



# **Sistemas de Información Empresariales**

## **Curso 2022-23**

---

### **Trabajo sobre Bonita BPM**

---

**Trabajo en equipo asociado: HÉRCULES SEGURIDAD**

**Integrantes:**

1. AVILÉS GIL, ELENA (ELEAVIGIL)
2. AGUILAR MARTÍN, ALBERTO (ALBAGUMAR)
3. GRIJUELA MUÑOZ, MANUEL (MANGRIMUN)
4. OGALLA LLAVERO, MARÍA DE LA PAZ (MARLLAOGA)

domingo, 14 de mayo de 2023

### **Indicaciones generales del trabajo para tener en cuenta:**

- Este documento debe ir acompañado del fichero .BOS completo del proyecto.
- Respetar el formato y actualizar los campos de la tabla de contenidos automática.
- Se debe procurar realizar un sistema lo más completo y realista posible.
- No se deben incluir en el proyecto elementos innecesarios que no tengan un fin práctico.
- El texto entre <> se debe sustituir por el contenido correspondiente.
- Las secciones empezando por texto numerado entre <> pueden replicarse para cubrir el número de elementos que sean necesarios.
- Las discrepancias entre las tablas resumen y los contenidos del proyecto podrán ser motivo de penalización.

### **Evaluación:**

- **NIVEL 1 (APROBADO):**
  - **Contenidos mínimos:** 2 procesos (ejecutables sin errores), 2 actores por proceso, tareas, contratos, formularios, compuertas, variables de negocio y filtros de actor.
  - **Diversidad:** los 2 procesos no deben ser idénticos ni demasiado parecidos. El uso del resto de contenidos debe ser relativamente variado (por ejemplo, usando dos filtros de actor distintos, o dos compuertas distintas, etc.).
  - **Calidad:** se valorará la limpieza en el diagrama, la calidad de la interfaz (formularios), la claridad en el código y en la nomenclatura de variables e identificadores, etc.
  - **Utilidad:** se valorará que los procesos presentados tengan sentido desde un punto de vista práctico, así como las tareas y demás integrantes de estos.
- **NIVEL 2 (NOTABLE):**
  - **Contenidos mínimos:** al menos 1 app y uso de documentos y conectores en los procesos (los 2 del NIVEL 1 o nuevos procesos ejecutables añadidos para este fin).
  - **Diversidad:** debe aumentarse la diversidad de elementos usados con respecto al nivel anterior. Por ejemplo, incluyendo nuevos procesos con nuevos filtros, nuevos tipos de tareas, nuevos eventos, nuevos widgets, etc.
  - **Calidad:** debe mantenerse la calidad a un nivel alto para obtener esta calificación.
  - **Utilidad:** se valorará que los añadidos incluidos tengan sentido desde un punto de vista práctico.
- **NIVEL 3 (SOBRESALIENTE):**
  - **Contenidos mínimos:** añadir elementos avanzados que requieran algún grado de personalización: tareas de script, consultas personalizadas, widgets personalizados, filtros de actor personalizados, conectores personalizados, y otras funcionalidades avanzadas de Bonita Studio que no hayan aparecido en los ejercicios de clase.
  - **Diversidad:** debe aumentarse la diversidad de elementos usados con respecto al nivel anterior. Por ejemplo, incluyendo nuevos procesos en los que aparezcan los elementos personalizados indicados anteriormente.
  - **Calidad:** debe mantenerse la calidad a un nivel muy alto para obtener esta calificación.
  - **Utilidad:** se valorará que los añadidos incluidos tengan sentido desde un punto de vista práctico.

## Contenido

### Descripción general

- Nivel al que se opta

- Elementos destacados incluidos en el proyecto

- Comentarios

- Contexto

- Añadir nuevo cliente

- Creación de sistemas

- Revisión de sistemas Instalados

### Organización

- Descripción

- Estructura

- alberto.aguilar

- Características

### Business Data Model (BDM)

- Descripción

- Características

- Cliente

- Sistema

### Procesos

- NuevoCliente

- Diagrama BPMN

- Características

- nuevoClienteForm

- revisarClienteForm

- llamadaForm

- presupuestoForm

- Revisión

- Diagrama BPMN

- Características

- visValidacion

- CrearSistema

- Diagrama BPMN

- Características

- nuevoSistemaForm

revisarSistemaForm

asignarDateForm

## Descripción general

### Nivel al que se opta

NIVEL: 3

### Elementos destacados incluidos en el proyecto

*Añade únicamente las aportaciones que no aparezca más adelante en las secciones posteriores*

Páginas	carteraClientes: Permite consultar y añadir clientes a la base de datos
Aplicaciones	Cartera de clientes: Aplicación dentro de la cual se encuentra la página anterior
Widgets personalizados	
Scripts (Groovy/Java)	Varios scripts dentro del proceso revisión encargados de procesar los resultados de las consultas a la base de datos externa, los cuales mapean a cada tipo de sistema el número de sistemas de tal tipo que se encuentran instalados
Código Javascript	
Consultas personalizadas	
Filtros de actor personalizados	
Conectores personalizados	-Consulta Odoo: realiza una consulta a la base de datos de Odoo (más elaborado en el proceso Revisión) -Insert Odoo: realiza un insert en la base de datos de Odoo, también dentro del proceso revisión, de una serie de datos procesados durante el proceso
Otros	A partir de un formulario y una plantilla de documento hemos podido enviar un presupuesto relleno al cliente (usamos dos conectores, uno de conversor a documento y otro de correo)

### Comentarios

Hemos hecho la mayoría de requisitos de nivel 2 aunque no todos pero también hemos hecho algunos de los requisitos de nivel 3.

### Contexto

Hemos diseñado este bpm para la empresa “Hércules Seguridad”, una empresa que se dedica a la instalación y mantenimiento de sistemas de videovigilancia, anti-robos y contraincendios.

La implementación se hace con el objetivo de hacer más sencillos los procesos de crear clientes nuevos e instalar nuevos sistemas, así como integrarlo con todo lo realizado en el trabajo de odoo para optimizar los procesos administrativos de la empresa.

## Añadir un nuevo cliente

Este proceso trata de crear un nuevo cliente a partir de un lead. Al crearse el lead, se introducen ciertos datos del usuario, se contacta con él para ver si sigue interesado y, en caso afirmativo, se le envía un presupuesto por correo electrónico. Es una forma muy eficaz de captar nuevos clientes.

## CREACIÓN DE SISTEMAS

Se permiten crear nuevos sistemas por un propio técnico, ser aceptados o rechazados y asignarles una fecha en la que realizar la instalación, así como añadirlo a la lista de sistemas que tiene cada cliente. Para ello simplemente se debe ejecutar el proceso CrearSistema.

## REVISIÓN DE SISTEMAS INSTALADOS

Nuestro sistema permite compatibilizar con nuestro trabajo de teoría, realizado en Odoo. Dentro de Odoo existe una tabla llamada sistemas que almacena todos los sistemas instalados hasta ese momento, así como el tipo de sistema (Sistema anti-incendios, Videovigilancia,...) Cuando se quiera realizar una revisión de estos datos, se inicia el proceso “revisión”, en el que se entra más en detalle más adelante en el trabajo.

Se creará una tarea para un administrador de la empresa en la que le aparecerá una gráfica con las proporciones de cada tipo de sistema en función del número de veces que aparece. Si todo es correcto, el usuario confirma la revisión y el resultado de la misma se almacena dentro de la base de datos de Odoo de nuevo en otra tabla personalizada.

## Organización

### Descripción

Nuestra organización posee la misma estructura que la empresa en la que nos basamos. Existen 3 departamentos: Administración, técnico y ventas. Dentro de los 3 departamentos anteriores existen 2 posibles roles: Jefe de departamento o empleado.

### Estructura

Resumen de la organización

Nombre:  Descripción:

Usuario por defecto:

Grupos de la organización	Roles en la organización	Organización de usuarios
<ul style="list-style-type: none"> <li>Administradores</li> <li>Técnicos</li> <li>Ventas</li> </ul>	<ul style="list-style-type: none"> <li>Jefe departamento</li> <li>Empleado</li> </ul>	<ul style="list-style-type: none"> <li>Alberto Aguilar</li> <li>Elena Avilés</li> <li>Ignacio Escolano</li> <li>Josefa Ramírez</li> <li>Manuel Grijuela</li> <li>Mariano López</li> <li>Mario de las Cuevas</li> <li>Maria de la Paz Llaverio</li> <li>Mónica Galindo</li> <li>Pascual Rodríguez</li> <li>Yasmine Feroe</li> <li>admin</li> </ul>

### Resumen de características

Número de grupos	3
Número de roles	2
Número de usuarios	12
Número de membresías	1

alberto.aguilar

Organización de usuarios

Añadir usuario

Eliminar

Búsqueda...

Alberto Aguilar

Elena Avilés

Ignacio Escolano

Josefa Ramírez

Manuel Grijuela

Mariano López

Mario de las Cuevas

María de la Paz Llaveró

Mónica Galindo

Pascual Rodríguez

Yasmine Feroe

admin

Alberto Aguilar

Nombre de usuario

alberto.aguilar

Contraseña

•••

Nombre

Alberto

Apellido

Aguilar

Título

Sr, Sra...

Profesión

CFO

Responsable

admin

Membresía

+ Añadir

Eliminar

Grupo	Rol
/Administradores	Empleado

Información

Datos profesionales

Datos personales

Personalizado

Contacto

Ubicación

Correo Electrónico

john.doe@company.com

Edificio

A

Oficina

6

Teléfono

456-874-415

Móvil

333-328-3548

Dirección

51 Federal St. Suite 305

Fax

789-123-456

Ciudad

San Francisco

Estado

Estado

Administrar información personalizada

8



## Business Data Model (BDM)

### Descripción

Hemos añadido dos objetos distintos: Cliente y Sistema

Cliente: Guarda los datos de contacto de un cliente y una lista de sistemas asociados.

Sistema: Guarda cada sistema que se ha instalado (Videovigilancia, seguridad perimetral,...)

### Resumen de características

Número de objetos	2
Group id utilizado	

### Cliente

- Ejecución

Modelo | Restricciones | Consultas | Índices | bom.xml

Desplegar | Despliegue limpio | Exportar | Importar | Explorar | Validar

Lista de Objetos de Negocio

Añadir paquete | Añadir objeto | Eliminar

Buscar

com.company.model

- Cliente
- Sistemas

Propiedades de los artefactos de Maven

Group id | com.company.model

Cliente

Descripción

Atributos

+ Añadir | Arriba | Abajo | Eliminar | Buscar

Nombre	Tipo	Múltiple	Requerido
nombre	STRING	<input type="checkbox"/>	<input checked="" type="checkbox"/>
direccion	STRING	<input type="checkbox"/>	<input checked="" type="checkbox"/>
empresa	BOOLEAN	<input type="checkbox"/>	<input checked="" type="checkbox"/>
telefono	STRING	<input type="checkbox"/>	<input type="checkbox"/>
correo	STRING	<input type="checkbox"/>	<input type="checkbox"/>
sistemas	Sistemas	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

- Restricciones

**Lista de Objetos de Negocio**

+

+

🔍

Buscar

▼

com.company.model

🗄️

Cliente

🗄️

Sistemas

**Restricciones únicas para `Cliente`**

+ Añadir

🗑️ Eliminar

🔍

Buscar

Nombre	Atributos
--------	-----------

- Consultas

**Lista de Objetos de Negocio**

+

+

🔍

Buscar

▼

com.company.model

🗄️

Cliente

🗄️

Sistemas

**Consultas para `Cliente`**

+ Añadir

🗑️ Eliminar

+

+

🔍

Buscar

>

Por defecto

Personalizado

- Índices

**Lista de Objetos de Negocio**

+

+

🔍

Buscar

▼

com.company.model

🗄️

Cliente

🗄️

Sistemas

**Índices para `Cliente`**

+ Añadir

🗑️ Eliminar

🔍

Buscar

Nombre	Atributos
--------	-----------

- Resumen de características del objeto de negocio

Número de campos de texto	4
Número de campos numéricos	
Número de campos booleanos	1
Número de campos relacionales	

Número de campos múltiples	1
Número de restricciones	
Número de consultas personalizadas	
Número de índices	

## Sistema

- Ejecución

Desplegar Despliegue limpio Exportar Importar Explorar Validar

Lista de Objetos de Negocio

Añadir paquete Añadir objeto Eliminar

Buscar

com.company.model

- Cliente
- Sistemas

Propiedades de los artefactos de Maven

Group id com.company.model

Cliente

Descripción

Atributos

+ Añadir Arriba Abajo Eliminar Buscar

Nombre	Tipo	Múltiple	Requerido
nombre	STRING	<input type="checkbox"/>	<input checked="" type="checkbox"/>
direccion	STRING	<input type="checkbox"/>	<input type="checkbox"/>
empresa	BOOLEAN	<input type="checkbox"/>	<input checked="" type="checkbox"/>
telefono	STRING	<input type="checkbox"/>	<input type="checkbox"/>
correo	STRING	<input type="checkbox"/>	<input type="checkbox"/>
sistemas	Sistemas	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

- Restricciones

Lista de Objetos de Negocio

com.company.model

- Cliente
- Sistemas

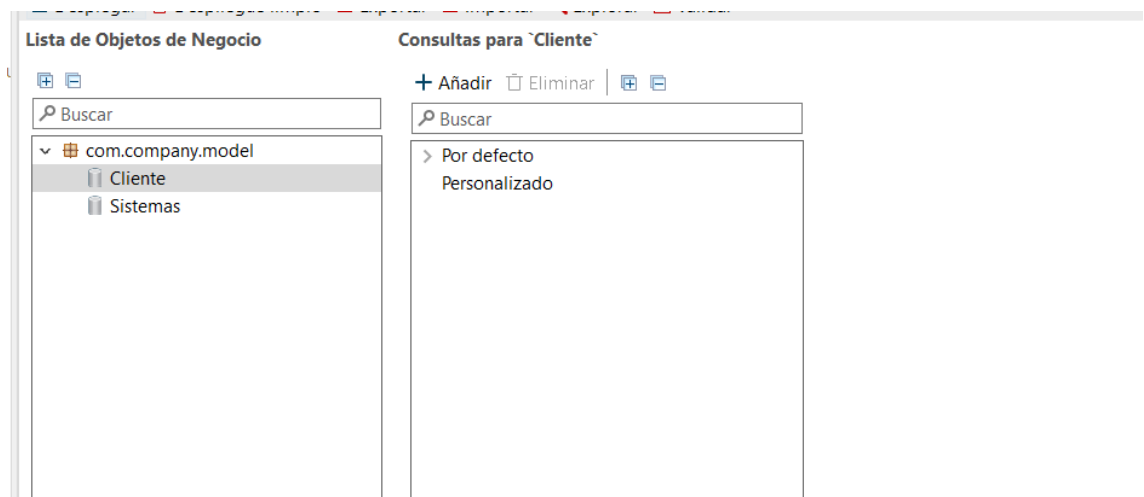
Restricciones únicas para `Cliente`

+ Añadir Eliminar

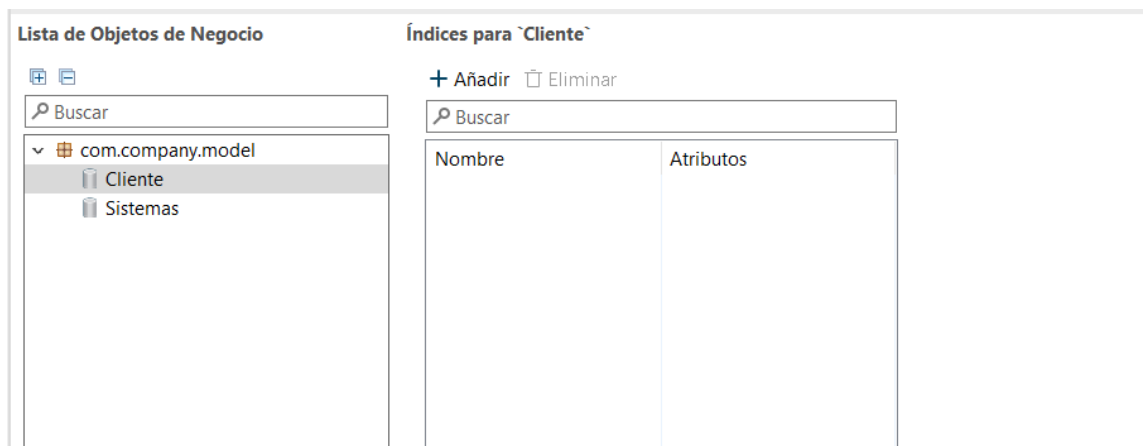
Buscar

Nombre	Atributos

- Consultas



- Índices



- Resumen de características del objeto de negocio

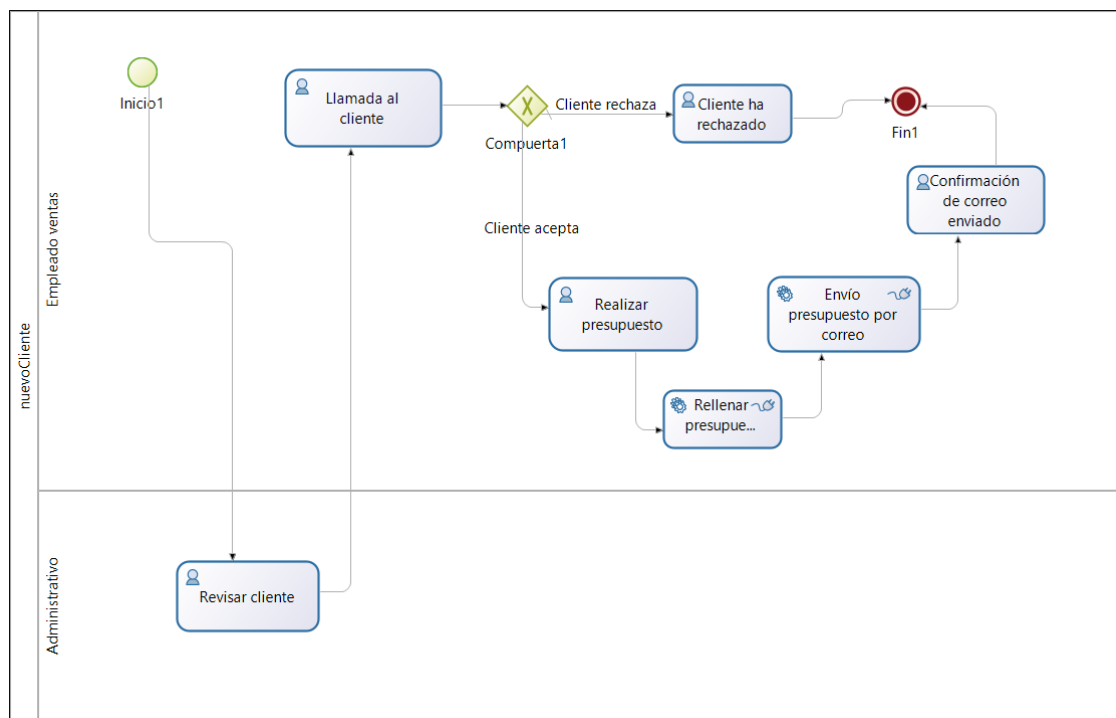
Número de campos de texto	3
Número de campos numéricos	1
Número de campos booleanos	1
Número de campos relacionales	1
Número de campos múltiples	
Número de restricciones	
Número de consultas personalizadas	
Número de índices	

## Procesos

### NuevoCliente

El proceso lo inicia un empleado de ventas que rellena su formulario de instanciación y esos datos le llegan a un administrador que confirma que los datos han llegado correctamente. Después, se produce una tarea en la que el vendedor que ha iniciado el proceso llama al cliente y confirma si quiere continuar con la venta. En caso negativo, se crea una tarea que avisa del rechazo y se acaba el proceso. En caso afirmativo, se crea un formulario con los datos a rellenar para crear un presupuesto. Una vez relleno, se crea una tarea automática en la que se usa un conector para pasar los datos del formulario anterior a una plantilla de word. Por último, se crea una tarea automática para enviar por correo el documento creado anteriormente, y se crea una tarea de confirmación, tras la que el proceso finaliza.

### Diagrama BPMN



### Resumen de características

¿Hay formulario de instanciación?	Si
Número de actores	2
Número de filtros de actor	1
Número de contratos	4


Número de variables de negocio	1
Número de variables de proceso Boolean	0
Número de variables de proceso Texto	2
Número de variables de proceso Fecha	1
Número de variables de proceso Numérica	3
Número de parámetros	0
Número de documentos	2
Número de tareas humanas (con formulario)	5
Número de tareas de servicio	2
Número de tareas de script	0
Número de compuertas XOR	1
Número de compuertas AND	0
Número de compuertas OR	0
Número de eventos temporizadores	0
Número de otros eventos (distintos de inicio, fin o terminación)	0
Número de conectores	2

### nuevoClienteForm

- Tarea asociada: nuevoCliente
- Ejecución

Bonita User Application Mónica Galindo

Tareas Casos Procesos



## Nuevo cliente

**Nombre del cliente \***

Pedro Piqueras

**Dirección**

Plaza Ramiro Fernández, 67

☐ ¿Se trata de una empresa?

**Teléfono**

654 98 29 00

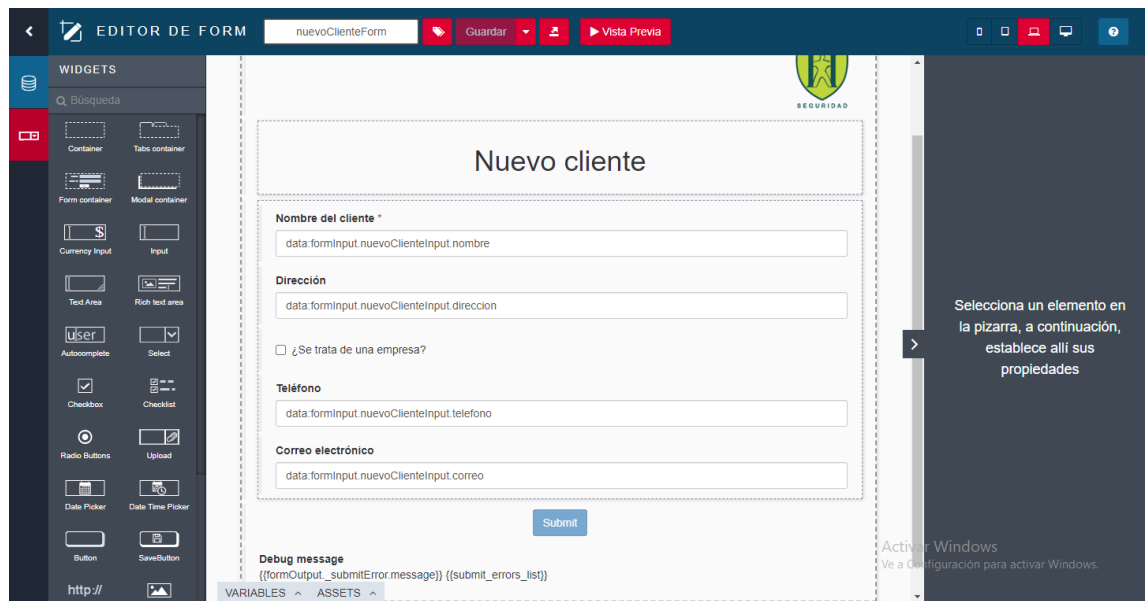
**Correo electrónico**

pedropiqueras@gmail.com

**Enviar**

Activar Windows

- Diseño



- Variables

- formInput

- Tipo: JSON
    - Contenido:
 

```
{
    "nuevoClienteInput" : {
      "nombre" : "",
      "direccion": "",
      "empresa" : false,
      "telefono" : "",
      "correo" : ""
    }
  }
```

- formOutput

- Tipo: Javascript
    - Contenido:
 

```
return {
    nuevoClienteInput: $data.formInput.nuevoClienteInput
  }
```

- Resumen de características del formulario

Número de widgets diferentes	5
Número de variables JSON	1
Número de variables String	0
Número de variables Javascript	2

Número de variables External API	0
Número de variables Business data	0
Número de variables URL Parameter	0
Número de assets usados	0
Número de widgets personalizados usados	0

### revisarForm

- Tarea asociada: Revisar cliente
- Ejecución

admin  

>

Formulario

Comentarios

Vista global





## Revisar cliente

Nuevo Cliente

**Nombre \***

Pedro Piqueras

**Dirección**

Plaza Ramiro Fernández, 67

☐ ¿Es una empresa?

**Teléfono**

654 98 29 00

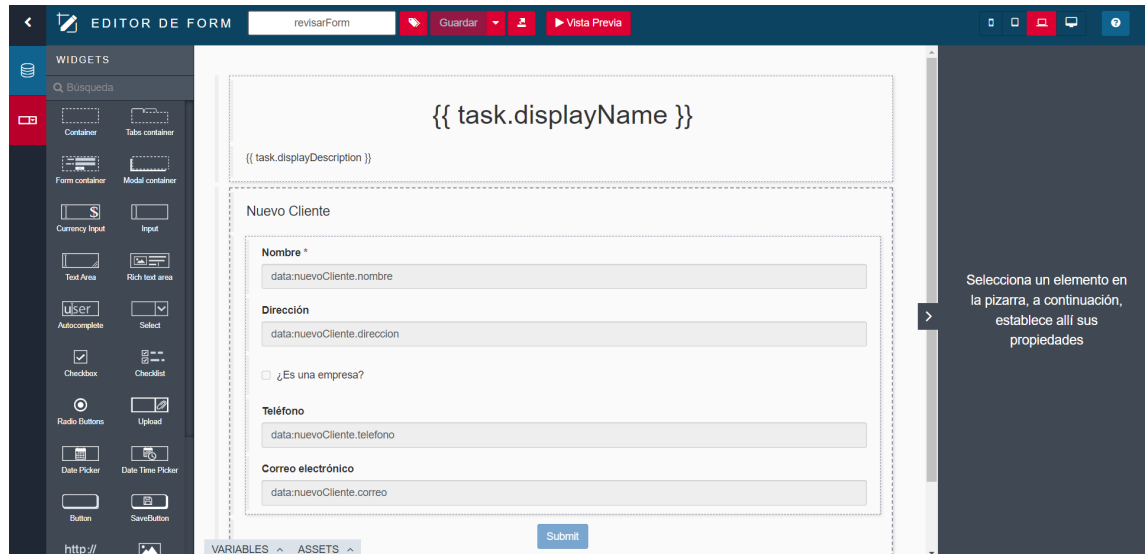
**Correo electrónico**

pedropiqueras@gmail.com

Enviar



- Diseño



- Variables



- nuevoCliente
  - Tipo: External API
  - Contenido: ../{{context.nuevoCliente\_ref.link}}
- nuevoCliente\_sistemas
  - Tipo: External API
  - Contenido: {{nuevoCliente | lazyRef:'sistemas'}}



- Resumen de características del formulario

Número de widgets diferentes	6
Número de variables JSON	0
Número de variables String	0
Número de variables Javascript	2
Número de variables External API	2
Número de variables Business data	0
Número de variables URL Parameter	0
Número de assets usados	0
Número de widgets personalizados usados	0

### LlamadaForm

- Tarea asociada: Llamada al cliente
- Ejecución

Mónica Galindo  

> **Formulario** Comentarios Vista global  

## Llamada al cliente

**Cliente**

**Nombre \***

Pedro Piqueras

**Dirección**

Plaza Ramiro Fernández, 67

☐ ¿Es una empresa?

**Teléfono**

654 98 29 00

**Correo electrónico**

pedropiqueras@gmail.com

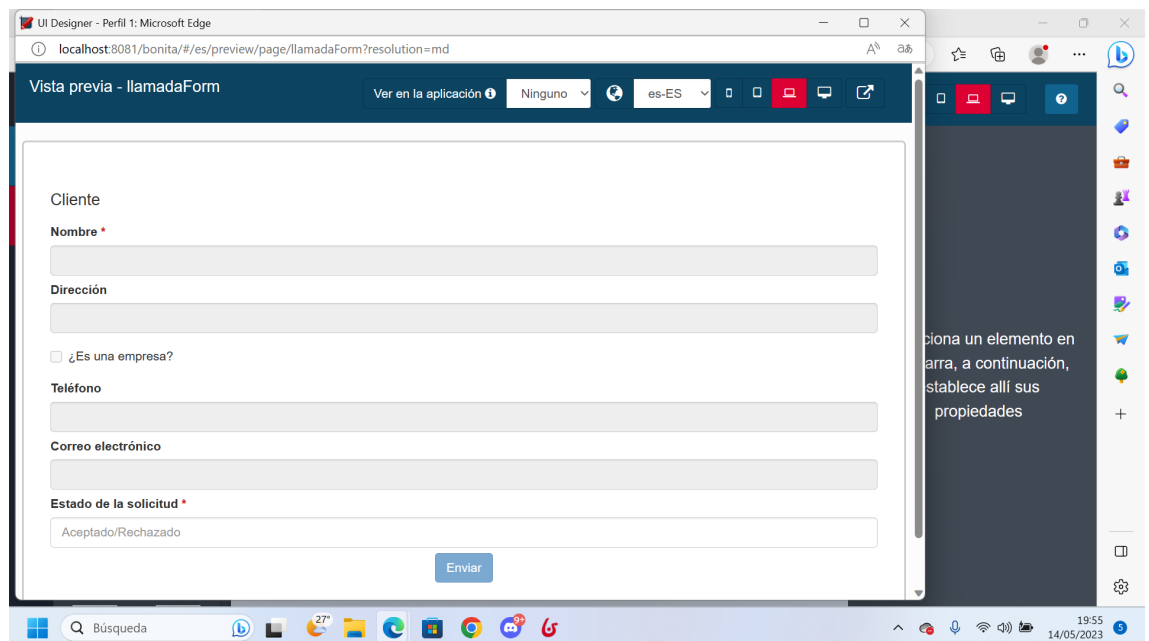
**Estado de la solicitud \***

Aceptado

**Enviar**

Activar Windows  
Vea la configuración para activar Windows.

- Diseño



- Variables
  - context
    - Tipo: External API
    - Contenido: ../API/bpm/userTask/{{taskId}}/context
  - formInput
    - Tipo: JSON
    - Contenido: { "estadoInput" : "" }
  - formOutput
    - Tipo: Javascript
    - Contenido: 

```
if( $data.nuevoCliente && $data.nuevoCliente_sistemas
){ //attach lazy references variables to parent variables
$data.nuevoCliente.sistemas = $data.nuevoCliente_sistemas;
return { //map nuevoCliente variable to expected task contract input
nuevoClienteInput: { nombre: $data.nuevoCliente.nombre !==
undefined ? $data.nuevoCliente.nombre : null, direccion:
$data.nuevoCliente.direccion !== undefined ?
$data.nuevoCliente.direccion : null, empresa:
$data.nuevoCliente.empresa !== undefined ?
$data.nuevoCliente.empresa : null, telefono:
$data.nuevoCliente.telefono !== undefined ?
$data.nuevoCliente.telefono : null, correo:
$data.nuevoCliente.correo !== undefined ?
$data.nuevoCliente.correo : null }, estadoInput:
$data.formInput.estadoInput } }
```
  - nuevoCliente
    - Tipo: External API
    - Contenido: ../{{context.nuevoCliente\_ref.link}}
  - nuevoCliente\_sistemas
    - Tipo: External API
    - Contenido: {{nuevoCliente|lazyRef:'sistemas'}}


#### Resumen de características del formulario



Número de widgets diferentes	3
Número de variables JSON	1
Número de variables String	
Número de variables Javascript	2
Número de variables External API	4
Número de variables Business data	

Número de variables URL Parameter	1
Número de assets usados	
Número de widgets personalizados usados	

### presupuestoForm

- Tarea asociada: Realizar presupuesto
- Ejecución

Mónica Galindo  

> Formulario Comentarios Vista global  

## Presupuesto Hércules Seguridad

**Cliente**

**Nombre \***

Pedro Piqueras

**Dirección**

Plaza Ramiro Fernández, 67

☐ ¿Es una empresa?

**Teléfono**

654 98 29 00


**Correo electrónico**

pedropiqueras@gmail.com

**Servicio \***

Instalación antirrobo

**Fecha límite de pago**

13/06/2023 

**Precio sin IVA \***

312,44€

**IVA \***

21

Activar Windows

ve a Configuración para activar Windows.

Enviar

- Diseño



- Variables
  - nuevoCliente
    - Tipo: External API
    - Contenido: ../{{context.nuevoCliente\_ref.link}}
  - nuevoCliente\_sistemas
    - Tipo: External API
    - Contenido: {{nuevoCliente | lazyRef:'sistemas'}}
- Resumen de características del formulario

Número de widgets diferentes	10
Número de variables JSON	0
Número de variables String	0
Número de variables Javascript	2
Número de variables External API	4
Número de variables Business data	0
Número de variables URL Parameter	0
Número de assets usados	0
Número de widgets personalizados usados	0

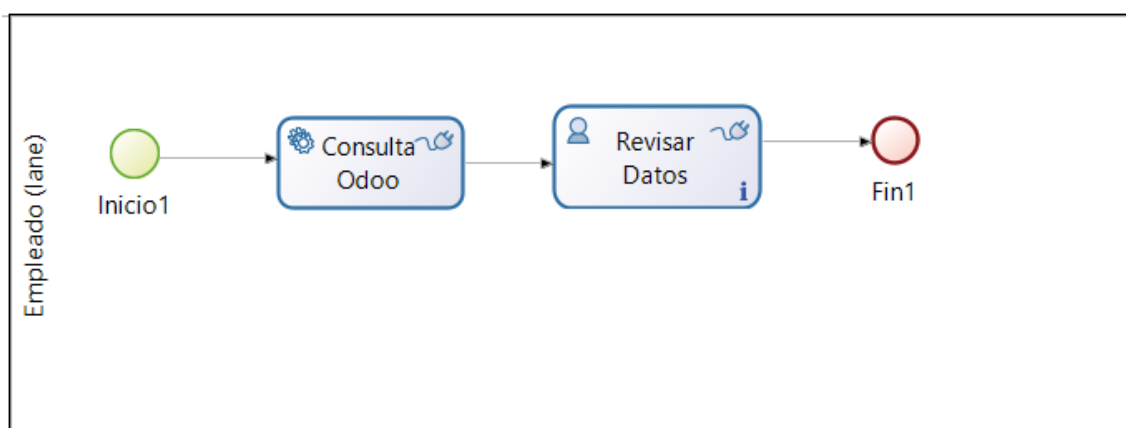
## REVISIÓN

El objetivo de este proceso consiste en realizar una revisión de los sistemas que tenemos instalados y que permita conectar con una base de datos externa. Para ello vamos a trabajar con la base de datos de Odoo. Al iniciar el proceso, una tarea de servicio realizará una consulta a la base de datos de Odoo en cuanto a los sistemas que tenemos instalados (tabla similar a la del objeto Sistema de este trabajo) Posteriormente mediante una operación dentro de esta tarea

se procesarán los datos y se filtrarán, creando una lista que relacione cada tipo de sistema con el número de sistemas de su tipo existentes en Odoo.

Posteriormente en una tarea humana aparecerá un gráfico mostrando dicha proporción calculada, para que el usuario pueda verlo y confirme que todo está correcto. Al final de esta tarea se realizará un INSERT en la base de datos de Odoo dentro de una tabla llamada “revisiones” que almacena el resultado de las revisiones que se han realizado con esta tarea.

### Diagrama BPMN



### Resumen de características

NOTA: EXISTEN 3 VARIABLES DE PROCESO DE TIPO java.util.list

¿Hay formulario de instanciación?	No
Número de actores	1
Número de filtros de actor	0
Número de contratos	0
Número de variables de negocio	0
Número de variables de proceso Boolean	0
Número de variables de proceso Texto	1
Número de variables de proceso Fecha	0
Número de variables de proceso Numérica	0
Número de parámetros	0
Número de documentos	0
Número de tareas humanas (con formulario)	1
Número de tareas de servicio	1
Número de tareas de script	0
Número de compuertas XOR	0
Número de compuertas AND	0
Número de compuertas OR	0
Número de eventos temporizadores	0
Número de otros eventos (distintos de inicio, fin o terminación)	0
Número de conectores	2

## visValidacion

- Tarea asociada: Consulta Odoo
- Ejecución



- Diseño

**EDITOR DE FORM** visValidacion Guardar Vista Previa

WIDGETS

Busqueda

Container Tabs container

Form container Modal container

Currency Input Input

Text Area Rich text area

User Autocomplete Select

Checkbox Checklist

Radio Buttons Upload

Date Picker Date Time Picker

Button SaveButton

http://

CHART

Clases CSS

Lista separada por espacios

Oculto

no si

Tipo

Pie

Datos

Lista separada por comas o variable

listaNumeroSistemas

Etiquetas

listaTipos

Leyenda oculta

no si

Colores

Por ejemplo #FF0000, #0000FF

Opciones Avanzadas

Ver para más detalles de las opciones de cada tipo de gráfico. Especifica una estructura JSON correspondiente a las opciones que desees

Nombre	Valor	Tipo
context	../API/bpm/userTask/{{taskId}}/context	External API
formOutput	return {}	Javascript expression
listaNumeroSistemasAPI	../API/bpm/activity/Variable/{{taskId}}/listaNumeroSistemas	External API
listaTipos	return \$data.tiposAPI.value;	Javascript expression

- Variables
  - listaNumeroSistemasAPI
    - Tipo: External API

- Contenido: Llamada a la API para conseguir una de las variables de proceso `../API/bpm/activityVariable/{{taskId}}/listaNumeroSistemas`
- listaNumeroSistemas
  - Tipo: Javascript Expression
  - Contenido: Valor resultante de la llamada a la API  
`return $data.listaNumeroSistemasAPI.value;`
- listaTiposAPI
  - Tipo: External API
  - Contenido: Llamada a la API para conseguir una de las variables de proceso `../API/bpm/activityVariable/{{taskId}}/listaTipos`
- listaTipos
  - Tipo: Javascript Expression
  - Contenido: Valor resultante de la llamada a la API  
`return $data.listaTipos.value;`
- Resumen de características del formulario

Número de widgets diferentes	3
Número de variables JSON	0
Número de variables String	0
Número de variables Javascript	2
Número de variables External API	2
Número de variables Business data	0
Número de variables URL Parameter	0
Número de assets usados	0
Número de widgets personalizados usados	0

## CrearSistema

Este proceso se debe ejecutar por un técnico. Además de este, también formará parte del proceso un administrador. Por ello, se han creado dos swimlanes, cada uno con un actor definido como técnico y como administrador. Cabe destacar que no se han utilizado filtros de actor porque no es necesario que ambas tareas de administrador las ejecute el mismo usuario.

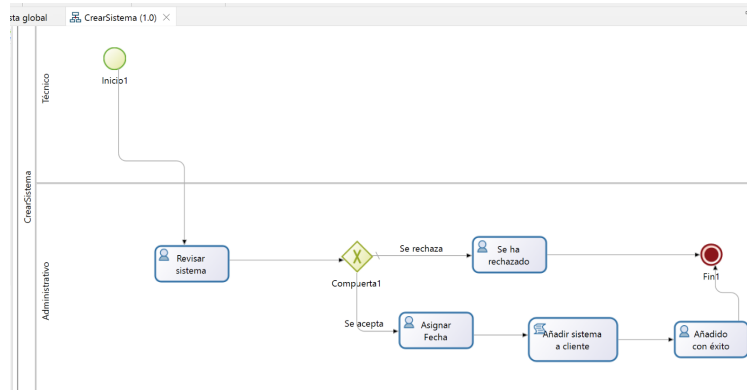
El proceso comienza siendo ejecutado por un técnico, el cual rellena un formulario de instanciación para crear un sistema (cabe destacar que el campo de “técnico” se rellena automáticamente con el técnico que inicia el proceso y que el campo de “fechaInstalacion” queda sin rellenar de momento).

Una vez se crea este sistema, se genera una tarea para un administrador en la cual se puede aceptar o rechazar. Si se acepta, el campo “aceptado” cambia a “true”.

El valor de dicho campo entra en la compuerta xor. Si es “false”, se crea una tarea indicando que el sistema ha sido rechazado y se finaliza el proceso. Si es “true”, se genera una tarea en la cual el administrador asigna una fecha para que se realice la instalación del sistema, seguida de una tarea de script que añade el sistema al cliente especificado y de una final que indica que el sistema ha sido añadido, después de esto hay otro fin.



## Diagrama BPMN



## Resumen de características

¿Hay formulario de instanciación?	Si
Número de actores	2
Número de filtros de actor	0
Número de contratos	1
Número de variables de negocio	2
Número de variables de proceso Boolean	0
Número de variables de proceso Texto	0
Número de variables de proceso Fecha	0
Número de variables de proceso Numérica	0
Número de parámetros	0
Número de documentos	0
Número de tareas humanas (con formulario)	4
Número de tareas de servicio	0
Número de tareas de script	1
Número de compuertas XOR	1
Número de compuertas AND	0
Número de compuertas OR	0
Número de eventos temporizadores	0
Número de otros eventos (distintos de inicio, fin o terminación)	0
Número de conectores	0

## nuevoSistemaForm

- Tarea asociada: "CrearSistema" (es de instanciación)
- Ejecución

Bonita User Application Pascual Rodríguez

Tareas Casos Procesos

Sistema

**Sistema \***  
Instalación anti-robo

**Cliente \***  
Pedro Piqueras

**Precio \***  
313,21€

☒ Garantía

Enviar

• Diseño

ORM nuevoSistemaForm Guardar Vista Previa

Sistema

**Sistema \***  
data:formInput.sistemaInput.sistema

**Fecha de Instalación**  
data:formInput.sistemaInput.fechaInstalacion Today

**Técnico \***  
data:formInput.sistemaInput.tecnico

VARIABLES ASSETS

Crear una nueva variable Búsqueda

Nombre	Valor	Tipo
Cliente	find [com.company.model.Cliente]	Business data
formInput	{ "sistemaInput" : { "sistema" : "", "fechaInstalacion" : null, "tecnico" : "", "c...	JSON
formOutput	\$data.formInput.sistemaInput.tecnico=\$data.tecnico_ini_username return...	Javascript expression

ORM nuevoSistemaForm Guardar Vista Previa

data:formInput.sistemaInput.tecnico

**Cliente \***

**Precio \***  
data:formInput.sistemaInput.precio

☐ Garantía

☐ Aceptado

Submit

VARIABLES ASSETS

Crear una nueva variable Búsqueda

Nombre	Valor	Tipo
Cliente	find [com.company.model.Cliente]	Business data
formInput	{ "sistemaInput" : { "sistema" : "", "fechaInstalacion" : null, "tecnico" : "", "c...	JSON
formOutput	\$data.formInput.sistemaInput.tecnico=\$data.tecnico_ini_username return...	Javascript expression

The screenshot shows the Bonita BPM Form Designer interface. At the top, there's a header bar with the title 'nuevoSistemaForm' and buttons for 'Guardar' (Save), 'Vista Previa' (Preview), and other icons. Below the header, the form design area contains a text input field labeled 'Precio \*' with the value 'data:formInput.sistemaInput.precio'. There are also two checkboxes labeled 'Garantía' and 'Aceptado', and a 'Submit' button. A 'Debug message' section shows the output of the submit action. At the bottom, there's a 'VARIABLES' tab with a search bar and a table of variables.

Nombre	Valor	Tipo
Cliente	find [com.company.model.Cliente]	Business data
formInput	{ "sistemaInput" : { "sistema" : "", "fechaInstalacion" : null, "tecnico" : "", "c..."	JSON
formOutput	\$data.formInput.sistemaInput.tecnico=\$data.tecnico_ini_username return...	Javascript expression

- Variables
  - Cliente
    - Tipo: Business data
    - Contenido: find [com.company.model.Cliente]
  - formInput
    - Tipo: JSON
    - Contenido: {
 

```
"sistemaInput" : {
                "sistema" : "",
                "fechaInstalacion" : null,
                "tecnico" : "",
                "cliente" : "",
                "precio" : 0,
                "garantia" : false,
                "aceptado" : false
              }
            }
```
  - formOutput
    - Tipo: Javascript Expression
    - Contenido: \$data.formInput.sistemaInput.tecnico=\$data.tecnico\_ini\_username

```
return {
  sistemaInput: $data.formInput.sistemaInput
```

- ```

    }
  ○ submit_errors_list
    ■ Tipo: Javascript expression
    ■ Contenido: if($data.formOutput && $data.formOutput._submitError
      && $data.formOutput._submitError.explanations){
      const liElements =
      $data.formOutput._submitError.explanations
      .filter(cause => cause !== null)
      .map(cause => "<li>" + cause + "</li>")
      .join("");
      if(liElements){
        return "<ul>" + liElements + "</ul>";
      }
    }
  ○ tecnico_ini
    ■ Tipo: External API
    ■ Contenido: ../API/system/session/1
  ○ tecnico_ini_username
    ■ Tipo: Javascript expression
    ■ Contenido: return $data.tecnico_ini.user_name;

```

- Resumen de características del formulario

|                                         |   |
|-----------------------------------------|---|
| Número de widgets diferentes            | 4 |
| Número de variables JSON                | 1 |
| Número de variables String              | 0 |
| Número de variables Javascript          | 3 |
| Número de variables External API        | 1 |
| Número de variables Business data       | 1 |
| Número de variables URL Parameter       | 0 |
| Número de assets usados                 | 0 |
| Número de widgets personalizados usados | 0 |

### revisarSistemaForm

- Tarea asociada: RevisarSistema

- Ejecución

admin  

> Formulario Comentarios Vista global  

## Revisar sistema

Sistema

**Sistema \***

Instalación anti-robo

**Técnico \***

pascual.rodriguez

**Cliente \***

Pedro Piqueras

**Precio \***

313,21€

☒ Garantía

☐ Aceptado

**Enviar**  Activar Windows  
Vea la Configuración para activar Windows.

- Diseño

{{ task.displayName }}


{{ task.displayDescription }}

Sistema


**Sistema \***

data:sistema.sistema

Fecha de Instalación

data:sistema.fechaInstalacion Today 

VARIABLES ▾ ASSETS

Crear una nueva variable 

Búsqueda

| Nombre ▾   | Valor                                                  | Tipo ▾                 |
|------------|--------------------------------------------------------|------------------------|
| context    | ../API/bpm/userTask/{{taskId}}/context                 | External API           |
| formOutput | if( \$data.sistema ){ return { //map sistema variab... | Javascript expressi... |
| sistema    | ../{{context.sistema_ref.link}}                        | External API           |

FORM revisarSistemaForm Guardar Vista Previa

**Sistema \***  
data:sistema.sistema

**Fecha de Instalación**  
data:sistema.fechaInstalacion Today

**Técnico \***  
data:sistema.tecnico

**Cliente \***  
data:sistema.cliente

VARIABLES ASSETS

Crear una nueva variable Búsqueda

| Nombre     | Valor                                                  | Tipo                   |
|------------|--------------------------------------------------------|------------------------|
| context    | ../API/bpm/userTask/{{taskId}}/context                 | External API           |
| formOutput | if( \$data.sistema ){ return { //map sistema variab... | Javascript expressi... |
| sistema    | ../{{context.sistema_ref.link}}                        | External API           |

FORM revisarSistemaForm Guardar Vista Previa

data:sistema.cliente

**Precio \***  
data:sistema.precio

☐ Garantía

☐ Aceptado

Submit

Debug message  
(formOutput, submitErrorMessage) {{submit\_errors\_list}}

VARIABLES ASSETS

Crear una nueva variable Búsqueda

| Nombre     | Valor                                                  | Tipo                   |
|------------|--------------------------------------------------------|------------------------|
| context    | ../API/bpm/userTask/{{taskId}}/context                 | External API           |
| formOutput | if( \$data.sistema ){ return { //map sistema variab... | Javascript expressi... |
| sistema    | ../{{context.sistema_ref.link}}                        | External API           |

- Variables
  - context
    - Tipo: External API
    - Contenido: ../API/bpm/userTask/{{taskId}}/context
  - sistema
    - Tipo: External API
    - Contenido: ../{{context.sistema\_ref.link}}
  - task
    - Tipo: External API
    - Contenido: ../API/bpm/userTask/{{taskId}}

- `taskId`
  - Tipo:URL parameter
  - Contenido: id

- Resumen de características del formulario

|                                         |   |
|-----------------------------------------|---|
| Número de widgets diferentes            | 4 |
| Número de variables JSON                | 0 |
| Número de variables String              | 0 |
| Número de variables Javascript          | 2 |
| Número de variables External API        | 3 |
| Número de variables Business data       | 0 |
| Número de variables URL Parameter       | 1 |
| Número de assets usados                 | 0 |
| Número de widgets personalizados usados | 0 |

### asignarDateForm

- Tarea asociada: Asignar fecha
- Ejecución

admin  

>

Formulario
Comentarios
Vista global




## Asignar Fecha

Sistema

**Sistema \***

Instalación anti-robo

**Fecha de Instalación**

14/06/2023

Hoy 

**Técnico \***

pascual.rodriguez

**Cliente \***

Pedro Piqueras

**Precio \***

313,21€

☒ Garantía

Activar Windows

Vea la Configuración para activar Windows.

Enviar

- Diseño

FORM asignarDateForm Guardar Vista Previa

{{ task.displayName }}

{{ task.displayDescription }}

Fila

Sistema

Sistema \*

data:sistema.sistema

Fecha de Instalación

data:sistema.fechaInstalacion Today

VARIABLES ASSETS

Crear una nueva variable Búsqueda

| Nombre     | Valor                                                  | Tipo                   |
|------------|--------------------------------------------------------|------------------------|
| context    | ../API/bpm/userTask/{{taskId}}/context                 | External API           |
| formOutput | if( \$data.sistema ){ return { //map sistema variab... | Javascript expressi... |
| sistema    | ../{{context.sistema_ref.link}}                        | External API           |

FORM asignarDateForm Guardar Vista Previa

Fecha de Instalación

data:sistema.fechaInstalacion Today

Técnico \*

data:sistema.tecnico

Cliente \*

data:sistema.cliente

Precio \*

data:sistema.precio

VARIABLES ASSETS

Crear una nueva variable Búsqueda

| Nombre     | Valor                                                  | Tipo                   |
|------------|--------------------------------------------------------|------------------------|
| context    | ../API/bpm/userTask/{{taskId}}/context                 | External API           |
| formOutput | if( \$data.sistema ){ return { //map sistema variab... | Javascript expressi... |
| sistema    | ../{{context.sistema_ref.link}}                        | External API           |



- Variables
  - context
    - Tipo: External API
    - Contenido: ../API/bpm/userTask/{{taskId}}/context
  - sistema
    - Tipo: External API
    - Contenido: ../{{context.sistema\_ref.link}}
  - task
    - Tipo: External API
    - Contenido: ../API/bpm/userTask/{{taskId}}
  - taskId
    - Tipo: URL parameter
    - Contenido: id

- Resumen de características del formulario

|                                         |   |
|-----------------------------------------|---|
| Número de widgets diferentes            | 4 |
| Número de variables JSON                | 0 |
| Número de variables String              | 0 |
| Número de variables Javascript          | 2 |
| Número de variables External API        | 3 |
| Número de variables Business data       | 0 |
| Número de variables URL Parameter       | 1 |
| Número de assets usados                 | 0 |
| Número de widgets personalizados usados | 0 |