



# Camunda: Propiedades de Ejecución

Gestión de Procesos y Servicios

# Introducción

# Camunda BPMS



# Introducción

## Conceptos básicos

# Expresiones

- Camunda BPM soporta el Unified Expression Language (EL). Pueden ser utilizadas en:
  - Tareas
  - Eventos
  - Condiciones en gateways
  - ...

## ¿Qué devuelven las siguientes expresiones?

- `${rfcName}`
- `${decision == "approved"}`
- `${!resolved}`
- `${amount>1000}`





Introducción  
Conceptos básicos  
**Tareas Automáticas**

# Tareas automáticas

- Service task
- Send task
- Business rule task
- Receive task



# Tareas automáticas

- **Service task** 
- **Send task** 
- **Business rule task** 
- **Receive task** 

## **Tareas automáticas - Envío de variables**

# Tareas automáticas

The screenshot displays the Camunda BPMN editor interface. On the left, a toolbar contains various BPMN symbols. The main workspace shows a BPMN diagram with a start event, a task named 'Charge Credit Card' (highlighted with a blue dashed border), and an end event. The task is connected to the start event by an arrow labeled 'Payment Retrieval Requested' and to the end event by an arrow labeled 'Payment Received'. A blue callout box with the text 'If it's not already visible, click on the **Properties Panel** tab' points to the 'Properties Panel' tab in the right-hand sidebar. The sidebar is currently showing the 'General' tab for 'serviceTask1'. The 'General' tab contains the following fields:

- Id:** serviceTask1
- Name:** Charge Credit Card
- Details:**
  - Implementation:** External
  - Topic:** charge-card
- External Task Configuration:**
  - Task Priority:** (empty field)
- Asynchronous Continuations:**
  - ☐ Asynchronous Before
  - ☐ Asynchronous After
- Documentation:** (empty field)

# Tareas automáticas (Class)

```
import java.util.logging.Logger;

public class ChargeCardWorker {
    private final static Logger LOGGER = Logger.getLogger(ChargeCardWorker.class.getName());

    public static void main(String[] args) {
        ExternalTaskClient client = ExternalTaskClient.create()
            .baseUrl("http://localhost:8080/engine-rest")
            .asyncResponseTimeout(10000) // long polling timeout
            .build();

        // subscribe to an external task topic as specified in the process
        client.subscribe("charge-card")
            .lockDuration(1000) // the default lock duration is 20 seconds, but you can override this
            .handler((externalTask, externalTaskService) -> {
                // Put your business logic here

                // Get a process variable
                String item = (String) externalTask.getVariable("item");
                Long amount = (Long) externalTask.getVariable("amount");

                LOGGER.info("Charging credit card with an amount of '" + amount + "'€ for the item '" + item + "'...");

                try {
                    Desktop.getDesktop().browse(new URI("https://docs.camunda.org/get-started/quick-start/complete"));
                } catch (Exception e) {
                    e.printStackTrace();
                }

                // Complete the task
                externalTaskService.complete(externalTask);
            })
            .open();
    }
}
```

## Tareas automáticas

- Para acceder valores de variables del proceso se utiliza:

```
String item = (String) externalTask.getVariable("ítem")
```

- Este código lo que hace es **acceder** a la variable *item* del proceso de tipo *string*

- Para guardar variables en el proceso se utiliza:

```
Map<String, Object> variables = new HashMap<String, Object>();  
variables.put("result", "Approved");
```

```
// Complete the task
```

```
externalTaskService.complete(externalTask, variables);
```

- Este código **guarda** en el proceso una variable *result* de tipo *string* con valor “Approved”.

# Tareas automáticas – Envío de variables (I)

```
// subscribe to an external task topic as specified in the process
client.subscribe("charge-card-var")
    .lockDuration(1000) // the default lock duration is 20 seconds, but you can override this
    .handler((externalTask, externalTaskService) -> {
        // Put your business logic here

        // Get a process variable
        String item = (String) externalTask.getVariable("item");
        Long amount = (Long) externalTask.getVariable("amount");

        /*****/
        Map<String, Object> variables = new HashMap<String, Object>();

        if(amount < 100) {
            System.out.println("Valor menor que 100");
            variables.put("result", "Approved");
        } else {
            System.out.println("Valor mayor que 100");
            variables.put("result", "Not approved");
        }

        LOGGER.info("Charging credit card with an amount of '" + amount + "'€ for the item '" + item + "'...");

        // Complete the task
        //externalTaskService.complete(externalTask);
        externalTaskService.complete(externalTask, variables);
    })
    .open();
```

# Tareas automáticas – Envío de variables (II)

Configura la tarea de servicio como **External** y topic “**charge-card-var**”

The screenshot displays the Camunda Platform BPMN editor interface. The main workspace shows a BPMN diagram for a process named "payment\_get.bpmn". The diagram consists of the following elements:

- Start Event: "Payment Retrieval Requested" (thin circle).
- Task: "Charge Credit Card" (rounded rectangle with a gear icon, highlighted with a blue dashed border).
- Intermediate Event: A complex event with multiple outgoing flows, including a timer event (circle with a clock face) and a message event (circle with a speech bubble).
- Task: "Form Result" (rounded rectangle with a document icon).
- End Event: "Payment Received" (thick circle).

The "Properties Panel" on the right is configured for the selected "Charge Credit Card" task:

- Activity\_Service** (Tab):
  - General** (Sub-tab):
    - Id**: Activity\_Service
    - Name**: Charge Credit Card
  - Details** (Sub-tab):
    - Implementation**: External
    - Topic**: charge-card-var
  - External Task Configuration** (Sub-tab):
    - Task Priority**: (Empty field)

The bottom status bar indicates the user is in the "XML" view of the "Camunda Platform" version 4.11.1.

# Tareas automáticas – Envío de variables (III)

Configura la tarea de usuario, utilizando un formulario externo con el tipo “**Embedded or External Task Forms**” donde se va a recibir el valor de “**variables**”.

Asigna el formulario al usuario “**demo**”.

Utiliza la clave: “**camunda-forms:deployment:confirmation.form**”

Cree un formulario utilizando **Camunda Modeler**, que se muestre como la siguiente imagen.

The screenshot displays the Camunda Modeler interface with a form titled "confirmation.form". The form is designed for displaying service task results. It features a "FORM ELEMENTS LIBRARY" on the left with various input types like Text Field, Number, Checkbox, Radio, Select, Text, and Button. The main canvas shows the form content: a heading "Este es un formulario creado desde Camunda Modeler para GPS-2021", a label "El resultado de la información proveniente de la tarea de servicio es:", a label "Valor de la variable:" followed by a large text input field, and a "Finalizar" button. On the right, a properties panel shows the form's ID as "Form\_06drprl". The bottom status bar indicates "JSON No platform selected" and the version "4.11.1".

payment\_get.bpmn ○ confirmation.form ×

**FORM ELEMENTS LIBRARY**

- Text Field
- Number
- ☒ Checkbox
- ☐ Radio
- Select
- ABC** Text
- Button

**Este es un formulario creado desde Camunda Modeler para GPS-2021**

El resultado de la información proveniente de la tarea de servicio es:

Valor de la variable:

**Finalizar**

**ABC** FORM Form\_06drprl

General

Id

JSON No platform selected ▼ Log 4.11.1 6



# Tareas automáticas – Envío de variables (III)

¡¡Recuerda!!

En el formulario hay que especificar dónde se va a mostrar el valor de la variable “**result**”.

The screenshot shows the Camunda Modeler interface with a form titled "confirmation.form". The form contains a text field labeled "Valor de la variable:" and a "Finalizar" button. The "FORM ELEMENTS LIBRARY" on the left lists various form elements: Text Field, Number, Checkbox, Radio, Select, Text, and Button. The right-hand configuration panel for the selected "TEXT FIELD" shows the "Key" property set to "result", which is highlighted with a red box. The "Field Label" is "Valor de la variable:". The status bar at the bottom indicates "JSON No platform selected" and the version "4.11.1".

Camunda Modeler

File Edit Window Help

send\_mails.bpmn x confirmation.form x

FORM ELEMENTS LIBRARY

- Text Field
- Number
- Checkbox
- Radio
- Select
- Text
- Button

Este es un formulario creado desde Camunda Modeler para GPS-2021

El resultado de la información proveniente de la tarea de servicio es:

Valor de la variable:

Finalizar

TEXT FIELD  
Valor de la variable:

General

Field Label  
Valor de la variable:

Field Description

Key  
result

Maps to a process variable.

Validation

Required

Minimum Length

Maximum Length

JSON No platform selected

Log 4.11.1

# Tareas automáticas – Envío de variables (IV)

Despliega el proceso. **No olvides adjuntar el archivo de formulario!**

**Deploy Diagram to Camunda Platform** [X]

Deployment Name  
Payment Retrieval T10

Tenant ID  
Optional

**Endpoint Configuration**

REST Endpoint  
http://localhost:8090/engine-rest

**Include additional files**

Select files

confirmation.form [X]

Cancel Deploy

# Tareas automáticas – Envío de variables (IV)

Inicia un nuevo proceso, definiendo las variables necesarias: *item* y *amount*

## Start process

You can set variables, using a generic form, by clicking the "Add a variable" link below.

**Business Key**

Add a varia... +	Name	Type	Value
Remo... x	<input type="text" value="item"/>	<input type="text" value="String"/>	<input type="text" value="Item_GPS_2021"/>
Remo... x	<input type="text" value="amount"/>	<input type="text" value="Long"/>	<input type="text" value="125"/>

[Back](#)[Close](#)[Start](#)

```
ChargeCardWorker [Java Application] C:\Users\bestrada\Desktop\eclipse GPS2021-22\eclipse\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64_16.0.2.v20210721
405 [main] INFO org.camunda.bpm.client - TASK/CLIENT-01026 Discovered data format provider: org.camunda.bpm.client.variable.impl.1
406 [main] INFO org.camunda.bpm.client - TASK/CLIENT-01025 Discovered data format: org.camunda.bpm.client.variable.impl.format.js
406 [main] INFO org.camunda.bpm.client - TASK/CLIENT-01026 Discovered data format provider: org.camunda.bpm.client.variable.impl.1
406 [main] INFO org.camunda.bpm.client - TASK/CLIENT-01025 Discovered data format: org.camunda.bpm.client.variable.impl.format.ser
407 [main] INFO org.camunda.bpm.client - TASK/CLIENT-01026 Discovered data format provider: org.camunda.bpm.client.variable.impl.1
427 [main] INFO org.camunda.bpm.client - TASK/CLIENT-01025 Discovered data format: org.camunda.bpm.client.variable.impl.format.xml
> El valor proporcionado es mayor o igual que 100.
dic 13, 2021 6:35:28 P. M. org.camunda.bpm.getstarted.chargecard.ChargeCardWorker lambda$2
INFO: Charging credit card with::
- Amount: '125'
- Item: 'Item_GPS_2021'
```

# Tareas automáticas – Envío de variables (V)

Inicia un nuevo proceso, definiendo las variables necesarias: *item* y *amount*

**Camunda Tasklist**

Create a filter + | Created ▾ + | 11 | Filter Tasks | Add Comment +

**All Tasks (11)**

- Form Result**  
Payment Retrieval T10 | Demo Demo | 50 | Created a minute ago
- Analizar solicitud**  
Prueba | 50 | Created 4 days ago
- Analizar solicitud**  
Prueba | 50 | Created 4 days ago
- Analizar solicitud**  
Prueba | Demo Demo | 50 | Created 4 days ago
- Approve Payment**  
Payment Retrieval | Demo Demo | 50 | Created 4 days ago
- Assign Reviewer**  
Review Invoice | Demo Demo | 50 | Created 10 days ago
- Prepare Bank Transfer**

**Form Result**  
Payment Retrieval T10

Set follow-up date | Set due date | Add groups | Demo Demo

Form | History | Diagram | Description

Este es un formulario creado desde Camunda Modeler para GPS-2021

El resultado de la información proveniente de la tarea de servicio es:

Valor de la variable:

Not approved

Finalizar

result: Not approved

Save | Complete

Date and Time displayed in local timezone: Europe/Madrid | Powered by Camunda Platform / v7.16.0

## **Tareas automáticas - Envío de mensajes**

# Envío de Mensajes

El envío de mensajes se podrá hacer de varias formas:

— **Tarea Automática:**

- Mail Automático: Cuando podemos automatizar el contenido del email.
- Mediante consumo de Servicio: Cuando el destinatario del correo provee de una API para comunicación (No lo vamos a ver en la asignatura)

— **Evento de mensaje:** solo tiene sentido cuando vamos a comunicar procesos que se ejecutan ambos en el mismo Camunda (NO es lo normal entre diferentes partes, p.e. Cliente y Proveedor).

— **Tarea Manual:** Cuando el envío de mensaje se realiza mediante un gestor de correo externo (Gmail, Outlook, Thunderbird,...), siguiendo los criterios vistos sobre tareas manuales.

# Envío de Mensajes

---

Utilizaremos Gmail

---

Configura la cuenta de correo

---

Modela el proceso y desplegarlo

---

Crea el servicio externo

---

Ejecuta el proceso desde Camunda

---

Interactúa con el proceso

---

# Envío de Mensajes

---

## Utilizaremos Gmail

---

Configura la cuenta de correo

---

Modela el proceso y desplegarlo

---

Crea el servicio externo

---

Ejecuta el proceso desde Camunda

---

Interactúa con el proceso

---



# Envío de Mensajes

---

*Utilizaremos Gmail*

---

**Configura la cuenta de correo**

---

Modela el proceso y desplegarlo

---

Crea el servicio externo

---

Ejecuta el proceso desde Camunda

---

Interactúa con el proceso


---

# Configura la cuenta de correo (I)

Configura la cuenta de correo para la **verificación en dos pasos** y la **creación de clave** para una aplicación.

Recomendamos crear una cuenta para las prácticas.

Seguiremos los pasos para configurar manualmente la cuenta de correo.



### Elige la configuración de personalización

☒


**Personalización rápida (1 paso)**  
Usa la configuración que proporciona anuncios y contenido personalizados. En unas semanas, te lo recordaremos para que revises tu configuración.

☐

**Personalización manual (5 pasos)**  
Define tu configuración de personalización paso a paso. Puedes activar o desactivar las opciones de configuración para administrar tu experiencia con el contenido y los anuncios.

Puedes modificar la configuración en cualquier momento, en [account.google.com](https://account.google.com)

Siguiente



Puedes controlar los datos que recopilamos y la manera en que se usan

# Configura la cuenta de correo (I)

Accede a la configuración de tu correo

The screenshot shows the Gmail web interface in a browser. The address bar displays the URL `https://mail.google.com/mail/u/0/?zx=sup2dhdj8tbz#inbox`. The Gmail logo and search bar are at the top. On the left sidebar, the 'Recibidos' (Inbox) tab is selected. The main area shows the 'Principal' (Primary) tab with an email from 'El equipo de la com.' with the subject 'Profesor, completa la configuración de tu nueva C'. A storage indicator shows '0 GB de 15 GB utilizado(s)'. On the right, a profile card for 'Profesor Estrada Torres' (practicass.ibet@gmail.com) is displayed. A red rectangle highlights the 'Administrar tu Cuenta de Google' button. Below it is the 'Agregar otra cuenta' button, and at the bottom of the card is the 'Salir' button.

Recibidos - practicas.ibet@gmail x +

https://mail.google.com/mail/u/0/?zx=sup2dhdj8tbz#inbox

Gmail

Buscar en el correo electrónico

Redactar

Recibidos

Destacados

Pospuestos

Enviados

Meet

Nueva reunión

Unirte a una reunión

Hangouts

Profesor

No hay chats recientes.

Inicia una nueva

Principal

Social

Promociones

El equipo de la com.

Profesor, completa la configuración de tu nueva C

0 GB de 15 GB utilizado(s)

Condiciones · Privacidad · Políticas de programa

Profesor Estrada Torres

practicass.ibet@gmail.com

Administrar tu Cuenta de Google

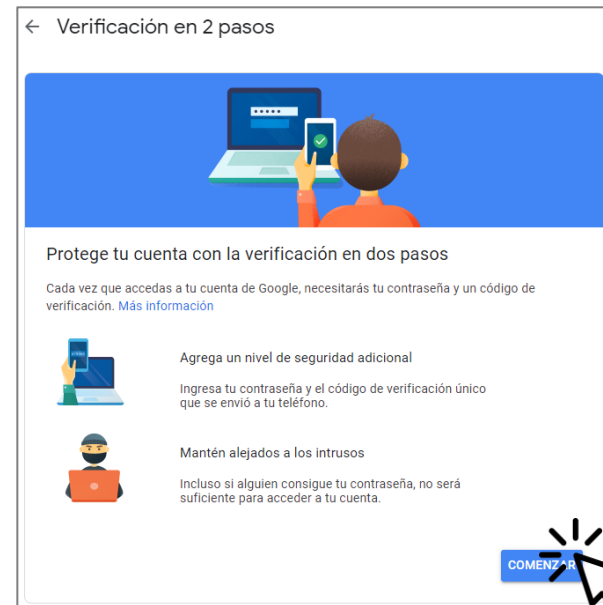
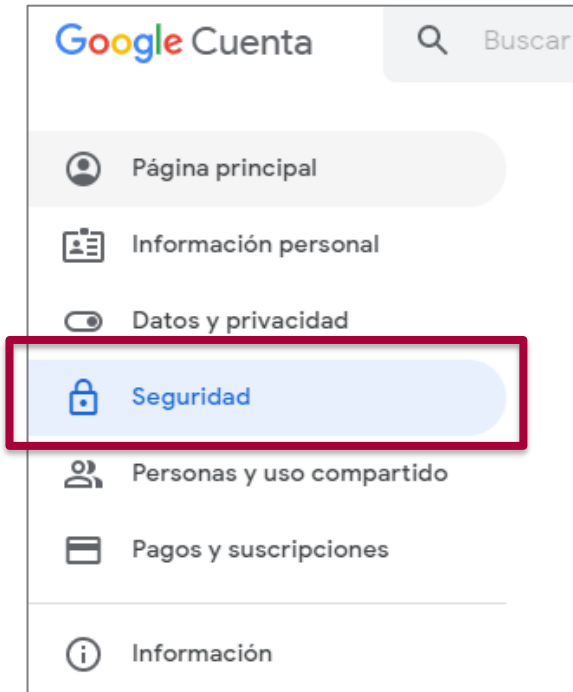
Agrega otra cuenta

Salir


https://mail.google.com/mail/u/0/#inbox

# Configura la cuenta de correo (II)

Seguir las instrucciones del asistente paso a paso.




# Configura la cuenta de correo – Verificación en dos pasos



## Configurar tu teléfono

¿Qué número de teléfono deseas usar?

 +34


Google solo usará este número como método de seguridad de la cuenta.  
No ingreses un número de Google Voice.  
Se pueden aplicar cargos por mensajes y datos.

¿Cómo deseas obtener los códigos?

☒ Mensaje de texto ☐ Llamada telefónica

[Mostrar más opciones](#)

Paso 1 de 3 [SIGUIENTE](#)



## Confirma que funciona


Google acaba de enviar un mensaje de texto con un código de verificación al

Ingresar el código

G-

¿No lo recibiste? [Volver a enviar](#)

[ATRÁS](#) Paso 2 de 3 [SIGUIENTE](#)



## ¡Funcionó! ¿Deseas activar la verificación en dos pasos?


Ahora que ya viste cómo funciona, ¿deseas activar la verificación en dos pasos para tu cuenta de Google practicas.ibet@gmail.com?

Paso 3 de 3 [ACTIVAR](#)



# Configura la cuenta de correo – Contraseña de aplicaciones

Acceso a Google




Contraseña Última modificación: 12:04 >

Verificación en 2 pasos ☒ Sí >

**Contraseñas de aplicaciones Ninguna >**

No tienes contraseñas de la aplicación.


Selecciona la app y el dispositivo para los que quieras generar la Contraseña de la aplicación.

Seleccionar app 

Otra (*nombre personalizado*)

No tienes contraseñas de la aplicación.

Selecciona la app y el dispositivo para los que quieras generar la Contraseña de la aplicación.

nombre-proyecto-app 

**GENERAR**

Asigna el nombre de la aplicación.  
Por ejemplo: gps-send-mails

# Configura la cuenta de correo (III)

## Contraseña de aplicación generada

Tu contraseña de aplicación para el dispositivo

wfef bdwa czwi fwip

Instrucciones de uso

Ve a la configuración de tu cuenta de Google en la aplicación o el dispositivo que quieres configurar. Ingresa los caracteres que aparecen en la imagen anterior.

Al igual que la contraseña de tu cuenta, esta clave es única y completa a tu configuración de seguridad. Es necesario que la guardes y la compartas con el dispositivo que quieres configurar.

**LISTO**

Email

Password

Esta clave se utilizará en la clase Java que funcionará como **tarea de servicio externa**.

# Envío de Mensajes

---

*Utilizaremos Gmail*

---

*Configura la cuenta de correo*

---

**Modela el proceso y desplegarlo**

---

Crea el servicio externo

---

Ejecuta el proceso desde Camunda

---

Interactua con el proceso

---



# Modela el proceso (I)

The screenshot displays the Camunda Modeler interface. The main canvas shows a BPMN process diagram for 'send\_mails.bpmn'. The process starts with a start event (thin circle), followed by a task 'Approve Data' (rounded rectangle with a person icon), then a task 'Send Information' (rounded rectangle with an envelope icon), and finally an end event (thick circle). Two red arrows point from labels below to the tasks: 'Tarea de usuario' points to 'Approve Data', and 'Tarea de envío de mensaje' points to 'Send Information'. On the left is a toolbar with various BPMN symbols. On the right is the 'Properties Panel' for the selected 'payment\_mail' task. The panel has tabs for 'General', 'Variables', 'Listeners', and 'Extensions'. The 'General' tab is active, showing fields for 'Id' (set to 'payment\_mail'), 'Name' (set to 'Payment Mail'), and 'Version Tag'. The 'Executable' checkbox is checked. Under 'External Task Configuration', there is a 'Task Priority' field.

Camunda Modeler

File Edit Window Help

send\_mails.bpmn x

payment\_mail

General Variables Listeners Extensions

General

Id  
payment\_mail x

This maps to the process definition key.

Name  
Payment Mail

Version Tag

☒ Executable

External Task Configuration

Task Priority

XML Camunda Platform Log 4.11.1

# Modela el proceso (II)

Agrega los inputs necesarios para el formulario (*interno*) **Type:<none>**

The screenshot displays the Camunda Modeler interface. The main canvas shows a BPMN process diagram with a start event, an 'Approve Data' activity (highlighted with a red dashed border), an intermediate event, a 'Send Information' activity, and an end event. The 'Approve Data' activity is selected, and the Properties Panel on the right is open, showing the 'Forms' tab. The 'Type' is set to '<none>' and the 'Form Fields' list contains 'mail', 'item', 'user', and 'amount'. The bottom status bar indicates 'XML Camunda Platform 4.11.1'.

Camunda Modeler

File Edit Window Help

send\_mails.bpmn

Approve Data

Send Information

Activity\_user

General Forms Listeners Input/Output Extensions

Forms

Type

<none>

Form Fields

mail  
item  
user  
amount

XML Camunda Platform 4.11.1

# Modela el proceso (III)

Agrega y configura los inputs del formulario

Forms

Type<none>

Form Fields

mail

item

user

amount

Form Field

ID (process variable name)mail

Typestring

LabelE-mail:

Default Value

Forms

Type<none>

Form Fields

mail

item

user

amount

Form Field

ID (process variable name)item

Typestring

LabelÍtem:

Default Value

Forms

Type<none>

Form Fields

mail

item

user

amount

Form Field

ID (process variable name)user

Typestring

LabelUsuario:

Default Value

Forms

Type<none>

Form Fields

mail

item

user

amount

Form Field

ID (process variable name)amount

Typelong

LabelMonto:

Default Value

# Modela el proceso (IV)

Configura la tarea de envío de mensaje.

Implementation: “*External*”, Topic: “*payment-mail*”

The screenshot displays the Camunda Modeler interface. The main canvas shows a BPMN process diagram with a start event, a task named 'Approve Data', and a task named 'Send Information'. The 'Send Information' task is highlighted with a red dashed border. The Properties Panel on the right shows the configuration for the 'Activity\_send' task. The 'General' tab is selected, and the 'Details' section is expanded. The 'Implementation' is set to 'External' and the 'Topic' is set to 'payment-mail'. The 'External Task Configuration' section is also visible.

Camunda Modeler

File Edit Window Help

send\_mails.bpmn

Activity\_send

General Listeners Input/Output

General

Id  
Activity\_send

Name  
Send Information

Details

Implementation  
External

Topic  
payment-mail

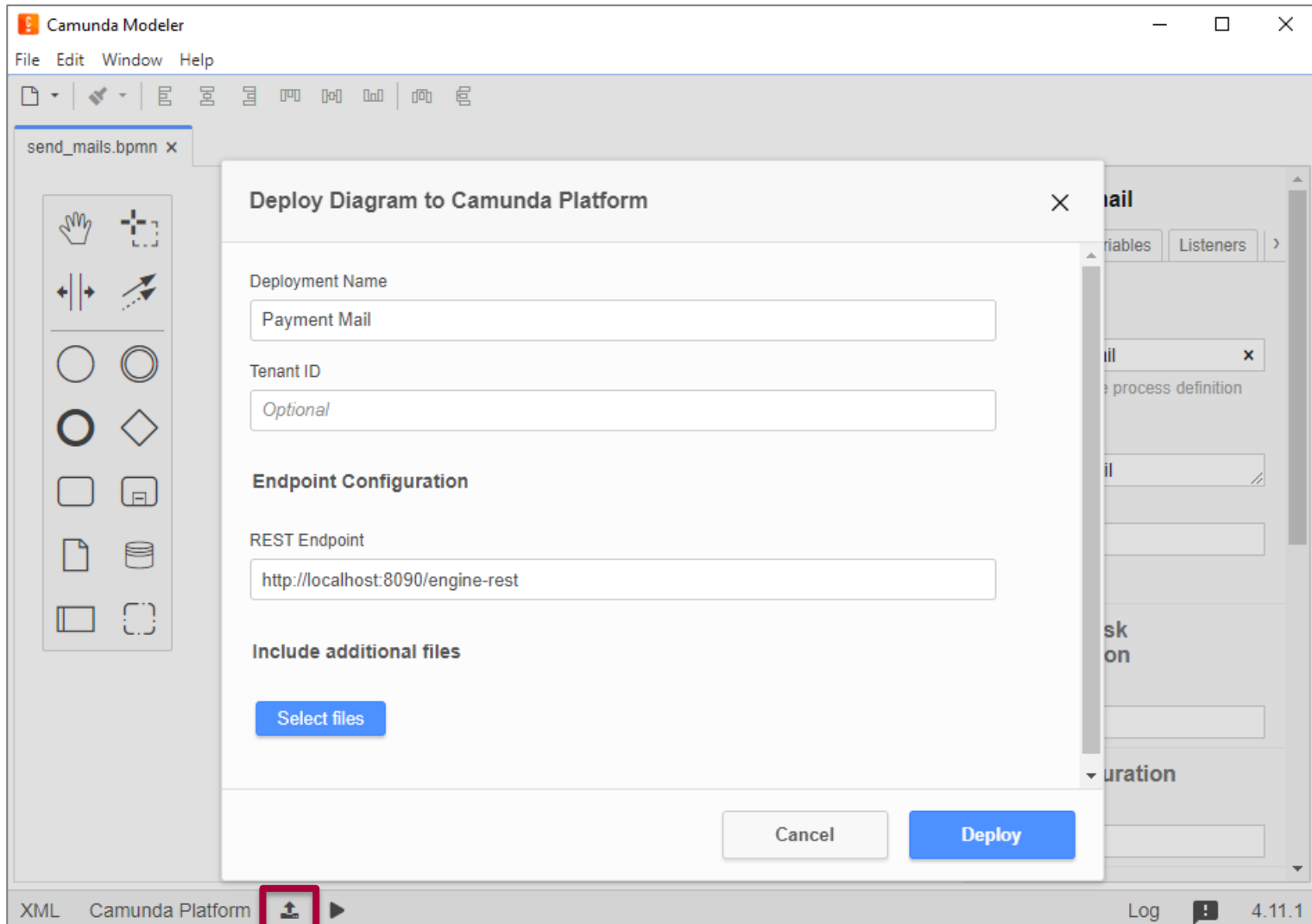
External Task Configuration

Task Priority

XML Camunda Platform Log 4.11.1

# Despliega el proceso

Despliega el proceso “*Payment Mail*”



# Envío de Mensajes

---

*Utilizaremos Gmail*

---

*Configura la cuenta de correo*

---

*Modela el proceso y desplegarlo*

---

**Crea el servicio externo**

---

Ejecuta el proceso desde Camunda

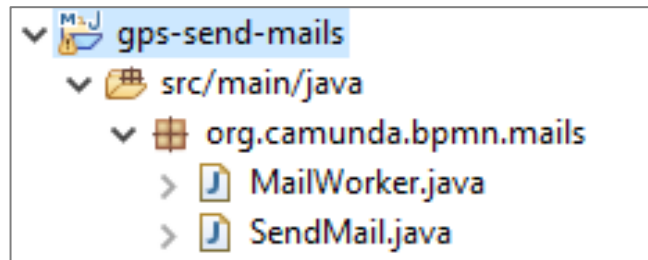
---

Interactua con el proceso

---

## Crea el servicio externo (I)

- Crea el proyecto Java siguiendo las directrices vistas anteriormente. Utiliza el nombre de la aplicación para el que se configuró la clave de Gmail: `gps-send-mails`



- En el *pomp.xml*, agrega la dependencia para el envío de correo.

```
<dependencies>
  <dependency>
    <groupId>org.apache.commons</groupId>
    <artifactId>commons-email</artifactId>
    <version>1.3.3</version>
  </dependency>
</dependencies>
```

# Crear el servicio externo (II)

Crea la clase Java “**SendMailWorker**”

```
MailWorker.java ×
1 package org.camunda.bpmn.mails;
2
3 import java.util.logging.Logger;
4 import org.camunda.bpm.client.ExternalTaskClient;
5
6 public class MailWorker {
7     final static Logger LOGGER = Logger.getLogger(MailWorker.class.getName());
8
9     public static void main(String[] args) {
10         ExternalTaskClient client = ExternalTaskClient.create()
11             .baseUrl("http://localhost:8090/engine-rest")
12             .build();
13
14         SendMail sendMail = new SendMail();
15
16         client.subscribe("payment-mail")
17             .lockDuration(1000)
18             .handler((externalTask, externalTaskService) -> {
19                 String mail = (String) externalTask.getVariable("mail");
20                 String user = (String) externalTask.getVariable("user");
21                 String item = (String) externalTask.getVariable("item");
22                 Long amount = (Long) externalTask.getVariable("amount");
23
24                 LOGGER.info("CONTENIDO DEL MENSAJE A ENVIAR:"
25                     + "\n Estimado/a " + user + " <" + mail + ">,\n"
26                     + "\n Se ha realizado el cargo a su tarjeta de crédito con los siguientes datos:
27                     + "\n - Item: " + item + "\n - Monto: " + amount);
28
29                 try {
30                     sendMail.sendMail(mail,user,item,amount);
31                     LOGGER.info(">> El correo electrónico ha sido enviado...");
32                 } catch (Exception e) {
33                     LOGGER.info(">> El correo electrónico no ha sido enviado...");
34                 }
35                 externalTaskService.complete(externalTask);
36             })
37             .open();
38     }
39 }
```



# Crear el servicio externo (III)

Crea la clase Java ***SendMail***

```
SendMail.java X
1 package org.camunda.bpmn.mails;
2
3 import java.util.logging.Logger;
4 import org.apache.commons.mail.DefaultAuthenticator;
5 import org.apache.commons.mail.Email;
6 import org.apache.commons.mail.SimpleEmail;
7
8 public class SendMail {
9     final static Logger LOGGER = Logger.getLogger(SendMail.class.getName());
10
11     private static final String HOST = "smtp.gmail.com";
12     private static final String USER = "practicas.ibet@gmail.com";
13     private static final String PWD = "wfefbdwaczwfip";
14     private static final int PORT = 465;
15
16     public void sendMail(String mail, String user, String item, long amount) {
17         if(mail != null && !mail.isEmpty()) {
18             Email email = new SimpleEmail();
19             email.setStartTLSEnabled(true);
20             email.setSSLOnConnect(true);
21             email.setHostName(HOST);
22             email.setAuthenticator(new DefaultAuthenticator(USER, PWD));
23             email.setSmtpPort(PORT);
24             email.setStartTLSEnabled(true);
25             try {
26                 email.setFrom(USER, "[GPS]");
27                 email.setSubject("Confirmación - Cargo tarjeta");
28                 email.setMsg("Estimado/a " + user + " <" + mail + ">,"
29                     + "\n\nSe ha realizado el cargo a su tarjeta de crédito. "
30                     + "\n - Item: " + item + "\n - Monto: " + amount
31                     + "\n\nSaludos cordiales."
32                     + "\nEquipo técnico");
33                 email.addTo(mail, user);
34                 email.send();
35             } catch (Exception e) {
36                 e.printStackTrace();
37                 LOGGER.warning("No fue posible enviar el correo a la dirección " + mail
38                     + " - " + e.getMessage());
39             }
40         }
41     }
42 }
```

Utiliza la dirección de correo configurada anteriormente

Esta es la clave obtenida al configurar el correo de Gmail

Ir a MailWorker

>> *Run as...*

>> *Java Application*

# Envío de Mensajes

---

*Utilizaremos Gmail*

---

*Configura la cuenta de correo*

---

*Modela el proceso y desplegarlo*

---

*Crea el servicio externo*

---

**Ejecuta el proceso desde Camunda**

---

Interactúa con el proceso

---

# Ejecuta el proceso desde Camunda (II)

Inicia una instancia del proceso *Payment Mail*

Proporciona las variables para el proceso *Payment Mail*

Camunda Tasklist

Create a filter +

Grupo GPS

My tasks (20)

All Tasks

### Start process

**You can set variables, using a generic form, by clicking the "Add a variable" link below.**

**Business Key**

	Name	Type	Value
Remo... x	mail	String	iestrada@us.es
Remo... x	user	String	Irene B. Estrada
Remo... x	item	String	Item GPS 2021
Remo... x	amount	Long	1275

Back Close Start

Created a day ago 50

Date and Time displayed in local timezone: Europe/Madrid

Powered by Camunda Platform / v7.16.0

Esta dirección de correo diferente al configurado en Gmail anteriormente

# Envío de Mensajes

---

*Utilizaremos Gmail*

---

*Configura la cuenta de correo*

---

*Modela el proceso y desplegarlo*

---

*Crea el servicio externo*

---

*Ejecuta el proceso desde Camunda*

---

**Interactúa con el proceso**

---

# Interactua con el proceso (I)

Accede a la tarea pendiente para el usuario *demo*

The screenshot displays the Camunda Tasklist web application. The interface is divided into several sections:

- Header:** Includes the Camunda logo, 'Camunda Tasklist' title, and navigation links for 'Keyboard Shortcuts', 'Create task', 'Start process', and a user profile 'Demo Demo'.
- Left Sidebar:** Contains navigation options: 'Create a filter +', 'Grupo GPS', 'My tasks', 'Filters (1)' (highlighted with a red bar), and 'All Tasks'.
- Task List:** A table showing a single task: 'Approve Data' with sub-task 'Payment Mail', created 'a minute ago' by 'Demo Demo' with a priority of '50'. A 'Filter Tasks' box is visible above the table.
- Task Detail View:** The 'Approve Data' task is selected, showing a form with the following fields:
  - E-mail:** iestrada@us.es
  - Ítem:** Item GPS 2021
  - Usuario:** Irene B. Estrada
  - Monto:** 1275
- Form Actions:** At the bottom right of the form are 'Save' and 'Complete' buttons. A mouse cursor is clicking the 'Complete' button, which is highlighted with a red starburst effect.
- Footer:** Displays 'Date and Time displayed in local timezone: Europe/Madrid' and 'Powered by Camunda Platform / v7.16.0'.

# Interactua con el proceso (II)

Se ejecuta la tarea externa de envío de mensaje “*Send Information*”

Camunda Cockpit Processes Decisions Human Tasks More ▾ Demo Demo ▾

Dashboard » Processes » Payment Mail : Runtime

Definition Version: 1  
Version Tag: null  
Definition ID: payment\_mail:1:fe71dc0e-5dae-11e...  
Definition Key: payment\_mail  
Definition Name: Payment Mail  
History Time To Live: null  
Tenant ID: null  
Deployment ID: fcb278cc-5dae-11ec-887f-049226b...  
Instances Running:  
• current version: 1  
• all versions: 1

Activity Instance Statistics: on ●

```
graph LR; Start(( )) --> ApproveData[Approve Data]; ApproveData --> SendInformation[Send Information]; SendInformation --> End(( ))
```

Process Instances Incidents Called Process Definitions Job Definitions

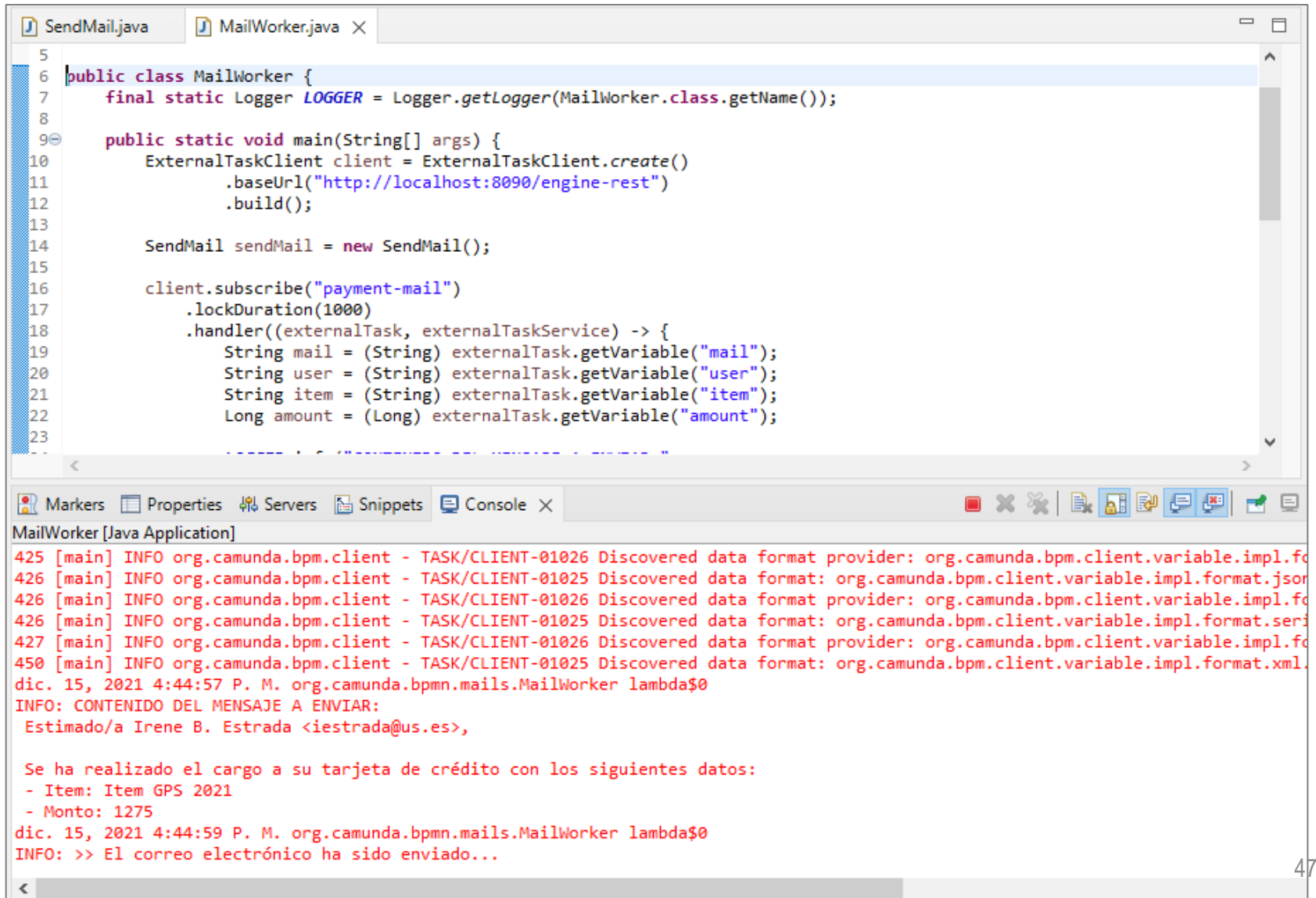
Add criteria 1 🔗 📄 ▾

State	ID	Start Time ▾	Business Key
✔	9770c16b-5db7-11ec-887f-049226be0810	2021-12-15T15:59:24	

Date and Time displayed in local timezone: Europe/Madrid Powered by Camunda Platform / v7.16.0

# Verifica el envío del correo

## Verifica la consola de Eclipse



The screenshot displays the Eclipse IDE interface. The top editor shows the `MailWorker.java` file with the following code:

```
5
6 public class MailWorker {
7     final static Logger LOGGER = Logger.getLogger(MailWorker.class.getName());
8
9     public static void main(String[] args) {
10         ExternalTaskClient client = ExternalTaskClient.create()
11             .baseUrl("http://localhost:8090/engine-rest")
12             .build();
13
14         SendMail sendMail = new SendMail();
15
16         client.subscribe("payment-mail")
17             .lockDuration(1000)
18             .handler((externalTask, externalTaskService) -> {
19                 String mail = (String) externalTask.getVariable("mail");
20                 String user = (String) externalTask.getVariable("user");
21                 String item = (String) externalTask.getVariable("item");
22                 Long amount = (Long) externalTask.getVariable("amount");
23
24                 // ... (code continues) ...
25             });
26     }
27 }
```

The bottom console window, titled "MailWorker [Java Application]", shows the following log output:

```
dic. 15, 2021 4:44:57 P. M. org.camunda.bpmn.mails.MailWorker lambda$0
INFO: CONTENIDO DEL MENSAJE A ENVIAR:
Estimado/a Irene B. Estrada <iestrada@us.es>,

Se ha realizado el cargo a su tarjeta de crédito con los siguientes datos:
- Item: Item GPS 2021
- Monto: 1275
dic. 15, 2021 4:44:59 P. M. org.camunda.bpmn.mails.MailWorker lambda$0
INFO: >> El correo electrónico ha sido enviado...
```

# Verifica la recepción del correo

## Verificar el correo del destinatario

The screenshot displays an email client interface. On the left is the 'Bandeja de entrada' (Inbox) with a star icon and a 'Filtrar' (Filter) dropdown. The selected email is from '[GPS]' with the subject 'Confirmación - Cargo tarjeta' and a timestamp of '16:46'. Below it are four other email entries with icons: a grey circle with 'P', a Microsoft logo, a person's profile picture, a purple circle with 'P', and a blue circle with 'S'.

The main area shows the open email titled 'Confirmación - Cargo tarjeta'. The sender is '[GPS] <practicas.ibet@gmail.com>' with a timestamp of 'Mié 15/12/2021 16:46'. The recipient is 'Para: IRENE BEDILIA ESTRADA TORRES'. The email body contains the following text:

Estimado/a Irene B. Estrada <iestrada@us.es>,  
  
Se ha realizado el cargo a su tarjeta de crédito.  
- Item: Item GPS 2021  
- Monto: 1275

Saludos cordiales.  
Equipo técnico

Below the text are three buttons: 'Saludos.', 'Me llegó esto.', and 'Gracias.'.

At the bottom, there is a feedback question: '¿Las sugerencias anteriores son útiles?' with 'Sí' and 'No' options.

At the very bottom are two buttons: 'Responder' and 'Reenviar'.



## **Tareas automáticas – Reglas de negocio**

# Tareas automáticas

- Service task
- Send task
- **Business rule task: Para decisiones usando DMN**
- Receive task



# Tareas automáticas – Business Rule Task

Configura la tarea de reglas de negocio “*Decidir reserva*”

The screenshot shows the Camunda Modeler interface. The main canvas displays a BPMN diagram with the following flow: Start Event → 'Rellenar solicitud' (Task) → 'Decidir reserva' (Task, highlighted with a blue dashed border) → 'Confirmar resultados' (Task) → End Event. The 'Decidir reserva' task is configured with a Decision Ref of 'dec-reserva'. The Properties Panel on the right shows the configuration for the 'Activity\_decision' task.

**Activity\_decision**

General

Id: Activity\_decision

Name: Decidir reserva

Details

Implementation: DMN

Decision Ref: dec-reserva

Binding: latest

Tenant Id:

Result Variable: response

Map Decision Result: singleEntry (TypedValue)

**DMN es el estándar de definición de decisiones**

**Decision Ref indica el Id de la decisión a ejecutar**

**Response variable nombre de la variable donde se almacena la respuesta**

**Map Decision Result define el tipo de variable que se espera**

XML Camunda Platform Log 51 4.11.1

# Tareas automáticas – Business Rule Task

Crea un DMN Diagram utilizando *Camunda Modeler* y crea la decisión “*Decidir reserva*”

The screenshot shows the Camunda Modeler interface. On the left, a palette contains various BPMN and DMN symbols. In the center, a DMN diagram titled 'reservas.dmn' is being edited. A decision task named 'Decidir reserva' is highlighted with a blue dashed border. On the right, the 'Properties Panel' for the 'dec-reserva' task is visible. The 'General' tab is active, showing the 'Id' field set to 'dec-reserva' and the 'Name' field set to 'Decidir reserva'. A red callout box with an arrow points to the 'Id' field, containing the text: 'Id debe coincidir con el Decision Ref de la tarea'.

The screenshot shows the 'Edit DRD' panel for the 'Reserva' decision. The 'Hit Policy' is set to 'Unique'. The table below defines the decision logic based on 'Turno' and 'Num Personas' to produce a 'response'.

	When	And	Then	Annotations
	Turno string	Num Personas long	response string	
1	"manana"	<5	"Reserva realizada"	
2	"manana"	>=5	"No hay mesas disponibles"	
3	"tarde"	<5	"No hay mesas disponibles"	
4	"tarde"	>=5	"No hay mesas disponibles"	
5	"noche"	<5	"Reserva realizada"	
6	"noche"	>=5	"Reserva realizada"	
+	-	-		

# Tareas automáticas – Business Rule Task

Configura los valores de la tabla de decisión

Turno|

Expression

Expression Language

Input Variable

Type:

Num Personas|

Expression

Expression Language

Input Variable

Type:

Response|

Output Name

Type:

Predefined Values  
No values  
[Clear predefined values.](#)

Add Predefined Values

## Tareas automáticas – Business Rule Task

- Despliega la **decision** y el **proceso** modelados desde “**Camunda Modeler**”
- Inicia una instancia del proceso “**Decision Process**”
  - No es necesario iniciar la decisión.
- Completa los datos del formulario “**Rellenar solicitud reserva**”
- Completa los datos del formulario “**Confirmar resultados**”
- Utiliza diferentes valores y comprueba los resultados de la decisión

## **Tareas automáticas – Receive task**

# Tareas automáticas

- Service task
- Send task
- Business rule task
- **Receive task:**

**No usar!**

**Utilizar eventos en su lugar**



Introducción  
Conceptos básicos  
Tareas Automáticas  
**Tareas de Usuario**

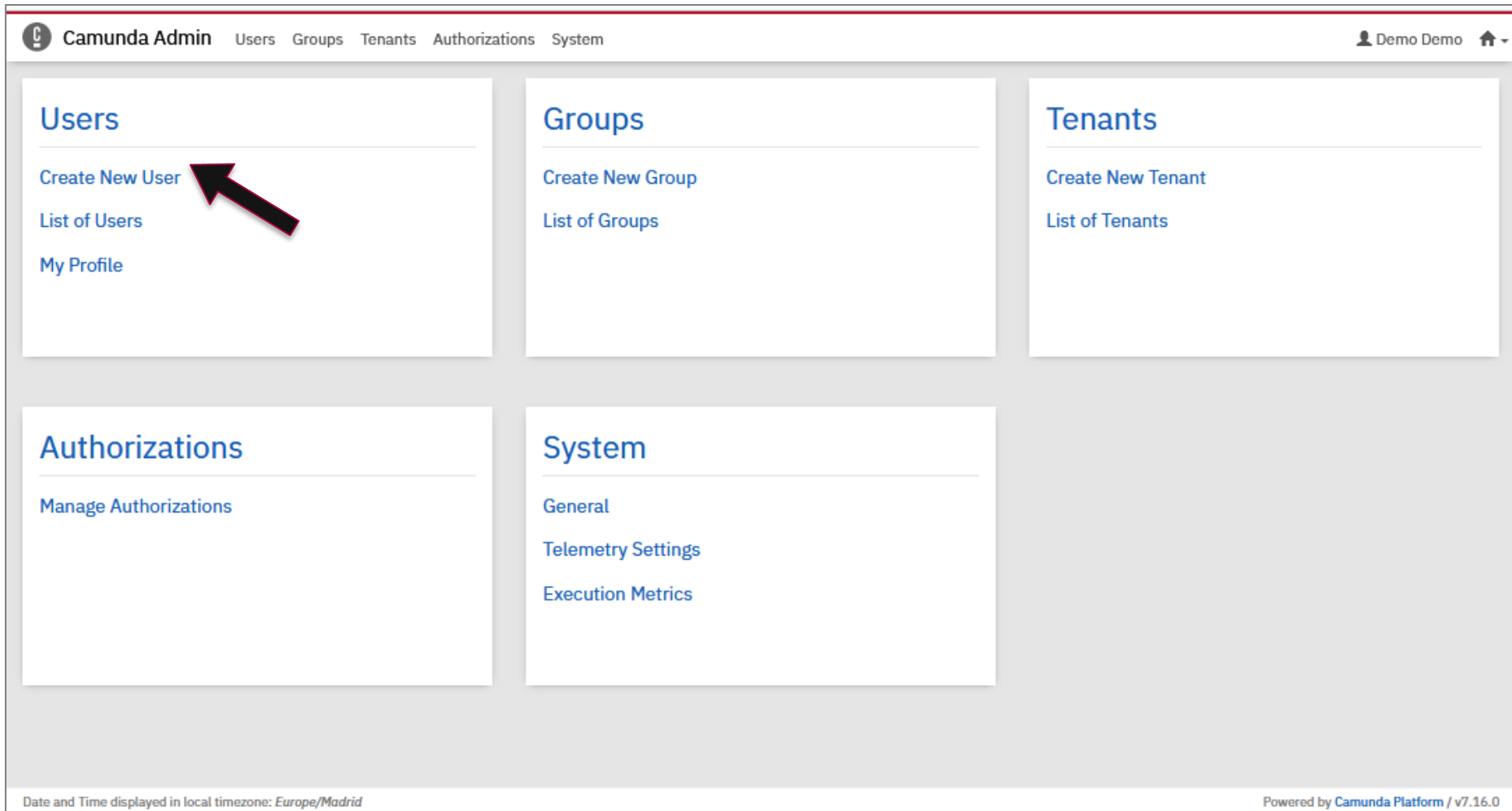
# Tareas de Usuario

- Asignación de usuarios y grupos
- Validación de formularios

**Tareas de usuario –Usuarios, grupos y filtros**

# Tareas de usuario – Crear usuarios (I)

Desde “**Camunda Admin**”, agrega tres nuevos usuarios



The screenshot displays the Camunda Admin web interface. The top navigation bar includes the Camunda logo, the text "Camunda Admin", and a series of tabs: "Users", "Groups", "Tenants", "Authorizations", and "System". On the right side of the navigation bar, there is a user profile icon labeled "Demo Demo" and a home icon. The main content area is divided into five sections, each with a title and a list of links:

- Users**:
  - Create New User (highlighted with a red arrow)
  - List of Users
  - My Profile
- Groups**:
  - Create New Group
  - List of Groups
- Tenants**:
  - Create New Tenant
  - List of Tenants
- Authorizations**:
  - Manage Authorizations
- System**:
  - General
  - Telemetry Settings
  - Execution Metrics

At the bottom of the interface, a footer bar contains the text "Date and Time displayed in local timezone: Europe/Madrid" on the left and "Powered by Camunda Platform / v7.16.0" on the right.

# Tareas de usuario – Crear usuarios (II)

Desde “**Camunda Admin**”, agrega dos nuevos usuarios: **alumnogps**, **estudiantegps**, **nogps**.

The screenshot shows the 'Create User' form in the Camunda Admin interface. The top navigation bar includes 'Camunda Admin', 'Users', 'Groups', 'Tenants', 'Authorizations', and 'System'. The breadcrumb trail is 'Dashboard » Users » Create'. The form is divided into two sections: 'User Account' and 'User Profile'.

**User Account**

- User ID\***: Input field containing 'alumnogps'.
- Password\***: Input field with masked characters (dots).
- Password (repeat)\***: Input field with masked characters (dots).

**User Profile**

- First Name\***: Input field containing 'Alumno'.
- Last Name\***: Input field containing 'Alumno GPS'.
- Email**: Input field containing 'alumno@us.es'.







At the bottom right, there are two buttons: 'Cancel' and 'Create new user'.

Date and Time displayed in local timezone: Europe/Madrid


Powered by Camunda Platform / v7.16.0

# Tareas de usuario – Crear grupo GPS



Verifica la lista de usuarios:

List of users				Add user +
Add criteria				5  
ID 	First Name* 	Last Name* 	Email 	Action
<a href="#">alumnogps</a>	Alumno	Alumno GPS	alumno@us.es	<a href="#">Edit</a>
<a href="#">demo</a>	Demo	Demo	demo@localhost	<a href="#">Edit</a>
<a href="#">estudiantegps</a>	Estudiante	Estudiante	estudiante@us.es	<a href="#">Edit</a>
<a href="#">nogps</a>	No GPS	No GPS	nogps@us.es	<a href="#">Edit</a>

Agrega un grupo “**GPS**”:

 Camunda Admin

UsersGroupsTenantsAuthorizationsSystem

 Demo Demo 

Dashboard » Groups » Create New Group

Create new group

Group ID\*

GPS

Group Name\*

GPS

Group Type

GPS



CancelCreate new group

Date and Time displayed in local timezone: Europe/Madrid

Powered by Camunda Platform / v7.16.0

# Tareas de usuario – Agregar usuarios al grupo GPS (I)

Selecciona el usuario

List of users				Add user +
Add criteria				5  
ID ^	First Name* -	Last Name* -	Email -	Action
alumnogps	Alumno	Alumno GPS	alumno@us.es	Edit
demo	Demo	Demo	demo@localhost	Edit
estudiantegps	Estudiante	Estudiante	estudiante@us.es	Edit
nogps	No GPS	No GPS	nogps@us.es	Edit

Selecciona “**Groups**”:

Camunda Admin

Users

Groups

Tenants

Authorizations

System

Demo Demo

Dashboard » Users » Alumno Alumno GPS

Profile

Account

Groups

Tenants

Edit user Alumno Alumno GPS

Profile

First Name\* Alumno

Agrega el usuario al grupo.

Camunda Admin

Users

Groups

Tenants

Authorizations

System

Demo Demo

Dashboard » Users » Alumno Alumno GPS

Profile

Account

Groups

Tenants

Edit user Alumno Alumno GPS

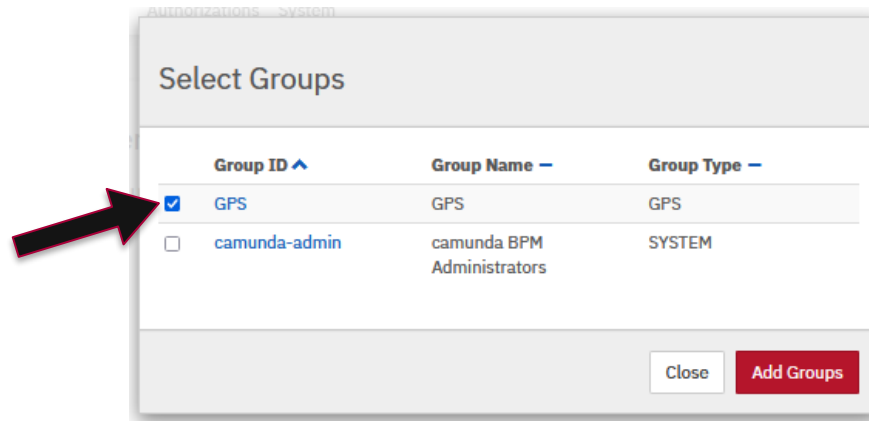
Alumno Alumno GPS's Groups

User Alumno Alumno GPS is currently not member of any groups.

Add to a group +

# Tareas de usuario – Agregar usuarios al grupo GPS (II)

Selecciona el grupo



Repite el proceso para el usuario “*estudiantegps*”. No agregue el usuario “*nogps*” a este grupo.

Verifica que los dos alumnos pertenecen al grupo “*GPS*”.

Camunda Admin Users **Groups** Tenants Authorizations System Demo Demo

Dashboard » Groups » GPS

Information  
Tenants  
**Users**

### Edit group GPS

#### Group users

ID	First Name*	Last Name*
alumnogps	Alumno	Alumno GPS
estudiantegps	Estudiante	Estudiante



# Tareas de usuario – Agregar usuarios al grupo GPS (III)

Crea un proceso con una tarea de usuario y asígnala al grupo GPS usando “**Candidate Groups**”

The screenshot displays the Camunda BPMN editor interface. On the left, a toolbar contains various BPMN symbols. The main canvas shows a process diagram with a start event (circle) leading to a task (rounded rectangle) labeled 'Prueba de usuario-grupo'. The task is highlighted with a blue dashed border. To the right of the task, there are several icons representing different types of events and connectors. On the far right, the 'Properties Panel' is open, showing the configuration for the selected task, 'Activity\_1481q3d'. The panel has tabs for 'General', 'Forms', 'Listeners', 'Input/Output', and 'Extension'. The 'General' tab is active, showing fields for 'Id' (Activity\_1481q3d), 'Name' (Prueba de usuario-grupo), 'Assignee', 'Candidate Users', 'Candidate Groups', and 'Due Date'. The 'Candidate Groups' field is highlighted with a red box and a red arrow, and it contains the value 'GPS'.

Configura la tarea de usuario para que reciba un campo “**nombre**” de tipo **string** para mostrar el nombre de un usuario.

Despliega el proceso con nombre “**Usuarios y grupos**”.

# Tareas de usuario – Agregar usuarios al grupo GPS (III)

Utilizando el usuario “**demo**”, lanza una nueva instancia del proceso “**Usuarios y grupos**”.

Como el usuario “**demo**” es **admin** podrá ver la tarea, y verá que está asignada a “**GPS**”. Al ser **admin**, podría reclamarla para ejecutarla (pero no lo hará).

The screenshot displays the Camunda Tasklist web application. The top navigation bar includes the Camunda logo, the title 'Camunda Tasklist', and links for 'Keyboard Shortcuts', 'Create task', 'Start process', and a user profile 'Demo Demo'. The main interface is divided into a left sidebar and a central task view. The sidebar shows a list of tasks under the heading 'All Tasks (11)', with the first task being 'Prueba de usuario-grupo' (50 points, created a few seconds ago) and the second being 'Analizar solicitud' (50 points, created 4 days ago). The central task view shows the details for 'Prueba de usuario-grupo'. It includes a title bar, a sub-header 'Usuarios y grupos', and a 'GPS' group icon highlighted with a red box. Below this, there are tabs for 'Form', 'History', 'Diagram', and 'Description'. The 'Form' tab is active, showing a text input field with the value 'Bedilia'. At the bottom right of the form, there are 'Save' and 'Complete' buttons. The footer of the interface indicates 'Date and Time displayed in local timezone: Europe/Madrid' and 'Powered by Camunda Platform / v7.16.0'.

# Tareas de usuario – Crea un filtro (I)

Crea un filtro de tareas:

### General

**Customize the appearance of the filter.**

<b>Name</b>	<input type="text" value="Grupo GPS"/>	<b>Color</b>	<input type="color" value="#000080"/>
<b>Description</b>	<input type="text" value="Shows the tasks I am assigned"/>	<b>Priority</b>	<input type="text" value="0"/>

☐ Auto-Refresh

### Criteria

**This section is aimed to set the parameters used to filter the tasks. Keys marked with a \* accept expressions as value.**

<b>Add criteri... +</b>	<b>Key</b>	<b>Value</b>
<b>Remo... x</b>	<input type="text" value="Candidate Groups *"/>	<input type="text" value="GPS"/>

*List of values separated by comma or an expression which evaluates to a list. E.g.: 'camunda-admin, accounting' or '\${ currentUserGroups() }'*

☐ Include assigned tasks

Also include tasks that are assigned to users in candidate queries.  
Default is to only include tasks that are not assigned to any user if you query by candidate user or group(s).

of the criteria are met.

# Tareas de usuario – Crea un filtro (II)

Accede a **Camunda** con el usuario “**estudiantegps**”.

Verifica que aparece una tarea asignada a su grupo.

The screenshot shows the Camunda Tasklist interface. On the left, a sidebar lists filters: 'Create a filter +', 'Grupo GPS (1)', and 'All Tasks'. The main area displays a task titled 'Prueba de usuario-grupo' with a description 'Usuarios y grupos' and 'Created 21 minutes ago'. A red box highlights the 'GPS' group assignment. The task details panel on the right shows 'Prueba de usuario-grupo' with options to 'Set follow-up date' and 'Set due date'. A red box highlights the 'GPS' group in the 'Usuarios y grupos' section. Below this, there are tabs for 'Form', 'History', 'Diagram', and 'Description'. The 'Form' tab is active, showing a text input field with the value 'Bedilia'. At the bottom right, there are 'Save' and 'Complete' buttons.

Reclama la tarea.

Finaliza el proceso.

Las asignaciones “**Candidate Groups**” y “**Candidate Users**” no son restrictivas. Para esto ve a: Camunda Admin > Authorizations y asigna las restricciones\*

Para asignar una tarea a varios posibles usuarios candidatos:

The diagram shows a process flow with a task named 'Application Form' highlighted by a red dashed box. To the right, a 'Details' panel is shown. The 'Assignee' field is empty. The 'Candidate Users' field is highlighted by a red box and contains the text 'ibet, demo, peter' with a clear 'x' button.

## **Tareas de usuario – Validación de formularios**

# Tareas de usuario – Formularios (I)

The screenshot shows the Camunda BPMN Editor interface. On the left is a toolbar with various BPMN symbols. The main canvas displays a BPMN diagram with a start event, a task labeled "Proporciona una form key", and an end event. The task is highlighted with a blue dashed border. On the right is the "Properties Panel" for the selected task, showing the "Forms" tab. The "Form Key" field is set to "camunda-forms:deployment:show\_form.form".

forms.bpmn

Activity\_Form

General Forms Listeners Input/Output Extensions

Forms

Type  
Embedded or External Task Forms

Form Key  
camunda-forms:deployment:show\_form.form

Form Fields

XML Camunda Platform

Log 4.11.1

**Form key:** FORM-TYPE:LOCATION:FORM.NAME

**FORM-TYPE:**

- "embedded"
- "camunda-forms"

**LOCATION:**

- "deployment": The file is part of your deployment
- "app":

# Tareas de usuario – Formularios (II)

Usa **Camunda Modeler** para crear un *formulario externo* como se muestra en la imagen.

forms.bpmn x show\_form.form

**FORM ELEMENTS LIBRARY**

- Text Field
- Number
- ☒ Checkbox
- us.es) está conectado
- Select
- ABC** Text
- Button

**Registro de alumnos - Multi Instancia**

DNI\*

Nombres:

Apellidos:\*

Correo electrónico:

Curso\*

Select\*

Listado de asignaturas disponibles

☐ Acepta las condiciones

Borrar todo

**ABC** FORM Form\_Registro

General

Id

Form\_Registro

Tipo Text Field - string

Tipo number - int

Tipo Select - enum

JSON No platform selected

Log 71 4.11.1

# Tareas de usuario – Validación de formularios (I)

Datos requeridos para un campo de formulario.

TEXT FIELD  
DNI

General

Field Label  
DNI

Field Description

Key  
dni

Maps to a process variable.

Validation

Required  
☐

Minimum Length

Maximum Length

Regular Expression Pattern

SELECT  
Select

General

Field Label  
Select

Field Description  
Listado de asignaturas disponibles

Key  
asignatura

Maps to a process variable.

Values

GPS  
Label  
GPS  
Value  
gps

AISS  
Label  
AISS

NUMBER  
Curso

General

Field Label  
Curso

Field Description

Key  
field\_1ad4otx

Maps to a process variable.

Validation

Required  
☐

Minimum

Maximum

☒ CHECKBOX  
Acepta las condiciones

General

Field Label  
Acepta las condiciones

Field Description

Key  
condiciones

Maps to a process variable.

BUTTON  
Borrar todo

General

Field Label  
Borrar todo

Action  
Reset

Validaciones

Validaciones



# Tareas de usuario – Validación de formularios (II)

Si el formulario es **embebido** los criterios de validación se agregan manualmente:

**Form Field**


ID (process variable name)

Type


Label

Default Value

**Validation**

Add Constraint 


**Properties**

Add Property 

Type

string  
long  
boolean  
date  
enum  
custom type

**Validation**

Add Constraint 

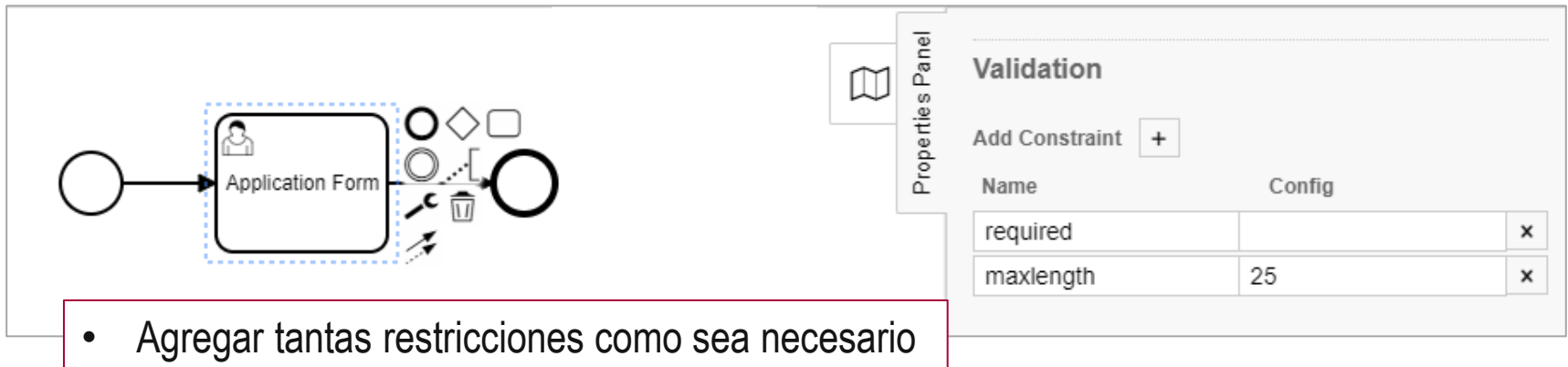
Name

Config

# Tareas de usuario – Validación de formularios (III)

Validación de campos de formulario (desde form embebido):

- **required** – Aplica a todos los tipos
- **minlength** (#) – Aplica a los campos *string*. Acepta 'null'
- **maxlength** (#) – Aplica a los campos *string*. Acepta 'null'
- **min** (#) – Aplica a valores numéricos. Acepta 'null'
- **max** (#) – Aplica a valores numéricos. Acepta 'null'
- **readonly** – Aplica a todos los tipos.



• Agregar tantas restricciones como sea necesario

# Tareas de usuario – Validación de formularios (IV)

Validación desde el formulario externo. Por ejemplo:

## *DNI*

Required  
☒

Minimum Length

Maximum Length

DNI\*

Field is required.

DNI\*

Field must have minimum length of 8.

DNI\*

Field must have maximum length of 10.

## *Curso*

Required  
☒

Minimum

Maximum

Curso\*

Field is required.

Curso\*

Field must have minimum value of 1.

Curso\*

Field must have maximum value of 4.

# Tareas de usuario – Validación de formularios (V)

Validación desde el formulario embebido:

**Forms**

Type  
<none>

Form Fields x +

- dni
- name

**Form Field**

ID (process variable name)  
dni x

Type  
string

Label

Default Value

**Validation**

Add Constraint +

Name	Config
readonly	<input type="checkbox"/>

**Forms**

Type  
<none>

Form Fields x +

- dni
- name

**Form Field**

ID (process variable name)  
name x

Type  
string

Label  
Nombre: x

Default Value

**Validation**

Add Constraint +

Name	Config
required	true x
minlength	5 x
maxlength	30 x

# Tareas de usuario – Validación de formularios (V)

Validación desde el formulario embebido:

Inicia un proceso proporcionando el valor de la variable **dni**

Name	Type	Value
dni	String	01234567X

## Formulario

**Form** History Diagram Description

01234567X

**Nombre:**

Required field

Save Complete

**Nombre:**

Use al menos 5 caracteres (actualmente está usando 4 caracteres).

Complete

**Nombre:**

Save Complete

# Tareas de usuario – Edición de formularios

A través del campo text se pueden incluir otros componentes al formulario.

Inicia un proceso proporcionando el

form\_1.form

FORM ELEMENTS LIBRARY

Text Field

Number

☒

Checkbox

☐

Radio

Select


ABC

Text

Button

This is a heading

This shows an image:



This is a sub-heading

Text can be shown for example using , or font.

- This is an unordered list...
- ...with two list items

- This is an ordered list...
- ...with two list items

HTML

ABC TEXT

<h1>This is a heading</h1> Th...

General

Text

<h1>This is a heading</h1>

This shows an image:



<h2>This is a sub-heading</h2>

Text can be shown for example using <b>bold</b>, or <i>italic</i> font.

<ul>

<li>This is an unordered list...</li>

<li>...with two list items</li>

</ul>

<ol>

<li>This is an ordered list...</li>

<li>...with two list items</li>

</ol>

Use Markdown or basic HTML to format.

Markdown

ABC TEXT

# This is a heading This show...

General

Text

# This is a heading

This shows an image:



## This is a sub-heading

Text can be shown for example using **bold**, or *italic* font.

\* This is an unordered list...

\* ...with two list items

1. This is an ordered list...

2. ...with two list items

Use Markdown or basic HTML to format.

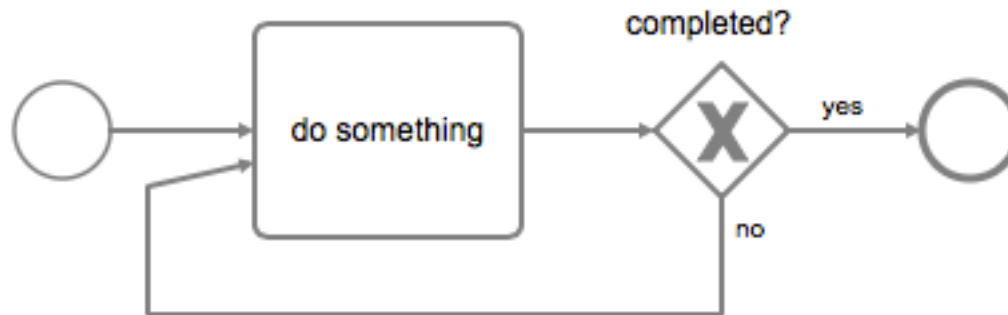
Introducción  
Conceptos básicos  
Tareas Automáticas  
Tareas de Usuario  
**Tareas de repetición**

## Tareas con Loops



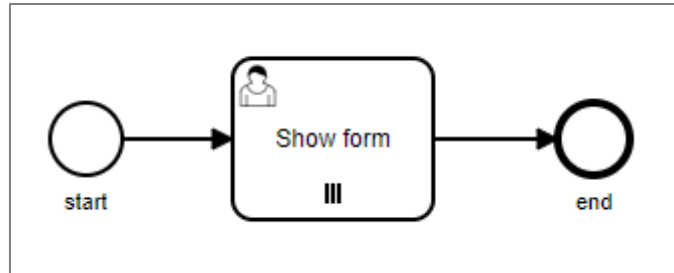
# Tareas con Loops

Las **tareas de loop no están implementadas** en Camunda. Por tanto, es necesario modificarlos a un ciclo con un gateway.



## Tareas Multi-Instance

# Tareas Multi-Instance



**Multi Instance**

Loop Cardinality

3| x

Collection

Element Variable

Completion Condition

☐ Multi Instance Asynchronous Before

☐ Multi Instance Asynchronous After

- **Cardinality:** Especifica el número de instancias.

# Tareas Multi-Instance

Candidate Groups

GPS

Camunda Tasklist

Keyboard Shortcuts Create task Start process Demo Demo

Create a filter + Created + Filter Tasks 3

Grupo GPS (3)

My tasks

All Tasks

Show form

Show Form Process

Created a few seconds ago

Show form

Show Form Process

Created a few seconds ago

Show form

Show Form Process

Created a few seconds ago

50

50

50

Show form

Show Form Process

Set follow-up date Set due date GPS Claim

Form History Diagram Description

Registro de alumnos - Multi Instancia

DNI\*

Nombres:

Apellidos:\*

Correo electrónico:

Curso\*

Select\*

Listado de asignaturas disponibles

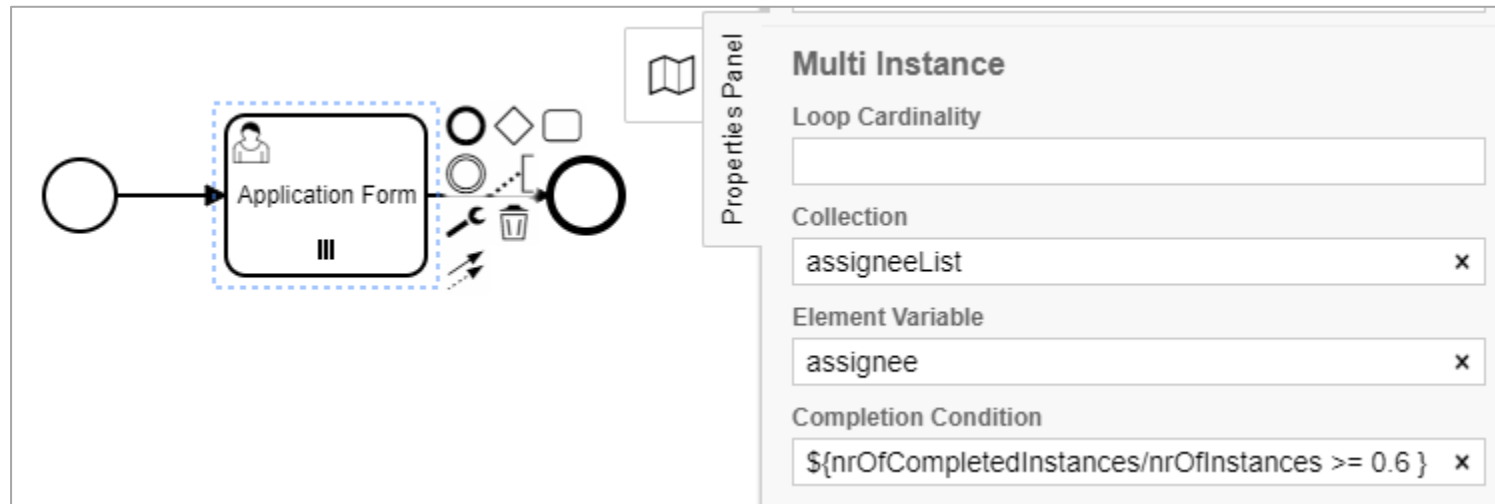
☐ Acepta las condiciones

Registrar

Save Complete

Se lanzan “en paralelo”


# Tareas Multi-Instance



- **Collection:** Variable que es un collection de Java sobre la que itera el multi-instance.
- **Element Variable:** Variable que contiene el elemento del Collection en cada iteración
- **Completion Condition:** Condición para terminar la tarea, antes de que terminen las instancias













# Tareas Multi-Instance

- Desde cada instancia se puede consultar las siguientes variables:
  - nrOfInstances***: El número total de instancias creadas
  - nrOfActiveInstances***: El número de instancias que quedan activas
  - nrOfCompletedInstances***: El número de instancias ya completas
  - loopCounter***: El índice en el iterador para cada instancia



The diagram shows a BPMN process starting with a 'start' circle, followed by a multi-instance task 'Show form' (indicated by a blue box with a person icon and a loop counter of 3), and ending with an 'end' circle.

Below the diagram, the 'Variables' tab is selected, showing a table of variables for the current instance. The table is highlighted with a red border.

Name	Type	Value	Scope	Actions
loopCounter	Integer	0	Show form	 
loopCounter	Integer	1	Show form	 
loopCounter	Integer	2	Show form	 
nrOfActiveInstances	Integer	3	Activity_Form#multiInstanceBody	 
nrOfCompletedInstances	Integer	0	Activity_Form#multiInstanceBody	 
nrOfInstances	Integer	3	Activity_Form#multiInstanceBody	 

Introducción  
Conceptos básicos  
Tareas Automáticas  
Tareas de Usuario  
Tareas de repetición  
**Fechas y eventos de tiempo**

# Fechas

Las fechas en Camunda se pueden definir de varias formas:

- **String**: Siguiendo el formato `yyyy-MM-dd'T'HH:mm:ss`
  - Por ejemplo: `2021-12-14T17:35:00`
- **String**: Siguiendo el formato ISO8601
  - Por ejemplo:
    - Duración de dos meses → `PT2M`
    - Duración de 10 días → `PT10D`
    - Time Cycle: 3 intervalos de repetición, cada uno de ellos de 10 horas de duración → `R3/PT10H`
- **Date**: Capturando el valor como variable de un tipo Date de Java
  - Por ejemplo: `${date}` en Camunda, `Date date`, en la clase Java



**Fechas de vencimiento y seguimiento**

# Fechas de vencimiento y seguimiento

- **Due Date** (Fecha de vencimiento)

Se puede especificar en las tareas de usuario

The screenshot shows a BPMN diagram with a task named 'Application Form' highlighted. The Properties Panel on the right is open, showing various configuration options. The 'Due Date' field is highlighted with a red arrow pointing to a text box containing '2021-12-15T23:59'.

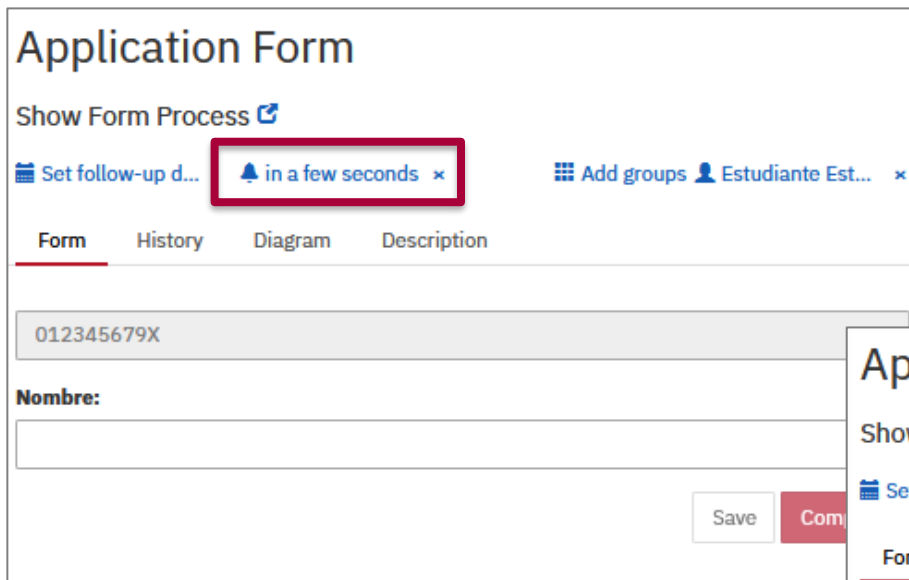
**Formatos permitidos:**

- **Variable:** `${someDate}` – Tipo Date de Java
- **String** en formato ISO8601: `2021-12-19T10:15`
- **Null** (no especificar)

# Fechas de vencimiento y seguimiento

- **Due Date** (Fecha de vencimiento)

Antes de que se cumpla el plazo



Application Form

Show Form Process [🔗](#)

Set follow-up d... **🔔 in a few seconds ×** Add groups [👤 Estudiante Est... ×](#)

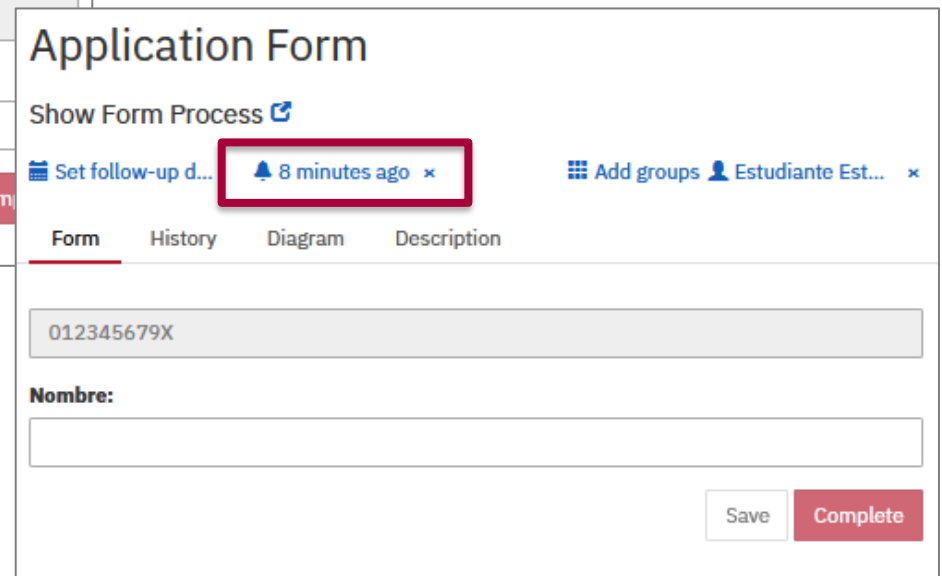
Form History Diagram Description

012345679X

Nombre:

Save Complete

Cuando se ha cumplido el plazo



Application Form

Show Form Process [🔗](#)

Set follow-up d... **🔔 8 minutes ago ×** Add groups [👤 Estudiante Est... ×](#)

Form History Diagram Description

012345679X

Nombre:

Save Complete

# Fechas de vencimiento y seguimiento

- **Follow Up Date** (Fecha de seguimiento)

Se pueden especificar en las tareas de usuario

The screenshot shows a BPMN editor window titled 'forms\_emb.bpmn'. The main canvas displays a process flow starting with a 'start' circle, followed by a task named 'Application Form' (represented by a rounded rectangle with a person icon). The task is highlighted with a blue dashed border. To the right of the task is a 'Properties Panel' with a 'Details' tab. The 'Details' tab contains several fields: 'Assignee' (set to 'estudiantegps'), 'Candidate Users', 'Candidate Groups', 'Due Date', and 'Follow Up Date'. The 'Follow Up Date' field is highlighted with a blue border and contains the value '2021-12-15T20:15'. Below this field, there is a text description: 'The follow up date as an EL expression (e.g. \${someDate}) or an ISO date (e.g. 2015-06-26T09:54:00)'. A red arrow points from the 'Follow Up Date' field in the Properties Panel to the 'Formatos permitidos' box.

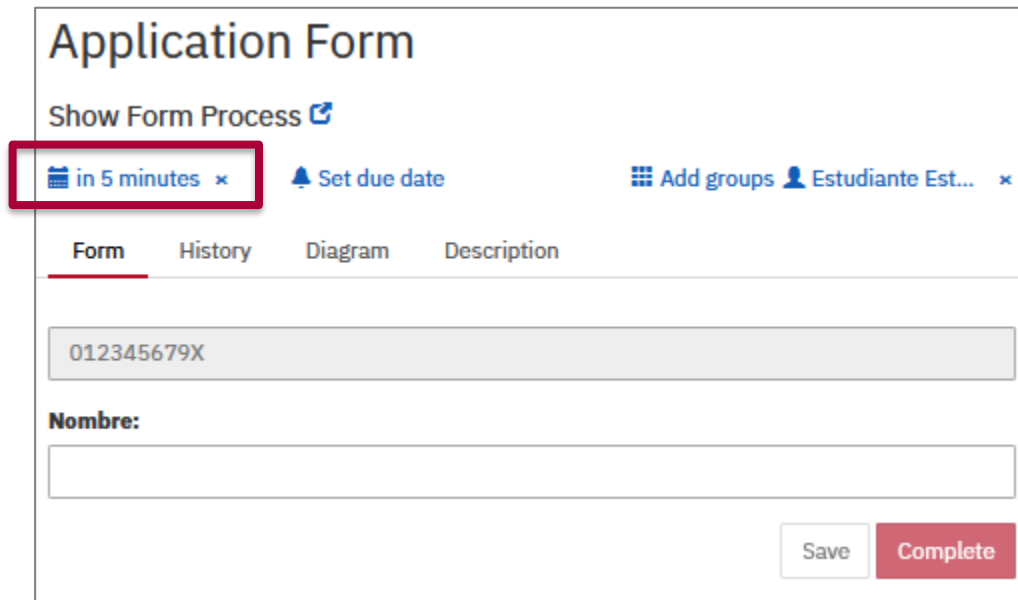
**Formatos permitidos:**

- **Variable:** `${someDate}` – Tipo Date de Java
- **String** en formato ISO8601: 2021-12-19T10:15
- **Null** (no especificar)

# Fechas de vencimiento y seguimiento

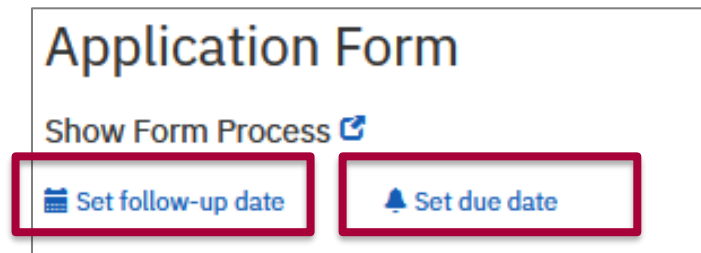
- **Follow Up Date** (Fecha de seguimiento)

Se muestra el tiempo restante a la fecha indicada.



The screenshot shows the 'Application Form' interface. At the top, there is a 'Show Form Process' link. Below it, a red box highlights the 'in 5 minutes' option next to a calendar icon. To the right of this, there is a 'Set due date' button with a bell icon. Further right, there is an 'Add groups' button and a user selection dropdown showing 'Estudiante Est...'. Below these options are tabs for 'Form', 'History', 'Diagram', and 'Description'. The 'Form' tab is active, showing a text input field with the value '012345679X'. Below this is a label 'Nombre:' followed by another text input field. At the bottom right, there are 'Save' and 'Complete' buttons.

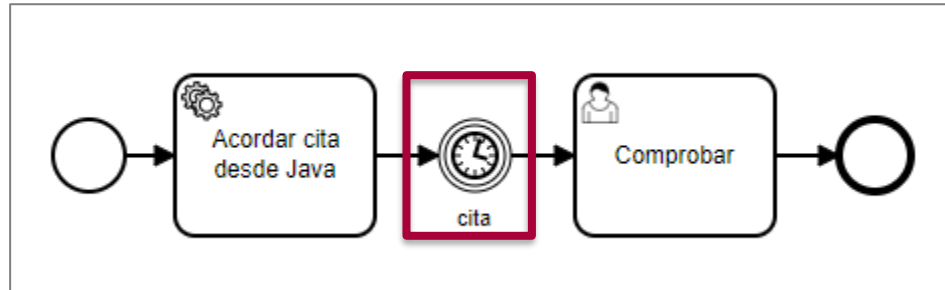
Si no se especifica la fecha antes del despliegue, puede asignarse posteriormente



This screenshot shows the same 'Application Form' interface, but with two buttons highlighted by red boxes: 'Set follow-up date' (with a calendar icon) and 'Set due date' (with a bell icon). The 'Set follow-up date' button is on the left, and the 'Set due date' button is on the right. The rest of the interface, including the 'Show Form Process' link and the form fields, is the same as in the previous screenshot.

## Eventos de tiempo

# Eventos de tiempo

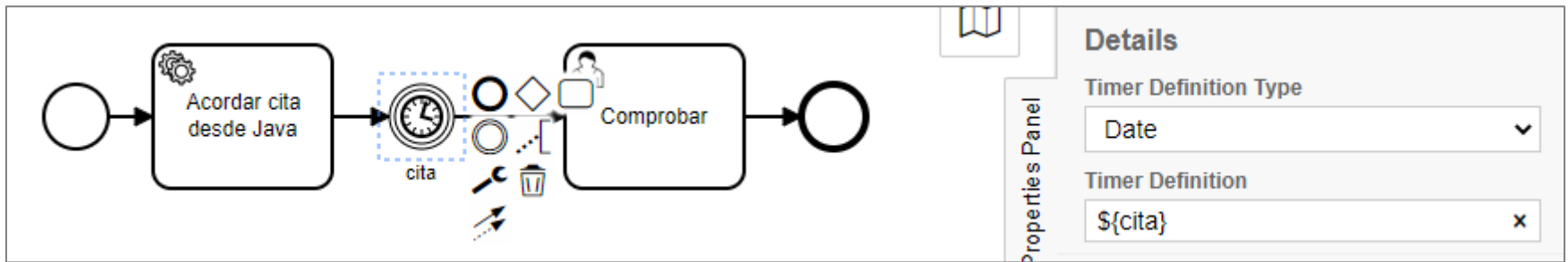


Es necesario definir un “***Timer Definition Type***”

- **Date**: Evalúa una fecha dada
- **Duration**: Analiza un periodo.
- **Cycle**: Define periodos de repetición

Los valores permitidos son strings que siguen el estándar ISO 8601 o variables que capturan tipos Date.

# Eventos de tiempo



## Ejemplos:

### Date

- Timer Definition: `${cita}`

### Date

- Timer Definition: `2021-02-03T16:08:00`

### Duration

- Intervalo de 10 días: `P10D`

### Duration

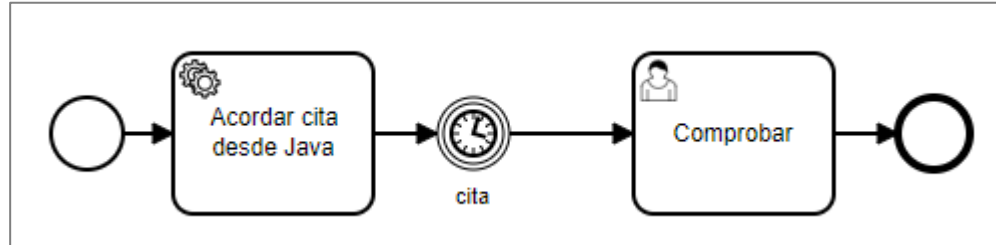
- Intervalo de 2 meses: `PT2M`

### Cycle

- 3 intervalos de repetición, cada uno de ellos de 10 horas de duración: `R3/PT10H`



# Eventos de tiempo con tareas externas de servicio (I)



Modela el proceso “**Process Dates**” que se muestra en la imagen.

Configura la tarea de servicio como:

- Implementation: **External**
- Topic: **defineJavaDate**

Configura el evento:

- Timer Definitio Type: **Date**
- Timer definition: **\${cita}**
- Marca la opción: **Asynchronous Before**

# Eventos de tiempo con tareas externas de servicio (I)

Asigna la tarea de usuario a **demo** y crea un formulario según se muestra:

**Forms**

Type  
<none>

Form Fields

- cita
- currentDate
- approve

**Form Field**

ID (process variable name)  
cita

Type  
date

Label  
Fecha

Default Value

**Validation**

Add Constraint +

Name	Config
readonly	

**Forms**

Type  
<none>

Form Fields

- cita
- currentDate
- approve

**Form Field**

ID (process variable name)  
currentDate

Type  
date

Label  
Fecha en que se registra la cita:

Default Value

**Validation**

Add Constraint +

Name	Config
readonly	

**Forms**

Type  
<none>

Form Fields

- cita
- currentDate
- approve

**Form Field**

ID (process variable name)  
approve

Type  
boolean

Label  
Aprobar?

Default Value

**Validation**

Add Constraint +

**Properties**

# Eventos de tiempo con tareas externas de servicio (II)

## Configura la tarea de servicio

Subscribe un nuevo cliente a la clase **Worker**, según se muestra:

```
client.subscribe("defineJavaDate")
    .lockDuration(1000)
    .handler((externalTask, externalTaskService) -> {

        LOGGER.info("Definiendo la fecha desde JAVA...");

        Map<String, Object> variables = new HashMap<String, Object>();

        Date cita = null;
        Date currentDate = null;

        SimpleDateFormat dateFormat = new SimpleDateFormat("yyyy-MM-dd'T'HH:mm:ss");
        try {
            cita = dateFormat.parse("2021-12-15T20:59:00");
            //Esto se puede definir como mejor convenga.
            currentDate = dateFormat.parse(LocalDate.now().toString());
        } catch (ParseException e) {
            e.printStackTrace();
        }

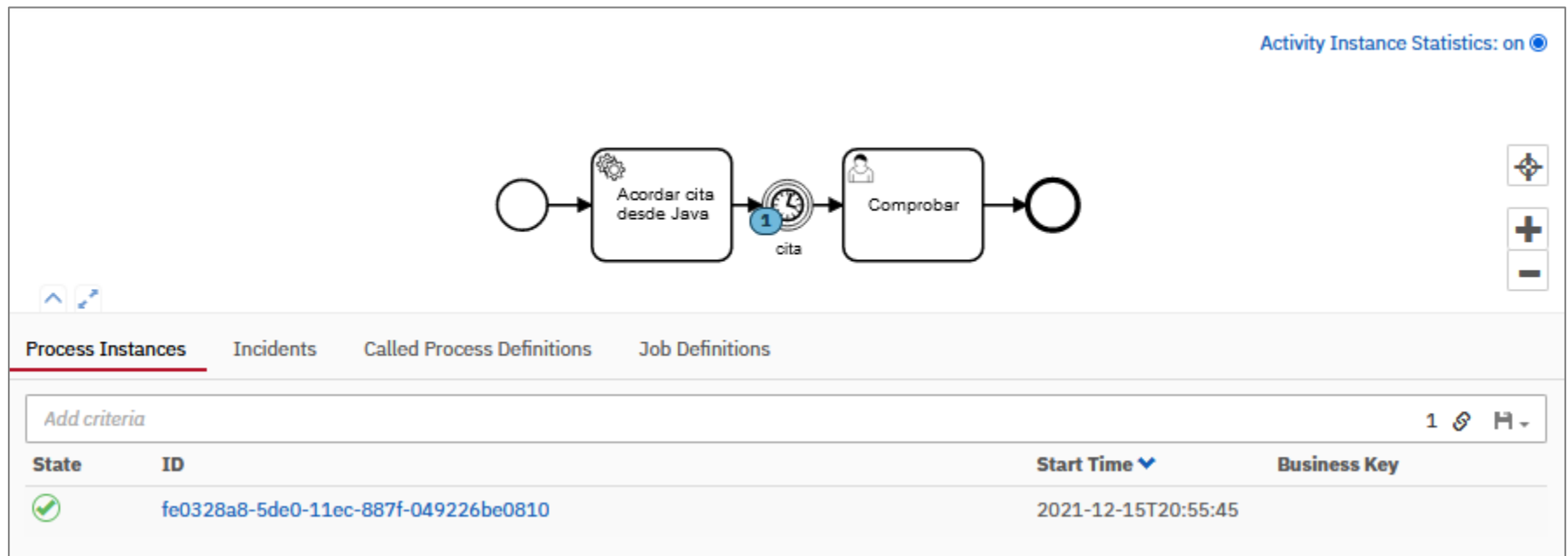
        variables.put("cita", cita);
        variables.put("currentDate", currentDate);

        LOGGER.info("Fin de tarea...\n  -" + cita + "\n  -"+currentDate);
        externalTaskService.complete(externalTask, variables);
    })
    .open();
```

# Eventos de tiempo con tareas externas de servicio (III)

## Proceso en funcionamiento

- Ejecuta la clase Java que acabamos de configurar
- Inicia una instancia del proceso “**Process Dates**”
- Verifica la consola de Eclipse
- Verifica la consola **Camunda Cockpit**



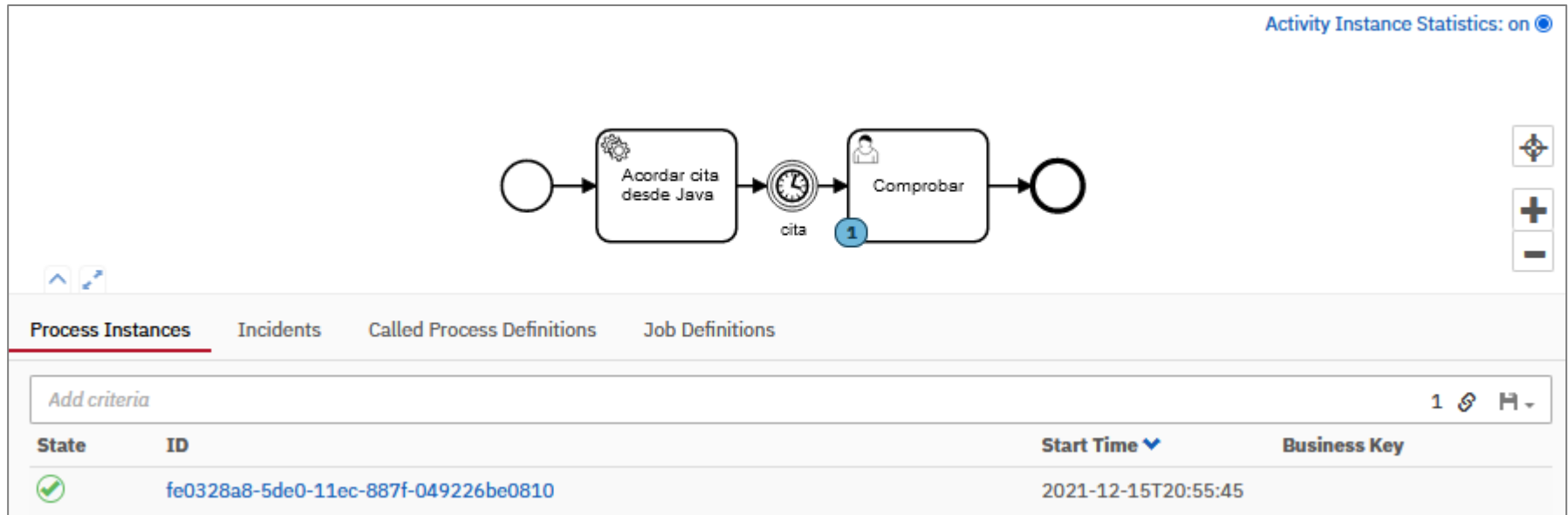
The screenshot displays the Camunda Cockpit interface for a running process instance. At the top right, it says "Activity Instance Statistics: on". The process diagram shows a flow from a start node to a task "Acordar cita desde Java", followed by a timer event "cita" (indicated by a clock icon and a blue circle with the number 1), then to a task "Comprobar", and finally to an end node. Below the diagram, there are tabs for "Process Instances", "Incidents", "Called Process Definitions", and "Job Definitions". The "Process Instances" tab is active, showing a table with one instance.

State	ID	Start Time	Business Key
✓	fe0328a8-5de0-11ec-887f-049226be0810	2021-12-15T20:55:45	

# Eventos de tiempo con tareas externas de servicio (III)

## Proceso en funcionamiento

- Cuando se cumpla el tiempo especificado en `${cita}`, el proceso avanza y el formulario está disponible para el usuario **demo**



## EJERCICIO

- Cree el siguiente modelo para que capture datos de una reserva en el formulario. Si la fecha está dentro de un periodo la acepte o de lo contrario rechace la reserva. Si la reserva es válida, cuando llegue la fecha dada, envíe un correo electrónico informando sobre la cita.
- Recuerde que las fechas deben proporcionarse en formatos válidos.
- Gestione las fechas en el formulario como string.

Introducción  
Conceptos básicos  
Tareas Automáticas  
Tareas de Usuario  
Tareas de repetición  
Fechas y eventos de tiempo  
**Consejos**

# Consejos: Usar Loggers

```
public class SendApprovalMail {
    final static Logger LOGGER = Logger.getLogger(SendApprovalMail.class
        .getName());

    public static void main(String[] args) {
        ExternalTaskClient client = ExternalTaskClient.create()
            .baseUrl("http://localhost:8080/engine-rest").build();

        SendMail sendMail = new SendMail();

        // subscribe to an external task topic as specified in the process
        client.subscribe("send-approval")
            .lockDuration(1000)
            // the default lock duration is 20 seconds, but you can override
            // this
            .handler((externalTask, externalTaskService) -> {
                // Get a process variable

                String mail = (String) externalTask.getVariable("item");
                Long amount = (Long) externalTask.getVariable("amount");
                String result = (String) externalTask
                    .getVariable("result");
                LOGGER.info(result + " amount of '" + amount
                    + "'€ for the item '" + mail + "'...");
            });
    }
}
```



# REFERENCIAS

# Manuales de referencia

- Sobre el uso de BPMN 2.0 en Camunda:  
<https://docs.camunda.org/manual/latest/reference/bpmn20/>
- Sobre el uso de código Java con Camunda con External Tasks:  
<https://docs.camunda.org/manual/latest/user-guide/ext-client/>
- Sobre los formularios: <https://docs.camunda.org/manual/latest/user-guide/task-forms/#generated-task-forms>
- <https://docs.camunda.org/manual/7.16/user-guide/task-forms/#camunda-forms>
- Sobre DMN en Camunda: <https://docs.camunda.org/get-started/quick-start/decision-automation/>