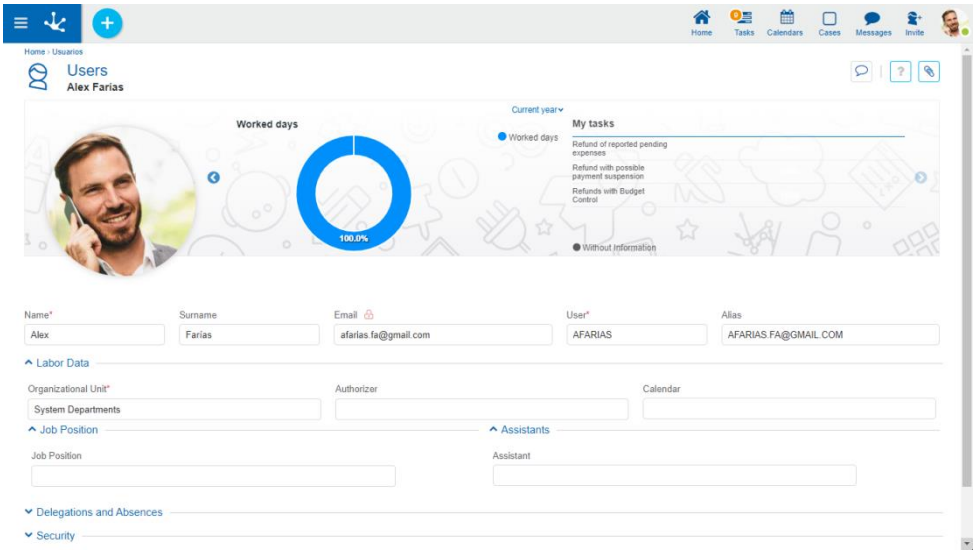
Chatbots are defined to interacts with users through the messages, delegating this way the execution of tasks. Human users can use the chatbots as their assistants, so that they automate the actions the user wants and reply to their messages.

Properties

For each user a profile is kept with the following information:

- User Identification
- Labor Data
- Delegations and Absences
- Security
- Personal Data.

For users type "Intelligent Thing", Configuration Data.



An asterisk "" on the label indicates that the property is required.*

User Identification

Profile Image

Each user can upload to their profile an image from a file. In case of not informing, an image with the user´s initials is automatically generated.

User

Code that uniquely identifies the user.

Alias

The value entered in this property works as an alternative code for the user. For example, email address can be used or any other coding that uniquely identifies the user. **Deyel** checks that there are not two users with the same alias.
When entering the portal, the user can enter either his code or his alias to authenticate.

Behavior Indicators

Information about usual tasks, days worked and the last tasks of the user are graphically displayed.

- With the indicator "Most usual activities" a ranking of the most executed activities is displayed.
- With the indicator "Worked days", it is seen how his presentism was. When clicking on this indicator, the profile form is displayed so that absences and licenses with their type and period covered are displayed.
- With "My tasks" a summary of the assigned tasks is displayed, together with their state regarding their due date. By clicking on these tasks, you can show the grid of the task with its cases.

Cover Image

It can be personalized selecting an image to use as background from the icon. .

Name - Surname

These properties may not be informed for "Intelligent Things" .

Email

The value entered in this field works as an alternative code for the user. **Deyel** checks that there are not two users with the same email.

Labor Data

Organizational Unit
Indicates the organizational unit in which the user works.

Authorizing
Determines the one in charge of authorizing or approving the procedures started by the user.
It is an optional property, that makes reference to another user, belonging to the same organizational unit or to a different one.
This property can be recovered from the business processes so as to assign the responsible to those activities that require an authorization, by using an agent. Authorizing.

Calendar
Identifies the calendar that establishes the dates and working hours of the user.
This property can be defined for each user individually.
When the user has not a special calendar defined, then the current calendar at the level of the organizational unit to which it belongs is considered. If there is no definition of a calendar in such unit, will be sought at the higher levels of the organizational structure, until detecting the calendar to apply.
The root organizational unit of all the hierarchy has a predetermined calendar defined.

Job Position

Optionally the job positions which correspond to the users can be indicated.

Position
Defines the job position occupied by the user.

Wizards

Optionally the chatbots can be selected to work as wizards of the user.

Wizard
A user can define one or more chatbots as their wizards.
Each time the user receives a message, each of their wizards verify if they recognize that message as a command they can execute.
Each chatbot exposes the commands they can execute and defines which participants can invoke their execution.
On the other hand, those who communicate with this user can see the commands they can use, by pressing the command icon on the chat window.

Delegations and Absences

Delegates

The list of delegated users is indicated and the period during which the delegation of tasks is valid.
When the user is inactive or absent then those current delegates can:

- Do the tasks assigned to the user.
- Initiate the processes authorized for the user.

Delegate of

Informs the user list that have delegated tasks in the user and the corresponding period.
This list is dynamic and is completed at the moment of opening the user profile.

Absences

Each element of the list indicates the type of absence and the corresponding period.
During these periods of absence the mechanisms of tasks delegation are activated.

> *Absences can be indicated considering a start and end time.*

Security

State and Expiration

User State
Indicates if the user is active or not.
An inactive user cannot enter the system, as their account is deactivated.

Expiration Date
The user account is automatically deactivated when the date entered in this property is exceeded. This automatic deactivation does not use the tasks delegation scheme.
This mechanism is not applied when the authentication of users is not made by **Deyel** but it is delegated to LDAP or Google.

Password Expires
It allows to establish a term of validity of 30, 60 or 90 consecutive days, starting to count from the last modification made on the user data.
After that deadline, when the user logs into the portal it is required they renew their password. It can also be stated that the password does not expire.
This mechanism is not applied when the authentication of users is not made by **Deyel** but it is delegated to LDAP or Google.

Licenses

Each environment of **Deyel** has use licenses that enable the use of licensed applications during a period of time and by a certain number of users.
In this section the list of licensed application the user can use is indicated and which type of user license they have assigned.

Product
Indicates the licensed application the user can use.

Type of License
Indicates the type of user license that is assigned to the user to use the product.

Permission

The access permits the user has assigned are listed. Each element from the list indicate the name of the permit and the application they belong to.

In the first positions the permits the user inherits for belonging to an organizational unit, playing a role or a job position are displayed. These elements are displayed protected and it is not possible to delete them from the list. Hovering over each of these elements, it is displayed from which object the permit is inherited.

Following the inherited permits, the permits assigned directly to the user are shown. When creating or modifying the user you can add or delete elements from the list.

First the application is chosen and then the permit of such application that you want to assign to the user.

When permits from a licensed application are assigned, this must be able to be used by the user. If the application is not available in the use license of the environment, the permit is seen as protected and it is possible to delete it from the permits list, but not modify the element

> *The permit "Account Manager" can only be assigned to a unique user.*
> *In On-Premise installations of **Deyel**, this permit can be assigned or removed from a user by the security administrator.*
> *In Cloud installations, you must use the site of **Deyel** to change the user defined as " Account Manager".*

In every installation there exists a set of predefined permits, that can be assigned to the users.

Roles

The roles the user performs are listed. The list is dynamically conformed when accessing the user profile, recovering the roles where the user and their unit are actors. Each element indicates the role name and the application it belongs to.
In the first positions the roles the user inherits for belonging to a unit are displayed and then the roles where the user is an actor.

Personal Data

The user profile includes the following properties:

Nationality
Birth Date
Identification Type / ID Number
Phone Number / Extension

Addresses

The user can have multiple addresses, with the following properties for each of them:

Country
State
City
Postal Code

Street
Number
Department

Media

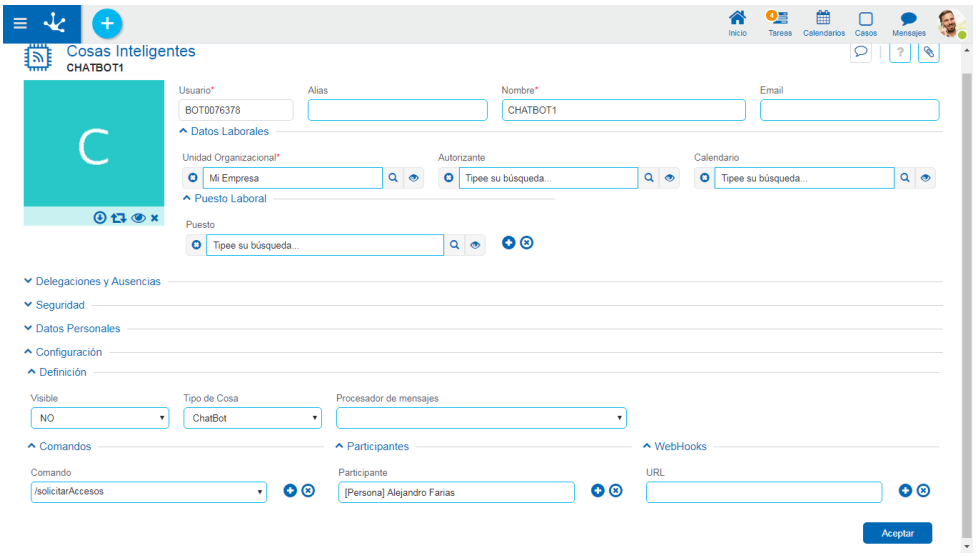
The user profile includes information about the identity of the user in the different social medias (Linkedin, Twitter, Facebook, YouTube y Skype).

Additional Data

Observations
Allows to load complementary data of the user.

Configuration Data

This section is only available for users type "Intelligent Thing".



Definition

Visible
Determines if the chatbot is visible in the user list of the business social media Tedis.
When a chatbot is not visible, it can work as a wizard, but it is not possible to send it messages directly in a chat.

Thing Type
Deyel incorporates the concept of chatbot, as a type of intelligent thing.
In future versions other types of intelligent things will be supported.

Message Processor
The chatbot can define the business process that implements the processing of the messages it receives. Each time a chatbot receives a message, starts a case of this process to generate the corresponding answer.

Commands

This section defines the list of commands the chatbot can interpret and answer.
Executing a command implies executing the business process that implements it.

Participants

This section defines the participants that can use the chatbot as wizard.
If an organizational unit is chosen, then any user from that unit can use the chatbot.

WebHooks

Webhooks are interfaces that allow to integrate applications, that is, they allow to connect and exchange data between applications.
A chatbot can use multiple webhooks, each of them defines the URL used to contact other application, sending it data in JSON format.
When a chatbot receives a message, it forwards it to each of its webhooks.

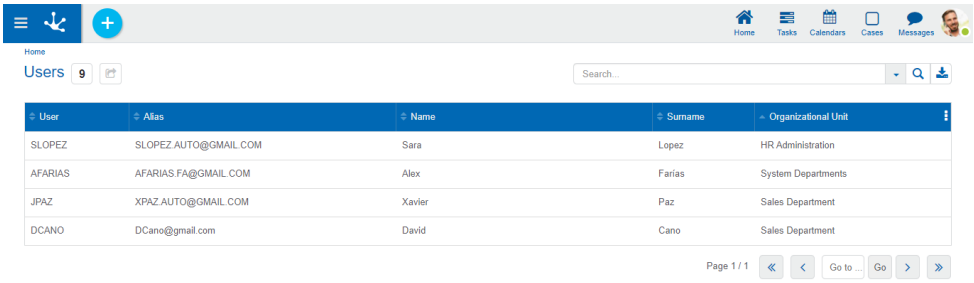
System User

To make internal operations, automatically, **Deyel** uses a predefined user called SYSTEM USER (SYSUSER).
This user is displayed in the user list, it is not possible to modify or deactivate it and it does not require user licenses.
The initial load of the use license or its subsequent update or the execution of programmed tasks, among others, are operations whose execution is made using this user and so it remains registered in the audit trails.
It is not possible for someone to enter the user portal authenticating as the system user. **Deyel** uses it only for the doing and registration of internal tasks.

3.10.2.4.1. Grid

To do the creation and administration of users, you can access the grid from the **Deyel menu**.

- From the option “Configuration”, selecting "Security" and then “Users”.
- From the searchfacility.



The users are visualized as a grid with the standard presentation of **Deyel** grids, with facilities of:

- Order
- Page
- Search Bar and Filters
- Download Data

The number displayed on the right of the grid title, indicates the amount of users included in the grid.

The following properties of the users are visualized as grid columns:

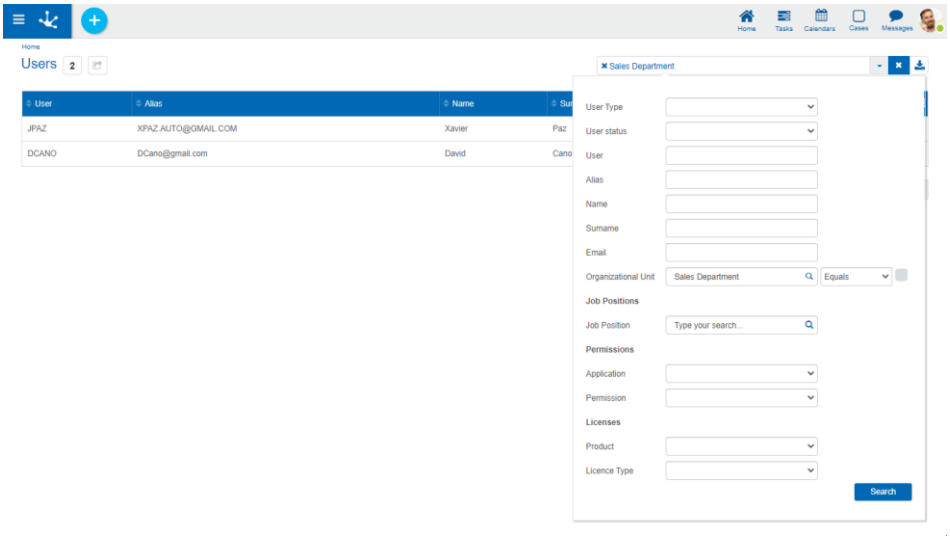
- User
- Alias
- Name
- Surname
- Organizational Unit

Operations

It is possible to realize operations on each user, depending on the access permits that the login user has. When hovering over each of the lines the buttons with available operations are visualized.

The exception is the operation "Create", that may be done from the context menu in different ways.

- Hover the mouse over the icon and select one of the icons on its right: Users or Smart Stuff.
- Click on the icon and select one of the options Users or Smart Stuff on the vertical panel displayed.



3.10.2.4.2. Operations

When hovering over each of the lines from the grid of users the buttons for making operations are enabled, with exception of the create operation, which is initiated from the icon corresponding to the context menu.

Show

Opens the properties panel of the selected user and the buttons to do operations on it are enabled, depending on the access permission that the login user has.

- Modify
- Delete

Create

Allows to create a new user. The creation is done when pressing the button "Accept" and the user receives a message indicating that the data has been saved.

> ***Deyel*** *can do the users authentication against LDAP directory, in this case some of the user properties can be disabled, given that such information can be recovered from the LDAP directory.*

Considerations

When creating users some considerations are applied to the entered properties.

Name - Surname
For user type "Intelligent Thing", these properties are not informed.

Email
There cannot be more than one user with the same email.

User Code
The user code is made up automatically of the name and surname. It is made up of the first initial of the name followed by a maximum of nine letters of the surname. It can be modified by the security administrator.
There cannot be more than one user with the same code.

Alias
Takes the same value the property Email does. Can be modified by the security administrator.
There cannot be more than one user with the same alias.

Profile Image
The profile image is entered by the user themselves. It is not a property that the security administrator can define when creating the user. **Deyel** assigns by default an image with the user´s initials.

Labor Data

Organizational Unit.
The organizational unit "My Business" is proposed by default. It can be modified by the security administrator.

Security

State and Expiration

Expiration Date.
It must be greater than or equal to the current date.

User License

In the properties Product and License Type the first product available in the use license is initially proposed. The security administrator can change this proposed value.

> *The list of applications only includes licensed applications available in the use license.*
> *When creating a new user, **Deyel** verifies that the amount of enabled users in the use license is not exceed.*

Permission

Considering the values entered for the properties Product and License Type, the initial permission is proposed by **Deyel**.

Type of Use License	Initially Proposed Permission
Deyel - Participant	Deyel End User
Deyel - Agile Modeler	Deyel.AgileModeler

Type of Use License	Initially Proposed Permission
Deyel - Deyel Modeler	Deyel.Deyel Modeler
CRM - Participant	CRM.Seller

The security administrator can change the permissions initially proposed.

> *The list of applications only includes licensed applications if they are available in the use license.*

When permissions of a licensed application are assigned, it is verifies that:

- The licensed application is on the list of applications the user can use.
- The license type of the user is compatible with the required one by the access permission.

> *The permission "Account Manager" can only be assigned to a unique user.*

Modify

Allows to modify the properties of a user. The modification is done when pressing the button "Accept" and the user receives a message indicating that the data has been saved.

> ***Deyel** can do the users authentication against LDAP directory, in this case some of the user properties can be disabled, given that such information can be recovered from the LDAP directory.*

Considerations

Users can upgrade their profile, modifying properties like their profile images, their delegates and inform absences, among others. However some properties, can only be modified by the account manager, who has permission to execute the security function "Deyel - Modify user data".

Name - Surname
For the user type "Intelligent Thing", these properties are not informed.

Email
There cannot be more than one user with the same email.

Alias
There cannot be more than one user with the same alias.

Profile Image
The profile image is entered by the user themselves. It is not a property that the security administrator can define when modifying the user.

Delegations and Absences

Delegates
When a user is modified:
- Delegations occurred in the past, cannot be modified nor deleted.
- Current delegations, that is the ones that have the current date within the period, can be modified but the date must be kept within the period.
- Future delegations can be freely deleted or modified.

Absences
This list can be administrated by the user itself or by the administrator user. Each element contains the absence type and the period of it.
Only the absences occurred during the last year are visualized, however the administrator user can see all of the absences.
When a user is modified:
- The absences occurred in the past, can be modified or deleted only by the security administrator.
- Current absences, that is the ones that have the current date within the period, can be modified but the date must be kept within the period. The security administrator can delete or modify current absences.
- Future absences can be freely deleted or modified.

Security

State and Expiration

Expiration Date.
It must be greater than or equal to the current date.

User License

When the user is modified you can add or remove elements from the list of licensed applications.

> *If an element is created or modified, only licensed applications available in the use license can be selected.*
> *Those elements from the list that make reference to applications available in the environment are displayed protected.*
> ***Deyel** verifies that it does not exceed the amount of enabled users in the use license.*

Permission

When modifying the user you can add or delete elements from the list.

You must select the Application and then select the Permission for such application.

When permissions of a licensed application are assigned, it is verified that:
- If the application is not available in the environment at the moment of the modification, the permission is seen as protected and it is possible to delete it from the permits list, but not modify it.
- If the application is available in the environment at the moment of the modification, it is verified:
  - The licensed application is one of the products the user can use.
  - The type of user license is compatible with the one required by the security permission.

The permission "Account Manager" can only be assigned to a unique user.

Delete

Allows to delete the selected user after viewing its properties. The deletion is done by pressing the button “Accept” and the user receives a message indicating that the data has been deleted.

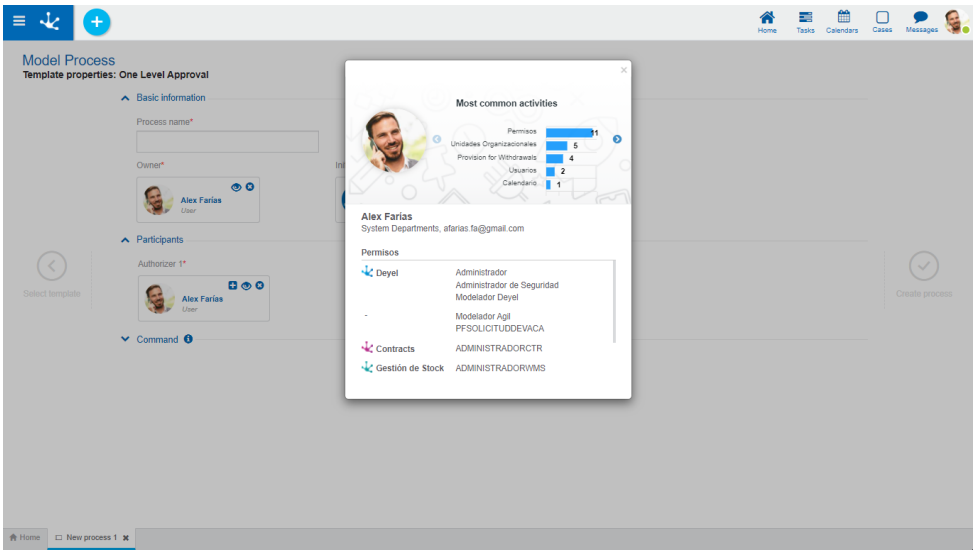
Considerations

When deleting users some verifications are made.

- The administrator of an organizational unit cannot be deleted.
- A coordinator or actor of a role cannot be deleted.
- A user with permission "Account Manager" cannot be deleted.
- A responsible or participant users of a process in use cannot be deleted.
- A logged in user cannot delete its own profile.

3.10.2.4.3. Show User Profile

From the different modelers, as well as from process execution and forms the query of a profile user can be made by pressing the icon and an informative panel is displayed.



Panel Elements

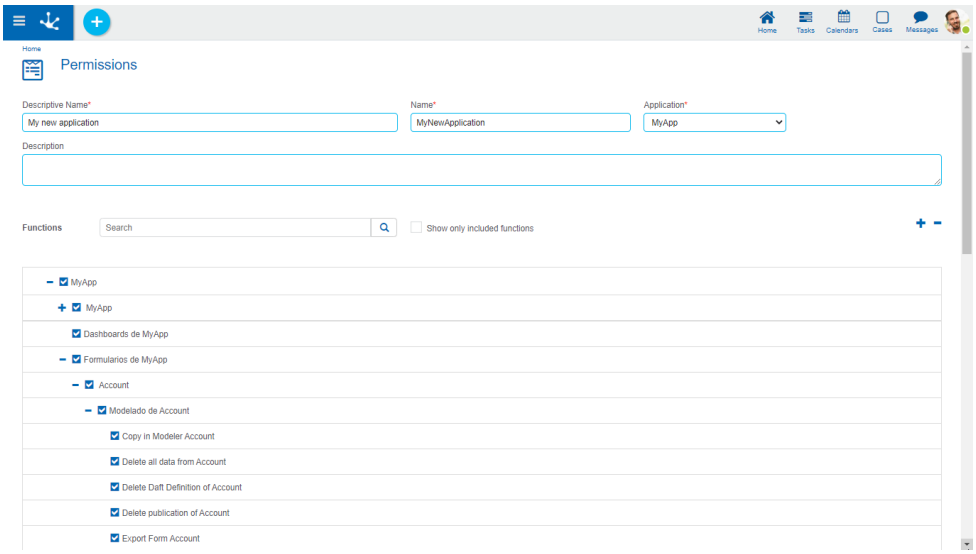
Some of the properties which were defined for the user are displayed, such as Name, Surname, Organizational Unit, Email and Permission. These last are seen within a container where the scroll bar can be used.

At the top of the panel the behavior indicators are seen, with the icon it advances to the following indicator and with the icon it returns to the previous indicator.

3.10.2.5. Permissions

An access permits is an entity that groups security functions.

Each user has assigned one or more security permits, that indicate which functions can they execute within the application. This way, the operations that a user can execute are determined by the access permits that they have assigned.



> *An asterisk "*" on the label indicates that the property is required.*

Properties

Descriptive Name

Complete name of the permit. This text is the one visualized on the permits grid.

Name

It is an abbreviated or reduced name. It is used when referring to the permit in an error message or in any mention that needs to abbreviate the descriptive name. This property does not admit blank spaces.

Application

Defines the application the permit belongs to, through the selection of a list of available applications. The permit is made up of security functions of such application.
The licensed applications are included in such list only if the use license of the application is current in the environment.
Permits that belong to licensed applications, can only be assigned to users that have user license of such application.

> *Any permit can contain security functions of the application **Deyel**.*
> *Any permit of the application **Deyel** can contain Global security functions.*

Description
Details the information about the permit, extending what is expressed in the property Descriptive Name.
Allows to enter a description of the facilities that the permit grants, without analyzing in detail the list of authorized functions.

Functions
Panel in which the set of security functions is defined which are authorized by the permit, they are visualized with the defined hierarchical structure.

Search
On top of the panel it is visualized a bar of search of functions, from where filters can be applied to the functions name to reduce the list.

The icon allows to delete the defined selection criteria and all lines on the panel are displayed again without highlighting.

View only included functions
When checking this property, only those included in the permit are displayed in the security functions panel.

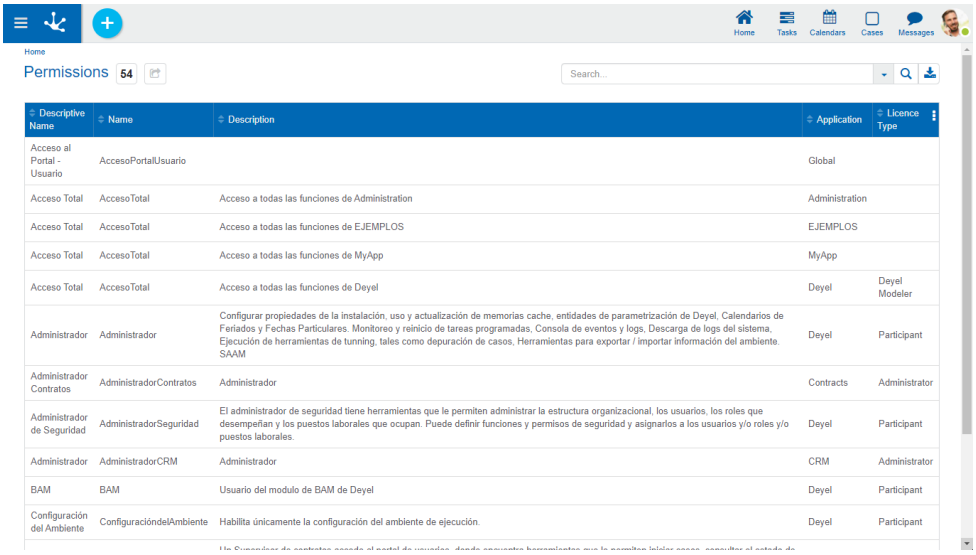
Expands the hierarchical structure of the security functions.

Reduces the hierarchical structure of the security functions only to higher- level ones.

3.10.2.5.1. Grid

To realize the creation and administration of permits, you can access the grid from the menu of **Deyel**.

- From the option “Configuration”, selecting "Security" and then “Permits”.
- From the menu search.



The permits are visualized as a grid, with the standard presentation of the grids of **Deyel**, with facilities of:

- Ordering
- Page
- Search Bar and Filters
- Data Download

The number displayed on the right of the grid title, indicates the amount of permits included in it.

The following properties of the permits are visualized as columns of the grid:

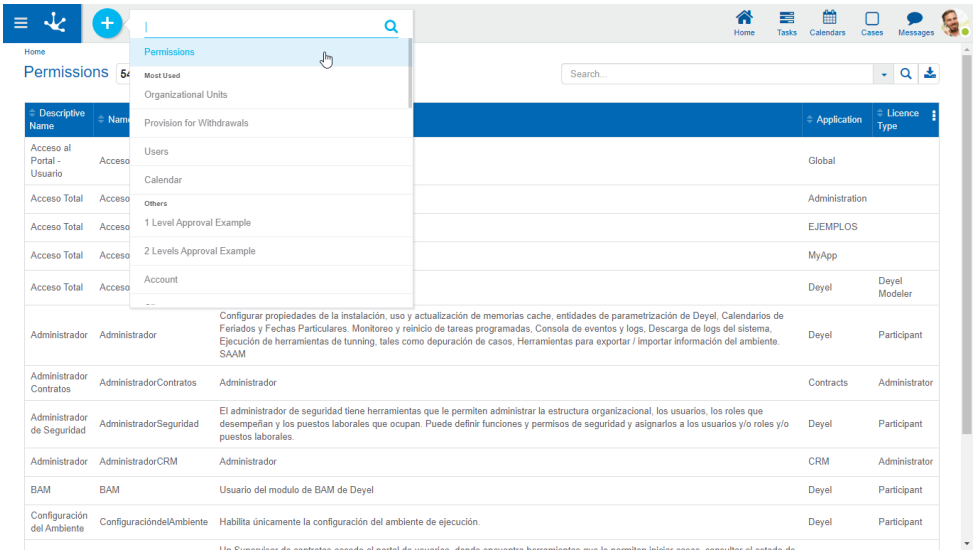
- Descriptive Name
- Name
- Description
- Application
- Type of License

Operations

It is possible to realize operations on each permit, depending on the security permits that the login user has defined. When hovering over each of the lines the buttons with available operations are visualized.

The exception is the operation "Create", that can be realized from the icon corresponding to the context menu in different ways.

- Hover the mouse over the icon and select the icon available on its right .
- Click on the icon and on the vertical panel displayed, select the option Permits.



3.10.2.5.2. Operations

The buttons to realize operations on the permits are enabled by hovering over one of them on the grid, with exception of the create operation, that is initiated from the icon corresponding to the context menu.

Create

Open properties panel of the new permit with all of them as editable. The creation is realized by pressing any of the buttons available and the user receives a message indicating that the data were saved.

Buttons

- Accept and Create: Confirms the creation of a permit and opens a new panel to create another one.
- Accept: Confirms the creation of a new permit.

Functions

In the panel in which the set of security functions is defined which are authorized through the permit, functions included in it can be selected.

Show

Opens the properties panel of the selected permit, where the properties are not editable and depending on the security permits of the user, the buttons available to operate with the permit that is being consulted are enabled.

Buttons

- Modify
- Copy
- Delete

Modify

Opens properties panel of the selected job position, with those properties that can be modified, as editable. The modification is realized by pressing the button “Accept” and the user receives a message indicating that the data were saved.

Copy

Opens a panel with the replica of the properties of the permit that is being consulted, with its properties as editable. The necessary modifications must be realized and when pressing the button “Accept”, the user receives a message indicating that the data were saved.
When the permit corresponds to a licensed application, it can only be copied if its license is valid.

Descriptive Name

This property takes the value of the origin permit, adding by default a suffix indicating the number of copy generated However this value can be modified.

Name

This property takes the value of the origin permit, adding by default a suffix indicating the number of copy generated However this value can be modified.

Delete

Opens the properties panel of the selected permit, where the properties are not editable. The elimination is realized by pressing the button “Accept” and the user receives a message indicating that the data was eliminated.

> *The permit cannot be deleted if there are users or roles who have it assigned.*
> *The permit "Account Manager" is a protected permit and it cannot be deleted.*

3.10.2.5.3. Predefined Permits

The predefined security permits are described below in **Deyel** and the feature of each of them.

Deyel

End User

- Accesses the users portal, where they find the tools which allow them to initiate cases, show the state of the cases they have initiated, show their pending tasks list and execute them, use the courier, etc.
- Uses forms to administer business entities, being able to show or modify each entity according to the access right they have on the entity.

Agile Modeler

- Includes the feature of the end user.
- Uses the assistant of forms modeling to build, document and implement the forms, processes and business entities of the organization.
- Defines roles and parameterization entities.
- Defines business rules to include logic in their processes.
- Uses tools to export/import their models and information of the environment.

Deyel Developer

- Includes the feature of the end user.
- Uses tools of graphical processes modeling and forms to build, document and implement the process and business entities of the organization.
- Defines roles, agents and parameterization entities.
- Implements more complex business logic using assistants that allow them to incorporate Java components.
- Implements the integration with other applications or external data bases.
- Implements behavior or more complex validations to the forms, using the advanced edition of scripting.
- Uses tools to export/import their models and information of the environment.

Tester

- Includes the feature of the end user.
- Has tools that allow they to define and maintain the test plans of the different processes, record individual test cases and define random scenarios.
- Uses tools to automate the execution of the test plans and monitors the results.

BAM

- User of BAM module of **Deyel**.

Guest in Tedis

- Includes the feature of the end user.
- It is used to limit the capacities of use of this collaborative platform.
  A guest user can participate in conversations, but not initiate them.

Security Administrator

- Includes the feature of the end user.
- It has tools that allow them to administer the organizational structure, the users and the roles they play.
- It can define functions and security permits and assign them to the users and roles.

Adapter Manager of Twitter Integration

- Allows to configure the adapter of integration with Twitter

Administrator

- Includes the feature of the end user.
- Uses tools to configure:
  - The properties of the installation (DB connection, integration with mail server, etc.).
  - The use and upgrade of cache memories.
  - The parameterization entities of **Deyel**.
  - Holiday calendars and particular dates
  - Glossaries
- Uses habitually the following features:
  - Monitoring and reboot of programmed tasks.
  - Events console and logs.
  - System log download.
  - Execution of tuning tools, such as case debugging.
  - Tools to export/import information of the environment.

Environment Configuration

- It only allows entry to the environment configuration.

Total Access

- Access to all the functions of **Deyel**.

Global

Account Manager

Allows to identify the user that performs as " Account Manager".
This is a protected security permit, that cannot be removed from the installation or modified.
Only a user can have this security permit assigned.. This is controlled in the creation and in the modification of users.

> *In On-Premise installations, this permit can be assigned or removed from a user by the security administrator.*
> *In Cloud installations, you must use the site of **Deyel** to change the user that performs as " Account Manager".*

Besides, this permit cannot be assigned to roles or job positions.

Modeler

Groups all the security functionsof modeling and of use of the objects, giving agility to the assignment of permits particularly at the moment of creating objects.
Every object created, has its security functions assigned to this permit.

CRM

Seller

- Includes the feature of the end user of **Deyel**, without access to agile forms.
- Includes access permits for CRM sellers.

Sales Manager

- Includes the feature of the end user of **Deyel**.
- Includes access permits for CRM sales managers.

Administrator

- Includes the feature of the end user of **Deyel**.
- Includes access permits for administration functions of Security and of CRM.

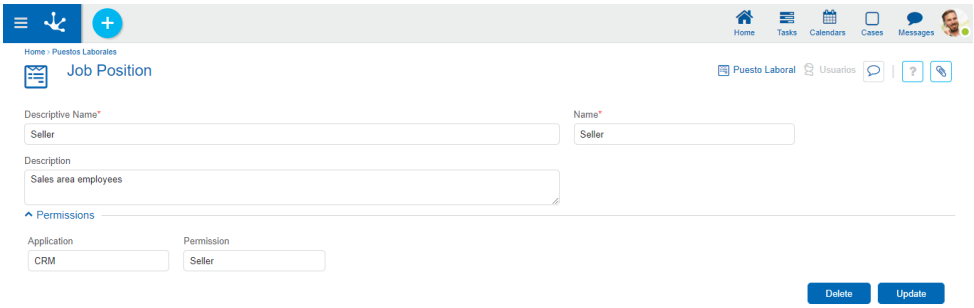
3.10.2.6. Job Positions

Phase 8: Configuration > Roles and job positions

A job position defines in **Deyel** the activities that a user or group of users develop within the organization.

Each job position can have assigned a set of security permits, that are inherited by the users who occupy that position, which allows to simplify assigning permissions to users.

This concept is used, to associate users to the job position they occupy and know which are their skills. From a job position can be visualized in the upper right section the relation with the users, with the indication of the number of users that are associated to the job position consulted. If you click on the name of the relation "Users", a grid is visualized of the users associated to the job position, if the number was greater than 1, or the properties panel of the user, in case that such number was 1.



An asterisk "" on the label indicates that the property is required.*

Properties

Descriptive Name
Complete name of the job position. This text is the one visualized on the grid of job positions.

Name
It is an abbreviated or reduced name. It is used when referring to the job position in an error message or in any mention that needs to abbreviate the descriptive name. This property does not admit blank spaces.

Description
Details the information about the job position, extending the property Descriptive Name.
Allows to include a more detailed description of the tasks that the user occupying that job position realizes.

Assigned permits
Security permits can be added to the job position.

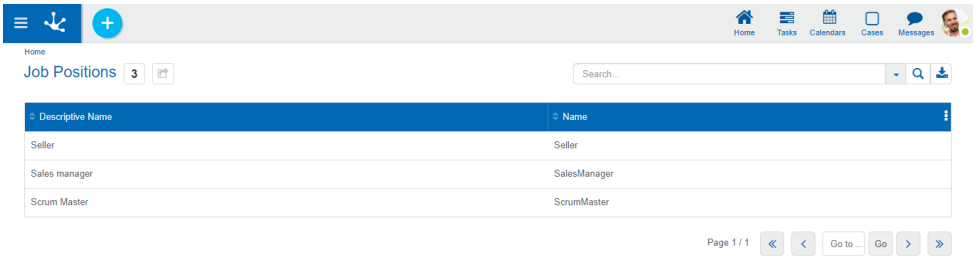
Application
Allows to define the application list that can be used by the users who have a job position assigned within their profile.

Permits
Allows to define a security permit predefined for each application entered in the property Application.

3.10.2.6.1. Grid

To realize the creation and administration of job positions, you can access the grid from the menu of **Deyel**.

- From the option “Configuration”, selecting "Security" and then “Job Positions”.
- From the menu search.



The job positions are visualized as a grid, with the standard presentation of the grids of **Deyel**, with facilities of:

- Ordering
- Page
- Search Bar and Filters
- Data Download

The number displayed on the right of the grid title, indicates the amount of job positions included in it.

The following properties of the job positions are visualized as columns of the grid:

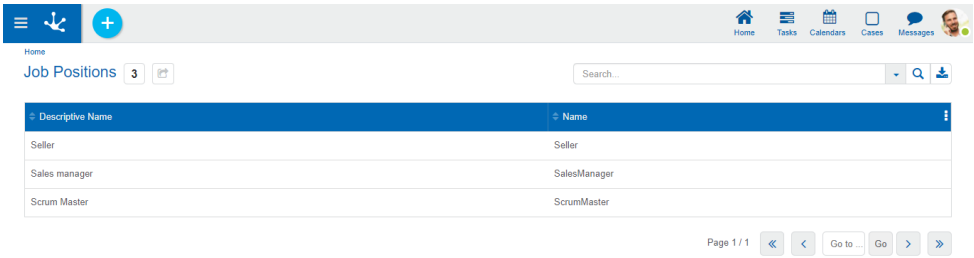
- Descriptive Name
- Name

Operations

It is possible to realize operations on each job position, depending on the security permits that the login user has defined. When hovering over each of the lines the buttons with available operations are visualized.

The exception is the operation "Create", that can be realized from the icon corresponding to the context menu in different ways.

- Hover the mouse over the icon and select the icon available on its right .
- Click on the icon and on the vertical panel displayed, select the option Job Positions.



3.10.2.6.2. Operations

The buttons to realize operations on the job positions are enabled by hovering over one of them on the grid, with exception of the create operation, that is initiated from the icon corresponding to the context menu.

Create

Opens properties panel of the new job position, where these are editable. The creation is realized by pressing any of the buttons available and the user receives a message indicating that the data were saved.

Buttons

- Accept and Create Confirm the creation of a job position and opens a new panel to create another one.
- Accept: Confirms the creation of a new job position.

Show

Opens the properties panel of the selected job position, where the properties are not editable and depending on the security permits of the user, the buttons available to operate with the job position that is being consulted are enabled.

Buttons

- Modify
- Delete

Modify

Opens properties panel of the selected job position, with those properties that can be modified, as editable. The modification is realized by pressing the button “Accept” and the user receives a message indicating that the data were saved.

Delete

Opens the properties panel of the selected job position, where its properties are not editable. The elimination is realized by pressing the button “Accept” and the user receives a message indicating that the data was eliminated.

3.10.2.7. Roles

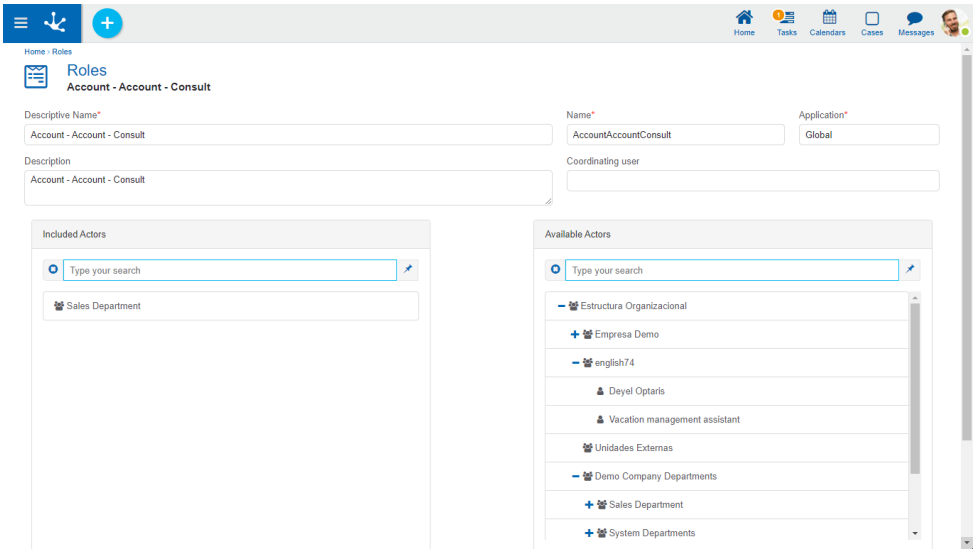
Phase 8: Configuration > Roles and job positions

While the organizational structure of **Deyel** defines the areas or offices of the company, the hierarchical relations between them and the actors that make it up, the roles allow to define groups of actors that have activities and common responsibilities, but are cross or parallel to the organizational structure.

The actors of a role can be users, organizational units or users that belong to these. Each role can have defined a set of permits of security, that are assigned to the users comprising such role or to the users whose organizational unit integrates the role.

In processes modeling of business, this concept is used to define that an activity can be executed by any of the integrants of the role. In agile forms, roles can be used to define who can realize operations on the form instances.

An organizational unit and a user can belong to multiple roles.



An asterisk "" on the label indicates that the property is required*

Properties

Descriptive Name
Complete name of the role. This text is the one visualized on the grid of roles.

Name
It is an abbreviated or reduced name. It is used when referring to the role in an error message or in any mention that needs to abbreviate the descriptive name. This property does not admit blank spaces.

Application
Defines the application the role belongs to, through the selection of a list of available applications.
The licensed applications are included in such list only if the use license of the application is current in the environment.

Description
Details the information about the role, extending the property Descriptive Name.
Allows to document the purpose for which the role has been generated, in which process is used, etc.

Coordinator User
A coordinator user of the role is optionally defined, realizing a selection from the user search.

Current Actors
Panel in which the users and/or organizational units that are role actors, can be defined and visualized in list form.
The role must have at least a current actor, that is the list of current actors cannot be empty.
On top of the panel it is visualized a search bar of actors, from where filters can be applied to the user names and organizational units to reduce the list.
It is possible to move elements from this panel from or to the panel Available Actors, as detailed under the title Transfer of Actors.

Available Actors
Panel from where you can select users and/or organizational units available to be defined as role actors.
On top of the panel it is visualized a bar of search of actors, from where filters can be applied to the user names and organizational units to reduce the list.
It is possible to move elements from this panel from or to the panel Current Actors, as detailed under the title Transfer of Actors.

Permits associated to the Role.
This property allows to define security permits to the role actors, that is that actors who integrate the role or that play in units which are role actors, they inherit the security permits defined here. It is an indirect way of assigning security permits.

It is not allowed for the permit "Account Manager" to be associated to roles.

Transfer of Actors

The transfer of actors between the panels Current Actors and Available Actors is realized by selecting the actors to transfer in the source panel and pressing the icon corresponding to the sense.

Transfers the selected actors in the panel of available actors to the panel of current actors.

Transfers the selected actors in the panel of current actors to the panel of available actors.

The icon allows to select all the actors of the corresponding panel, or remove the selection.

Search of Actors

The search of actors is realized by clicking on the chosen lines, or using the search facility of each panel of actors.

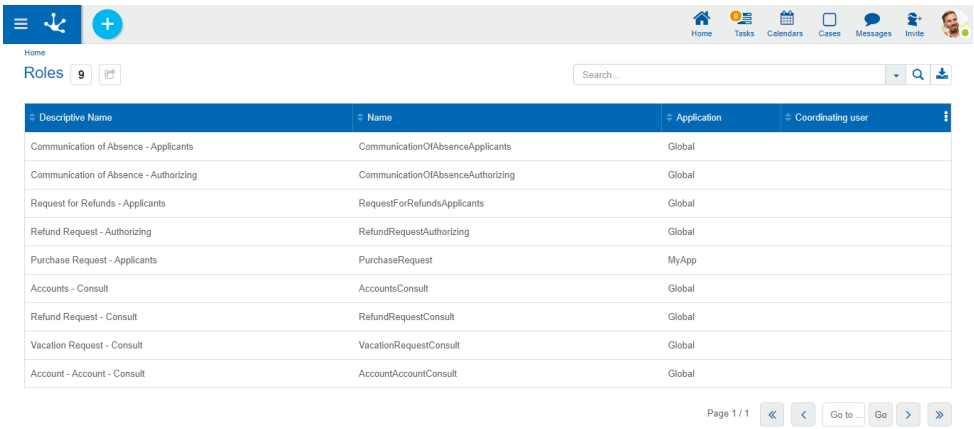
If a text is entered and the key "Enter" is pressed, the panel updates showing only the actors whose name contains the text entered. The results are features in yellow.

The icon allows to delete the defined selection criteria and all lines on the panel are displayed again.

3.10.2.7.1. Grid

To realize the creation and administration of roles, you can access the grid from the menu of **Deyel**.

- From the option “Configuration”, selecting "Security" and then “Roles”.
- From the menu search.



The roles are visualized as a grid, with the standard presentation of the grids of **Deyel**, with facilities of:

- Ordering

- Page
- Search Bar and Filters
- Data Download

The number displayed on the right of the grid title, indicates the amount of roles included in it.

The following properties of the roles are visualized as columns of the grid:

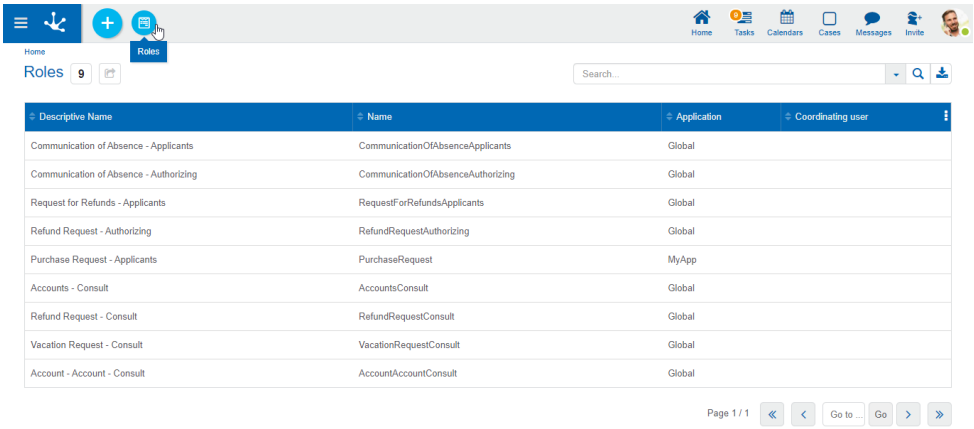
- Descriptive Name
- Name
- Application
- Coordinator User

Operations

It is possible to realize operations on each role, depending on the security permits that the login user has defined. When hovering over each of the lines the buttons with available operations are visualized.

The exception is the operation "Create", can be realized from the icon corresponding to the context menu in different ways.

- Hover the mouse over the icon and select the icon available on its right .
- Click on the icon and on the vertical panel displayed, select the option Roles.



3.10.2.7.2. Operations

The buttons to realize operations on the roles are enabled by hovering over one of them on the grid, with exception of the create operation, that is initiated from the icon corresponding to the context menu.

Create

Opens properties panel of the new role, where these are editable. The creation is realized by pressing the button “Accept” and the user receives a message indicating that the data were saved.

Show

Opens properties panel of the selected role, where these are not editable and the icons are not visualized for the transfer of actors, however the search of actors is enabled. Depending on the security permits of the user, the buttons available to operate with role that is being consulted are enabled.

Buttons

- Modify
- Delete

Modify

Open properties panel of the selected role, with those properties that can be modified as editable. The modification is realized by pressing the button “Accept” and the user receives a message indicating that the data were saved.

Application

It is not allowed to modify the role if the application is licensed and its use license is not available.

Coordinator User

When the role is responsible or participant of a process that is in use, it is not possible to delete the definition of the coordinator, although it is possible to define a new one.

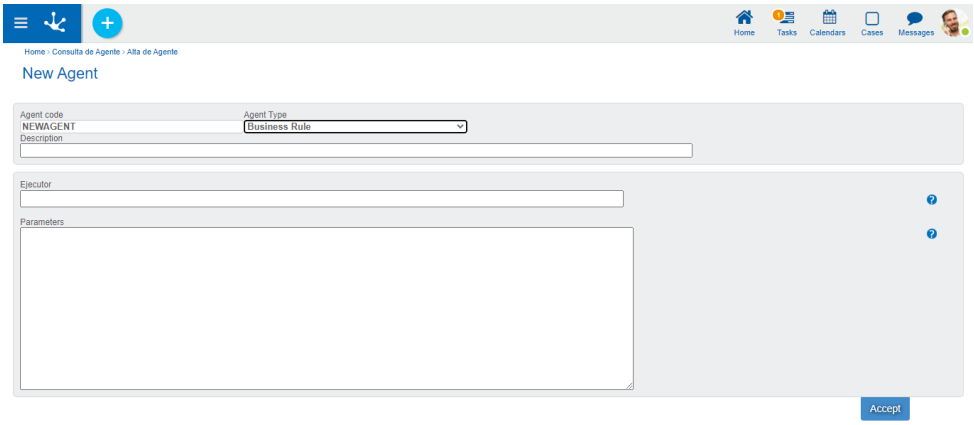
Delete

Opens properties panel of the selected role, where these are not editable and the icons are not visualized for the transfer of actors, however the search of actors is enabled.. The elimination is realized by pressing the button “Accept” and the user receives a message indicating that the data were eliminated.

> *A role cannot be deleted if it is responsible or participant of a process in use.*

3.10.2.8. Agentes

Un agente es un tipo de participante que permite determinar dinámicamente un usuario, una unidad organizacional o un rol, en base a cierta lógica durante la ejecución de un proceso.
Los agentes pueden utilizarse, al igual que los demás tipos de participantes, en la definición de un proceso para establecer quienes pueden ejecutar las actividades que se encuentran en un lane.



Propiedades

Código de Agente

Nombre que identifica al agente. Este texto se visualiza en la grilla de agentes.

Tipo de Agente

Se informa Regla de Negocio de manera predeterminada. Este tipo de agente invoca una regla de negocio que debe estar publicada.

Descripción

Nombre completo del agente. Este texto se visualiza en la grilla de agentes.

Ejecutor

Contiene el nombre y versión de la regla a invocar, separando estos datos por un carácter '-'. Por ejemplo, DeyelAgentExample-1.

Parámetros

Varían de acuerdo a la definición de la regla a utilizar. En principio la regla debe tener como mínimo los siguientes parámetros:
-Dos parámetros de salida obligatorios:

- 'participantCode'
  De tipo java.util.String, para indicar el código del participante seleccionado.
- 'participantType'
  De tipo java.util.String, para indicar el tipo de participante seleccionado. Solo se admite informar USER, ORG_UNIT y ROLE).

-Un parámetro de salida opcional:

- 'errors'
  De tipo java.util.List o compatible, para indicar mensajes de error en caso de ser necesario.

-Cuatro parámetros de entrada opcionales:

- 'caseNumber'
  De tipo java.lang.String donde recibe el número de caso en ejecución.

- 'process'
  De tipo java.lang.Integer donde recibe el código de proceso del caso en ejecución
- 'processVersion'
  De tipo java.lang.String donde recibe el código de versión de proceso del caso en ejecución
- 'activity'
  De tipo java.lang.Integer donde recibe el código de actividad para la cual se evalúa el agente.

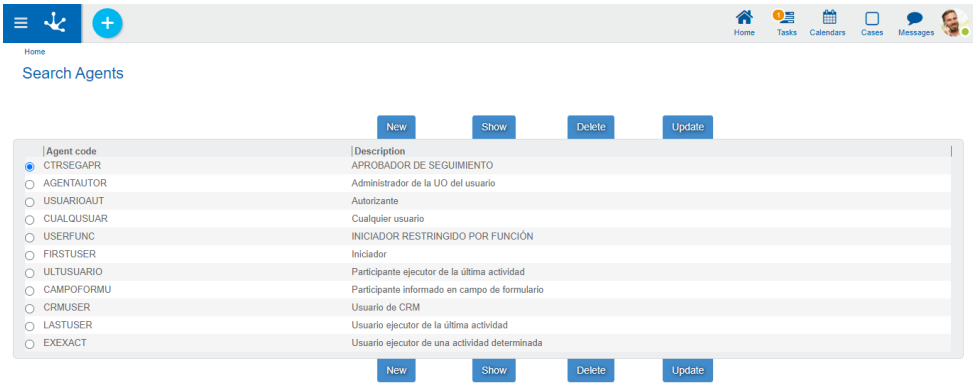
La regla puede tener parámetros de entrada adicionales, en cuyo caso se especifican en la sección de parámetros.

Presionando este ícono se despliega un asistente que analiza los parámetros indicados en la regla a ejecutar y asiste al usuario en su configuración.
El formato de los parámetros y sus valores es el siguiente: [param]=[value] por línea, donde [param] es el nombre de alguno de los parámetros y [value] puede ser un valor fijo o una variable del workflow, que se evalúa con información del caso actual al momento de la ejecución. Por ejemplo, pCdUser=|ONLINE-USER|.

3.10.2.8.1. Grilla

Para realizar la creación y administración de los agentes, se puede acceder a la grilla desde el menú de **Deyel**.

- Desde la opción “Configuración”, seleccionando "Seguridad" y a continuación “Agentes”.
- Desde la facilidad de búsqueda.



Se visualizan las siguientes propiedades de los agentes como columnas de la grilla:

- Código.
- Descripción.

Operaciones

Se pueden realizar operaciones sobre cada agente, dependiendo de los permisos de seguridad que tenga definidos el usuario conectado, seleccionando el agente y presionando el botón correspondiente.

3.10.2.8.2. Operaciones

Los botones para realizar cada una de las operaciones se encuentran arriba y debajo de la grilla. Con excepción de la operación crear, se debe seleccionar el agente antes de presionar el botón.

Crear

Abre el panel de propiedades del nuevo agente con todas ellas como editables. La creación se realiza al presionar el botón "Aceptar" y el usuario recibe un mensaje indicando que los datos fueron guardados.

Consultar

Abre el panel de propiedades del agente seleccionado. Sus propiedades se visualizan como no editables.

Modificar

Abre el panel de propiedades del agente seleccionado, con aquellas propiedades que pueden modificarse, como editables. La modificación se realiza al presionar el botón "Aceptar" y el usuario recibe un mensaje indicando que los datos fueron guardados.

Eliminar

Abre el panel de propiedades del agente seleccionado. Sus propiedades se visualizan como no editables. La eliminación se realiza al presionar el botón "Aceptar" y el usuario recibe un mensaje indicando que los datos fueron eliminados.

> *No se puede eliminar un agente si es responsable o participante de un proceso que está en uso.*

3.10.2.8.3. Agentes Predefinidos

Deyel provee un conjunto de agentes predefinidos que permiten resolver la mayoría de los casos en que se requiere asignar participantes dinámicamente. Estos agentes se pueden utilizar libremente.

- Usuario ejecutor de una actividad determinada

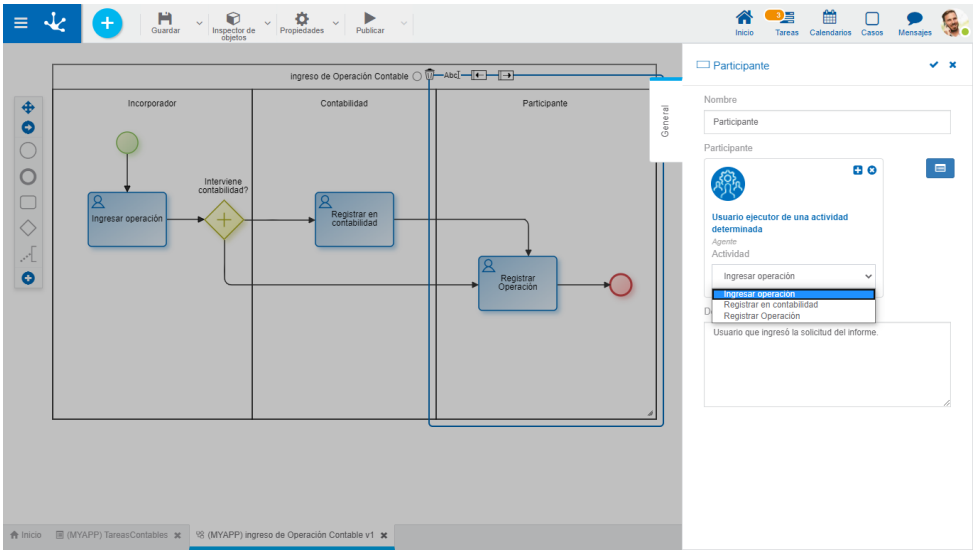
- Participante informado en campo de formulario
- Administrador de la UO del usuario
- Cualquier usuario
- Autorizante
- Usuario ejecutor de la última actividad
- Participante ejecutor de la última actividad
- Iniciador
- Usuario de CRM
- Iniciador restringido por función

3.10.2.8.3.1. Usuario Ejecutor de una Actividad Determinada

Este agente retorna el usuario que realizó una determinada actividad. En general este agente se utiliza cuando se debe indicar que una actividad se realiza por el mismo usuario que ejecutó la actividad definida.

Ejemplo de Uso

La utilización del agente en este proceso indica que la actividad "Registrar operación" es ejecutada por el usuario que ejecutó la actividad "Ingresar operación".

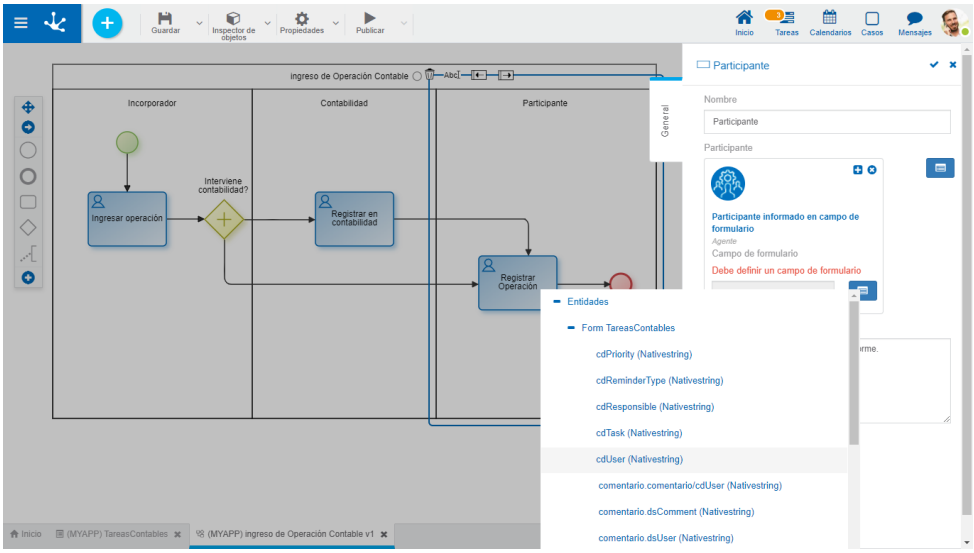


3.10.2.8.3.2. Participante Informado en Campo de Formulario

Este agente retorna un participante desde un campo de formulario. En general este agente se utiliza cuando se debe indicar que una actividad es realizada por el usuario que se indica en el campo del formulario definido.

Ejemplo de Uso

En el siguiente proceso, la utilización del agente indica que la actividad de "Registrar operación" es ejecutada por el usuario que se muestra en el campo "cdUser" del formulario.



3.10.2.8.3.3. Administrador de la UO del Usuario

Este agente retorna el usuario que es administrador de la unidad organizacional a la que pertenece el usuario ejecutor de la última actividad.

3.10.2.8.3.4. Cualquier Usuario

Este agente retorna como participante al usuario conectado actualmente. Si un proceso tiene como participante del lane inicial a este agente, todo usuario puede iniciar el caso. Se recomienda utilizar este agente solamente para definir el participante de la actividad inicial del proceso.

3.10.2.8.3.5. Autorizante

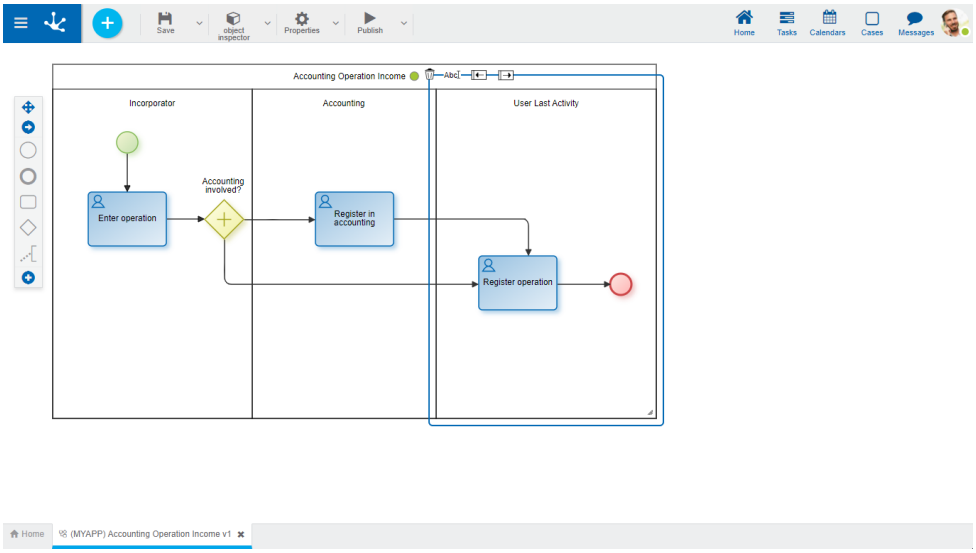
Este agente retorna el usuario configurado como autorizante para el usuario que ejecutó la última actividad.

3.10.2.8.3.6. Usuario Ejecutor de Ultima Actividad

Este agente retorna el usuario que inició la ejecución de la última actividad.
En general este agente se utiliza cuando se debe indicar que una actividad se realiza por el mismo usuario que ejecuta la actividad que le precede.

Ejemplo de Uso

La utilización del agente en este proceso indica que la actividad "Registrar operación" es ejecutada por el usuario que inició el caso o por el usuario que ejecutó la tarea "Registrar en contabilidad", si el caso tuvo intervención del área contable.

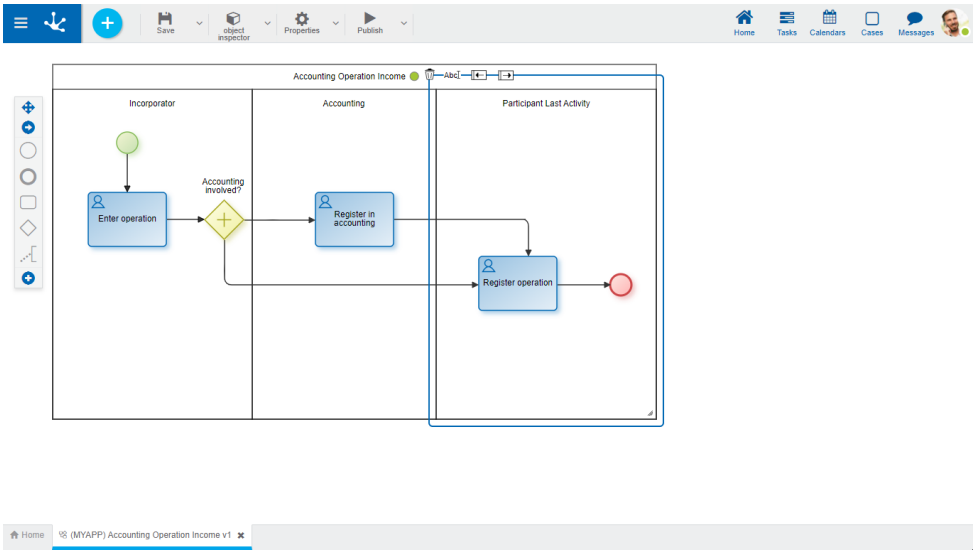


3.10.2.8.3.7. Participante Ejecutor de Ultima Actividad

Este agente retorna el participante que inició la ejecución de la última actividad.
En general se utiliza para indicar que una actividad se realiza por el mismo participante.

Ejemplo de Uso

En el siguiente proceso, la utilización del agente indica que la actividad de “Registrar operación” es ejecutada por el usuario que inició el caso o por la unidad organizacional “Contable”, si el caso tuvo intervención del área contable.



A diferencia del agente “Usuario ejecutor de la última actividad”, que siempre retorna el usuario que ejecutó la última actividad, este agente puede asignar el caso a la unidad organizacional o al rol que estén configurados en la última actividad ejecutada.

3.10.2.8.3.8. Iniciador

Este agente retorna el usuario que ejecutó la primera actividad del proceso. No se puede utilizar en la primera actividad de un proceso.

3.10.2.8.3.9. Usuario de CRM

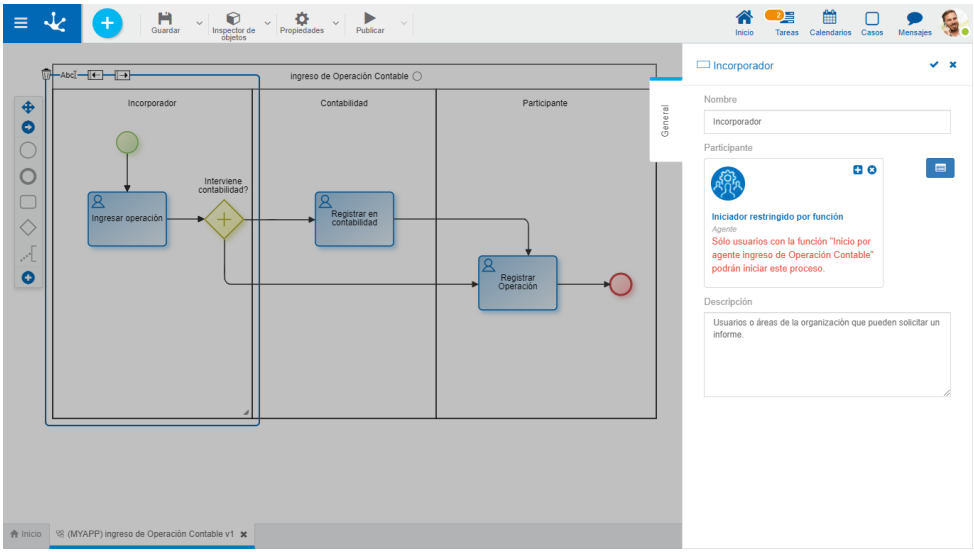
Retorna como participante al usuario conectado actualmente solo si éste tiene licencia de CRM. Si un proceso tiene como participante del lane inicial a este agente, todo usuario con licencia de CRM puede iniciar el caso. Se recomienda utilizar este agente solamente para definir el participante de la actividad inicial del proceso.

3.10.2.8.3.10. Iniciador Restringido por Función

Este agente evalúa si el usuario actual tiene en sus permisos la función de uso "Iniciar proceso" del actual proceso. Solo puede usarse este agente en el primer lane.

Ejemplo de Uso

La utilización del agente en este proceso indica que la actividad "Registrar operación" debe ser ejecutada por un usuario que tenga habilitado el permiso con la función "Inicio por agente ingreso de operación contable".



3.10.2.9. Security Functions

A security function represents an operation that can be realized on an object. It can be included in one or more permits thus authorizing, the users that have those permits, to realize such operation.

Every operation available in **Deyel** or in its solutions, has a corresponding security function defined. Besides, every object that is modeled has a set of security functions, that is automatically created, when saving and publishing the object.

Of Modeling

They authorize the execution of the modeling operations, such as save, publish, export and delete.

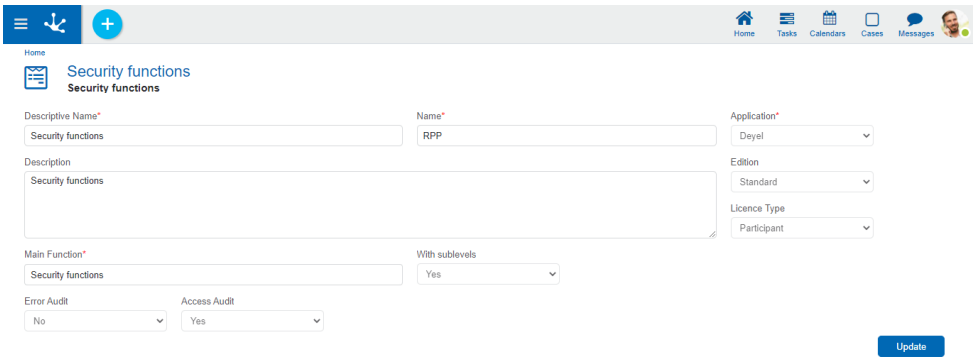
They are generated when saving the object and its assignment to permits has an immediate impact when saving or publishing an object.

Of Use

They authorize the execution of the executions on the instances of the objects, like create, modify, delete, show, when they are used from the portal.

They are generated when the object is published. The modifications in the assignment of use functions do not have a real impact on the use of the object, until it is published.

The security functions can be shown or assigned to a permit from the option “Permits” on the menu and particularly the security functions of the modeled objects, from the object´s modeler as well.



Properties

Descriptive Name
Complete name of the security function.This text is the one visualized on the grid of security functions..

Name
It is an abbreviated name. It is used when referring to the security function in an error message or in any mention that needs to abbreviate the descriptive name.

Application
Indicates the applications to which the security function belongs to, it coincides with the application of his father function.

Edition
It indicates the edition of **Deyel** or the licensed application required for the execution of the function. Some functions are available only in the Enterprise edition of **Deyel** or Professional of the solutions.

Type of License
Indicates the type of license the user needs to have assigned in order to execute the function. It is used for functions of Deyel or licensed applications.

Description
Property where more detail of the security function is found, extending what was expressed in the property Descriptive Name.

Father Function

Allows to identify the superior security function inside the hierarchical structure.
There exists a father function for each available solution and for each modeled application.

It has Sublevels

Indicates that the security function has subordinated functions.

Access Audits

When the value "YES" is defined an entry is generated in the audit register each time a user executes the security function.
Using the Logs Console these audit trails can be recovered and evaluate the use of the different security functions.

Audits for Errors

When the value "YES" is defined an entry is generated in the audit register each time a mistake occurs during the execution of the security function.
Using the Logs Console these audit trails can be recovered and evaluate the occurrence of these type of mistakes.

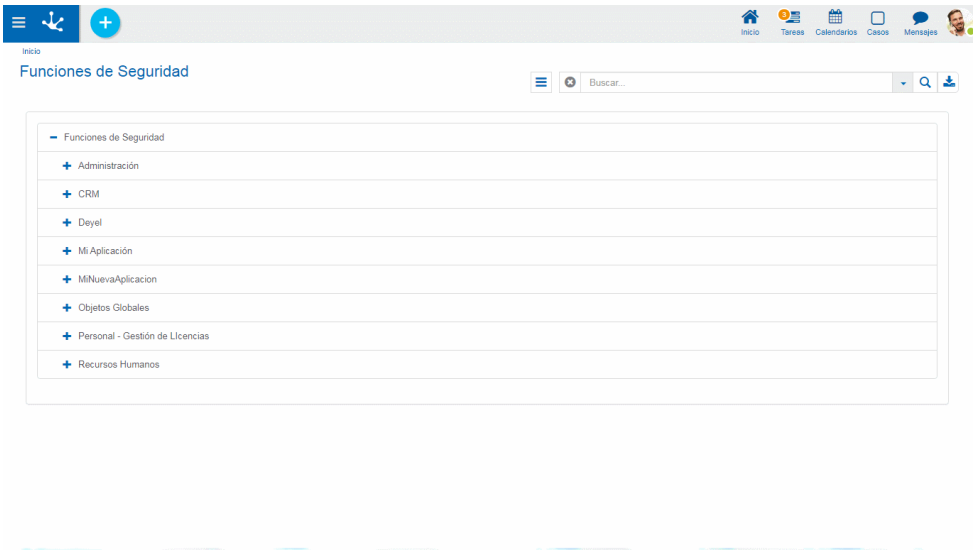
3.10.2.9.1. Functional Structure

For inquiries of the security functions, it is possible access the functional structure from the menu of **Deyel.**

- From the optionConfiguration", selecting "Security" and then "Security Functions".
- From the option of menu search.

The security functions are identified from its property Descriptive Name and automatically generated when the objects. The visualization of such functions can be realized in the form of hierarchical view or as an object grid.

Hierarchical View



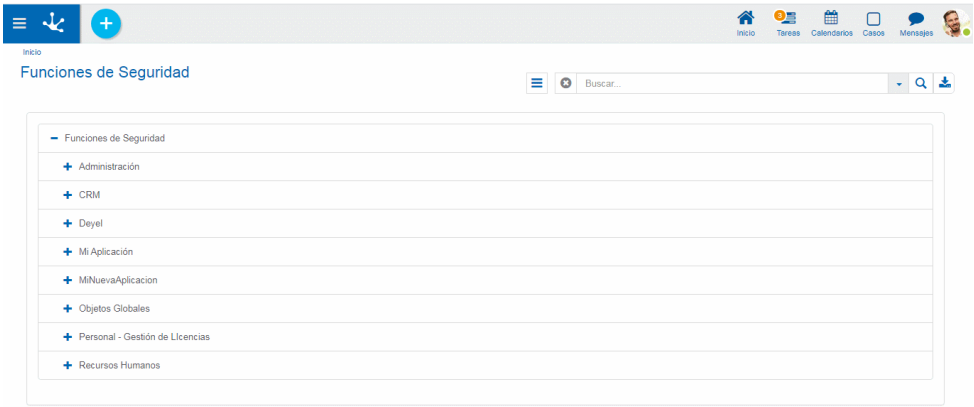
The father element, which name is the name of the application, is located as sublevel of the stop element "Security Functions" and depending on it the security functions of the application and its objects are visualized. For each modeled object, its corresponding modeling security functions and of use are created.

The icon allows to change the visualization to grid mode.

Grid

The grid of security functions allows to visualize them in the same way as the objects in the Deyel modeler and in the results grid of user forms, with facilities of:

- Ordering
- Page
- Search Bar and Filters
- Data Download



The icon allows to change the visualization to hierarchical view mode.

The number displayed on the right of the grid title, indicates the amount of security functions included in it.

The following properties of the security functions are visualized as columns of the grid:

- Name
- Descriptive Name
- Application
- Father Function
- Edition
- Type of License
- Code

Operations

From the hierarchical view and from the grid inquiries can be made. When hovering over each of the lines the button "Show" is visualized which allows to display all the properties of the selected security function.

3.10.2.9.2. Operations

The buttons to realize operations on the security functions are enabled when hovering over one of them in the functional structure, whether it is visualized as hierarchical or as a grid.

Show

Opens properties panel of the selected security function. The properties are visualized as not editable and depending on the security permits of the user, the button to modify the security function that is being consulted is enabled.

Modify

Opens properties panel of the selected security function, with those properties that can be modified, as editable. The modification is realized by pressing the button "Accept" and the user receives a message indicating that the data were saved.

> *When the security function belongs to a licensed application, only the attributes that define audits for access and audits for errors can be modified.*

3.10.2.10. Security in Entities

Phase 2: Forms Modeling > Advanced Tips > Entities Privacy

Security in an entity is related to the access the users have to the instances of such entity, through their user interface which is named form. An instance of an entity corresponds to a created form, that contains data in its fields.

Entities Classification according to the Security Level

Public
The only security that controls the access to a public entity is the one given by the user profile.

Private
An entity is private when besides the defined security in the user profiles, it exists an additional verification per entity instance.

The verification takes into account the following concepts:

- Instance Owner
  Identification of the user that has access permits to display, modify and delete the particular instance of an entity.

- Access according to Hierarchy
  The access to data of instance besides the owner, takes into account the definition of the organizational unit the owner user belongs to. If a user belongs to a unit that has the access properties to the private data defined, the coordinator of such unit can have some of the defined access for the instances of their subordinates.

- Security to functions
  Allows the user that is not owner of instances, through security functions assigned to a permit, to have access to these instances despite being private. The user is allowed to show, modify or delete the private instances.

3.10.3. Localización

Deyel permite configurar ciertos aspectos que dependen de la localización geográfica de la instalación cliente.
Por ejemplo, los calendarios de feriados o la traducción de etiquetas de la interfaz, son aspectos que pueden variar en cada país o región.

Calendarios

Permite administrar los calendarios laborales.
Un calendario laboral se utiliza para configurar la duración de las jornadas laborales y los horarios de trabajo. Esta definición es necesaria para que el sistema pueda calcular plazos de vencimiento, cuando se considera tiempo laboral.
En general **Deyel** cuenta con un calendario por defecto, que define los horarios en la organización. No obstante, cuando se trata de organizaciones que funcionan con oficinas distribuidas y que tienen diferentes usos horarios o diferentes días festivos, se pueden definir calendarios particulares y asociarlos a las unidades organizacionales que corresponda.

3.10.3.1. Calendarios

Un calendario se utiliza para definir periodos laborables. **Deyel** provee un calendario estándar, que se utiliza por defecto, cuando no se definen otros calendarios.
El usuario que administra la configuración del sistema, puede cambiar este calendario estándar, o definir otros, para establecer los horarios laborales, los días feriados, las fechas particulares, etc.

Cuando **Deyel** realiza un cálculo de tiempos laborales, por ejemplo para calcular la duración de una determinada actividad o determinar si corresponde enviar un alerta, se tienen en cuenta estos calendarios.

> *Calendario Estándar*
> *En toda instalación de **Deyel** existe el calendario con código 01 – Calendario Estándar, que no puede ser eliminado y que tampoco define zona horaria, ya que siempre tendrá asignada la zona horaria configurada en el equipo donde está instalado el sistema.*
> *Este es el calendario por defecto que rige para todo usuario y/o unidad que no defina su propio calendario.*

Propiedades

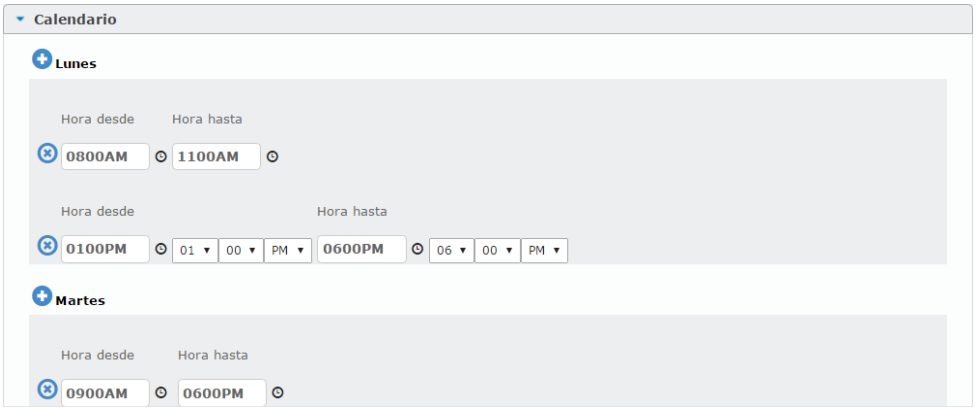
Código del Calendario
Identificador único del calendario.

Descripción del Calendario
Es el nombre o descripción del calendario.

Duración del día laboral,
Expresado en cantidad de horas y minutos. La duración del día laboral no puede ser cero.

Zona horaria
Deyel soporta la actividad de usuarios en diversas bandas horarias haciendo conversiones automáticas y brindando la comodidad al usuario de trabajar con datos de tipo fecha / hora según la zona horaria de su región. Cualquier dato de tipo fecha / hora se almacena respetando la fecha y hora real del servidor central, no obstante, cuando ese dato debe ser mostrado al usuario, se lo convierte automáticamente según la zona horaria del calendario usado por dicho usuario.
Escribiendo en el campo con autocompletar se podrá buscar la zona horaria correspondiente al calendario que se desea configurar.

Horario laboral
Para cada día de la semana se indica la hora inicial y final de cada periodo laborable. (Hora desde – Hora hasta).
Se puede especificar más de un intervalo para cada día.



Presionando el ícono se mostrará un asistente para la selección de hora, el cual podrá ser ocultado presionando el ícono nuevamente.

Definición de Feriados del Calendario
En esta sección se define la lista de feriados que impactarán en el calendario.
Para cada feriado se indica la fecha y la descripción del mismo.

Incluir Feriados Generales:
Este indicador establece si el calendario toma en cuenta los feriados definidos en la sección de Fechas Generales.

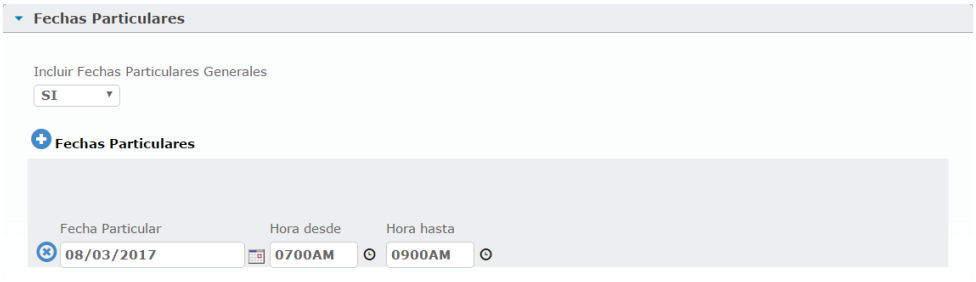
Definición de Fechas Particulares del Calendario
En esta sección se define la lista de fechas particulares que se consideran en el calendario.

En estas fechas, el horario laboral difiere del habitual, por tal razón se denominan particulares.
Para cada fecha particular se indica la fecha y los rangos horarios que se establecen como laborables para esa fecha particular.

> *Cuando una fecha es definida como particular y también como feriado, se toma dicha fecha como laborable, en los horarios establecidos.*

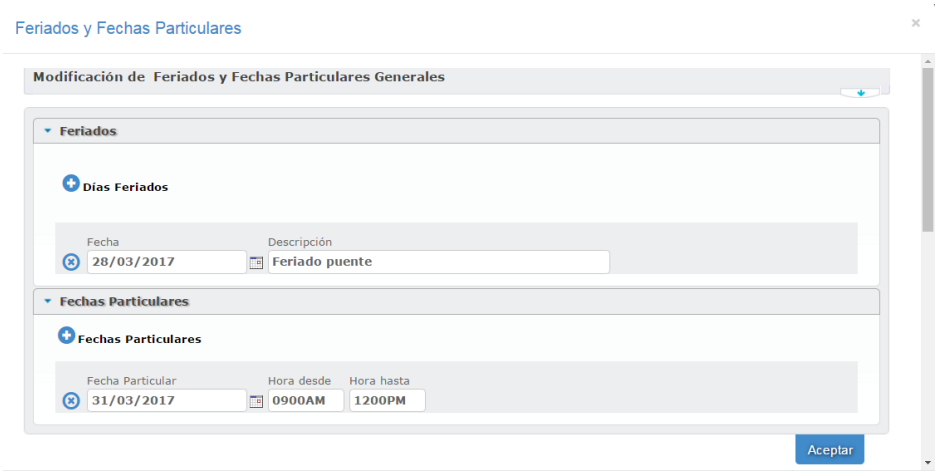
Incluir Fechas Particulares Generales:

Este indicador establece si el calendario toma en cuenta las fechas particulares definidas en la sección de Fechas Generales.



Fechas Generales.

Esta sección permite definir feriados o fechas particulares que son globales a toda la organización.
Se definen una única vez y luego pueden ser considerados por los distintos calendarios o no, según se defina en cada calendario.



La creación de feriados y fechas particulares generales es similar a la explicada en la configuración de un calendario particular.
Para los feriados generales se deberá ingresar la fecha y descripción del feriado, mientras que para las fechas particulares, se especificarán los días y los rangos horarios laborales.

> *Una vez que se configuran los calendarios, existen visualizaciones que cambian en torno a esta configuración.*
> *Principalmente pantallas donde se muestran fecha y hora, por ejemplo:*
>
> - *Mis Tareas*
> - *Consulta del caso y datos adicionales*
> - *Consulta de anexos del caso*
> - *Consulta múltiple de actividades ejecutadas*
> - *Consulta puntual de actividades ejecutadas*
>
> *En todas estas pantallas el usuario verá la hora en que ocurrieron las acciones de acuerdo a su zona horaria, por ejemplo, supongamos:*

-El usuario A, trabaja con un calendario cuya zona horaria es GMT-3.
-El usuario A finaliza la ejecución de una actividad X a las 10:00 AM

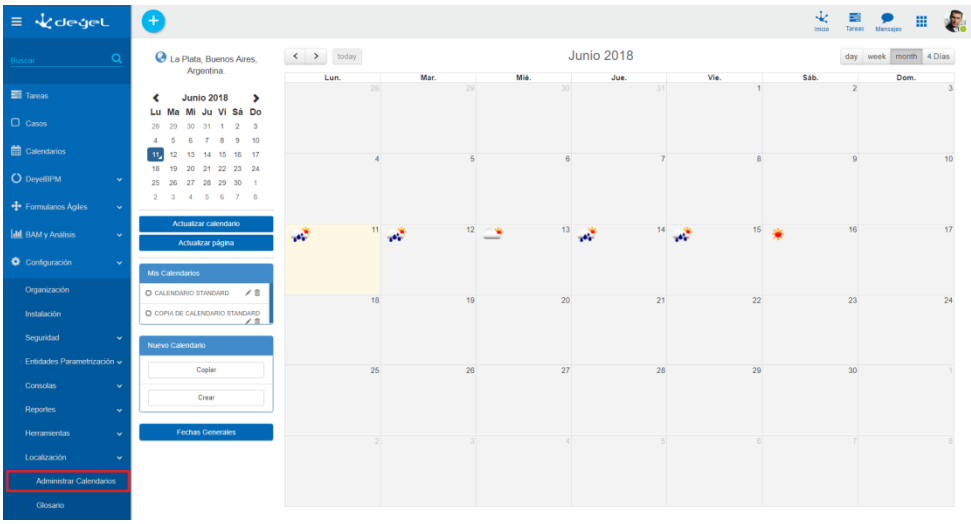
-El usuario B, trabaja con un calendario cuya zona horaria es GMT+1.
-El usuario B consulta la actividad X y verá que la misma finalizó a las 14:00 PM

3.10.3.1.1. Administrar Calendarios

Para acceder a la administración de calendarios existen distintas alternativas:

- Hacer clic sobre la opción “Configuración” del menú principal y dentro del sub menú "Localización", hacer clic en la opción "Administrar Calendarios".
- En la *búsqueda rápida del menú* ingresar “Administrar Calendarios” y hacer clic en la opción listada.

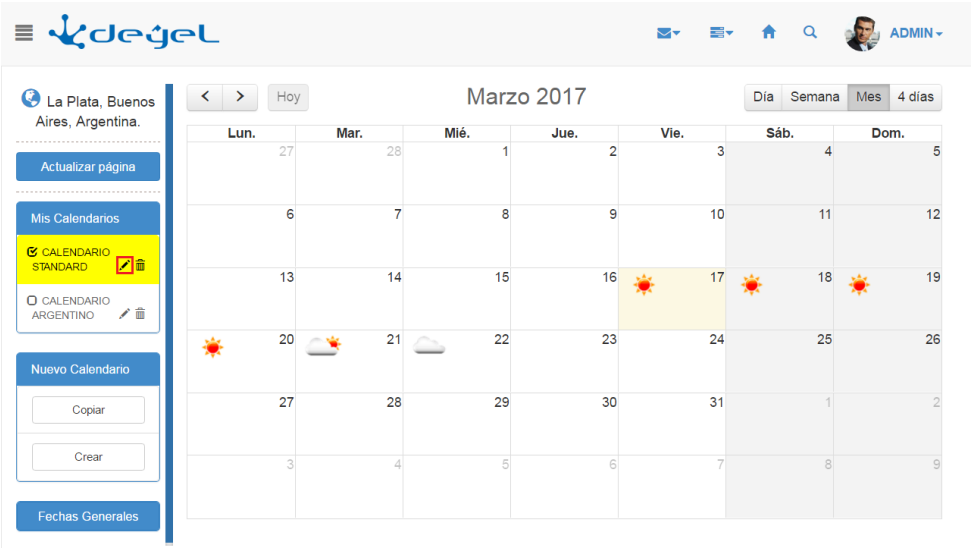
Como resultado se visualiza la siguiente ventana de administración de calendarios, que presenta los siguientes componentes principales.



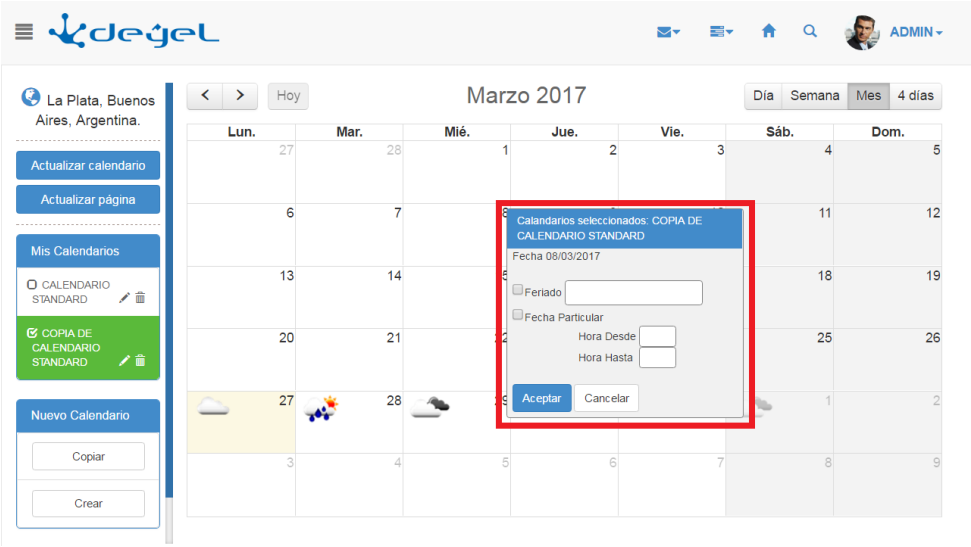
Actualizar Calendario
Actualiza el contenido del panel central.

Actualizar Página
Actualiza la página de configuración de calendarios.

Mis calendarios
Muestra la lista de los calendarios definidos.
Seleccionando cualquiera de ellos, en el panel central se visualizan los feriados y fechas particulares que se han definido en dicho calendario.
Para modificar o eliminar un calendario, se debe presionar los botones que se encuentran a su derecha



Otro mecanismo que permite modificar un calendario es definir feriados / fechas particulares utilizando el panel central.
Haciendo clic en un día determinado, se accede a la siguiente ventana



donde se indica si la fecha se considera feriado y/o fecha particular en los calendarios seleccionados (En el tope de la ventana se indica cuales son estos calendarios).

> *Calendario Estándar*
> *En toda instalación de **Deyel** existe el calendario con código 01 – Calendario Estándar, que no puede ser eliminado y que tampoco define zona horaria, ya que siempre tendrá asignada la zona horaria configurada en el equipo donde está instalado el sistema. Este es el calendario por defecto que rige para todo usuario y/o unidad que no defina su propio calendario.*

Creación de Nuevos Calendarios

Se ofrecen dos mecanismos que permiten generar un nuevo calendario:

- Crear

Permite crear un nuevo calendario.

- Copiar

Crea una copia del calendario seleccionado en "Mis Calendarios". El nombre del nuevo calendario coincide con el original, pero se le antepone el prefijo “Copia"

3.10.4. Ambiente

Deyel considera un conjunto de propiedades que permiten adecuar el funcionamiento del producto según necesidades particulares del cliente.

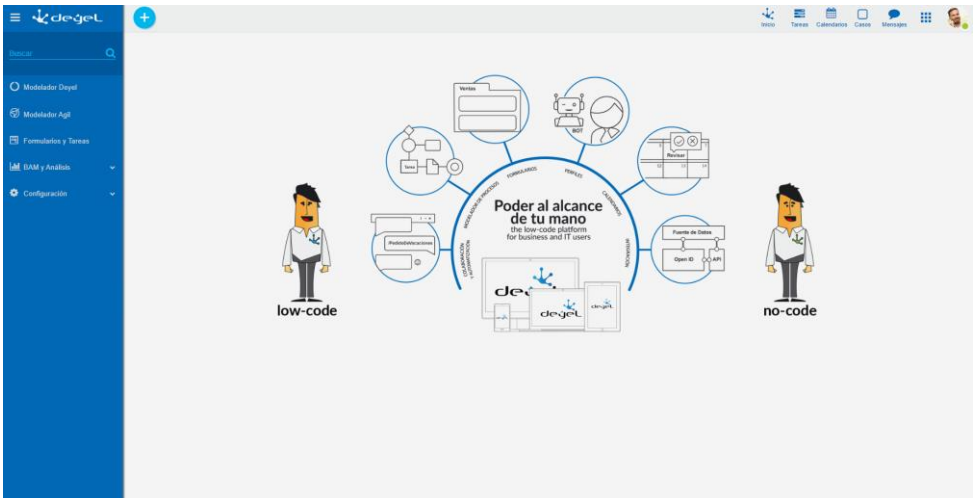
Para acceder a la configuración de propiedades del ambiente existen distintas alternativas:

- Hacer clic sobre la opción “Configuración” del menú lateral y hacer clic en la opción "Ambiente".
- En la búsqueda del menú ingresar “Ambiente” y hacer clic en la opción listada.

Algunas propiedades, permiten además que cada usuario pueda seleccionar el valor que prefiere.

Cuando **Deyel** recupera el valor de una propiedad, considera las siguientes precedencias

- Valor preferido del usuario conectado.
- Valor configurado en el ambiente.
- Valor predeterminado de la propiedad.

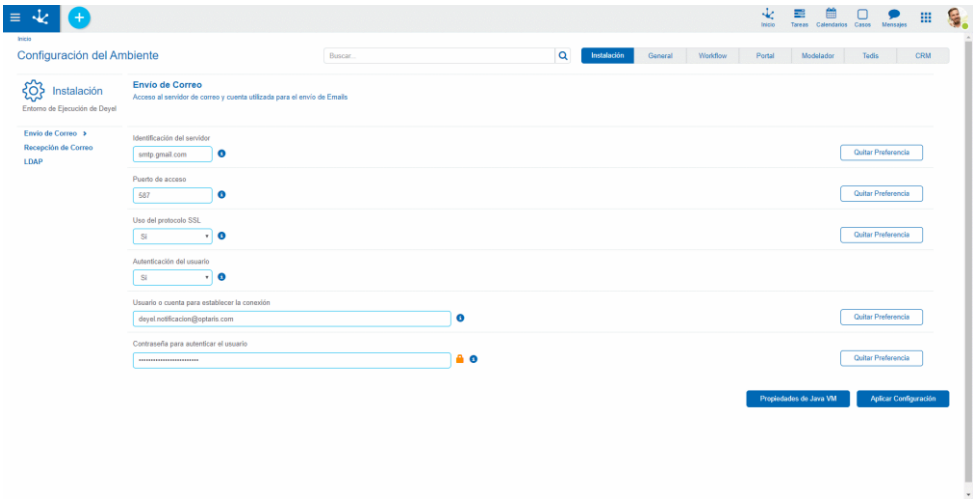


Descripción

En la parte superior de la pantalla existen botones que permiten acceder a diferentes grupos de propiedades

- Instalación, para configurar el entorno de ejecución de **Deyel**
- General, para configurar el comportamiento general de **Deyel**
- Workflow, para configurar el motor de Workflow
- Portal, para configurar el portal de usuarios
- Modelador, para configurar el modelador
- Tedis, para configurar la red social empresarial
- CRM, para configurar comportamientos específicos de la aplicación CRM

Cuando se presiona uno de estos botones, se despliegan las propiedades agrupadas por tema.
A la izquierda se visualiza una lista que permite acceder a las propiedades propias de cada tema.



Propiedades Configurables

Para cada propiedad se visualiza su nombre descriptivo y el valor actual, que puede ser modificado por el usuario. En algunos casos este valor se puede ingresar y en otros se selecciona desde una lista de valores posibles.
Todas las propiedades tienen un valor predeterminado, que puede ser nulo.

Cuando el usuario modifica el valor de una propiedad, puede presionar el botón "Deshacer Cambio" para volver atrás la modificación.
Esta opción de deshacer un cambio está disponible hasta que se presiona el botón "Aplicar Configuración". En ese momento, se almacenan todas las propiedades el repositorio de **Deyel** y se notifica al usuario con un mensaje de operación exitosa.

Cuando una propiedad tiene establecido un valor diferente del valor predeterminado, se visualiza el botón "Quitar preferencia", que cambia el valor actual por el predeterminado.

Existen propiedades que se pueden modificar dinámicamente y otras cuyo nuevo valor se aplica cuando el ambiente es reiniciado.

Las propiedades que requieren reinicio se visualizan con el ícono .

Algunas propiedades se encuentran encriptadas y su contenido se visualiza protegido. Se visualizan con el ícono , por ejemplo las contraseñas de usuarios.

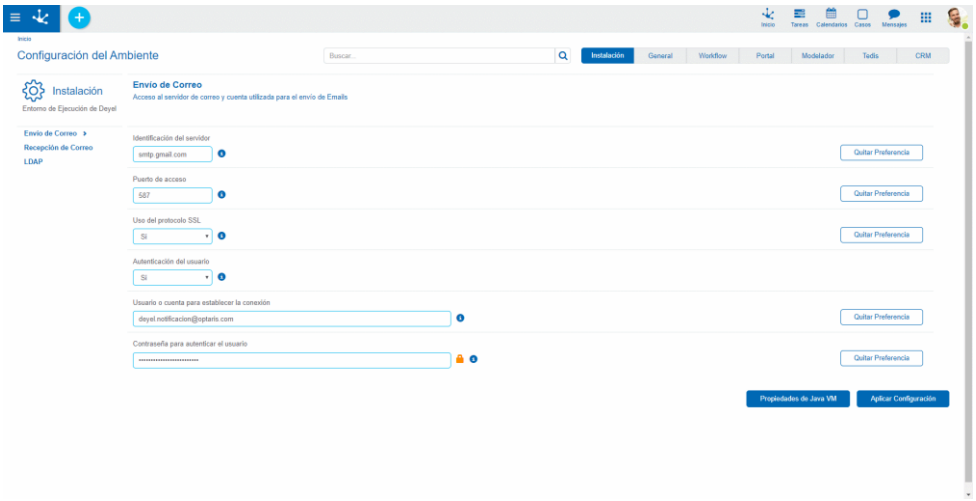
Presionando se accede a una descripción detallada de la propiedad.

Búsqueda

También se puede utilizar la herramienta "Buscar", indicando el código, nombre o descripción de la propiedad buscada, en forma total o parcial.

Los resultados se muestran en una lista, donde cada una de las propiedades expone la información descripta anteriormente,

Sobre cada una de las propiedades se visualiza un breadcrump, que indica a que tema pertenece la propiedad. Puede utilizarse ese breadcrump para dirigirse al módulo o tema indicado.



Propiedades de Java VM

Se visualizan presionando el botón "Mostrar Propiedades" y corresponden a la Java Virtual Machine correspondiente al servidor de aplicaciones en uso (IBM Websphere, Apache Tomcat). Estas propiedades no pueden modificarse y dependen de la versión de Java que se está ulilizando (Java para Linux, Windows, etc.).

3.10.4.1. Instalación

Permite configurar el ambiente de ejecución de **Deyel**.
Se trata de propiedades que establecen cómo está instalado el producto y cómo se integra con otros componentes de software, por ejemplo, con el servidor de email o con un servidor LDAP.

Temas

- Envío de Correo

- Recepción de Correo

- LDAP - Conexión con el Servidor
- LDAP - Busqueda de Usuarios
- LDAP - Sincronización de Atributos

- JAVA

3.10.4.1.1. Envío de Correo

Acceso al servidor de correo y cuenta utilizada para el envío de Emails.

> *IMPORTANTE .*
> *Si la propiedad* **ServidorEnvioMail** no *es informada, Deyel no envía notificaciones por correo. Cuando la propiedad es informada, Deyel intenta conectarse con el servidor de correo y autenticarse usando el usuario / clave indicados.*
> *De este modo se asegura al administrador que al momento de realizar la configuración el acceso al servidor de correo funciona correctamente.*

Propiedades Configurables

Identificación del servidor

Indica el nombre o dirección IP del servidor de correo utilizado para el envío de Emails. Por ejemplo, your.domain.com

Nombre	ServidorEnvioMail
Código	MAIL_SERVER
Niveles de Configuración	
• Instalación	✔
• Aplicación	—
• Unidad Organizacional	—
• Usuario	—
Es dinámica	✔
Es encriptada	—

Valor Predeterminado	your.domain.com

Puerto de acceso

Indica el puerto de acceso al servidor de correo. Por ejemplo, 25

Nombre	PuertoEnvioMail
Código	MAIL_SERVER_PORT
Niveles de Configuración	
• Instalación	✔
• Aplicación	—
• Unidad Organizacional	—
• Usuario	—
Es dinámica	✔
Es encriptada	—
Valor Predeterminado	25

Uso del protocolo SSL

Determina si utiliza SSL para el envío de mails (habilitar para cuentas tipo Gmail o similares).

Nombre	SSLEnvioMail
Código	SMTP_USES_SSL
Niveles de Configuración	
• Instalación	✔
• Aplicación	—
• Unidad Organizacional	—
• Usuario	—
Es dinámica	✔
Es encriptada	—
Valores Posibles	• Si • No (Predeterminado)

Autenticación del usuario

Indica si el servidor de correo requiere autenticación del usuario.

Nombre	AutenticacionEnvioMail
Código	MAIL_SERVER_ AUTHENTICATE
Niveles de Configuración	
• Instalación	✔
• Aplicación	—
• Unidad Organizacional	—
• Usuario	—
Es dinámica	✔
Es encriptada	—
Valores Posibles	• Si (Predeterminado) • No

Usuario o cuenta para establecer la conexión

Usuario o cuenta con la cual se realiza el envío de correo.

Nombre	UsuarioEnvioMail
Código	MAIL_SERVER_USER
Niveles de Configuración	
• Instalación	✔
• Aplicación	—
• Unidad Organizacional	—
• Usuario	—
Es dinámica	✔
Es encriptada	—
Valor Predeterminado	your@user.com

Contraseña para autenticar el usuario

Contraseña utilizada para autenticar al usuario.

Nombre	ContraseñaEnvioMail
Código	MAIL_SERVER_PASSWORD
Niveles de Configuración	

• Instalación	✔
• Aplicación	—
• Unidad Organizacional	—
• Usuario	—
Es dinámica	✔
Es encriptada	✔
Valor Predeterminado	

—

3.10.4.1.2. Recepción de Correo

Acceso al servidor de correo entrante y cuenta utilizada para la recepción de Emails.

Propiedades Configurables

Protocolo utilizado por el monitor de correo entrante para conectarse al servidor

Nombre	ProtocoloRecepcionMail
Código	MAIL_SERVER_MONITOR_PROTOCOL
Niveles de Configuración	
• Instalación	✔
• Aplicación	—
• Unidad Organizacional	—
• Usuario	—
Es dinámica	✔
Es encriptada	—
Valores Posibles	• IMAP (Predeterminado) • POP3

IMAP - Identificación del servidor

Identificación del Servidor IMAP

Nombre	IMAPServidorRecepcionMail
Código	MAIL_SERVER_IMAP

Niveles de Configuración	
• Instalación	✔
• Aplicación	—
• Unidad Organizacional	—
• Usuario	—
Es dinámica	✔
Es encriptada	—
Valor Predeterminado	your.domain.com

IMAP - Puerto de acceso al servidor

Puerto de acceso al servidor IMAP

Nombre	IMAPPuertoRecepcionMail
Código	MAIL_SERVER_PORT_IMAP
Niveles de Configuración	
• Instalación	✔
• Aplicación	—
• Unidad Organizacional	—
• Usuario	—
Es dinámica	✔
Es encriptada	—
Valor Predeterminado	993

IMAP - Nombre de usuario de la cuenta

Nombre del usuario para establecer la conexión con el servidor IMAP

Nombre	IMAPUsuarioRecepcionMail
Código	MAIL_SERVER_USER_IMAP
Niveles de Configuración	
• Instalación	✔
• Aplicación	—
• Unidad Organizacional	—

• Usuario	—
Es dinámica	✔
Es encriptada	—
Valor Predeterminado	

IMAP - Contraseña de usuario de la cuenta

Contraseña para establecer la conexión con el servidor IMAP

Nombre	IMAPContraseñaRecepcionMail
Código	MAIL_SERVER_PASSWORD_IMAP
Niveles de Configuración	
• Instalación	✔
• Aplicación	—
• Unidad Organizacional	—
• Usuario	—
Es dinámica	✔
Es encriptada	✔
Valor Predeterminado	

POP3 - Identificación del servidor

Identificación del Servidor POP3

Nombre	POP3ServidorRecepcionMail
Código	MAIL_SERVER_POP3
Niveles de Configuración	
• Instalación	✔
• Aplicación	—
• Unidad Organizacional	—
• Usuario	—
Es dinámica	✔
Es encriptada	—
Valor Predeterminado	pop.your.domain.com

POP3 - Puerto de acceso al servidor

Puerto de acceso al servidor POP3

Nombre	POP3PuertoRecepcionMail
Código	MAIL_SERVER_PORT_POP3
Niveles de Configuración	
• Instalación	✔
• Aplicación	—
• Unidad Organizacional	—
• Usuario	—
Es dinámica	✔
Es encriptada	—
Valores Posibles	• Si • No (Predeterminado)

POP3 - Uso de protocolo SSL para autenticación del usuario

Determina si utiliza SSL para la recepción de mails (habilitar para cuentas tipo Gmail o similares).

Nombre	POP3SSLProtocol
Código	POP3_SSL_PROTOCOL
Niveles de Configuración	
• Instalación	✔
• Aplicación	—
• Unidad Organizacional	—
• Usuario	—
Es dinámica	✔
Es encriptada	—
Valor Predeterminado	

POP3 - Autenticación del usuario

Indica si debe autenticar al usuario de la cuenta del mail reader.

Nombre	POP3AutenticacionUsuario
Código	MAIL_SERVER_AUTHENTICATE_POP3
Niveles de Configuración	
• Instalación	✔
• Aplicación	—
• Unidad Organizacional	—
• Usuario	—
Es dinámica	✔
Es encriptada	—
Valor Predeterminado	

POP3 - Nombre de usuario de la cuenta

Nombre del usuario para establecer la conexión con el servidor POP3

Nombre	POP3UsuarioRecepcionMail
Código	MAIL_SERVER_USER_POP3
Niveles de Configuración	
• Instalación	✔
• Aplicación	—
• Unidad Organizacional	—
• Usuario	—
Es dinámica	✔
Es encriptada	—
Valor Predeterminado	

POP3 - Contraseña de usuario de la cuenta

Contraseña para establecer la conexión con el servidor POP3

Nombre	POP3ContraseñaRecepcionMail
Código	MAIL_SERVER_PASSWORD_POP3
Niveles de Configuración	

• Instalación	✔
• Aplicación	—
• Unidad Organizacional	—
• Usuario	—
Es dinámica	✔
Es encriptada	✔
Valor Predeterminado	

Carpeta de Mails

Indica el nombre de la carpeta que se usa como bandeja de entrada. Si no se establece valor, se utiliza la bandeja de entrada por defecto (INBOX).

Nombre	MailServerInbox
Código	MAIL_SERVER_INBOX
Niveles de Configuración	
• Instalación	✔
• Aplicación	—
• Unidad Organizacional	—
• Usuario	—
Es dinámica	✔
Es encriptada	—
Valor Predeterminado	INBOX

Eliminar mails procesados

Determina si se conservan o se eliminan de la bandeja de entrada aquellos mails procesados correctamente.

Nombre	MailCheckerDeleteFromServer
Código	MAIL_CHECKER_DELETE_FORM_SERVER
Niveles de Configuración	
• Instalación	✔
• Aplicación	—
• Unidad Organizacional	—
• Usuario	—

Es dinámica	✔
Es encriptada	—
Valor Predeterminado	• Eliminar (Predeterminado) • Conservar

3.10.4.1.3. LDAP Conexión

Configuración del acceso al servidor LDAP. Todas las propiedades son requeridas para poder activar la autenticación LDAP

Propiedades Configurables

URL de conexión con LDAP

Establece la URL de acceso al servidor LDAP.
Es requerida para poder habilitar la autenticación LDAP. Ejemplo: ldap://localhost:389/

Nombre	LDAPUrl
Código	LDAP_PROVIDER_URL
Niveles de Configuración	
• Instalación	✔
• Aplicación	—
• Unidad Organizacional	—
• Usuario	—
Es dinámica	✔
Es encriptada	—
Valor Predeterminado	

Usuario para conexión con LDAP

Identificación del usuario con el cual **Deyel** accede a LDAP para realizar las consultas.
Se indica el nombre distinguido (DN) de tal usuario. Ejemplo:
cn=admin,ou=miempresa,dc=miempresa,dc=com

Nombre	LDAPUsuario
Código	LDAP_USER
Niveles de Configuración	

• Instalación	✔
• Aplicación	—
• Unidad Organizacional	—
• Usuario	—
Es dinámica	✔
Es encriptada	—
Valor Predeterminado	

Contraseña para conexión con LDAP.

Nombre	LDAPContrasenia
Código	LDAP_PASSWORD
Niveles de Configuración	
• Instalación	✔
• Aplicación	—
• Unidad Organizacional	—
• Usuario	—
Es dinámica	✔
Es encriptada	—
Valor Predeterminado	

3.10.4.1.4. LDAP Búsqueda

Configuración de la búsqueda de usuarios dentro del Directorio LDAP.
Establece el subárbol de búsqueda, los atributos LDAP que se consideran claves de búsqueda y filtros adicionales de selección de usuarios

Propiedades Configurables

Nodo Raíz del Directorio LDAP

Establece el nodo raíz del árbol LDAP considerado por **Deyel** para la búsqueda de usuarios.
Ejemplo: ou=miempresa,dc=miempresa,dc=com

Nombre	LDAPDirectorioRaiz
Código	LDAP_ROOT_DIR
Niveles de Configuración	
• Instalación	✔
• Aplicación	—
• Unidad Organizacional	—
• Usuario	—
Es dinámica	✔
Es encriptada	—
Valor Predeterminado	

Filtro de búsqueda de usuarios

Deyel utiliza el código de usuario y/o el alias como campos predeterminados de búsqueda.
Esta propiedad, permite establecer condiciones adicionales al criterio de búsqueda.
Se puede ingresar una condición en la cual participen operadores y atributos del usuario.
El valor debe ser indicado respetando el estándar LDAP . Por ejemplo: (objectClass=inetOrgPerson)

Nombre	LDAPUserFilter
Código	LDAP_USER_FILTER
Niveles de Configuración	
• Instalación	✔
• Aplicación	—
• Unidad Organizacional	—
• Usuario	—
Es dinámica	✔
Es encriptada	—
Valor Predeterminado	

Código de Usuario

Indica el atributo LDAP cuyo valor se mapea en el código de usuario de **Deyel**.
El atributo informado debe identificar unívocamente al usuario, dentro del directorio LDAP.
Si esta propiedad no es informada, entonces el código del usuario debe ser ingresado en **Deyel**.

Nombre	LDAPCodigoUsuario
Código	LDAP_USER_FIELD_cdUsuario
Niveles de Configuración	
• Instalación	✔
• Aplicación	—
• Unidad Organizacional	—
• Usuario	—
Es dinámica	✔
Es encriptada	—
Valor Predeterminado	

Alias del Usuario

Indica el atributo LDAP cuyo valor se mapea en el Alias del usuario de **Deyel**.
El atributo informado debe identificar unívocamente al usuario dentro del directorio LDAP.
No puede haber dos usuarios con el mismo Alias.
Si esta propiedad no es informada, entonces el alias del usuario puede ser ingresado en **Deyel**.

Nombre	LDAPAliasUsuario
Código	LDAP_USER_FIELD_dsAlias
Niveles de Configuración	
• Instalación	✔
• Aplicación	—
• Unidad Organizacional	—
• Usuario	—
Es dinámica	✔
Es encriptada	—
Valor Predeterminado	

3.10.4.1.5. LDAP Sincronización

Configuración de las propiedades del usuario de **Deyel** que se sincronizan con atributos LDAP

Propiedades Configurables

Apellido

Indica el atributo LDAP cuyo valor se mapea en el apellido del usuario de **Deyel**.
Si no es informado, entonces el apellido del usuario debe ser ingresado en **Deyel**.

Nombre	LDAPApellidoUsuario
Código	LDAP_USER_FIELD_dsApellido
Niveles de Configuración	
• Instalación	✔
• Aplicación	—
• Unidad Organizacional	—
• Usuario	—
Es dinámica	✔
Es encriptada	—
Valor Predeterminado	sn

Nombre

Indica el atributo LDAP cuyo valor se mapea en el nombre del usuario de **Deyel**.
Si no es informado, entonces el nombre del usuario debe ser ingresado en **Deyel**.

Nombre	LDAPNombreUsuario
Código	LDAP_USER_FIELD_dsNombre
Niveles de Configuración	
• Instalación	✔
• Aplicación	—
• Unidad Organizacional	—
• Usuario	—
Es dinámica	✔
Es encriptada	—
Valor Predeterminado	givenName

Email

Indica el atributo LDAP cuyo valor se mapea en el Email del usuario de **Deyel**.
No puede haber dos usuarios con el mismo EMail.
Si no es informado, entonces el Email del usuario puede ser ingresado en **Deyel**.

Nombre	LDAPEmailUsuario
Código	LDAP_USER_FIELD_dsEmail
Niveles de Configuración	
• Instalación	✔
• Aplicación	—
• Unidad Organizacional	—
• Usuario	—
Es dinámica	✔
Es encriptada	—
Valor Predeterminado	mail

Estado

Indica el atributo LDAP cuyo valor se mapea en el estado del usuario de **Deyel**.
Si no es informado, entonces el estado del usuario puede ser ingresado en **Deyel**.

Nombre	LDAPEstadoUsuario
Código	LDAP_USER_FIELD_dsActivo
Niveles de Configuración	
• Instalación	✔
• Aplicación	—
• Unidad Organizacional	—
• Usuario	—
Es dinámica	✔
Es encriptada	—
Valor Predeterminado	

Teléfono

Indica el atributo LDAP cuyo valor se mapea en el teléfono del usuario de **Deyel**.
Si no es informado, entonces el teléfono del usuario puede ser ingresado en **Deyel**

Nombre	LDAPTelefonoUsuario

Código	LDAP_USER_FIELD_dsTelefono
Niveles de Configuración	
• Instalación	✔
• Aplicación	—
• Unidad Organizacional	—
• Usuario	—
Es dinámica	✔
Es encriptada	—
Valor Predeterminado	

Unidad Organizacional

Indica el atributo LDAP cuyo valor se mapea en la unidad organizacional del usuario de **Deyel**.
El valor debe ser un código de unidad existente en **Deyel**.
Si no es informado, entonces la unidad del usuario debe ser ingresada en **Deyel**.

Nombre	LDAPUnidadUsuario
Código	LDAP_USER_FIELD_cd_oRG_Unit
Niveles de Configuración	
• Instalación	✔
• Aplicación	—
• Unidad Organizacional	—
• Usuario	—
Es dinámica	✔
Es encriptada	—
Valor Predeterminado	

3.10.4.1.6. JAVA

Configuración de JAVA Virtual Machine, que utiliza **Deyel** para compilación de reglas de negocio.

Propiedades Configurables

Directorio de Instalación de JDK

JDK_HOME" permite ubicar los archivos "jar" que vienen en la instalación del JDK y que son necesarios para el "deploy" y "testing" de las reglas de negocio.

Nombre	DirectorioInstalacionJDK
Código	JDK_HOME
Niveles de Configuración	
• Instalación	✔
• Aplicación	—
• Unidad Organizacional	—
• Usuario	—
Es dinámica	✔
Es encriptada	—
Valor Predeterminado	

3.10.4.2. General

Permite configurar propiedades que no pertenecen a un módulo en particular, sino que afectan el uso general del producto.

Temas

Herramientas
- Estructura Organizacional
- Tareas Programadas
- Logs

3.10.4.2.1. Herramientas

Utilización de Herramientas.

Propiedades Configurables

Habilitar el registro de casos ejecutados durante las pruebas automatizadas

Cuando se ejecutan los casos de prueba automatizados, esta propiedad determina si se debe resguardar la información de cada caso ejecutado, en el formulario de Ejecución de Casos de Usuario (UEC).

Nombre	HabilitarRegistroEjecucionDeCasos
Código	ENABLE_STRESS

Niveles de Configuración	
• Instalación	✔
• Aplicación	—
• Unidad Organizacional	—
• Usuario	—
Es dinámica	✔
Es encriptada	—
Valores Posibles	• Si • No (Predeterminado)

Habilitar el registro de objetos importados

Habilitando esta propiedad, cada operación de import que se realiza queda registrada y puede visualizarse en detalle la información incorporada al sistema, el origen de la misma, etc.

Nombre	HabilitarRegistroObjetosImportados
Código	REGISTER_IMPORTS
Niveles de Configuración	
• Instalación	✔
• Aplicación	—
• Unidad Organizacional	—
• Usuario	—
Es dinámica	✔
Es encriptada	—
Valores Posibles	• Si • No (Predeterminado)

Habilitar el registro de objetos exportados

Habilitando esta propiedad, cada operación de export que se realiza queda registrada en el formulario de Registro de Export y puede visualizarse en detalle.

Nombre	HabilitarRegistroObjetosExportados
Código	REGISTER_EXPORTS
Niveles de Configuración	
• Instalación	✔

• Aplicación	—
• Unidad Organizacional	—
• Usuario	—
Es dinámica	✔
Es encriptada	—
Valores Posibles	• Si • No (Predeterminado)

3.10.4.2.2. Notificaciones

Configuración de las notificaciones automáticas emitidas por el motor de workflow

Propiedades Configurables

Notificación de Errores - Emisor

Cuando se produce algún error de ejecución, el sistema presenta al usuario la opción de reportar ese error al administrador del sistema. Se enviará en tal caso un correo, con los detalles del error ocurrido. El emisor de dicho correo será el indicado en esta propiedad.

Nombre	NotificacionErroresEmisor
Código	ERROR_REPORT_EMAIL_FROM
Niveles de Configuración	
• Instalación	✔
• Aplicación	—
• Unidad Organizacional	—
• Usuario	—
Es dinámica	✔
Es encriptada	—
Valor Predeterminado	

Notificación de Errores - Destinatario

Cuando se produce algún error de ejecución, el sistema presenta al usuario la opción de reportar ese error al administrador del sistema. Se enviará en tal caso un correo, con los detalles del error ocurrido. El destinatario de dicho correo será el indicado en esta propiedad.

Nombre	NotificacionErroresDestinatario
Código	ERROR_REPORT_EMAIL_TO
Niveles de Configuración	
• Instalación	✔
• Aplicación	—
• Unidad Organizacional	—
• Usuario	—
Es dinámica	✔
Es encriptada	—
Valor Predeterminado	

3.10.4.2.3. Organización

Administración de la estructura organizacional

Propiedades Configurables

Tipo de codificación de unidades organizacionales

Establece el mecanismo que se utiliza para generar el código de las unidades organizacionales.

Nombre	TipoCodificacionUnidades
Código	AUTOMATIC_ORG_UNIT_ID
Niveles de Configuración	
• Instalación	✔
• Aplicación	—
• Unidad Organizacional	—
• Usuario	—
Es dinámica	✔
Es encriptada	—
Valores Posibles	• Manual • Semi Automático • Automático (Predeterminado)

- Manual

El código de unidad es visible en la interfaz y debe ser informado por el usuario.

- Semi Automático

El código de unidad es visible en la interfaz y puede ser informado por el usuario.
Si no es informado se genera un código de manera automática.

- Automático

El código de unidad no es visualizado en la interfaz y es asignado por el Deyel.

3.10.4.2.4. Tareas Programadas

Estas propiedades definen el inicio automático de las tareas programadas y parámetros considerados por tareas programadas especificas del sistema.

Propiedades Configurables

Inicio automático de las tareas programadas

Define si al iniciar la ejecución de **Deyel** comienza automáticamente la ejecución de tareas programadas.
Cuando se indica No, las tareas programadas deben iniciarse manualmente, accediendo al monitor de tareas programadas.

Nombre	InicioAutomaticoTareasProgramadas
Código	TASK_SCHEDULER
Niveles de Configuración	
• Instalación	✔
• Aplicación	—
• Unidad Organizacional	—
• Usuario	—
Es dinámica	✔
Es encriptada	—
Valores Posibles	• Si (Predeterminado) • No

Periodo de activación del depurador de archivos en deshuso

Intervalo de activación (expresado en segundos) del proceso depurador de archivos en deshuso.
Este proceso detecta y elimina archivos que han sido desvinculados del caso o del formulario y ya no son referenciados.
Esta es una de las tareas programadas del sistema, cuya periodicidad de ejecución se define con esta propiedad.

Nombre	PeriodoActivacionDepuradorArchivos
Código	CLEAN_UPLOADED_FILES_PROCESS_PERIODICITY
Niveles de Configuración	
• Instalación	✔
• Aplicación	—
• Unidad Organizacional	—
• Usuario	—
Es dinámica	✔
Es encriptada	—
Valor Predeterminado	18.000 (5 horas)

Tiempo mínimo de resguardo considerado por el depurador de archivos en deshuso

Tiempo mínimo (expresado en segundos) durante el cual un archivo subido a **Deyel** no puede ser eliminado.
Aunque el proceso depurador de archivos detecte que el archivo no tiene relación con ningún caso ni formulario, no será eliminado hasta que no haya transcurrido el tiempo indicado en esta propiedad.

Nombre	TiempoMinimoResguardo
Código	UPLOADED_FILES_MIN_TIME
Niveles de Configuración	
• Instalación	✔
• Aplicación	—
• Unidad Organizacional	—
• Usuario	—
Es dinámica	✔
Es encriptada	—
Valor Predeterminado	3.600 (1 hora)

3.10.4.2.5. Logs

Configura la generación y depuración automática de archivos de logs.

Propiedades Configurables

Tiempo máximo de ejecución para consultas SQL

Define el tiempo máximo en milisegundos que se considera admisible para una consulta SQL.
Si la ejecución de una consulta supera dicho tiempo, se graba la sentencia ejecutada en un archivo de log.
Si el valor definido es 0 no se controla el tiempo de las consultas.

Nombre	SQLTiempoMáximoConsulta
Código	LOG_SQL_MAX_TIME
Niveles de Configuración	
• Instalación	✔
• Aplicación	—
• Unidad Organizacional	—
• Usuario	—
Es dinámica	✔
Es encriptada	—
Valor Predeterminado	100

Internacionalización - Registrar las claves no encontradas en los archivos de traducción.

Esta propiedad permite activar un log de textos no registrados en el archivo de traducción que se está usando en la instalación.

Nombre	BTRBLogNotFound
Código	BTRBLogNotFound
Niveles de Configuración	
• Instalación	✔
• Aplicación	—
• Unidad Organizacional	—
• Usuario	—
Es dinámica	✔
Es encriptada	—
Valor Predeterminado	• Sin Log (Predeterminado) • Standard • LogNotFound

Puede tomar los siguientes valores:

Sin Log: No registra las traducciones no encontradas;
Standard: Registra las traducciones no encontradas en el archivo SystemOut_YYYY-MM-DD.log;
LogNotFound: Registra las traducciones no encontradas en el archivo LogNotFound_YYYY-MM-DD.log.

Estos archivos pueden ser recuperados con la función de descarga de logs.

Tipos de logs habilitados

Establece los tipos de logs que son informados en el archivo de logs.
Para ambientes productivos se recomienda utilizar la configuración por defecto.

Nombre	LogsTiposHabilitados
Código	ENABLED_LOG_TYPES
Niveles de Configuración	
• Instalación	✔
• Aplicación	—
• Unidad Organizacional	—
• Usuario	—
Es dinámica	✔
Es encriptada	—
Valores Posibles	• ERROR (Predeterminado) • SERVLET (Predeterminado) • SERVLET_TIME (Predeterminado) • RULE_TIME (Predeterminado) • SQL • SQL_TIME (Predeterminado) • CORRECT_PROCESS (Predeterminado)

Los valores se ingresan separados por ";".
Por ejemplo, el valor predeterminado es el siguiente:
"ERROR;SERVLET;SERVLET_TIME;RULE_TIME;SQL_TIME;;CORRECT_PROCESS"

- ERROR

Reporta errores de ejecución.

- SERVLET

Reporta los parámetros recibidos por todas las servlet.

Debería estar activado solamente cuando se esta analizando una situación de error, para visualizar los datos transferidos entre las servlets.

- SERVLET_TIME

Reporta los tiempos de ejecución de las servlet que superan el umbral definido.

- RULE_TIME

Reporta todas las reglas cuyo tiempo de ejecución supera el umbral establecido en la propiedad LOG_LOGIC_MAX_TIME.

- SQL

Reporta las sentencias SQL junto al tiempo que transcurre al ejecutarse.

- SQL_TIME

Reporta todas las sentencias SQL cuyo tiempo de ejecución supera el umbral establecido en la propiedad LOG_SQL_MAX_TIME

- CORRECT_PROCESS

Reporta los resultados de la ejecución del proceso corrector del pool de sesiones.

Directorio de registro de logs

Directorio en el cual se generan los archivos de logs.
Si no se informa valor, se utiliza el directorio **/logs**, dentro del contexto.

> *Solamente se puede configurar en instalaciones On-Premise*

Nombre	DirectorioRegistroLogs
Código	LOG_DIRECTORY
Niveles de Configuración	
• Instalación	✔
• Aplicación	—
• Unidad Organizacional	—
• Usuario	—
Es dinámica	—
Es encriptada	—
Valor Predeterminado	

Persistencia de los archivos de logs

Los archivos de logs generados son eliminados automáticamente por el proceso depurador de logs del sistema. Esta propiedad establece la cantidad de días que deben persistir, para poder eliminarlos.

Si el valor es cero nunca se eliminarán archivos.

Nombre	LogsPersistenciaArchivos
Código	LOG_FILES_DAYS_OLD
Niveles de Configuración	
• Instalación	✔
• Aplicación	—
• Unidad Organizacional	—
• Usuario	—
Es dinámica	✔
Es encriptada	—
Valor Predeterminado	15

Tiempo máximo de ejecución de las reglas de negocio

Define un tiempo, expresado en milisegundos. Cuando la ejecución de una regla de negocios supera dicho umbral, se guardan los parámetros de entrada y salida en un archivo de log.
Si el valor definido es 0 no se controla el tiempo de ejecución de las reglas.

Nombre	ReglasTiempoMaximoEjecucion
Código	LOG_LOGIC_MAX_TIME
Niveles de Configuración	
• Instalación	✔
• Aplicación	—
• Unidad Organizacional	—
• Usuario	—
Es dinámica	✔
Es encriptada	—
Valor Predeterminado	400

Depuración de la consola de logs

Establece la cantidad de días, durante los cuales se mantiene la información en la consola de logs.
La tarea programada de depuración de logs antiguos del sistema eliminará aquellos registros que sean anteriores a la cantidad de días indicada.

Nombre	LogsDepuracionConsola

Código	LOG_DATA_DAYS_OLD
Niveles de Configuración	
• Instalación	✔
• Aplicación	—
• Unidad Organizacional	—
• Usuario	—
Es dinámica	✔
Es encriptada	—
Valor Predeterminado	15

3.10.4.3. Motor de Workflow

Permite configurar propiedades que definen el comportamiento del Motor de Workflow.

Temas

- Notificaciones

3.10.4.3.1. Notificaciones

Configuración de las notificaciones automáticas emitidas por el motor de workflow.

Propiedades Configurables

Notificación de acciones automáticas canceladas

Indica las direcciones de correo a las que se enviará un mail por cada acción automática cancelada (separadas por ';').

Nombre	NotificacionAccionesAutomáticas Canceladas
Código	ADDRESSES_FOR_CANCELED_ACTIVITY
Niveles de Configuración	
• Instalación	✔
• Aplicación	—
• Unidad Organizacional	—

• Usuario	—
Es dinámica	✔
Es encriptada	—
Valor Predeterminado	

Valor de la función emailWorkflow()

Al definir acciones automáticas de Email o alertas, puede usarse la función emailWorkflow() para modelar el emisor de la notificación. Cuando se recibe el correo, se visualiza como Emisor el valor modelado, en lugar de visualizar el emisor real, que es el configurado en la propiedad UsuarioEnvioMail. Puede ocurrir que algunos servidores de correo, como Google por ejemplo, no permitan visualizar un valor diferente del emisor real.

Nombre	EmailWorkflow
Código	WORKFLOW_MAIL
Niveles de Configuración	
• Instalación	✔
• Aplicación	—
• Unidad Organizacional	—
• Usuario	—
Es dinámica	✔
Es encriptada	—
Valor Predeterminado	deyel@deyel.com

Valor de la función nombreMotorWorkflow()

La función nombreMotorWorkflow() retorna un valor alfanumérico que indica el nombre del motor de Workflow. Se utiliza habitualmente en el modelado de notificaciones. El valor predeterminado es Deyel Workflow Engine, pero puede configurarse un valor diferente si fuese necesario.

Nombre	NombreMotorWorkflow
Código	WORKFLOW_NAME
Niveles de Configuración	
• Instalación	✔
• Aplicación	—
• Unidad Organizacional	—
• Usuario	—

Es dinámica	✔
Es encriptada	—
Valor Predeterminado	Deyel Workflow Engine

3.10.4.4. Portal de Usuarios

Permite configurar propiedades que definen la apariencia y el comportamiento general del portal de usuarios.

Temas

- General
- Sesión de Trabajo
- Grilla de Casos
- Grilla de Tareas
- Internacionalización

3.10.4.4.1. General

Apariencia general y condiciones de uso del portal de usuarios.

Propiedades Configurables

Nombre del portal de usuarios

Texto visualizado en las pestañas del navegador, precediendo los datos del Usuario. Por ejemplo, Deyel o Portal Deyel. Permite identificar rápidamente las pestañas del navegador que gestan accediendo al portal.

Nombre	NombrePortal
Código	SYSTEM_NAME
Niveles de Configuración	
• Instalación	✔
• Aplicación	—
• Unidad Organizacional	—
• Usuario	—
Es dinámica	✔
Es encriptada	—

Valor Predeterminado	Portal Usuarios

Tamaño máximo para archivos adjuntos

Cuando el usuario adjunta archivos al sistema, ya sea como campos de un formulario o como documentos adjuntos a un caso, se controla que el tamaño de los mismos no supere el limite establecido en esta propiedad.
Se indica un número, expresado en MB. El máximo valor admisible es 20 MB

Nombre	UploadDocumentosTamañoMaximo
Código	MAX_UPLOAD_SIZE
Niveles de Configuración	
• Instalación	✔
• Aplicación	—
• Unidad Organizacional	—
• Usuario	—
Es dinámica	✔
Es encriptada	—
Valor Predeterminado	5

Cantidad de líneas por página en la búsqueda de formularios

Establece la cantidad de líneas por página cuando se consultan instancias de un formulario.

Nombre	BusquedaFormulariosCantidadLineas-PorPagina
Código	DEFAULT_PAGE_SIZE
Niveles de Configuración	
• Instalación	✔
• Aplicación	—
• Unidad Organizacional	—
• Usuario	—
Es dinámica	✔
Es encriptada	—
Valor Predeterminado	13

Tema

Define el tema utilizado por el portal de usuarios.

Nombre	Tema
Código	THEME_DIR
Niveles de Configuración	
• Instalación	✔
• Aplicación	—
• Unidad Organizacional	—
• Usuario	—
Es dinámica	✔
Es encriptada	—
Valores Posibles	• Azul (Predeterminado) • Verde

Orden predeterminado para la grilla de formularios

Permite definir el orden predeterminado para la grilla de formularios.

Nombre	Orden predeterminado para la grilla de formularios
Código	BROWSE_FORM_ORDER
Niveles de Configuración	
• Instalación	✔
• Aplicación	—
• Unidad Organizacional	—
• Usuario	—
Es dinámica	✔
Es encriptada	—
Valores Posibles	• Ascendente • Descendente (Predeterminado)

Campo de orden predeterminado para la grilla de formularios

Permite definir el campo usado para el orden predeterminado de la grilla de formularios.

Nombre	Campo de orden predeterminado para la grilla de formularios
Código	BROWSE_FORMS_ORDER_FIELD
Niveles de Configuración	
• Instalación	✔
• Aplicación	—
• Unidad Organizacional	—
• Usuario	—
Es dinámica	✔
Es encriptada	—
Valores Posibles	• Fecha de Creación (Predeterminado) • Ultima modificación • Identificador del formulario

3.10.4.4.2. Sesión de Trabajo

Características de la sesión de trabajo.

Propiedades Configurables

Tipo de Autenticación

Define el método de autenticación de usuarios utilizado por Deyel.

Nombre	TipoAutenticacion
Código	TP_AUTHENTICATION_LOGIN
Niveles de Configuración	
• Instalación	✔
• Aplicación	—
• Unidad Organizacional	—
• Usuario	—
Es dinámica	✔
Es encriptada	—
Valores Posibles	• Nativa (Predeterminado) • LDAP • Google

	• Personalizada • Mixta

Autenticación Nativa
El usuario informa su código de usuario o su alias y su contraseña, en la página de acceso al portal de usuarios.
Deyel verifica que exista un usuario en su registro, con el código de usuario o alias informados y verifica la contraseña, que se almacena encriptada en la base de datos de **Deyel**.
Si el usuario o la contraseña son incorrectos, se deniega el uso del producto.

Autenticación LDAP
Cuando una organización utiliza servicios de directorio para registrar a sus usuarios, puede configurarse **Deyel**, para delegar en LDAP el proceso de autenticación.
El usuario informa su código de usuario o su alias y su contraseña, en la página de acceso al portal de usuarios.
Deyel verifica que exista un usuario en su registro, con el código de usuario o alias informados y luego delega en LDAP la autenticación..
Sólo cuando el usuario está registrado en **Deyel** y LDAP reporta una correcta autenticación se permite utilizar el producto.

Autenticación Federada
Deyel implementa este tipo de autenticación integrándose con **Google**.
Cuando una organización utiliza cuentas de Google para sus usuarios, se puede delegar en Google el proceso de autenticación.
La página de acceso al portal de usuarios, redirige a los usuarios al inicio de sesión de Google.
El usuario ingresa sus credenciales en Google. La clave de acceso no es informada nunca en **Deyel**.
Cuando el usuario ingresa a su cuenta, Google informa a **Deyel** que el usuario se ha autenticado exitosamente.
Deyel verifica que exista un usuario en su registro, cuya cuenta de correo sea la cuenta de Google indicada.
Sólo cuando el usuario se autentica correctamente en Google y está registrado en **Deyel** se permite utilizar el producto.

Autenticación Personalizada
El usuario informa su código de usuario o su alias y su contraseña, en la página de acceso al portal de usuarios.
Deyel invoca a esta regla de negocios, pasándole los datos ingresados por el usuario.
La regla de negocios implementa la lógica de autenticación y retorna si la misma es correcta o no.
Cuando **Deyel** recibe que la autenticación es correcta, se verifica que el usuario exista en **Deyel** y se permite el ingreso.
Las contraseñas NO persisten en Deyel.

Autenticación Mixta
Permite que una organización pueda utilizar simultáneamente diferentes tipos de autenticación. Se establecen cuáles son los tipos de autenticación habilitados y en qué orden deben aplicarse

Validaciones

En la configuración del ambiente, se ejecutan validaciones especificas para cada mecanismo y validaciones conjuntas cuando se utiliza un esquema Mixto.

De este modo el administrador se asegura que la autenticación queda correctamente configurada y en funcionamiento.

- Para habilitar la autenticación LDAP, se verifica que funcione correctamente la conexión con LDAP y se realiza la búsqueda de usuarios dentro del directorio LDAP.
- Para habilitar la autenticación Google, se verifica que esté correctamente configurada la conexión con Google.
- Para habilitar la autenticación Personalizada, se verifica que exista la regla de negocios que controla la autenticación.
- Para habilitar la autenticación Mixta se deben indicar cuáles métodos se requieren y en qué orden se aplican.

Permitir múltiples sesiones de usuario con el mismo navegador

Cuando se habilita esta propiedad, es posible mantener múltiples sesiones de trabajo, con usuarios diferentes, dentro del mismo navegador.

Nombre	PermitirMultiplesSesiones
Código	ALLOW_MULTISESSION
Niveles de Configuración	
• Instalación	✔
• Aplicación	—
• Unidad Organizacional	—
• Usuario	—
Es dinámica	✔
Es encriptada	—
Valores Posibles	• Si • No (Predeterminada)

Autenticación Personalizada

Nombre de la regla definida para ejecutar la Autenticación Personalizada.

Nombre	CustomAuthentication
Código	CUSTOM_AUTHENTICATION
Niveles de Configuración	
• Instalación	✔
• Aplicación	—
• Unidad Organizacional	—
• Usuario	—

Es dinámica	✔
Es encriptada	—
Valores Posibles	

Se valida que la regla exista al configurar el tipo de autenticación personalizado o mixta que lo contenga.
Si la regla existe se valida que tenga los parámetros en entrada y salida necesarios.

Métodos de Autenticación Mixta

Estas propiedades se utilizan para configurar el mecanismo de autenticación mixta.
Establecen cuales son los métodos de autenticación habilitados y el orden en que deben utilizarse.

Son 3 propiedades (Primer Método de Autenticación, Segundo Método de Autenticación, Tercer Método de Autenticación) que funcionan de manera similar.

Nombre	AuthenticacionMixta1 AuthenticacionMixta2 AuthenticacionMixta3
Código	AUTHENTICACION_MIXTA_1 AUTHENTICACION_MIXTA_2 AUTHENTICACION_MIXTA_3
Niveles de Configuración	
• Instalación	✔
• Aplicación	—
• Unidad Organizacional	—
• Usuario	—
Es dinámica	✔
Es encriptada	—
Valores Posibles	• Nativa • LDAP • Google

Cuando se configura el uso de autenticación Mixta, se validar que:

- Al menos una de estas propiedades esté indicada.
- Que no existan repeticiones en los valores establecidos.

Si alguna propiedad asume el valor LDAP, se validan todas las propiedades relacionadas con LDAP

LDAP - Conexión con el Servidor
LDAP - Búsqueda de Usuarios
LDAP - Sincronización de Atributos

Si alguna propiedad asume el valor Google, se validan todas las propiedades relacionadas con Google.
Si alguna propiedad asume el valor Personalizada, se validan todas las propiedades relacionadas con la autenticación personalizada.

Solamente se puede aplicar la configuración cuando todas las validaciones son correctas.

3.10.4.4.3. Grilla de Casos

Se presenta un conjunto de propiedades que permiten establecer las columnas que se visualizan inicialmente en la grilla de casos.

Todas ellas tienen las siguientes características generales.

Niveles de Configuración	
• Instalación	✔
• Aplicación	—
• Unidad Organizacional	—
• Usuario	✔
Es dinámica	✔
Es encriptada	—
Valor Predeterminado	• Si (Predeterminado) • No

Propiedades Configurables

Nombre del proceso

Nombre	GrillaCasoProceso
Código	SHOW_PROCESS_CASES

Descripción del caso

Nombre	GrillaCasoDescripcion
Código	SHOW_DESCRIPTION_CASES

Chat

Visualizar acceso a la conversación asociada al caso.

Nombre	GrillaCasoChat
Código	SHOW_CHAT_CASES

Nombre de la actividad actual

Nombre	GrillaCasoActividad
Código	SHOW_ACTIVITY_CASES

Fecha de inicio del caso

Nombre	GrillaCasoInicio
Código	SHOW_BEGIN_CASES

Responsable de la actividad actual

Participante responsable por la ejecución de la actividad actual del caso.

Nombre	GrillaCasoResponsable
Código	SHOW_RESPONSIBLE_CASES

Vencimiento de la actividad actual.

Nombre	GrillaCasoVencimiento
Código	SHOW_EXPIRED_CASES

Prioridad del caso

Nombre	GrillaCasoPrioridad
Código	SHOW_PRIORITY_CASES

Finalización del caso

Fecha de finalización del caso.

Nombre	GrillaCasoFinalizacion
Código	SHOW_ENDED_CASES

Inicio de la actividad actual

Nombre	GrillaCasoInicioActividad
Código	SHOW_BEGINACT_CASES

Estado del caso

Nombre	GrillaCasoEstado
Código	SHOW_STATE_CASES

Identificación del caso

Nombre	GrillaCasoCaso
Código	SHOW_CASE_CASES

3.10.4.4.4. Grilla de Tareas

Se presenta un conjunto de propiedades que permiten establecer las columnas que se visualizan inicialmente en la grilla de tareas.

Todas ellas tienen las siguientes características generales.

Niveles de Configuración	
• Instalación	✔
• Aplicación	—
• Unidad Organizacional	—
• Usuario	✔
Es dinámica	✔
Es encriptada	—
Valor Predeterminado	• Si (Predeterminado) • No

Propiedades Configurables

Identificación del caso

Nombre	GrillaTareasCaso
Código	SHOW_CASE_TODOLIST

Nombre del proceso

Nombre	GrillaTareasProceso
Código	SHOW_PROCESS_TODOLIST

Descripción del caso

Nombre	GrillaTareasDescripcionCaso
Código	SHOW_DESCRIPTION_TODOLIST

Chat

Visualizar acceso a la conversación asociada al caso.

Nombre	GrillaTareasChat
Código	SHOW_CHAT_TODOLIST

Nombre de la tarea

Nombre	GrillaTareasNombre
Código	SHOW_ACTIVITY_TODOLIST

Inicio de la tarea

Nombre	GrillaTareasInicio
Código	SHOW_BEGIN_TODOLIST

Responsable de la tarea

Nombre	GrillaTareasResponsable
Código	SHOW_RESPONSIBLE_TODOLIST

Vencimiento de la tarea

Nombre	GrillaTareasVencimiento
Código	SHOW_EXPIRED_TODOLIST

Prioridad de la tarea

Nombre	GrillaTareasPrioridad
Código	SHOW_PRIORITY_TODOLIST

3.10.4.4.5. Internacionalización

Configure el lenguaje y formatos de fechas o campos numéricos que le resulten más adecuados.

Propiedades Configurables

Lenguaje

Indica el lenguaje o idioma para la utilización del producto.
En función de esta propiedad, Deyel presenta la interfase de usuario en el idioma correspondiente.

Nombre	Lenguaje
Código	BTRB.Locale.language
Niveles de Configuración	
• Instalación	✔
• Aplicación	—
• Unidad Organizacional	—
• Usuario	—
Es dinámica	✔
Es encriptada	—
Valores Posibles	• Español (Predeterminado) • Inglés • Portugués

País

Indica el país a considerar para las traducciones.
Deyel considera que pueden existir variantes del idioma o lenguaje según el país en que se utiliza el producto.
Se realizan traducciones específicas para el lenguaje y país configurados.

Nombre	Pais
Código	BTRB.Locale.country
Niveles de Configuración	
• Instalación	✔
• Aplicación	—
• Unidad Organizacional	—
• Usuario	—

Es dinámica	✔
Es encriptada	—
Valores Posibles	• Argentina (Predeterminado) • Estados Unidos • Brasil

Validaciones
Cada texto que el Deyel visualiza en la interfaz de usuario, ya sea el texto visualizado en un menú, o en un botón, o en el nombre de un campo, o un mensaje de error, etc. es traducido, antes de presentarlo al usuario.
Esta traducción es automática y utiliza los archivos de traducción incluidos en el producto.
Al configurar estas propiedades se verifica que la combinación de lenguaje y país sea valida para determinar el archivo de traducciones a utilizar.

Símbolo Decimal - Edición de valores numéricos

Indica el símbolo que separa la parte entera de la parte decimal para formato de ingreso. Por ejemplo: 10,99.
Esta propiedad debe estar informada para que se puedan ingresar correctamente los valores numéricos.
El valor del campo puede ser ingresado sin digitar el separador de parte entera, pero en caso de tipearse algún separador, debe ser el que se especifica en esta propiedad.

Nombre	SimboloDecimalEdicion
Código	DECIMAL_CHARACTER_SEPARATOR_IN
Niveles de Configuración	
• Instalación	✔
• Aplicación	✔
• Unidad Organizacional	—
• Usuario	✔
Es dinámica	✔
Es encriptada	—
Valores Posibles	, (Predeterminado) .

Símbolo de Miles - Edición de valores numéricos

Indica el símbolo que separa grupos de miles (1.000, 1.000.000, etcétera) para formato de ingreso. Por ejemplo: 999.999.
El valor del campo puede ser ingresado sin digitar separador de miles, pero en caso de tipearse algún separador, debe ser el que se especifica en esta propiedad.

Nombre	SimboloMilesEdicion
Código	THOUSANDS_CHARACTER_SEPARATOR_IN
Niveles de Configuración	
• Instalación	✔
• Aplicación	✔
• Unidad Organizacional	—
• Usuario	✔
Es dinámica	✔
Es encriptada	—
Valores Posibles	, . (Predeterminado)

Símbolo Decimal - Visualización de valores numéricos

Indica el símbolo que separa la parte entera de la parte decimal para formato de salida. Por ejemplo: 10,99.
Esta propiedad debe estar informada para que se puedan visualizar correctamente los valores numéricos.

Nombre	SimboloDecimalVisualizacion
Código	DECIMAL_CHARACTER_SEPARATOR_OUT
Niveles de Configuración	
• Instalación	✔
• Aplicación	✔
• Unidad Organizacional	—
• Usuario	✔
Es dinámica	✔
Es encriptada	—
Valores Posibles	, (Predeterminado) .

Símbolo de Miles - Visualización de valores numéricos

Indica el símbolo que separa grupos de miles (1.000, 1.000.000, etcétera) para formato de salida. Por ejemplo: 999.999.
Esta propiedad debe estar informada para que se puedan visualizar correctamente los valores numéricos.

Nombre	SimboloMilesVisualizacion
Código	THOUSANDS_CHARACTER_SEPARATOR_OUT
Niveles de Configuración	
• Instalación	✔
• Aplicación	✔
• Unidad Organizacional	—
• Usuario	✔
Es dinámica	✔
Es encriptada	—
Valores Posibles	, . (Predeterminado)

3.10.4.5. Modelador

Permite configurar propiedades que definen comportamientos en el modelador de **Deyel.**

Temas

- Modelador de Procesos

3.10.4.5.1. Modelador de Procesos

Preferencias de uso para el Modelador de Procesos.

Propiedades Configurables

Mostrar los identificadores de los símbolos

Activando esta propiedad se visualiza, junto al nombre de cada símbolo, su identificador único.

Nombre	MostrarIdentificadores
Código	SHOW_IDS
Niveles de Configuración	
• Instalación	✔

• Aplicación	—
• Unidad Organizacional	—
• Usuario	✔
Es dinámica	✔
Es encriptada	—
Valores Posibles	• Si • No (Predeterminado)

Tamaño de la grilla

Cantidad de pixeles utilizada para mover los símbolos en el diagrama.

Nombre	TamañoGrilla
Código	GRID_SIZE
Niveles de Configuración	
• Instalación	✔
• Aplicación	—
• Unidad Organizacional	—
• Usuario	✔
Es dinámica	✔
Es encriptada	—
Valores Posibles	• 1 (Predeterminado) • 10 • 20

Visualizar Minimap

Visualiza el minimap en la parte inferior derecha del área de modelado.

Nombre	VisualizarMinimap
Código	SHOW_MINIMAP
Niveles de Configuración	
• Instalación	✔
• Aplicación	—
• Unidad Organizacional	—
• Usuario	✔

Es dinámica	✔
Es encriptada	—
Valores Posibles	• Si (Predeterminado) • No

Visualizar validaciones sobre las figuras

Visualiza el signo de admiración sobre las figuras que tengan errores de validación.

Nombre	ValidacionesSobreFiguras
Código	INLINE_VALIDATIONS
Niveles de Configuración	
• Instalación	✔
• Aplicación	—
• Unidad Organizacional	—
• Usuario	✔
Es dinámica	✔
Es encriptada	—
Valores Posibles	• Si (Predeterminado) • No

3.10.4.6. Tedis

Permite configurar propiedades que definen la apariencia y comportamiento de la Red Social Empresarial - Tedis.

Temas

- General
- Notificaciones

3.10.4.6.1. General

Apariencia general de las conversaciones.

Propiedades Configurables

Tamaño de visualización de conversaciones

Tamaño inicial de las conversaciones. Cada una puede visualizarse inicialmente en una ventana reducida o expandida.

Nombre	TamañoConversaciones
Código	INITIAL_CONV_SIZE
Niveles de Configuración	
• Instalación	✔
• Aplicación	—
• Unidad Organizacional	—
• Usuario	—
Es dinámica	✔
Es encriptada	—
Valores Posibles	• Reducida (Predeterminado) • Expandida

3.10.4.6.2. Notificaciones

Emisión de notificaciones automáticas.

Propiedades Configurables

Notificación de nuevos mensajes

Recibir una notificación cuando se recibe un nuevo mensaje.

Nombre	NotificacionNuevosMensajes
Código	ALERT_ON_NEW_MESSAGE
Niveles de Configuración	
• Instalación	✔
• Aplicación	—
• Unidad Organizacional	—
• Usuario	✔
Es dinámica	✔
Es encriptada	—

Valores Posibles	• Si (Predeterminado) • No

Notificación de usuarios conectados

Recibir una notificación cuando un usuario se conecta al portal.

Nombre	NotificacionUsuariosConectados
Código	ALERT_USER_ONLINE
Niveles de Configuración	
• Instalación	✔
• Aplicación	—
• Unidad Organizacional	—
• Usuario	✔
Es dinámica	✔
Es encriptada	—
Valores Posibles	• Si (Predeterminado) • No

Descripción	Di-nám.	Inst	Aplic	Usuario	Valores Posibles	Valor Predefinido
Notificación de nuevos mensajes (ALERT_ON_NEW_MESSAGE) Recibir una notificación cuando se recibe un nuevo mensaje.	SI	X		X	Si/No	Si
Notificación de usuarios conectados (ALERT_USER_ONLINE) Recibir una notificación cuando un usuario se conecta.	SI	X		X	Si/No	Si

3.10.4.7. CRM

Permite configurar propiedades que definen comportamientos específicos y preferencias de uso de la solución **CRM**.

Temas

- Presupuestos

3.10.4.7.1. Presupuestos

Configuración del formulario Presupuesto.

Propiedades Configurables

Numeración de presupuestos

Permite elegir la forma en que se genera la numeración de los presupuestos

Nombre	NumeracionPresupuesto
Código	QUOTE_NUMBER
Niveles de Configuración	
• Instalación	✔
• Aplicación	—
• Unidad Organizacional	—
• Usuario	✔
Es dinámica	—
Es encriptada	—
Valores Posibles	• Manual (Predeterminado) • Automática

- Manual
Debe ser informado por el usuario.
- Automática
La numeración sera autoincremental.

3.10.5. Consolas

Para acceder al menú de Administración de Consolas se debe hacer clic sobre la opción "**Consolas**" en la solapa "**Configuración**" del menú principal

Otra alternativa para acceder es utilizar la búsqueda rápida del menú.
Ingresando las frases "Tareas Programadas", "Consola de eventos", "Consola de logs", "Procesamiento de alerta (Formulario ALR)" o "Monitor de cachés", se podrán visualizar las distintas opciones.

Tareas Programadas
Permite acceder al monitor de tareas programadas.

Consola de Eventos
Permite ver una lista con todos los Eventos generados por actividades en **Deyel**.

Consola de Logs

Permite ver una lista con todos los Logs generados al ejecutarse distintas funciones del sistema.

Procesamiento de Alertas
Permite ver una lista con las alertas procesadas por la tarea programada de emisión de alertas de las actividades.

Monitor de Cachés
Invoca al Monitor de Cachés, que permite consultar la cantidad de objetos de cada tipo que están alocados en el área de caché de memoria del servidor, o solicitar la actualización de este área de memoria, releyendo la información desde la base de datos.

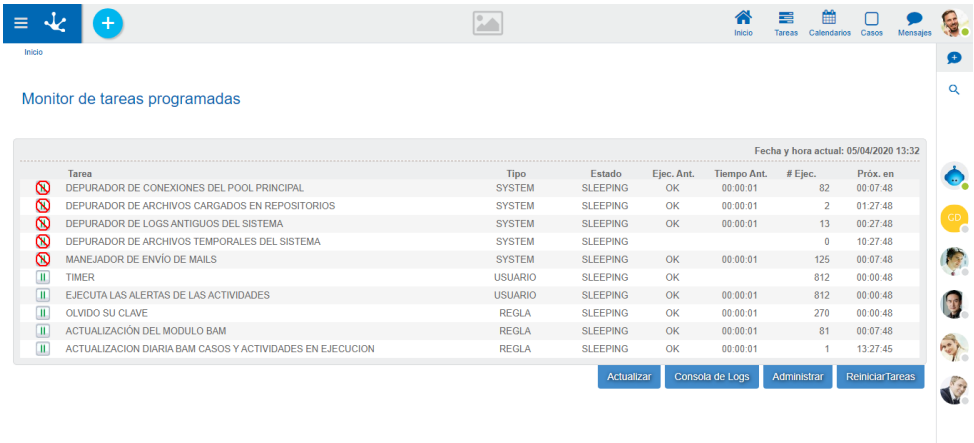
3.10.5.1. Monitoreo de Tareas Programadas

Esta función tiene por objetivo visualizar el estado de ejecución de las tareas programadas activas en el sistema.

Para acceder a esta función existen distintas alternativas:

- Hacer clic sobre la opción "**Configuración**" del menú principal y dentro del sub menú "**Consolas**", hacer clic en la opción "**Tareas Programadas**".
- En la búsqueda del menú ingresar "**Tareas Programadas**" y hacer clic en la opción listada.

Al ejecutar esta función se muestra el estado de ejecución de las tareas programadas:



En esta pantalla se puede ver de manera tabular cada una de las tareas activas en el sistema, el tipo de tarea, su estado, resultado de la ultima ejecución (OK o ERROR), tiempo que tardó en la última ejecución, cantidad de ejecuciones desde el último reinicio de las tareas programadas y finalmente el tiempo restante para la próxima ejecución en HH:MM:SS.

Debajo de la tabla se ven cuatro botones:

- **Actualizar**: refresca la pantalla para obtener una vista del estado actual del sistema.
- **Consola de logs**: ejecuta la función de consola de logs.
- **Administrar**: permite administrar las tareas programadas registradas en el sistema.
- **Reiniciar Tareas**: ejecuta la función de reinicio de tareas programadas.

Reinicio de las Tareas Programadas

Esta función tiene por objetivo reiniciar la ejecución de las tareas programadas.

El sistema recupera la configuración vigente de cada tarea programada y comienza a ejecutarlas según la periodicidad definida.
Debe utilizarse cuando se modifican y/o agregan tareas.

Se obtendrá como resultado un mensaje indicando el resultado de la operación.

> *Cuando **Deyel** está configurado para iniciar sin ejecución de tareas programadas (propiedad TASK_SCHEDULER=False),*
> *la ejecución de esta función provocará que se inicien la tareas programadas .*

3.10.5.2. Consola de Eventos

Para acceder a la función se debe hacer clic sobre la opción **"Consola de Eventos"** del submenú **"Consolas"** en el menú de operaciones del módulo de configuración.

Otra alternativa para acceder a dicho menú, es utilizar la sección **"búsqueda rápida del menú"**,en la cual ingresando la palabra **"Eventos"**,se podrán visualizar las distintas opciones.

Objetivo

Esta función tiene por objetivo permitir la consulta del registro de eventos ocurridos y la información relacionada a los mismos.

Descripción de la Función

El procedimiento a seguir es el siguiente:

1. Establecer los criterios a considerar:

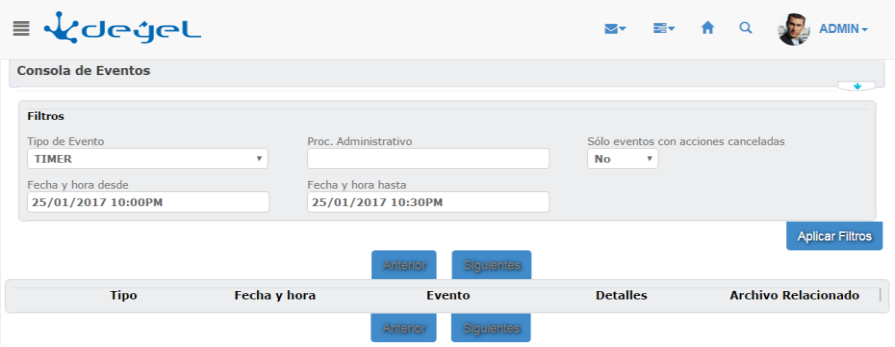
   - Tipo de evento.
     Permite una búsqueda por el tipo de Evento buscado.

   - Identificar el Evento por el Nº del Caso al que está asociado.

Se puede restringir la consulta de Eventos a un período, especificando los siguientes datos:
- Fecha y hora desde.
- Fecha y hora hasta.

- Además se puede filtrar por Eventos con acciones canceladas.

2. Pulsando el botón **"Aplicar Filtros"**, el sistema recuperará aquellos Eventos que cumplan con las condiciones establecidas y visualizará la siguiente información:



Para cada uno de los Eventos recuperados se visualizarán los siguientes datos:

- Tipo del Evento
- Fecha y hora de ocurrencia
- Identificación del Evento.
- Detalles.
- Archivo Relacionado.

Pulsando sobre la imagen de la columna *Detalles* se podrá acceder a la consulta detallada del evento de la fila seleccionada.

3.10.5.3. Consola de Logs

Para acceder a la función se debe hacer clic sobre la opción **"Consola de Logs"** del submenú **"Consolas"** en el menú de operaciones del módulo de configuración.

Otra alternativa para acceder a dicho menú, es utilizar la sección **"búsqueda rápida del menú"**,en la cual ingresando la palabra **"Logs"**,se podrán visualizar las distintas opciones.

Objetivo

Esta función tiene por objetivo permitir la consulta del registro de Logs producidos en el sistema. Diferentes funciones del sistema dejan registro de errores o simplemente informan la ocurrencia de determinados eventos, alimentando un repositorio que puede ser luego consultado por este mecanismo.

Descripción de la Función

El procedimiento a seguir es el siguiente:

1. Establecer los criterios a considerar:
   Todos los campos de filtro son de uso opcional

- Tipo de Log. Error, Información, Warning

- Tipo de Origen
  Es el tipo de funcionalidad que se ejecutaba cuando se originó el log.

- Modo de Acceso
  Modalidad de acceso al sistema que se utilizaba cuando se originó el log.
  Puede ser: Modelador, Batch, Cliente WEB, web Service, Otros

- Caso Relacionado
  Permite buscar incidentes relacionados con un determinado caso.

- Fecha desde. Por defecto el sistema propone la fecha actual.
- Fecha hasta.

- Hora desde. Por defecto el sistema propone la hora actual.
- Hora hasta.

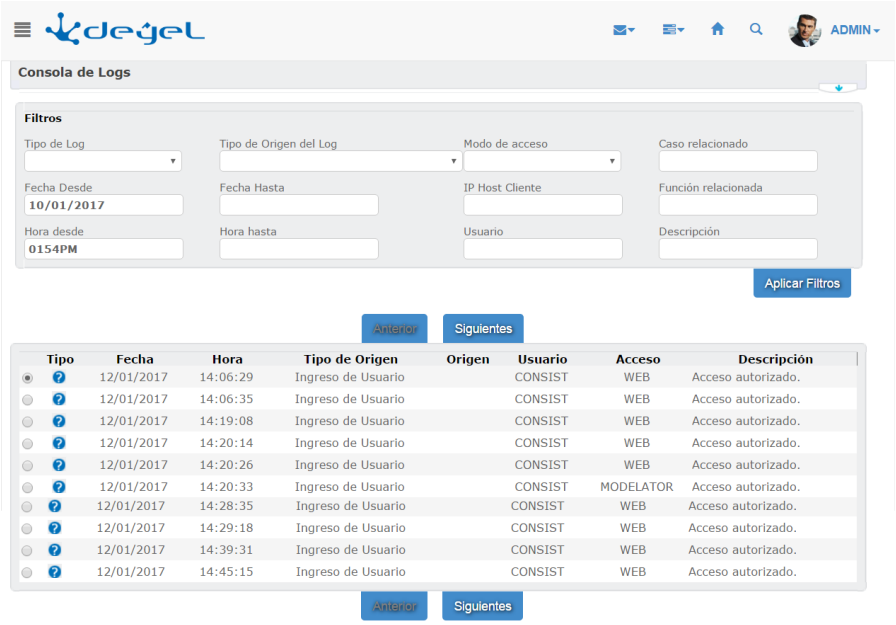
- IP HOST Cliente.
  Dirección IP desde la cual se solicitó la ejecución de la funcionalidad que originó el log.

- Función Relacionada
  Permite búsquedas por este atributo, que es alimentado por el sistema cuando se originó el log.

- Usuario.
  Es el usuario que estaba ejecutando la funcionalidad que originó el log.

- Descripción.
  Permite búsquedas por este atributo. El valor de este campo es generado por el sistema cuando se originó el log.
  Contiene una descripción del error producido.

2. Pulsando el botón **"Aplicar Filtros"**, el sistema recuperará aquellos Logs que cumplan con las condiciones establecidas y visualizará la siguiente información:



Para cada uno de los Logs recuperados se visualizarán los siguientes datos:

- Tipo de Logs. - Pulsando sobre la imagen visualizada en esta columna se accede a la consulta detallada del registro de log.
- Fecha y hora.
- Tipo de origen.
- Origen - Pulsando sobre la imagen visualizada en esta columna se accede a la consulta del Log seleccionado.
- Usuario
- Modo de Acceso
- Descripción. - Pulsando sobre la imagen de la columna se visualiza la descripción completa del Log seleccionado.

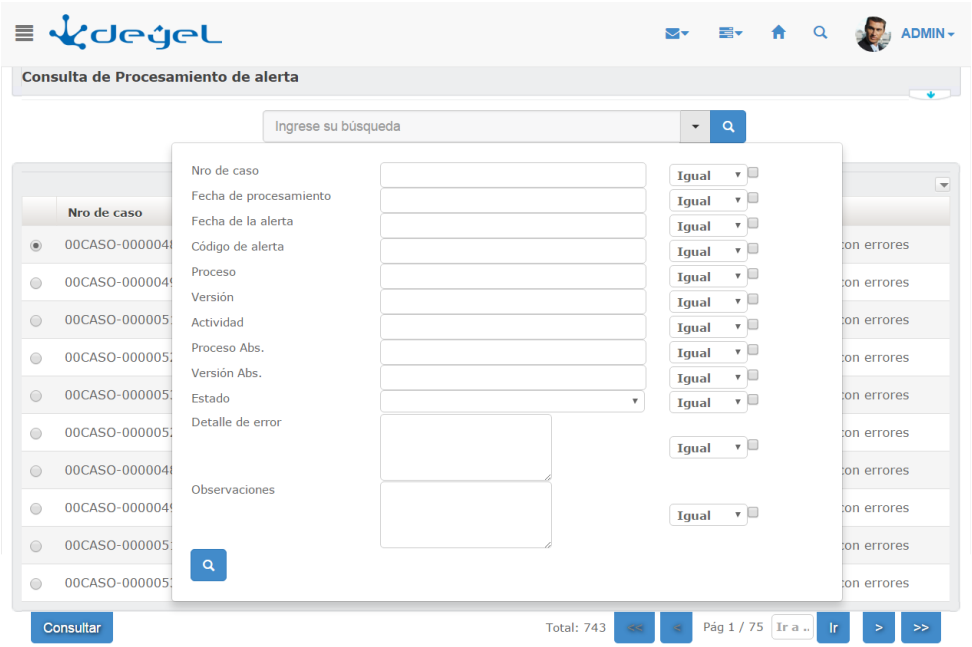
Para el caso de errores registrados por actividades automáticas se tiene opcionalmente la posibilidad de re ejecutarla. Esto dependerá del tipo de actividad automática y las características del error.

3.10.5.4. Consola de Procesamiento de Alertas

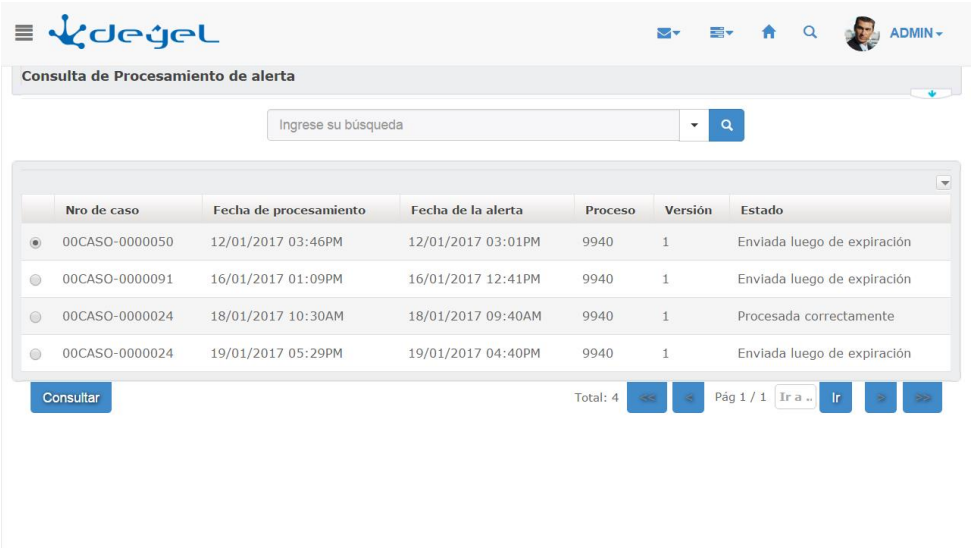
Para acceder a la función se debe hacer clic sobre la opción **"Procesamiento de Alertas"** en el sub-menú "**Consolas**" del menú de operaciones del módulo de **Configuración**

Objetivo
Esta función permite consultar, por diferentes criterios, las alertas que han sido procesadas por la tarea programada Ejecución de Alertas de las Actividades.

Descripción de la Función
Cuando se selecciona esta función se muestra una pantalla como la siguiente:

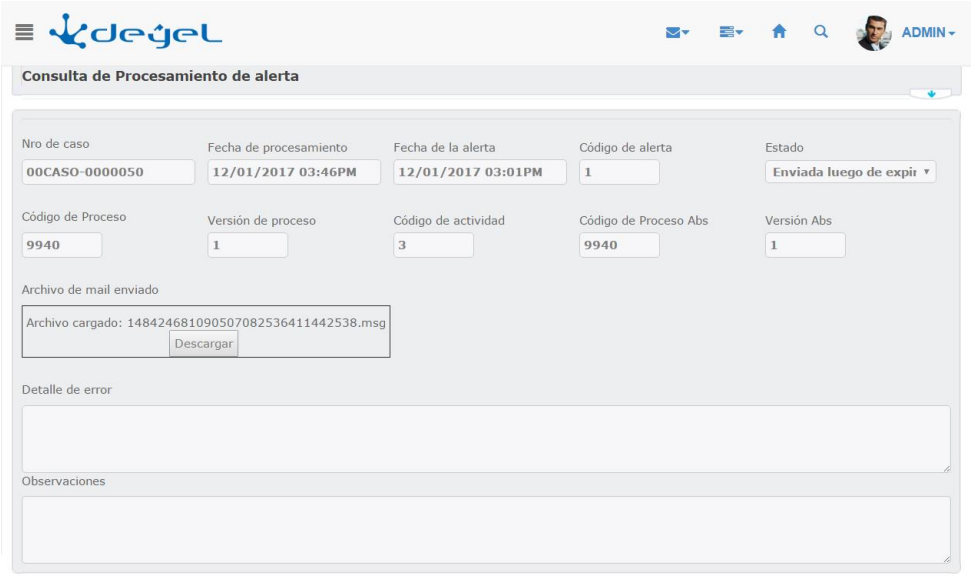


Se establecen los criterios de selección de la consulta a realizar.
Presionando el botón Consultar se ejecuta la consulta y se muestran los resultados en la siguiente pantalla



Se visualiza la lista de Alertas que se registraron al ser procesadas por el proceso emisor de las alertas.

Los detalles de cada alerta se pueden obtener seleccionándola de la lista y presionando el botón Consultar, que expondrá la siguiente pantalla:



En la que se visualiza la siguiente información
Nro de Caso.
Número de Caso a que corresponde la alerta.

Fecha de Procesamiento
Fecha y hora en que activó y ejecutó la tarea programada de emisión de alertas.

Fecha de Alerta
Fecha en que correspondía emitir la alerta.
Esta es la fecha que se almacena en la tabla de alertas pendientes en el campo DT_ALERT.
Siempre se cumple que la Fecha de Procesamiento es mayor o igual que la fecha de Alerta.

Código de Proceso . Versión de Proceso
Código y versión del proceso a que corresponde el Caso.

Código de Proceso Abst. Versión Abst.
Si la alerta corresponde a una actividad que está definida en un subproceso, estos campos indican el código y versión del subproceso.

Código de Actividad
Código de la actividad en la cual está definida el alerta.

Código de Alerta
Identifica a la alerta. En una misma actividad se pueden definir múltiples alertas.

Estado
Indica el estado con que finalizó el procesamiento de la alerta y el correspondiente envío de la notificación por mail.
Los estados posibles se almacenan en la tabla auxiliar T084 - Estados del Procesamiento de Alerta.

- Enviando

Este es un estado temporal. Cuando el programa emisor de alertas registra una alerta, lo hace con este estado inicial. Cuando termina de procesarla, le designa el nuevo estado según corresponda.

- Procesada Correctamente

Indica que la alerta se procesó correctamente y se envió la notificación por mail.

- Procesada con Errores

Indica que la alerta se procesó pero ocurrieron errores mientras se armaba el mail o se lo enviaba. Cualquier error de ejecución queda registrado en el campo Descripción del Error.

- Enviada luego de Expiración

Indica que la alerta se procesó correctamente pero el correo electrónico se envió cuando la alerta ya había expirado.
En estos casos, luego de enviar el mail se elimina la alerta de la tabla de alertas pendientes.

Archivo de Mail Enviado
Se almacena copia del mail enviado. Puede descargarse y consultarse, abriéndolo con su cliente de correo.

Detalle del Error
Cuando se producen errores durante la emisión de la alerta, por ejemplo al enviar el mail, se deja registro del mismo en este campo.

Observaciones
Este campo es de uso futuro.
El proceso de emisión de alertas podrá utilizarlo para dar información detallada al usuario sobre condiciones de error.

3.10.6. Herramientas

En la opción "**Configuración**" del menú principal, dentro del sub menú "**Herramientas**", están disponibles las siguiente opciones.

Administración de Tareas Programadas
Definición y configuración de las tareas programadas.

Exportar e Importar
Esta funcionalidad se utiliza fundamentalmente para resguardo de la información o para traspaso de la misma entre diferentes instalaciones o ambientes de **Deyel**.

Consola de Logs
Se incorpora a **Deyel** una herramienta para aplicar modificaciones sobre un ambientes.
Se utiliza para incorporar en un ambiente un conjunto de actualizaciones, que pueden incluir mejoras funcionales, correcciones de errores, actualización de versiones, etc.

3.10.6.1. Administración de Tareas Programadas

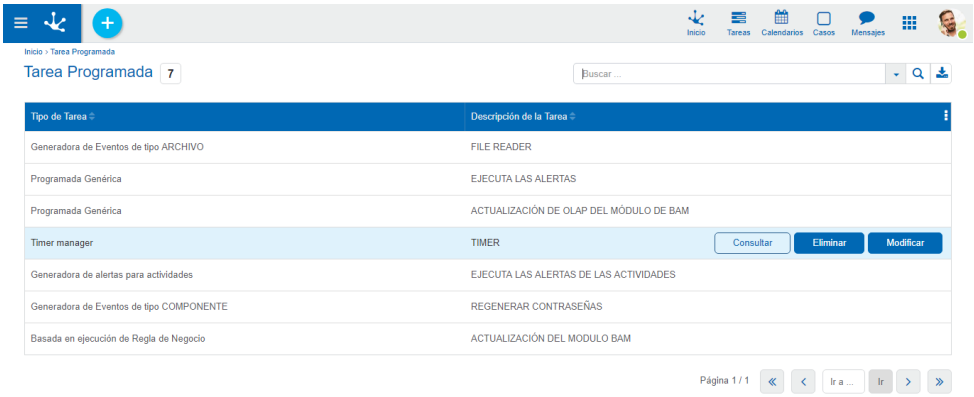
Esta función tiene por objetivo permitir la administración y configuración de las tareas programadas del sistema.

Cuando **Deyel** inicia su ejecución, se analizan las tareas programadas definidas y se dispara la ejecución de cada una de ellas en segundo plano y de acuerdo a la periodicidad configurada en cada caso.

Para acceder a esta función existen distintas alternativas:

- Hacer clic sobre la opción "Configuración" del menú principal y dentro del sub menú "**Consolas**", hacer clic en la opción "**Tareas Programadas**".
- En la búsqueda del menú ingresar "Tareas Programadas" y hacer clic en la opción listada.

Se presenta la lista de tareas programadas definidas, desde la cual se puede crear una nueva tarea programada o realizar el mantenimiento de las existentes.



Consultar Tarea Programada

Para acceder a la función se debe hacer clic sobre el botón "*Consulta*" desde la ventana de Administración de Tareas Programadas.
Permite consultar la Tarea Programada seleccionada.
Al seleccionar la función, aparecerá la ventana de Definición de Tarea Programada, donde se visualiza la definición de la tarea programada seleccionada.

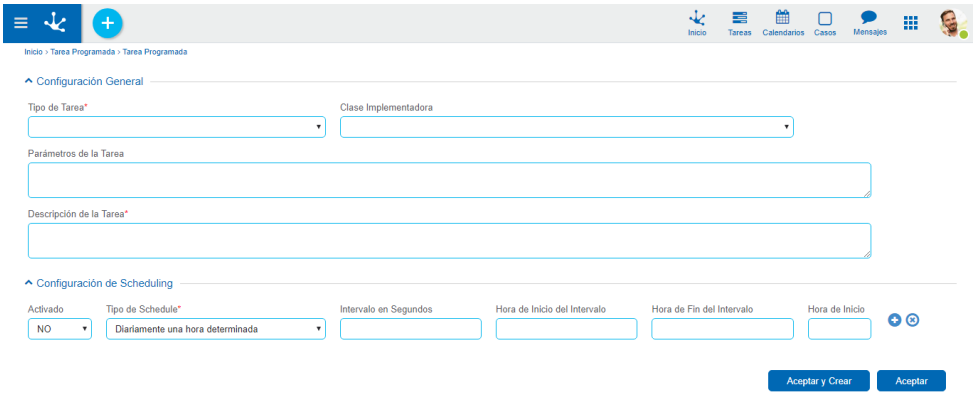
Modificar Tarea programada

Para acceder a la función se debe hacer clic sobre el botón "*Modificar*" desde la ventana de Administración de Tareas Programadas.
Permite modificar la Tarea Programada seleccionada.
Al seleccionar la función, aparecerá la ventana de Definición de Tarea Programada.
Se puede modificar la tarea teniendo en cuenta las mismas consideraciones que en alta en cuanto a los tipos de tareas, scheduling, etc.
Una vez cargados los datos, haciendo clic sobre el botón "*Aceptar*", se modifica la Tarea Programada y aparece una ventana informando el resultado de la operación.

Eliminar Tarea Programada

Para acceder a la función se debe hacer clic sobre el botón "*Eliminar*" desde la ventana de Administración de Tareas Programadas.
Permite eliminar la Tarea Programada seleccionada.

Nueva Tarea Programada
Desde la administración de tareas programadas, se puede crear una nueva haciendo clic sobre el botón y luego sobre el icono
Aparece la siguiente ventana de Definición de Tarea Programada:



Ingresar los datos correspondientes a la tarea a definir:

Configuración general

Tipo de tarea. (Obligatorio)
Se debe seleccionar el tipo de tarea.
Los tipos disponibles se obtienen desde la tabla auxiliar "T076 – Tipos de Tareas Programadas".

Clase Implementadora.
Se debe seleccionar la clase Java que define el funcionamiento de la tarea. Estas clases están registradas en la tabla auxiliar "T077 – Clases registradas para el Task Scheduler"
El dato es obligatorio para algunos tipos de tarea. Ej: Programada Genérica. Y es opcional para otros tipos, como por ejemplo: Basada en reglas de negocio

Parámetros.
En esta sección se informan los parámetros que se le pueden pasar a la tarea. Se indican en líneas separadas usando el formato:

paramX = valorX
paramY = valorY
paramZ = valorZ

Descripción.
Texto que describe el objetivo de la tarea.

Configuración de Scheduling

Se pueden definir varias líneas de scheduling, las cuales ejecutarán la tarea de manera independiente.

Por cada línea se deberán especificar los siguientes datos:

Activo. (Obligatorio)
Determina si la línea de scheduling es tenida en cuenta por el sistema de tareas programadas.

Tipo de Schedule. (Obligatorio)
Puede ser “Intervalos regulares” o “Diariamente una hora determinada”.

Intervalo. (Obligatorio - Si el tipo es “Intervalos regulares”).
Cantidad de segundos que transcurre entre ejecuciones de la tarea.

Hora de inicio. (Obligatorio si el tipo es “Diariamente una hora determinada”).
Hora del día en la que se inicia la tarea.

Configuración del tipo de tarea
Esta configuración solo se encuentra habilitada para los siguientes tipos de tareas:

- Basada en ejecución de Regla de Negocio.
- Ejecución condicional de acciones automáticas.
- Ejecución de regla y envío de resultado por mail.
- Generadora de Eventos de tipo COMPONENTE.

Para más información, ver "Tipos de tareas Programadas".

Una vez cargados los datos, haciendo clic sobre el botón **“*Aceptar*”**, se dará de alta la Tarea Programada y aparecerá una ventana informando el resultado de la operación.

3.10.6.1.1. Tipos de Tareas Programadas

Existen diferentes tipos de tareas programadas.

Generadora de eventos de tipo Mail

Al activar una tarea de este tipo, el sistema verifica la recepción de mails en la cuenta que se indica en la "Configuración de la recepción de Correo".
Por cada email recibido, se dispara un evento de tipo MAIL, que será anunciado a todos los procesos workflow que estén esperando la ocurrencia de eventos de este tipo.
Deyel provee, como parte de la instalación estándar, una tarea de este tipo. Ver el punto "Tareas Programadas del Sistema".

Generadora de eventos de tipo Archivo

Al activar una tarea de este tipo el sistema verifica la creación de nuevos archivos en una determinada carpeta.
La carpeta a verificar se indica en la propiedad FILE_READER_WORK_FOLDER, en la configuración del ambiente.
Por cada nuevo archivo detectado, se dispara un evento de tipo ARCHIVO, que será anunciado a todos los procesos workflow que estén esperando la ocurrencia de eventos este tipo.
Deyel provee, como parte de la instalación estándar, una tarea de este tipo. Ver el punto Tareas Programadas del Sistema.

Timer Manager

Siempre que haya procesos que utilicen eventos de tipo TIMER será necesario definir una tarea de este tipo.
Su función es analizar los distintos procesos modelados y, por cada evento de tipo TIMER que esté habilitado, dar inicio a una tarea de tipo "Generadora de eventos de tipo Timer".

Deyel provee, como parte de la instalación estándar, una tarea de este tipo. Ver el punto Tareas Programadas del Sistema.

Generadora de eventos de tipo Timer

Este tipo de tareas tiene relación con la anterior.
El usuario no crea este tipo de tareas programadas, sino que son creadas automáticamente por **Deyel**, cuando se ejecuta una tarea de tipo "Timer Manager".
Cada una de estas tareas queda asociada con un evento de tipo TIMER en un proceso particular. La función que cumple esta tarea programada es controlar el tiempo transcurrido y, de acuerdo a la periodicidad indicada en dicho evento, comunicar al proceso su ocurrencia.

Generadora de eventos de tipo Componente

Al activar una tarea de este tipo el sistema ejecuta una regla de negocios (también denominada "componente") y con los resultados de dicha ejecución se dispara un evento de tipo COMPONENTE, que será anunciado a todos los procesos workflow que estén esperando la ocurrencia de eventos de este tipo.

La regla de negocios a ejecutar se especifica al crear y configurar la tarea.

Basada en reglas de negocio

Este tipo de tarea ejecuta una regla de negocios y luego, opcionalmente, realiza una operación con el resultado de la ejecución.
La regla de negocios a ejecutar se especifica al crear y configurar la tarea, así como la clase Java que aporta el comportamiento adicional o procesamiento de los resultados.

Ejecución de reglas de negocio y envío de resultados por mail

Este tipo de tareas es similar a la anterior pero tiene la particularidad de que, después de ejecutar la regla de negocios, se envía un mail con los resultados obtenidos al ejecutar la regla.
Cuando se crea una regla de este tipo, en el atributo Parámetros se especifican los siguientes:

- Una línea obligatoria DS_EMAIL_TO = direcciones de correo a las cuales enviar los resultados. Pueden ser varias, separadas por ";"
- Una línea opcional DS_EMAIL_ADDITIONAL_TEXT = Texto que se debe incluir en el cuerpo del MAIL

Los resultados de la ejecución de la regla se adjuntan al mail como un archivo de valores separados por coma (formato .CVS)

Tarea de ejecución de archivos de comandos

Una tarea de este tipo permite ejecutar un archivo de comandos (por ejemplo: bat, cmd, sh) que se especifica en el cuadro de parámetros de la siguiente forma:
- Una línea obligatoria FILENAME=<nombre de archivo> para especificar el archivo a ejecutar. El nombre puede ser absoluto, relativo al directorio de instalación del contexto o a la carpeta "batch" del mismo.
- Una línea opcional ARGUMENTS=<argumentos del archivo> donde se especifican parámetros que se le pasarán a la ejecución del archivo de comandos.

Generadora de Alertas para Actividades

Al activar una tarea de este tipo el sistema verifica las actividades que se encuentran en ejecución y analiza la necesidad de emitir alertas. Considerando la definición de alertas que se modeló en cada actividad, se envían las notificaciones por correo electrónico que corresponda.
Deyel provee, como parte de la instalación básica, una tarea de este tipo. Ver, el punto "Tareas Programadas del Sistema".

Programada Genérica

Una tarea de este tipo permite programar la ejecucion periodica de una clase Java.
Requiere que el usuario realice un desarrollo adicional que aporte el comportamiento o la función específica de la tarea. Este desarrollo adicional, consiste en crear una clase Java que extienda de **com.consist.taskScheduler.Task** y redefina los métodos necesarios para que cumpla la función deseada.
Esta clase, luego debe estar en el ambiente de **Deyel** ya sea como archivo .class bajo el directorio **WEB-INF/classes** o bien dentro de un jar perteneciente a la carpeta WEB-INF/lib.

Finalmente, para poder utilizar esta clase, su nombre completo debe registrase en la tabla auxiliar "T077 – Clases registradas para el Task Scheduler".

3.10.6.1.2. Tareas Programadas del Sistema

Deyel provee un conjunto predefinido de tareas programadas, que se describen a continuación.

Tareas programadas requeridas

Un primer grupo conformado por tareas programadas cuya ejecución es requerida por **Deyel**, para asegurar su correcto funcionamiento.
Inician automáticamente al iniciar el servicio y no pueden ser pausadas por el administrador del Sistema.

- **DEPURADOR DE CONEXIONES DEL POOL PRINCIPAL**

Esta tarea monitorea las conexiones del pool principal y verifica su correcto desempeño.
Elimina conexiones que pudieran no estar activas o que dan time-out.
Garantiza que la cantidad de conexiones se encuentre dentro de los rangos indicados en la configuración del sistema.

- **DEPURADOR DE ARCHIVOS CARGADOS EN REPOSITORIOS.**

Esta tarea se encarga de eliminar definitivamente contenidos que han quedado en desuso. Por ejemplo, archivos o imágenes que estaban contenidos dentro de un campo de un formulario y que fueron reemplazados por otro. O archivos de distinto tipo que estaban anexados a un caso y han dejado de estarlo.
Al ejecutar esta tarea se detectan este tipo de condiciones y se eliminan estos contenidos que pueden considerarse obsoletos o en desuso.

- **DEPURADOR DE LOGS ANTIGUOS DEL SISTEMA**

Esta tarea elimina archivos de logs antiguos e ítems de la "Consola de Logs". En la configuración del ambiente puede indicarse la antigüedad de los datos a depurar:

LogsPersistenciaArchivos: establece la antigüedad, en días, de los archivos de log de la aplicación que serán eliminados por esta tarea. Si el valor es cero nunca se eliminarán archivos.

LogsDepuracionConsola: establece la antigüedad, en días, de los ítems de la consola de logs que serán eliminados por esta tarea. Si el valor es cero nunca se eliminarán ítems.

- **DEPURADOR DE ARCHIVOS TEMPORALES DEL SISTEMA**

Esta tarea elimina los directorios temporales de **Deyel** (Temp, BusinessRuleDeployment, BusinessRuleXML, EventFiles, Export, Import, IntegrationDeployment, IntegrationXML y Upload).
Ejecuta cada 24 Hs, a partir de la hora de inicio del sistema o de reinicio de las Tareas Programadas.

- **MANEJADOR DE ENVÍO DE MAILS.**

Esta tarea se asegura el envío de mails por parte del **Deyel**. Se implementa un esquema de reenvíos que permiten superar fallas en un servidor de correo saliente, etc.

Tareas programadas opcionales

Un segundo grupo de tareas programadas, admiten que el administrador del sistema las active o desactive o configure la periodicidad de ejecución, de acuerdo a sus necesidades.

- **MAIL READER**

Esta tarea es de tipo Generadora de Eventos de tipo MAIL.
En una instalación inicial de **Deyel**, esta tarea esta inactiva. Debe ser activada cuando se modelan procesos que usen este tipo de eventos.
Deyel verificará la recepción de correos en la cuenta que se indica en la "Configuración de la recepción de Correo".

- **FILE READER**

Esta tarea es de tipo Generadora de Eventos de tipo ARCHIVO.
En una instalación inicial de **Deyel**, esta tarea esta inactiva. Debe ser activada cuando se modelan procesos que usen este tipo de eventos.
Por cada nuevo archivo detectado, se dispara un evento de tipo ARCHIVO, que será anunciado a todos los procesos workflow que estén esperando la ocurrencia de eventos este tipo.

- **TIMERS**

Esta tarea es de tipo Timer Manager.
En una instalación inicial de **Deyel**, esta tarea esta inactiva. Siempre que haya procesos que utilicen eventos de tipo TIMER será necesario activar esta tarea.

- **EJECUCIÓN DE ALERTAS DE LAS ACTIVIDADES.**

Esta tarea es de tipo "Generadora de Alertas para Actividades".
En una instalación inicial de **Deyel**, esta tarea esta inactiva. Debe activarse su ejecución si se requiere que **Deyel** considere las alertas modeladas en las actividades y envíe las notificaciones pertinentes vía correo electrónico.

- **ACTUALIZACIÓN DEL MODULO BAM**

Esta tarea es de tipo "Basada en Reglas de Negocio".
En una instalación inicial de **Deyel**, esta tarea está activa y ejecuta cada 10 minutos.
Su misión es tomar todas las novedades de casos y actividades que se registraron desde la última ejecución de esta tarea, y actualizar automáticamente la información del módulo de BAM (Business Activity Monitoring). La ejecución de esta tarea provocará que tanto la información de BAM (casos y actividades en ejecución) como de Análisis de Procesos (casos y actividades finalizados) sea actualizada.

La información de actividades y de casos nuevos o modificados, es detectada por esta tarea para que de **manera incremental**, impacte en el módulo de BAM.
Entonces, cada 10 min. todo cambio o novedad es detectada y trasmitida al módulo de BAM, para que al realizar la visualización de los reportes de BAM, se pueda contar con información reciente y actualizada.

- **ACTUALIZACION DIARIA BAM CASOS Y ACTIVIDADES EN EJECUCION**

Esta tarea es de tipo "Basada en Reglas de Negocio".
En una instalación inicial de **Deyel**, esta tarea está activa y ejecuta solo 1 vez por día. No obstante, esta tarea se puede programar para ejecutarse con mayor frecuencia, por ejemplo, cada 4 hs. según lo requiera la instalación.
Su misión es mantener actualizada la duración (tiempo de ejecución) y la situación (atrasado, en riesgo, a tiempo) de **todos los casos y actividades** que están "en ejecución", según la frecuencia con que

la tarea fue programada: diaria o cada 4 hs.
Entonces, esta tarea solo contempla la actualización de casos y actividades “en ejecución”, que no son tomados por la tarea anterior que corre cada 10 min.

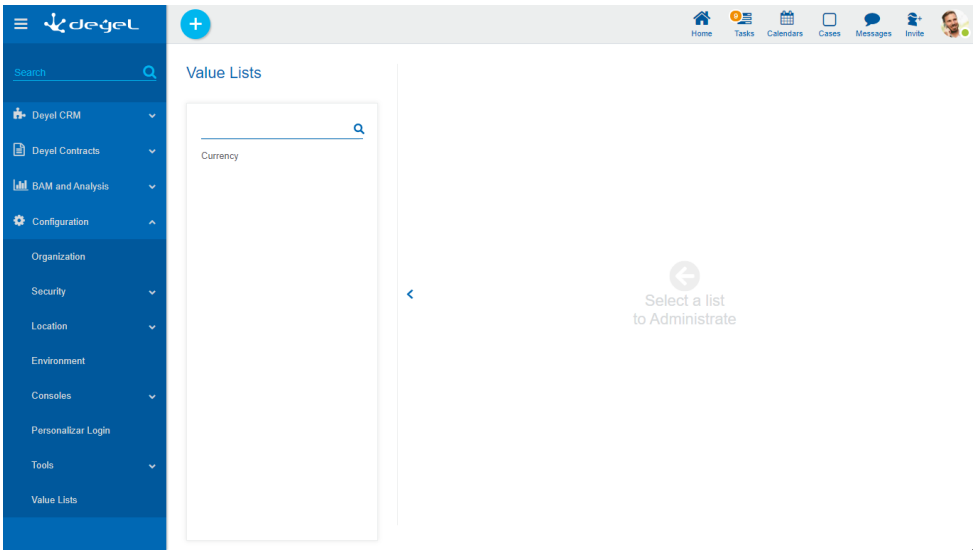
Por ejemplo, un caso en ejecución que fue cargado el día de ayer y fue tomado por la actualización incremental cada 10 min., quedará sin actualizar su duración y situación hasta tanto no sea modificado nuevamente y sea detectado por la carga incremental. Para evitar que la información quede desactualizada, esta tarea se encarga de barrer toda la información “en ejecución” y actualizar su duración y situación.

De esta manera, esta tarea se complementa con la anterior, para que al realizar la visualización de los reportes de BAM, se pueda contar con información completa y actualizada.

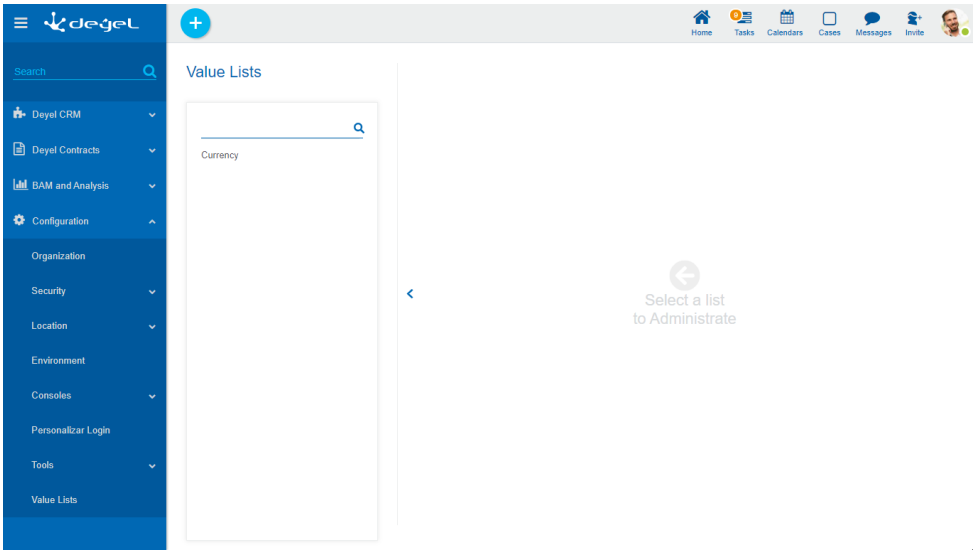
3.10.7. Value Lists

The "Value List" option of the menu allows application users to manage the lists values, if they have the corresponding permissions assigned.

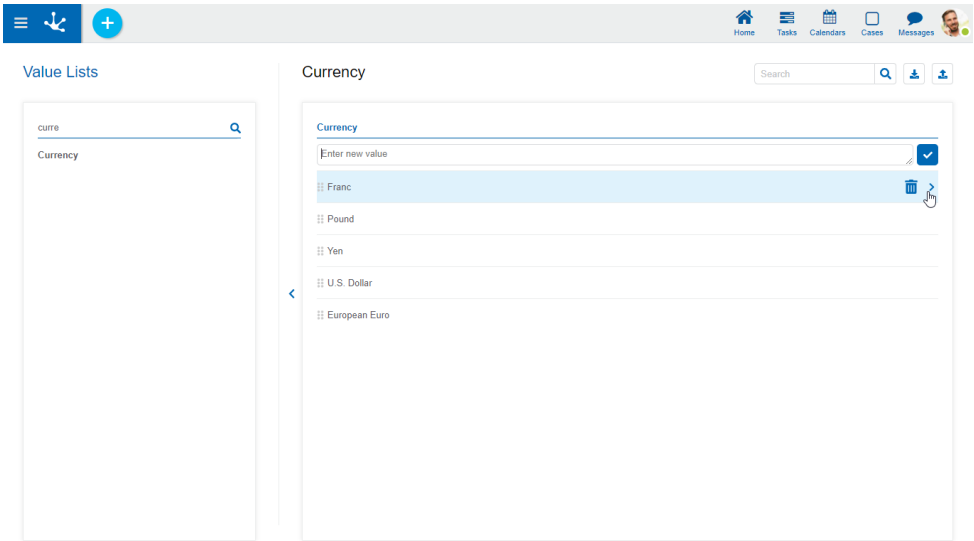
The names of the value lists in the different applications are displayed.



Allows to filter values from the list based on the characters entered. If a list is very long it helps users to easily visualize the desired values.



When selecting a value list, the values of the selected list are displayed in the right panel, thus making it possible to search, enter, delete and restore their values.



It allows to export the value list in an Excel file. This file contains the following columns:

- CODE
- VALUE
- ORDER
- DELETED
- FILTER_CODE

Where DELETED indicates low logic, FILTER_CODE contains the code to establish the correspondence between the value of the list being downloaded and the list that filters it.

It allows to import a value list from an Excel file. A panel opens where the location and name of the same is selected.

Conditions for Export and Import

- The download file format must be respected.
- New values must have a blank Code column.
- Blank rows must be ignored.
- Partial downloading and uploading of values is not allowed.

Operations on Values

- Allows adding each entered value to the list of values.
- Double click: Allows to modify a value in the list.
- Move: Allows to change the position of a value within the list by dragging the value with the mouse.

Hovering the cursor over each of the values entered, a set of icons is displayed and this allows to perform different operations.

- Allows to delete a value from the list of values. Once deleted, it is displayed in gray and crossed out.
- Allows to restore a previously deleted value.

Display the Selected Line

- Hides the icons that are displayed.
- Shows hidden icons.

3.11. Requirements

A continuación se indican los requerimientos de software para el uso y la instalación de **Deyel**.

Puestos de Trabajo (Modalidad On-Premise y Cloud)

HARDWARE	RAM	4 GB mínimo
	Resolución de pantalla	• Usuario final: 1024x768 mínima • Modelador: 1920x1080 mínima
SOFTWARE	Browser	• Chrome 79.0 mínima • Edge Chromium 79.0 mínima

		• Firefox 74.0 mínima

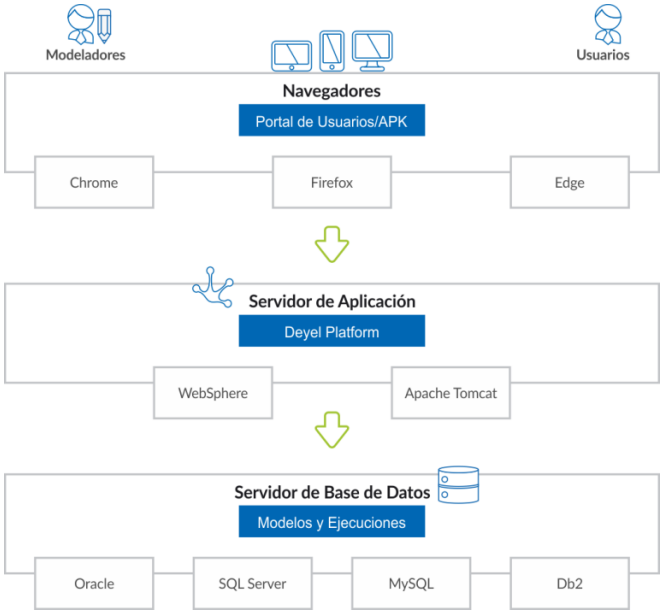
Las versiones de los navegadores es recomendable mantenerlas actualizadas dado que se corrigen periódicamente vulnerabilidades de seguridad.

Servidor de Aplicación (Modalidad On-Premise)

HARDWARE	RAM	6 GB mínimo
	HD	5 GB tamaño mínimo libre.
	Procesador	64-bit / 2 cores / 2.4GHz mínimo
SOFTWARE	Web server	• WebSphere 8 / WebSphere 9 • Apache Tomcat 8 / Apache Tomcat 9
	Java	JRE 1.8

Servidor de Base de Datos (Modalidad On-Premise)

HARDWARE	RAM	4 GB mínimo
	HD	10 GB tamaño mínimo libre
	Procesador	64-bit / 2 cores / 2.4GHz mínimo
SOFTWARE	Database	• Mysql 5.7 • DB2 9.5 o DB2 10.5 • Sql Server 2012 • Oracle 12c release 12.2



3.12. On-Premise Installation

Se describen a continuación los pasos para realizar una nueva instalación o una actualización de versión en instalaciones On-Premise.

Distribución

La distribución de **Deyel** y sus soluciones se componen de los siguientes archivos:

Descripción	Archivo
Aplicación web	deyel-***Versión***.war
Base de datos inicial	• Oracle: deyeldb-***Versión***.dmp • Mysql: deyeldb-***Versión***.sql • Sql Server: deyeldb-***Versión***.bak • DB2: deyeldb-***Versión***.zip
Licencia	licence.xml

> ***Versión***
> *Cada versión de **Deyel** tiene la siguiente estructura V.R.F.B. donde*
> *V - Versión*
> *R - Release*
> *F - Fix*
> *B - Build*

Ejemplo: deyel-6.5.0.3.war

Paso 1: Crear la Base de Datos Inicial para Deyel

a. Instalar un motor de base de datos sobre la que se quiere instalar **Deyel**.

b. Crear un usuario con permisos que pueda crear tablas, vistas, funciones, stored procedures y triggers. Se deben conocer las siguientes variables:

- **host**: dirección ip del motor
- **port**: puerto del motor
- **user**: usuario con los derechos necesarios
- **password**: clave del usuario con los derechos necesarios

c. Restaurar la base inicial de **Deyel** con los siguientes comandos:

- Oracle
  impdp **user/password@host:port**/deyel dumpfile=deyeldb-***Versión***.dmp

- Mysql
  mysql -u **user** -p **password** -h **host** --port **port** --default-character-set=utf8 deyel < deyeldb-***Versión***.sql

- Sql Server
  SqlCmd -S **host** -U **user** -P **password** -Q "RESTORE DATABASE deyel FROM DISK='deyeldb-***Versión***.bak' WITH REPLACE"

- DB2
  Descomprimir el archivo deyeldb-***Versión***.zip
  Ejecutar el comando en la carpeta donde se encuentran los archivos del export
  db2move deyeldb import -io replace_create > restore_deyel.log

Paso 2: Instalar Deyel en el Servidor de Aplicaciones

- Instalación en Apache Tomcat

- Instalación en WAS

Paso 3: Configurar la Base de Datos

Para realizar la configuración de la base de datos, se recomienda utilizar la variable de ambiente DEYEL_DB_PROPERTIES_PATH para definir el directorio donde se encuentra el archivo con los datos de conexión. En caso que no se defina la variable de ambiente, se utiliza el archivo dentro del contexto WEB-INF/clasess/ConsistEnv_es_AR.properties.

La variable de ambiente se debe crear antes de hacer el deploy del contexto.

Instructivo para la Configuración de la Base de Datos

a. Ingresar a https://<servidor:puerto>/deyel/Config.

> *Donde servidor: es el nombre del servidor donde se instala o su ip y puerto corresponde al puerto de acceso.*
> *Ejemplo: http://miequipo:8080/deyel*

> *Se accede sin usuario y clave hasta que se defina uno. Es recomendable definir el usuario y la clave en el primer acceso*

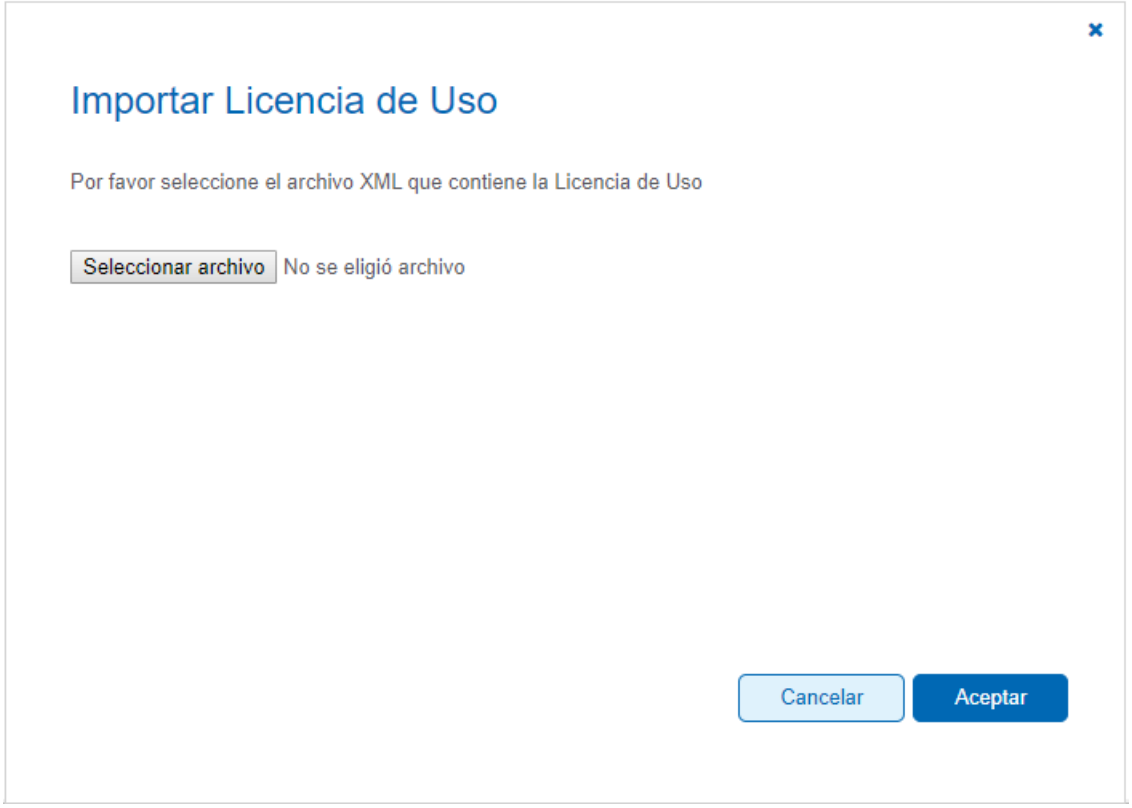
b. Completar los datos de la sección Propiedades de la Base de Datos y aplicar la configuración. Si se definió la variable de ambiente DEYEL_DB_PROPERTIES_PATH, en la propiedad Path de Archivo de Configuración se muestra el directorio donde se encuentra el archivo con los datos de conexión, si esta propiedad no se muestra informada, esto significa que el archivo se encuentra en el contexto.

c. Reiniciar el contexto.

Paso 4: Importar la Licencia

a. Ingresar a https://<servidor:puerto>/deyel.

b. Importar la licencia.



c. Ingresar a **Deyel** con https://<servidor:puerto>/deyel utilizando el usuario de ejemplo"afarias" con clave "deyel123".

3.12.1. Estructura de Directorios

Una vez realizado el deploy del war queda definida la siguiente estructura de directorio.

Directorio	Descripción
batch • log	Directorio con las clases Java que permiten ejecución de procesos batch. • log: Directorio donde se guarda el resultado de la ejecución de las clases ejecutadas.

Directorio	Descripción
businessRuleDeployment	Directorio donde se almacenan temporalmente los archivos generados por los deploys de reglas de negocio.
businessRuleXML	Directorio donde se almacenan temporalmente las definiciones de reglas de negocio.
docGenerator • app • staticHTML • themes • themesDefault • themesResponsive • themesResponsiveMin	Se almacenan todos los archivos relacionados con la definición de los formularios. • app: Se encuentran los documentos generados. Deprecado. • staticHTML: Archivos básicos que se utilizan para componer otras pantallas. • themes / themesDefault / themesResponsive / themesResponsiveMin: Los temas que se utilizan al modelar formularios.
eventFiles	Directorio temporal de archivos vinculados con eventos que ocurren en los procesos (mails, archivos del File Reader, etc.).
export	Directorio temporal donde se almacenan los archivos exportados.
Import	Directorio temporal donde se almacenan los archivos importados.
integrationDeployment	Directorio para almacenar temporalmente los archivos generados por los deploys de reglas de integración (SQL y web services).
integrationXML	Directorio para almacenar temporalmente las definiciones de reglas de integración (SQL y web services).
jasperReportFiles	Archivos xml que contienen templates del framework Jasper Report utilizado en la impresión de reportes e impresiones en general.
logs	Directorio de almacenamiento de los logs de **Deyel**. Al instalar se pueden especificar los tipos de logs y la cantidad de días que se mantienen.
META-INF	Información de versión de **Deyel**.
modeler	Archivos con imágenes, javascript y estilos del modelador web
script	Archivos JavaScript que se utilizan en el portal web.
Temp	Directorio temporal de **Deyel**.
themes	Directorio de almacenamiento de los temas de **Deyel**.

Directorio	Descripción
updaters	Directorio que contiene los archivos necesarios para actualizar la versión de **Deyel**.
upload	Directorio para almacenar temporalmente los archivos subidos a **Deyel**.
vendors	Directorio con los archivos de los vendors utilizados por **Deyel**.
WEB-INF	Directorio de almacenamiento de las clases que definen el funcionamiento de **Deyel**.
widget	Directorio con los templates de los widgets utilizados por el portal web.

Los directorios temporales (Temp, BusinessRuleDeployment, BusinessRuleXML, EventFiles, Export, Import, IntegrationDeployment, IntegrationXML y Upload) se eliminan automáticamente cada 24hs. Las 24 hs se calculan a partir de la hora de inicio de las Tareas Programadas.

3.12.2. Instalación en Apache Tomcat

Una de las opciones para instalar **Deyel** en un servidor de aplicaciones es hacerlo bajo Apache Tomcat.

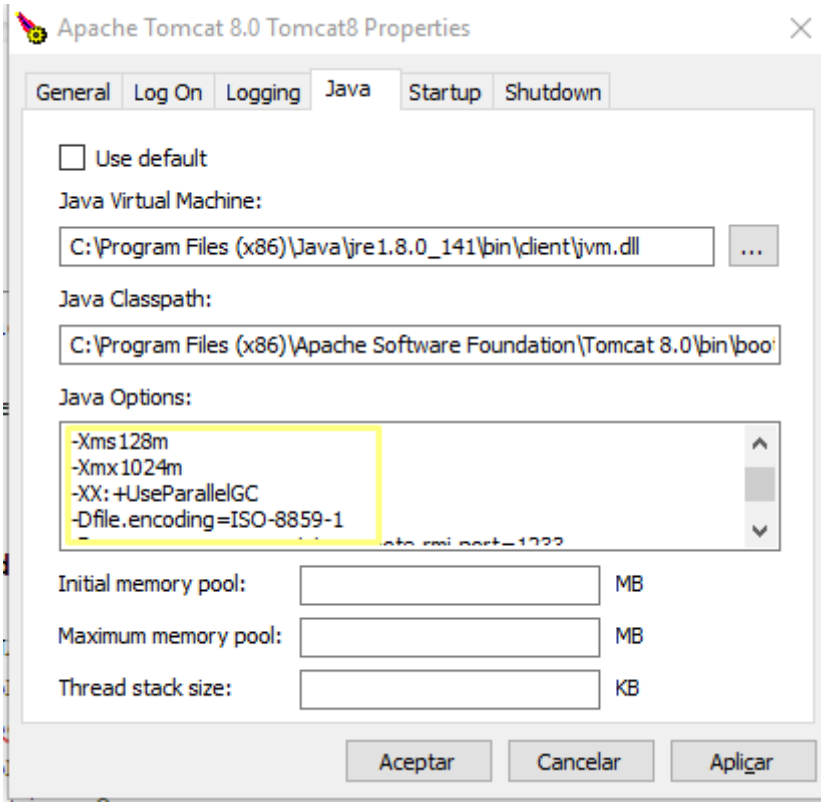
Configuración de Apache Tomcat para Deyel

En Linux

En el archivo setenv.sh dentro de la carpeta bin de Apache Tomcat se pueden definir los siguientes valores mínimos:

- `Xms128m`
- `Xmx1024m`
- `XX:+UseParallelGC`
- `Dfile.encoding=ISO-8859-1`

En Windows



> *Los valores Xms y Xmx son valores mínimos que pueden ser incrementados según el uso que se de a**Deyel**.*

Dockerizadas

- `XX:+UnlockExperimentalVMOptions`
- `XX:+UseCGroupMemoryLimitForHeap`
- `XX:+UnlockExperimentalVMOptions`
- `XX:+UseCGroupMemoryLimitForHeap`
- `XX:MaxRAMFraction=2`
- `XX:+UseParallelGC`
- `Dfile.encoding=ISO-8859-1`

Uso de Caracteres Especiales, Encoding y Tamaño de Post

Configurar los siguientes archivos con las líneas indicadas para cada caso.

Apache Tomcat 8.0

- conf/catalina.properties
  Incluir la línea:
  `tomcat.util.http.parser.HttpParser.requestTargetAllow=|`
- conf/server.xml

```
<Connector port="8080" protocol="HTTP/1.1"
     connectionTimeout="20000"
     maxHttpHeaderSize="32768"
     maxPostSize="-1"
```

```
redirectPort="8443"
URIEncoding="UTF-8" />
```

Apache Tomcat 8.5 y superiores

- conf/server.xml

```
<Connector port="8080" protocol="HTTP/1.1"
  connectionTimeout="20000"
  maxHttpHeaderSize="32768"
  maxPostSize="-1"
  redirectPort="8443"
  URIEncoding="UTF-8"
  relaxedPathChars="[ \ ] { | }"
  relaxedQueryChars="[ \ ] { | }"  />
```

> *El tamaño máximo de post requerido se deshabilita con maxPostSize="-1" para guardar o publicar procesos.*

Protección clickjacking, HSTS header force y browser content sniffing

Habilitar en el archivo conf/web.xml el filtro de seguridad HTTPHeader que posee Tomcat. Para instalaciones de Tomcat 8 y superiores debe ser:

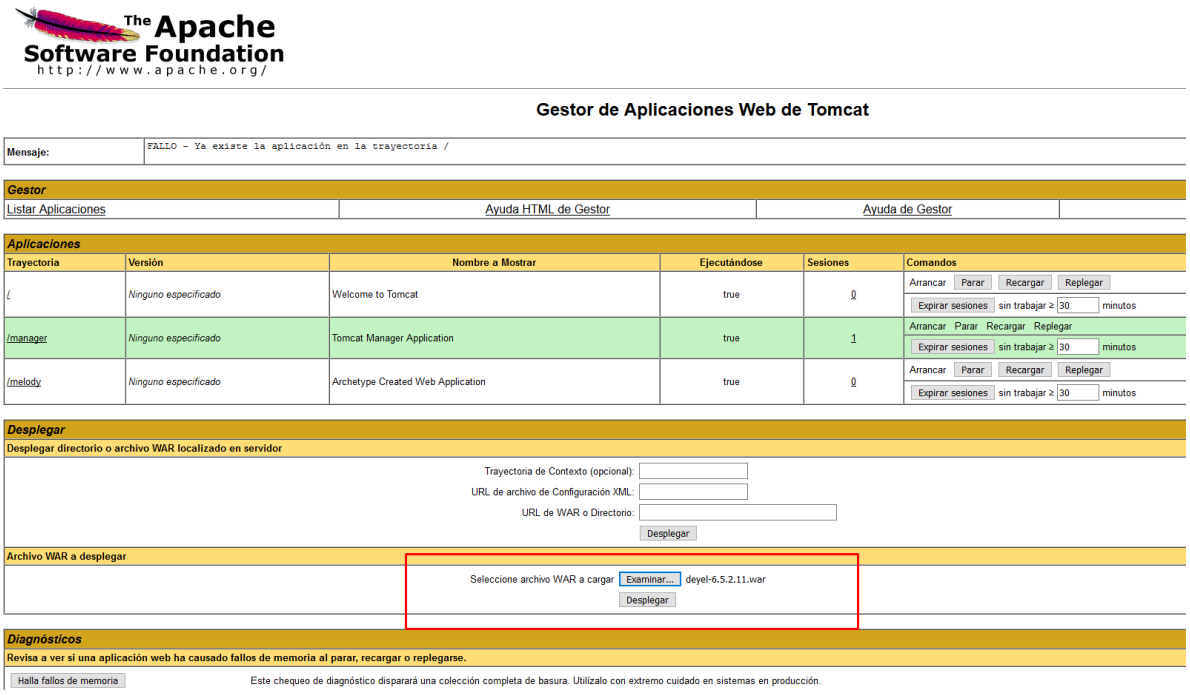
```
<filter>
        <filter-name>httpHeaderSecurity</filter-name>
      <filter-
class>org.apache.catalina.filters.HttpHeaderSecurityFilter</filter-
class>
        <async-supported>true</async-supported>
        <init-param>
            <param-name>antiClickJackingOption</param-name>
            <param-value>SAMEORIGIN</param-value>
        </init-param>
        <init-param>
            <param-name>hstsEnabled</param-name>
            <param-value>true</param-value>
        </init-param>
</filter>

 <filter-mapping>
        <filter-name>httpHeaderSecurity</filter-name>
        <url-pattern>/*</url-pattern>
        <dispatcher>REQUEST</dispatcher>
</filter-mapping>
```

Deploy del war

Para que el deploy se realice en un contexto con el nombre deyel se debe renombrar el war asignado-le el nombre deyel.war

Apache Tomcat tiene un máximo predeterminado para subir archivos war de 50Mb, incrementarlo a 500Mb en el archivo <TOMCAT>\webapps\manager\WEB-INF\web.xml

```
51      </init-param>
52      -->
53      <multipart-config>
54        <!-- 500MB max -->
55        <max-file-size>524288000</max-file-size>
56        <max-request-size>524288000</max-request-size>
57        <file-size-threshold>0</file-size-threshold>
58      </multipart-config>
59    </servlet>
60    <servlet>
61      <servlet-name>Status</servlet-name>
```



Continuar con la instalación.

3.12.3. Actualización de Versión

Se describen a continuación los pasos para realizar una actualización de versión en instalaciones On-Premise.

Distribución

La distribución de **Deyel** y sus soluciones para realizar una actualización de versión incluye el siguiente archivo:

Descripción	Archivo
Aplicación web	deyel-***Versión***.war

> ***Versión***
> *Cada versión de **Deyel** tiene la siguiente estructura V.R.F.B. donde*
> *V - Versión*
> *R - Release*
> *F - Fix*
> *B - Build*
>
> *Ejemplo: deyel-6.5.0.3.war*

Paso 1: Realizar el Resguardo de la Base de Datos Existente

- Oracle
  expdp deyel/**password**@**host**:**port**/deyel dumpfile=deyeldb.dmp schemas=deyel

- Mysql
  mysqldump -u root -p -h **host** --routines --skip-add-locks --single-transaction deyel > deyeldb.sql

- Sql Server
  BACKUP DATABASE deyel TO DISK = 'deyeldb.bak'"

- DB2
  db2move deyeldb export -aw -sn deyel > deyel_export.log
  Los archivos del export quedan en la carpeta donde se ejecuta el comando.
  Generar un archivo comprimido llamado deyeldb.zip con todos los archivos generados

Paso 2: Realizar el Resguardo del Contexto Existente

a. Comprimir todos los archivos del contexto existente y moverlos fuera del servidor de aplicaciones.

b. Realizar en el servidor de aplicación el undeploy del contexto existente.

Paso 3: Instalar Deyel en el Servidor de Aplicaciones.

- Instalación en Apache Tomcat

- Instalación en WAS

Paso 4: Configurar la Base de Datos

Si se definió la variable de ambiente DEYEL_DB_PROPERTIES_PATH con el directorio donde se encuentra el archivo con los datos de conexión, no es necesario realizar este paso.

Si no se definió la variable de ambiente DEYEL_DB_PROPERTIES_PATH, el archivo de configuración de propiedades de base de datos se encuentra dentro del contexto y en ese caso se debe ingresar a la configuración de **Deyel** y volver a completar los valores para las propiedades de conexión.

Instructivo

a. Ingresar a https://<servidor:puerto>/deyel/Config.

> *Donde servidor: es el nombre del servidor donde se instala o su ip y puerto corresponde al puerto de acceso.*
> *Ejemplo: https://miequipo:8080/deyel*

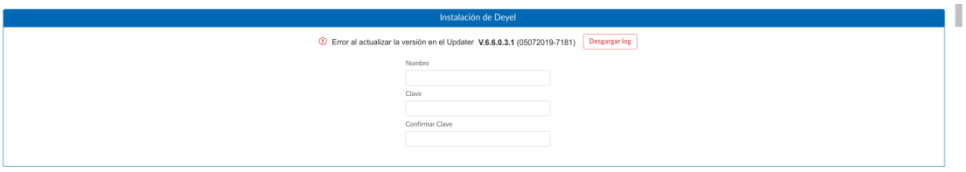
> *Se accede sin usuario y clave hasta que se defina uno. Es recomendable definir el usuario y la clave en el primer acceso*

b. Completar los datos de la sección Propiedades de la Base de Datos y aplicar la configuración.

c. Reiniciar el contexto.

Paso 5: Verificar si se Realiza la Actualización Correctamente

Ingresar a https://<servidor:puerto>/deyel/Config y verificar si hay error en https://<servidor:puerto>/deyel/Config.



Si hubo error al actualizar realizar lo siguiente:

1. Descargar el log del updater que falla y el log completo del dia (SystemOut*.log) y enviarlo al equipo **Deyel**.
2. Restaurar la base de datos resguardada en el paso 1.
3. Detener el servidor de aplicación y copiar todos los archivos del contexto resguardados en el paso 2.

3.13. Phased Learning

Fases de Capacitación

La capacitación en **Deyel** comprende un conjunto de fases progresivas que permiten al usuario, modelar sus aplicaciones y utilizar el producto en toda su potencialidad, sin necesidad de conocimientos previos.

Las fases se componen de un conjunto de videos de autoestudio y propuestas de ejercicios para refor-

zar los conocimientos. Se incluyen también videos de tips avanzados, para usuarios más experimentados.

Fases de Capacitación

- Introducción a Deyel
- Modelado de Formularios
- Portal
- Modelado de Procesos
- BAM
- Configuración

www.deyel.com
Argentina: +54-11-5219-4600
Chile: +56-2-2233-5901
México: +52-55-5980-9528
Email: hola@deyel.com
Síguenos

Argentina +54-11-5219-4600
hola@optaris.com
www.optaris.com