



P8 – Primeros pasos con Camunda

Gestión de Procesos y Servicios

INTRODUCCIÓN

INSTALACIÓN

MODELADO DE UN PROCESO

**IMPLEMENTAR UN “EXTERNAL TASK
WORKER”**

DESPLIEGUE

TAREAS DE USUARIO

GATEWAYS

EJEMPLO EXTRA

RESUMEN

INTRODUCCIÓN

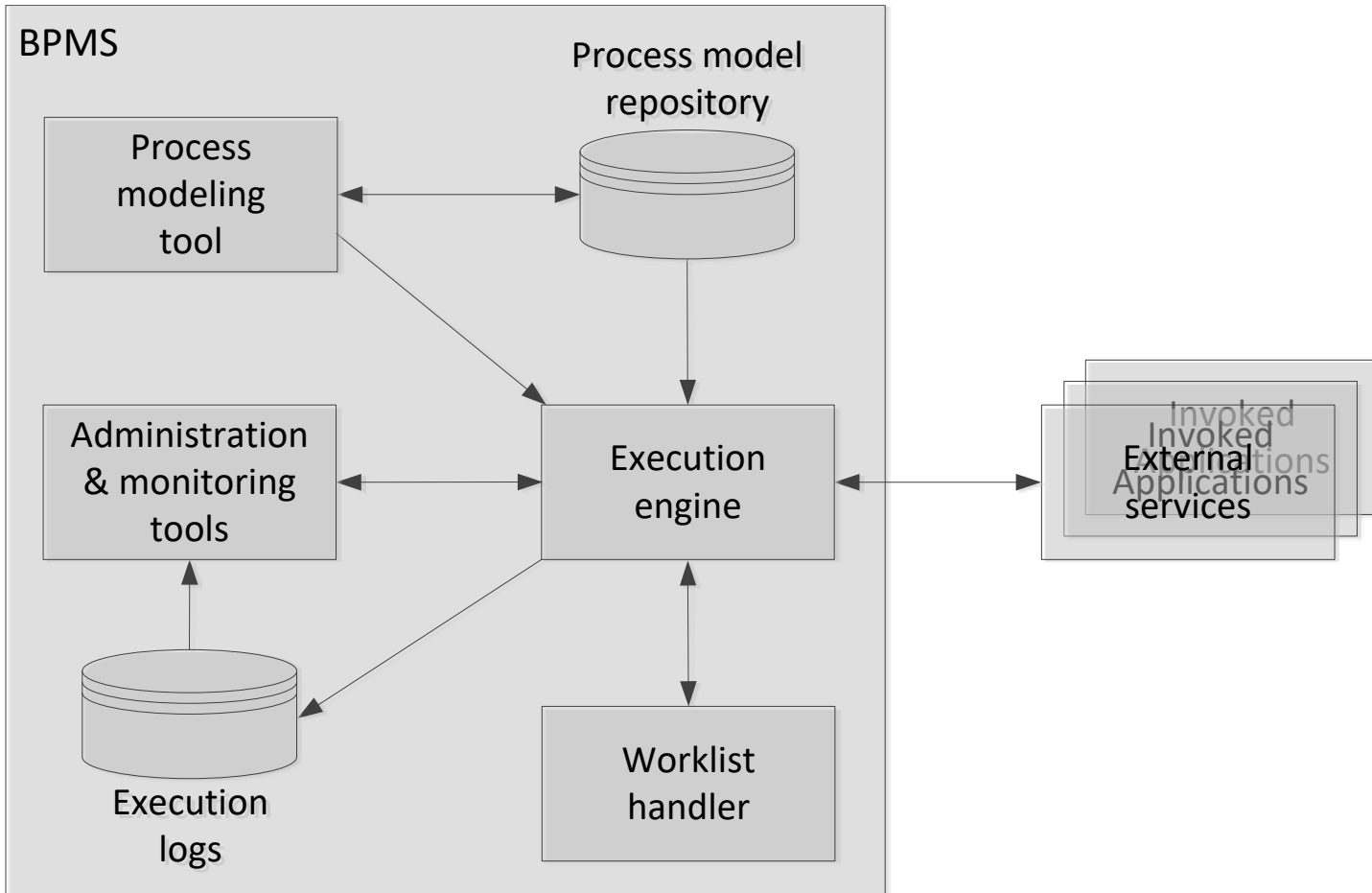
Business Process Management System (BPMS)

Sistema software genérico que está dirigido por la representación explícita de representaciones de procesos para coordinar la realización de procesos de negocio

BPMS debe dar soporte a:

- Modelado de procesos de negocio
 - Incluye visualización, simulación, definición de reglas de negocio
- Ejecución de procesos de negocio
 - Es habitual usar sistemas de workflow
- Medición de procesos de negocio
- Análisis, monitorización y auditoría de procesos de negocio
- Optimización de procesos de negocio

La arquitectura de un BPMS



Camunda BPMS



Herramientas a utilizar

Para trabajar con Camunda usaremos dos herramientas:

- Modelador de Procesos
- Motor de ejecución de Procesos

INSTALACIÓN

¡IMPORTANTE!

Todo lo que viene a continuación es válido para la versión **7.16.0 de Camunda**.

Si usas otra versión, las cosas pueden funcionar de manera distinta.

Requisitos

- Java JDK 11+
- Apache Maven o Eclipse IDE for Java and Web Developers
<https://www.eclipse.org/downloads/eclipse-packages/>

Eclipse IDE for Enterprise Java and Web Developers

517 MB 775,864 DOWNLOADS



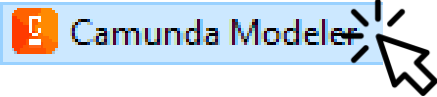
Tools for developers working with Java and Web applications, including a Java IDE, tools for JavaScript, TypeScript, JavaServer Pages and Faces, Yaml, Markdown, Web Services, JPA and Data Tools, Maven and Gradle, Git, and more.



Windows x86_64
macOS x86_64
Linux x86_64 | AArch64

- Navegador reciente (Firefox, Chrome, Microsoft Edge)


Instalación del Modelador de procesos (Camunda Modeler)


- Es una aplicación de escritorio para modelar diagramas de proceso BPMN 2.0 (y también reglas de decisión, aunque nosotros no lo veremos)
- Descarga **Camunda Modeler (4.11.1)** del siguiente enlace:
 - <https://camunda.org/download/modeler/>
- Descomprímelo → A screenshot of a file explorer window showing a folder named 'Camunda Modeler'. The folder icon is a blue square with a white 'C' and an orange square. A mouse cursor is pointing at the folder.
- Puedes encontrar los requisitos e instrucciones en:
 - <https://docs.camunda.org/manual/latest/installation/camunda-modeler/>

Instalación del Motor de procesos (Camunda BPM Platform)

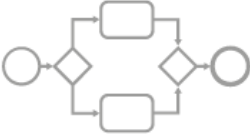
- Es un BPMS de Java para la automatización de procesos.
- Pasos
 1. Descarga Camunda BPM Platform (Open source community platform, versión 7.16.0): <https://camunda.org/download/>
 2. Descomprímelo
 3. Ejecuta “start.bat” (“start.sh” si usas Mac o Linux)
 4. Si no se abre ninguna página en tu navegador, ve a <http://localhost:8080/>
 5. Accede utilizando las credenciales
 - *username*: demo
 - *password*: demo

Instalación del Motor de procesos (Camunda BPM Platform)


 Camunda Welcome

 Demo Demo


Applications



Cockpit



Tasklist



Admin

Profile

Demo Demo
demo@localhost
[Edit profile](#) [Change password](#)

Groups

- camunda BPM Administrators

Links

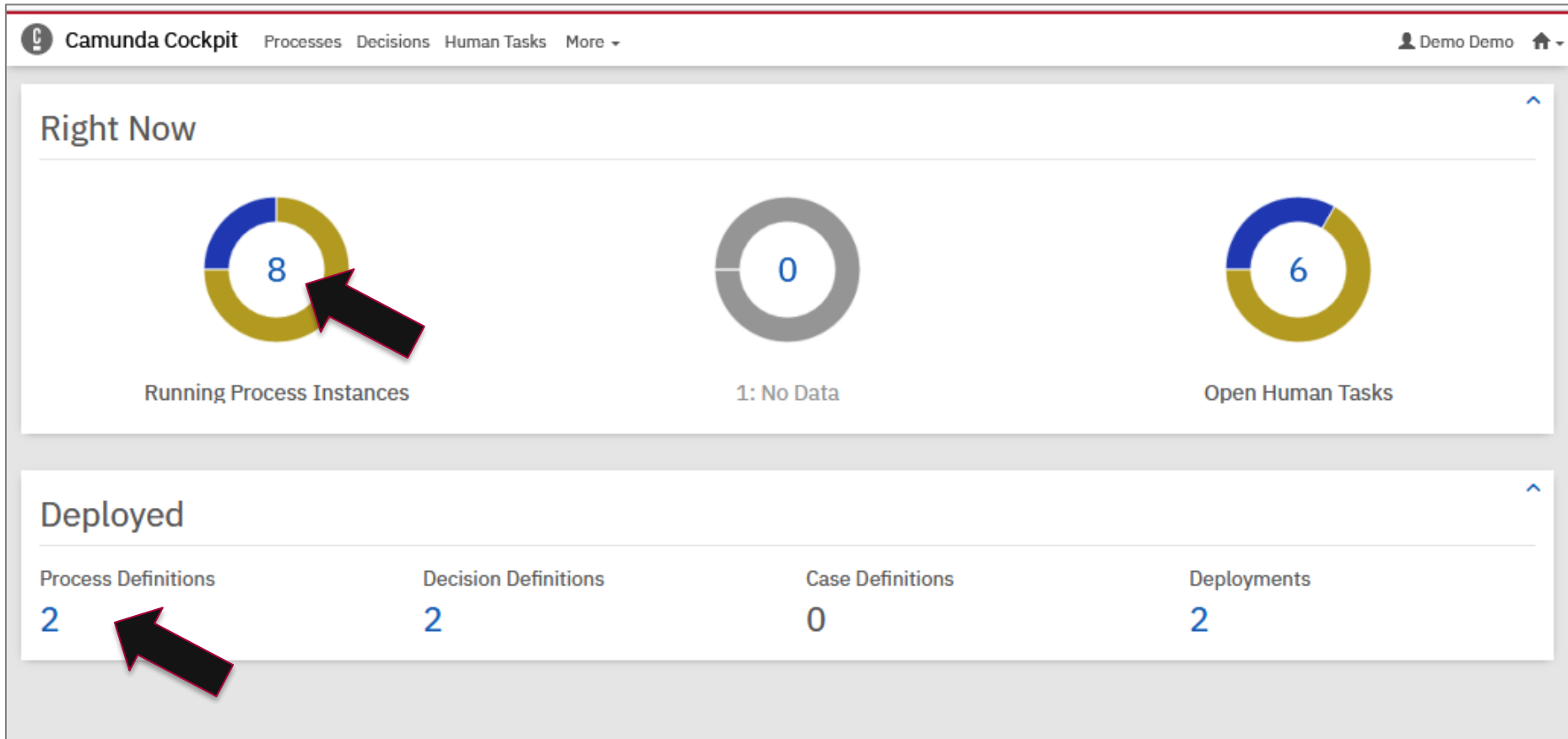
[Documentation](#)
Camunda webapps user documentation

Resolución de problemas

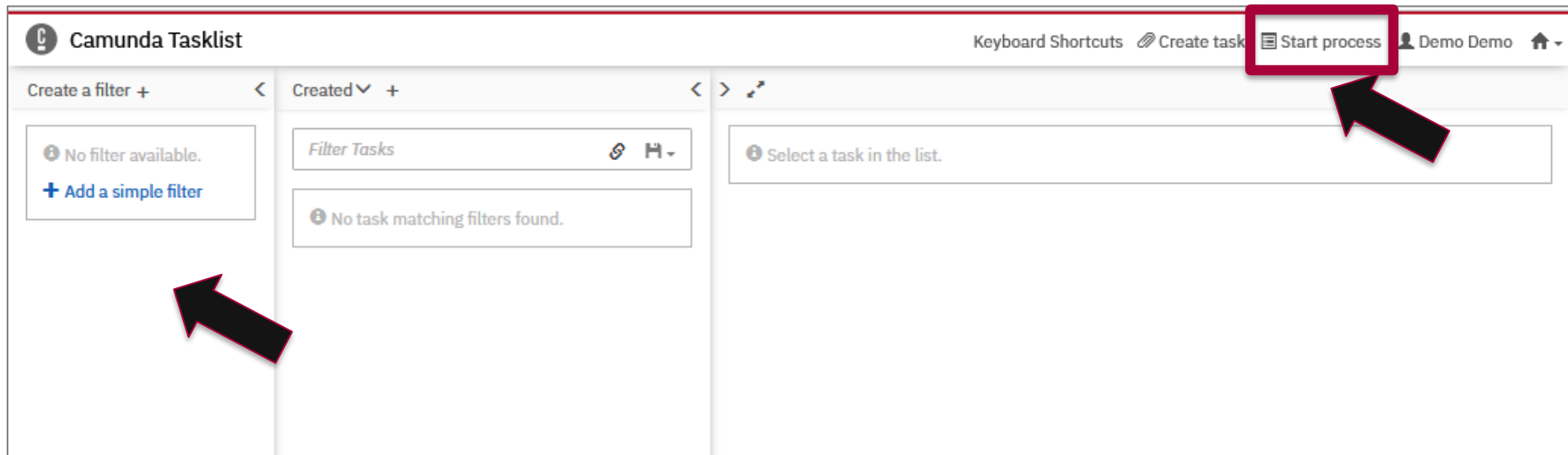
- Si no funciona, lo más probable es que el puerto 8080 esté ocupado por otro programa que use apache, como, por ejemplo, Alfresco. Para resolverlo:
 1. En `../<carpeta_camunda_7.16.0>\configuración` abre el fichero `“default.yml”`
 2. Edítalo. Agrega como última línea la siguiente instrucción, especificando el puerto que se desea utilizar, por ejemplo, el 8090:
`server.port: 8090`
 3. Vuelve a ejecutar `“start.bat”` (`“start.sh”` para Mac o Linux)
 4. Ahora prueba a abrir la aplicación en <http://localhost:8090> (o el puerto que hayas puesto).

Familiarízate con el Cockpit y la Tasklist del motor de procesos de Camunda.

Camunda BPM Platform - Cockpit



Camunda BPM Platform - Tasklist



MODELADO DE UN PROCESO

Crea tu primer proceso BPMN 2.0 con Camunda Modeler

Los pasos que se describen a continuación están basados en el tutorial disponible en:

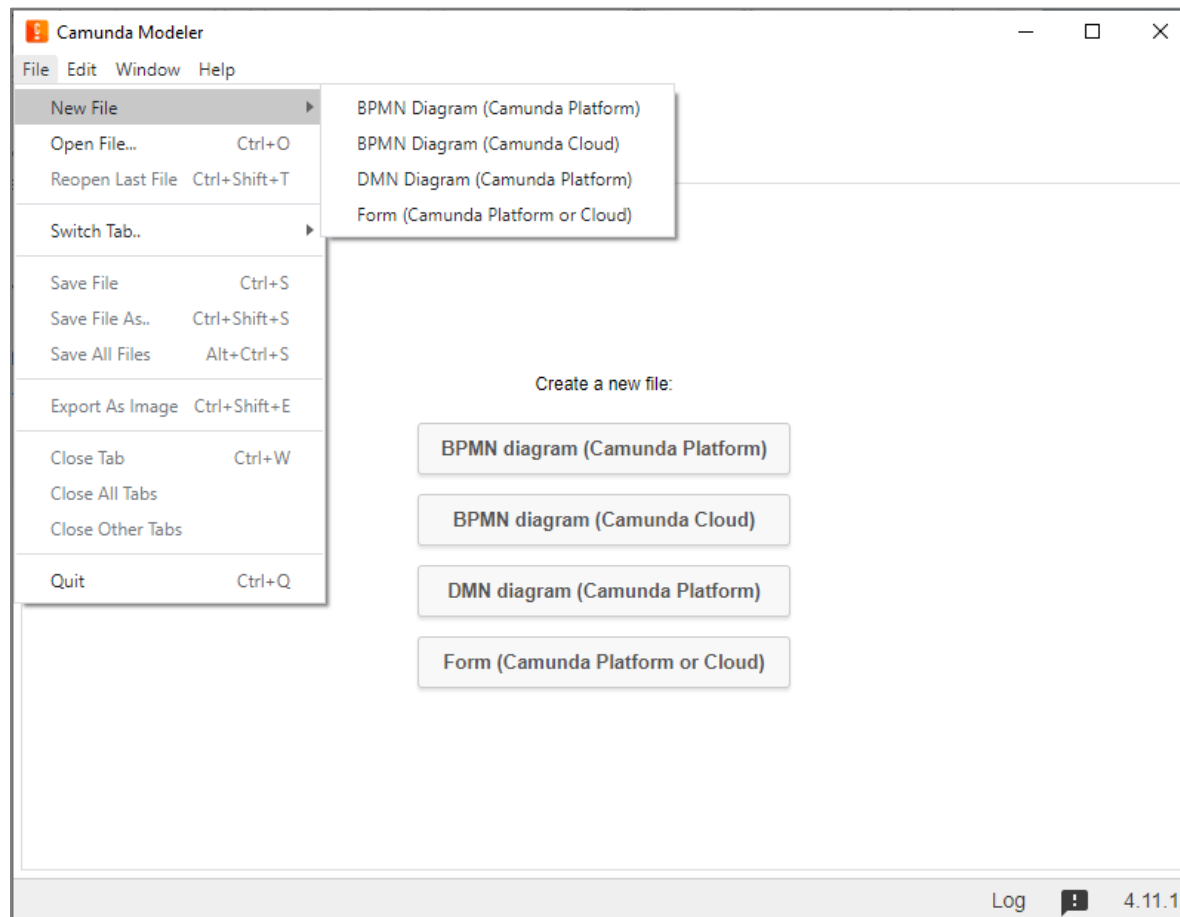
<https://docs.camunda.org/get-started/quick-start/>

Crea un diagrama nuevo BPMN 2.0 usando Camunda Modeler

1. Empieza con un proceso simple
2. Configura una tarea de servicio
3. Configura un proceso ejecutable
4. Guarda el diagrama BPMN

1. Crea un diagrama nuevo BPMN 2.0

Crea un nuevo diagrama BPMN haciendo click en **File > New File > BPMN Diagram**



2. Empieza con un proceso simple (I)

Double-click **Start Event** to edit label
(Use SHIFT + ENTER for line break)

Select activity shape (rectangle)

Payment Retri Requested

StartEvent_1

General Forms Listeners Extensions

General

Id
StartEvent_1

Name
Payment Retri Requested

Details

Initiator

Asynchronous Continuations

☐ Asynchronous Before
☐ Asynchronous After

Documentation

Element Documentation

2. Empieza con un proceso simple (II)

The screenshot shows a BPMN editor interface with a process diagram and a properties panel. The process diagram includes a start event labeled "Payment Retrieval Requested" connected to a task labeled "Charge Credit Card". The "Charge Credit Card" task is highlighted with a dashed blue border. A wrench icon is positioned next to it, and a blue callout box contains the instructions: "1. Select activity" and "2. Click wrench icon and change type to Service Task". Below the diagram, a task palette lists various activity types, with "Service Task" selected and highlighted. On the right, the "Properties Panel" for "serviceTask1" is open, showing tabs for "General", "Listeners", "Input/Output", and "Extensions". The "General" tab is active, displaying the "Name" field with the value "Charge Credit Card", the "Asynchronous Continuations" section with two unchecked checkboxes ("Asynchronous Before" and "Asynchronous After"), and the "Documentation" section with an "Element Documentation" text area.

step1.2.bpmn x

Payment Retrieval Requested

Charge Credit Card

1. Select activity
2. Click wrench icon and change type to Service Task

serviceTask1

General Listeners Input/Output Extensions

Name
Charge Credit Card

Asynchronous Continuations

☐ Asynchronous Before
☐ Asynchronous After

Documentation
Element Documentation

Properties Panel

- Send Task
- Receive Task
- User Task
- Manual Task
- Business Rule Task
- Service Task
- Script Task
- Call Activity
- Sub Process

2. Empieza con un proceso simple (III)

The screenshot displays a BPMN editor interface. On the left, a toolbar contains various BPMN symbols. The main workspace shows a process diagram with the following elements:

- Start Event: A circle labeled "Payment Retrieval Requested".
- Task: A rounded rectangle labeled "Charge Credit Card" with a gear icon.
- End Event: A circle labeled "Payment Received" with a checkmark icon. It is highlighted with a blue dashed border.
- Flow: An arrow connects the start event to the task, and another arrow connects the task to the end event.
- Annotation: A blue box labeled "Add an End Event" with an arrow pointing to the end event.

On the right, the "Properties Panel" is open, showing the configuration for the selected "endEvent1".

endEvent1

- General** (selected tab):
 - Id**: endEvent1
 - Name**: Payment Received
- Asynchronous Continuations**:
 - ☐ Asynchronous Before
 - ☐ Asynchronous After
- Documentation**:
 - Element Documentation**: (empty text area)

3. Configura una tarea de servicio

The screenshot displays the Camunda BPMN Designer interface. On the left, a palette of BPMN symbols is visible. The main canvas shows a BPMN diagram with a start event, a task named 'Charge Credit Card', and an end event. The 'Charge Credit Card' task is highlighted with a blue dashed border. To the right, the 'Properties Panel' is open, showing the configuration for 'serviceTask1'. The 'General' tab is selected, and the 'Name' is 'Charge Credit Card'. The 'Details' section shows the 'Implementation' set to 'External' and the 'Topic' set to 'charge-card'. The 'External Task Configuration' section is also visible.

step1.3.bpmn x

Payment Retrieval Requested

Charge Credit Card

Payment Received

If it's not already visible, click on the **Properties Panel** tab

Select **Implementation**

Set a topic

serviceTask1

General Listeners Input/Output Field Injections Extension >

General

Id
serviceTask1 x

Name
Charge Credit Card

Details

Implementation
External v

Topic
charge-card x

External Task Configuration

Task Priority

Asynchronous Continuations

☐ Asynchronous Before

☐ Asynchronous After

Documentation

4. Configura las propiedades de ejecución

The screenshot displays the Camunda BPMN Editor interface. The main workspace shows a BPMN diagram for a process named 'payment-retrival'. The diagram includes a start event 'Payment Retrival Requested', a task 'Charge Credit Card', and an end event 'Payment Received'. A red dashed box highlights the configuration area, which includes three blue boxes: 'Set Process ID', 'Set Process Name', and 'Mark the process to be executable'. A mouse cursor points to the 'Mark the process to be executable' box, with a note 'To configure the process properties'.

The right-hand panel shows the configuration for the 'payment-retrival' process. The 'General' tab is selected, showing the following settings:

- Id:** payment-retrival (This maps to the process definition key.)
- Name:** Payment Retrival
- Version Tag:**
- Executable:** ☒
- External Task Configuration:**
 - Task Priority:**
- Job Configuration:**
 - Job Priority:**
- Candidate Starter Configuration:**
 - Candidate Starter Groups:**
 - Specify more than one group as a comma separated list.

5. Guarda el diagrama BPMN

The screenshot displays a BPMN editor interface. On the left, the **File** menu is open, showing options like **New File**, **Open File...**, **Reopen Last File**, **Switch Tab..**, **Save File** (highlighted with **Ctrl+S**), **Save File As..**, **Save All Files**, **Export As Image**, **Close Tab**, **Close All Tabs**, **Close Other Tabs**, and **Quit**. The main workspace shows a BPMN diagram with a task labeled **Charge Credit Card** connected to an event labeled **Payment Received**. A blue callout box with an arrow pointing to the **Save File** menu item contains the text: **Assign a name to the file** (*payment.bpmn*). On the right, the **Properties Panel** is visible for the selected task, titled **payment-retrival**. It includes tabs for **General**, **Variables**, **Listeners**, and **Extensions**. The **General** tab is active, showing fields for **Id** (payment-retrival), **Name** (Payment Retrival), and **Version Tag**. There is a checked **Executable** checkbox. Below are sections for **External Task Configuration** (Task Priority), **Job Configuration** (Job Priority), and **Candidate Starter Configuration** (Candidate Starter Groups).

**IMPLEMENTAR UN “EXTERNAL TASK
WORKER”**

Pasos para crear un “External Task Worker”

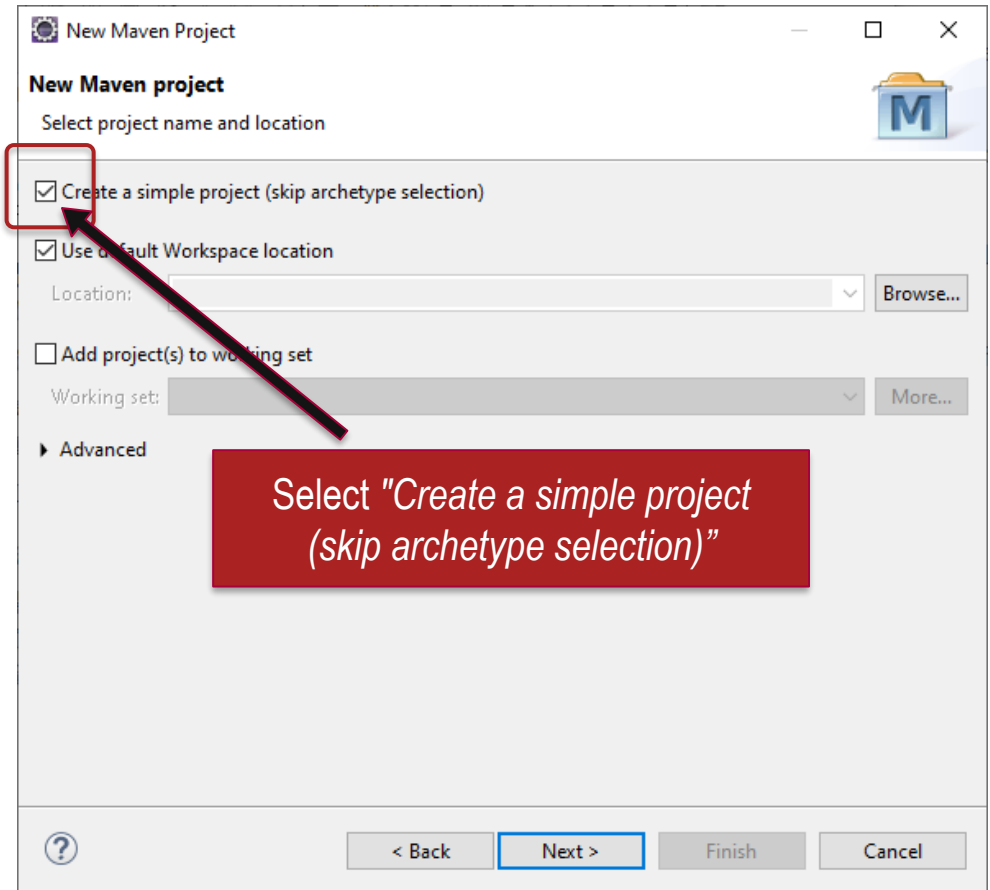
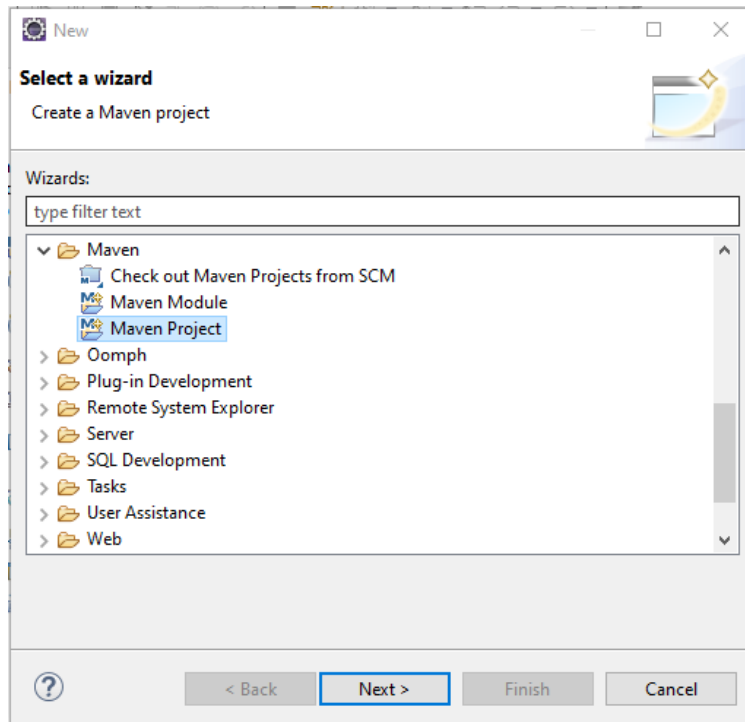
Para crear un **External Task Worker**, encargado de gestionar las tareas externas, debes realizar los siguientes pasos:

1. Crea un proyecto Maven, por ejemplo Eclipse
2. Añade las dependencias de Maven a Camunda
3. Añade una clase Java **ExternalTaskClient**
4. Ejecuta la clase Java

1. Crea un proyecto Maven en Eclipse (I)

- En Eclipse:

**File > New > Other >
Maven > Maven Project**



1. Crea un proyecto Maven nuevo en Eclipse (II)

New Maven Project
Configure project

Artifact

Group Id:

Artifact Id:

Version:

Packaging:

Name:

Description:

Parent Project

Group Id:

Artifact Id:

Version:

► Advanced

Second page:
Configure the project

Finish

2. Añade las dependencias de Maven a Camunda

```
<project xmlns="http://maven.apache.org/POM/4.0.0" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/xsd/maven-4.0.0.xsd">
  <modelVersion>4.0.0</modelVersion>

  <groupId>org.camunda.bpm.getstarted</groupId>
  <artifactId>charge-card-worker</artifactId>
  <version>0.0.1-SNAPSHOT</version>

  <properties>
    <camunda.external-task-client.version>7.16.0</camunda.external-task-client.version>
    <maven.compiler.source>1.8</maven.compiler.source>
    <maven.compiler.target>1.8</maven.compiler.target>
  </properties>

  <dependencies>
    <dependency>
      <groupId>org.camunda.bpm</groupId>
      <artifactId>camunda-external-task-client</artifactId>
      <version>${camunda.external-task-client.version}</version>
    </dependency>
    <dependency>
      <groupId>org.slf4j</groupId>
      <artifactId>slf4j-simple</artifactId>
      <version>1.6.1</version>
    </dependency>
    <dependency>
      <groupId>javax.xml.bind</groupId>
      <artifactId>jaxb-api</artifactId>
      <version>2.3.1</version>
    </dependency>
  </dependencies>
</project>
```

Añade las siguientes propiedades y dependencias
al fichero "**pom.xml**"

(cópialas del enlace del tutorial)

3. Añade una clase Java `ExternalTaskClient` (I)

- Crea una `ExternalTaskClient` que se suscribirá al topic “charge-card”.
- Cuando el motor de procesos encuentra una tarea de servicio que está configurada para ser gestionada externamente, crea una instancia de tarea externa sobre la cual nuestro handler reaccionará. Estamos usando “Long Polling” en el `ExternalTaskClient` para hacer que la comunicación sea más eficiente.
- Crea un paquete, por ejemplo `org.camunda.bpm.getstarted.chargecard`, y añádele una clase Java, por ejemplo, `ChargeCardWorker`

3. Añade una clase Java ExternalTaskClient (II)

New Java Class

Java Class
Create a new Java class.

Source folder:

Package:

☐ Enclosing type:

Name:

Modifiers: ☒ public ☐ package ☐ private ☐ protected
☐ abstract ☐ final ☐ static

Superclass:

Interfaces:

Which method stubs would you like to create?

☐ public static void main(String[] args)

☐ Constructors from superclass

☒ Inherited abstract methods

Do you want to add comments? (Configure templates and default value [here](#))

☐ Generate comments

Class:
ChargeCardWorker

Package:
org.camunda.bpm.getstarted.chargecard

3. Añade una clase Java ExternalTaskClient (III)

```
package org.camunda.bpm.getstarted.chargecard;

import java.util.logging.Logger;
import java.awt.Desktop;
import java.net.URI;

import org.camunda.bpm.client.ExternalTaskClient;

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();
    }
}
```

3. Añade una clase Java ExternalTaskClient (IV)

The screenshot shows the Eclipse IDE with the file `ChargeCardWorker.java` open. The code is as follows:

```
1 package org.camunda.bpm.getstarted.chargecard;
2
3 import java.util.logging.Logger;
4 import java.awt.Desktop;
5 import java.net.URI;
6
7 import org.camunda.bpm.client.ExternalTaskClient;
8
9 public class ChargeCardWorker {
10     private final static Logger LOGGER = Logger.getLogger(ChargeCardWorker.class.getName());
11
12     public static void main(String[] args) {
13         ExternalTaskClient client = ExternalTaskClient.create()
14             .baseUrl("http://localhost:8080/engine-rest")
15             .asyncResponseTimeout(10000) // long polling timeout
16             .build();
17
18         // subscribe to an external task topic as specified in the process
19         client.subscribe("charge-card")
20             .lockDuration(1000) // the default lock duration is 20 seconds but you can override this
21             .run(() -> {
22                 // Put your logic here
23
24                 // Get a String i
25                 String i
26                 Long amo
27
28                 LOGGER.i
29
30                 try {
31                     Desk
32                 } catch
33                 e.pr
34             }
35
36             // Compl
37             external
38         })
39         .open();
40     }
41 }
```

A red arrow points to the lambda expression `run(() -> {` on line 21, which is highlighted in blue. A tooltip is visible over this line, showing the following options:

- Change project compliance and JRE to 1.8
- Assign statement to new local variable (Ctrl+2, L)
- Assign statement to new field (Ctrl+2, F)
- Convert to anonymous class creation

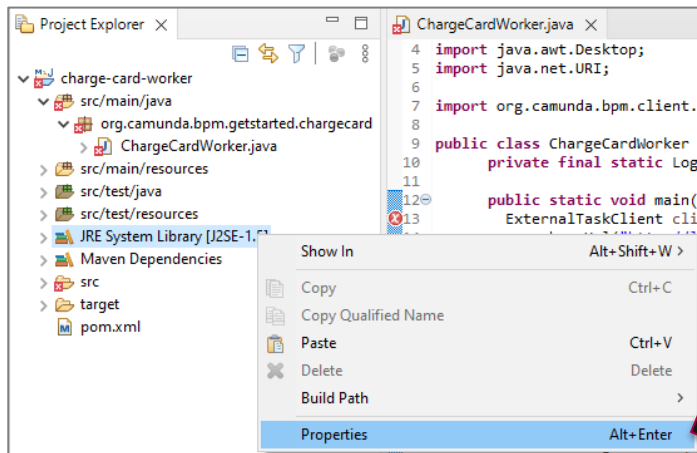
Below the tooltip, a yellow box contains the following text:

Set project compiler compliance settings to 1.8
Set project JRE build path entry to 'JavaSE-1.8'

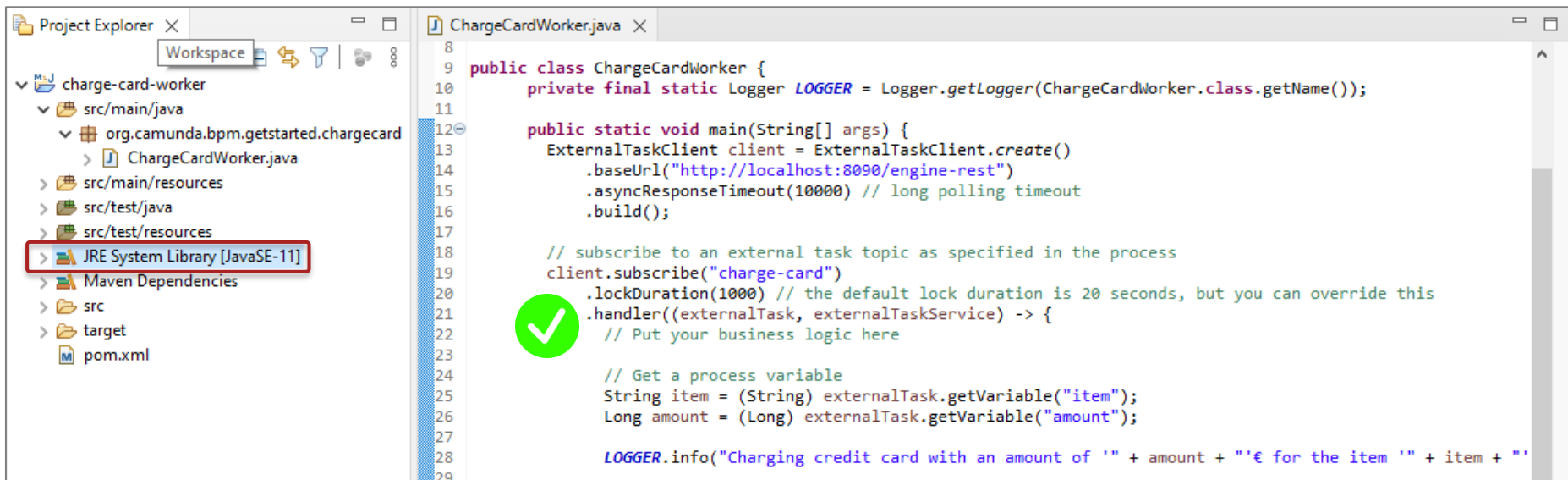
At the bottom of the yellow box, it says: "Press 'Tab' from proposal table or click for focus".

A red box with a black border contains the text: "Si hay error, realizar el cambio." (If there is an error, make the change.)

3. Añade una clase Java ExternalTaskClient (V)



Usar una versión reciente de Java (1,8 +)



4. Ejecuta la clase Java

ws-us-gps - charge-card-worker/src/main/java/org/camunda/bpm/getstarted/chargecard/ChargeCardWorker.java - Eclipse IDE

File Edit Source Refactor Navigate Search Project Run Window Help

Project Explorer

- charge-card-worker
 - src/main/java
 - org.camunda.bpm.getstarted.chargecard
 - ChargeCardWorker.java

ChargeCardWorker.java

```
1 package org.camunda.bpm.getstarted.chargecard;  
2  
3 import java.util.logging.Logger;  
4 import java.awt.Desktop;  
5 import java.net.URI;  
6  
7 import org.camunda.bpm.client.ExternalTaskClient;  
8  
9 public class ChargeCardWorker {  
10     private final static Logger LOGGER = Logger.getLogger(ChargeCardWorker.class);  
11  
12     public static void main(String[] args) {  
13         ExternalTaskClient client = ExternalTaskClient.create()  
14             .baseUrl("http://localhost:8090/engine-rest")  
15             .asyncResponseTimeout(10000) // long timeout  
16             .build();  
17  
18         // subscribe to external task topic  
19         client.subscribe("charge-card")  
20             .callback((String externalTaskId, Long amount) // the default  
21                 // business logic here  
22                 // process variable  
23                 // ...  
24                 // ...  
25                 // ...  
26                 // ...  
27                 // ...  
28                 // ...  
29                 // ...  
30                 // ...  
31                 // ...  
32                 // ...  
33                 // ...  
34                 // ...  
35                 // ...  
36                 // Complete the task  
37                 externalTaskService.complete(externalTaskId, amount);  
38     }  
}
```

El número de puerto debe coincidir con el configurado para Camunda

Ten en cuenta que el “Worker” debe permanecer en funcionamiento durante todo el recorrido de esta guía.

Run As

- 1 Java Application Alt+Shift+X, J

Run Configurations...

DESPLIEGUE

Pasos para desplegar un modelo

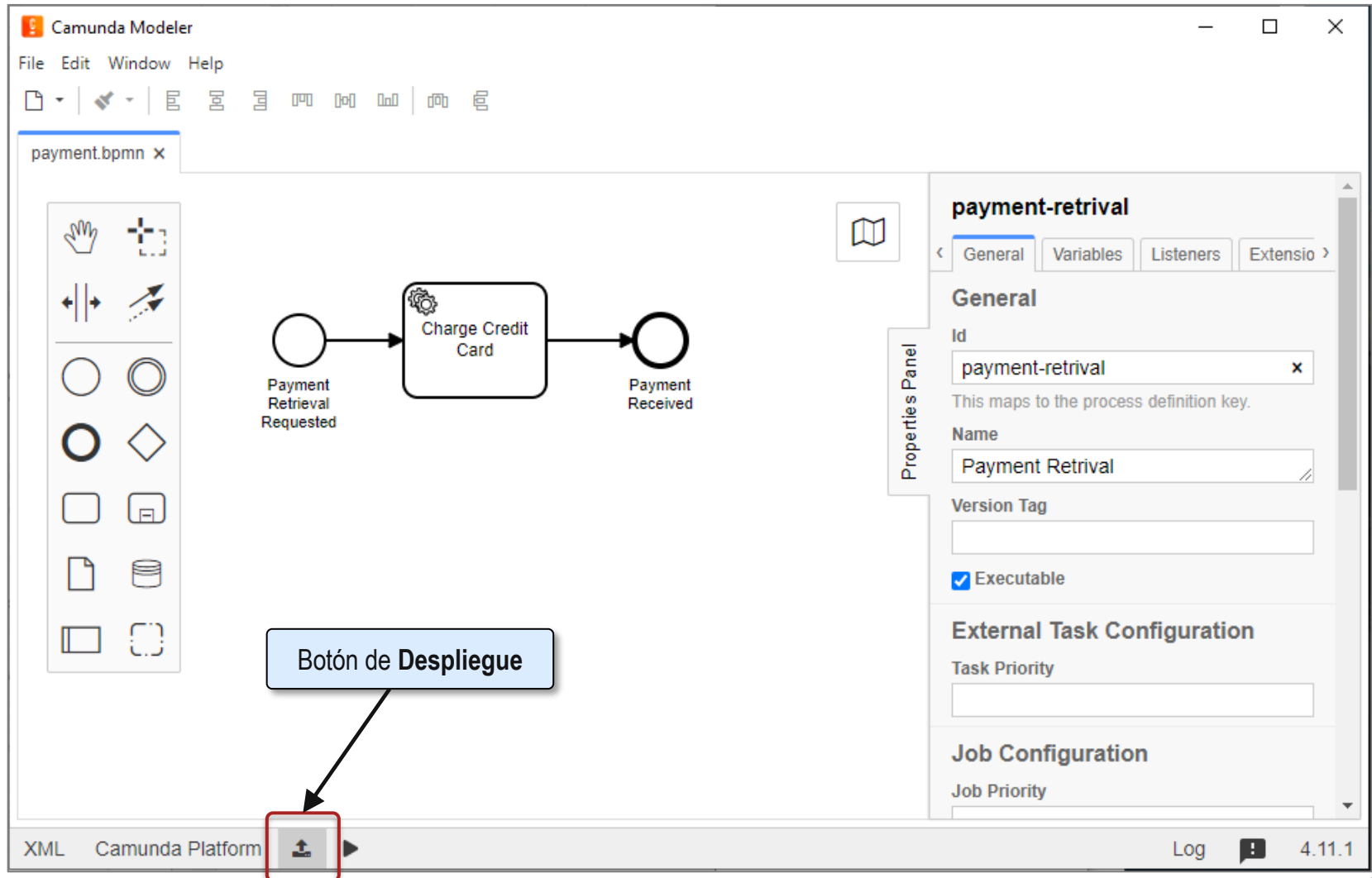
Los pasos que se describen a continuación están basados en el tutorial disponible en:

<https://docs.camunda.org/get-started/quick-start/deploy/>

1. Despliega el modelo
2. Verifica el despliegue en **Camunda Cockpit**
3. Inicia una nueva instancia de proceso

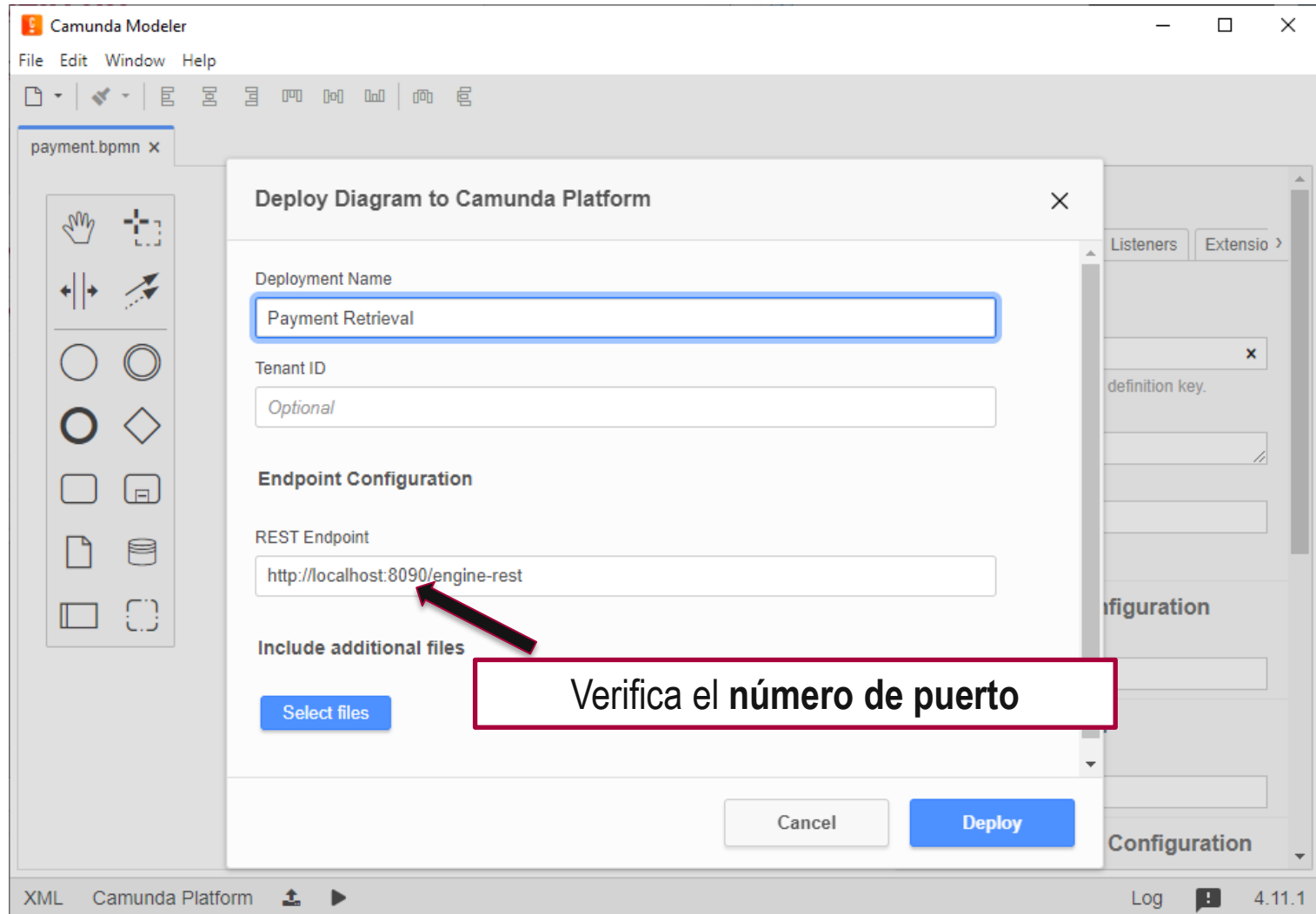
1. Despliega el modelo (I)

- Utilice el Modelador de Camunda para desplegar el proceso



1. Despliega el modelo (II)

Asigna “*Payment Retrieval*” como valor de **Deployment Name**



1. Despliega el modelo (III)

Si todo ha ido bien, aparecerá un mensaje de éxito.

The screenshot displays the Camunda Modeler interface. The main canvas shows a BPMN diagram for 'payment-retrival' (note the typo in the diagram). The diagram consists of a start event 'Payment Retrieval Requested', followed by a task 'Charge Credit Card' (indicated by a gear icon), and ending with an event 'Payment Received'. A red rectangular box highlights a message box that says 'Deployment succeeded' with a close button (X). On the right, the 'Properties Panel' is open for the 'payment-retrival' process. It shows the 'General' tab with fields for 'Id' (payment-retrival), 'Name' (Payment Retrival), and 'Version Tag'. The 'Executable' checkbox is checked. Below are sections for 'External Task Configuration', 'Job Configuration', and 'Candidate Starter Configuration'. The bottom status bar indicates 'XML Camunda Platform' and 'Log 4.11.1'.

1. Despliega el modelo (IV)

Si hay problemas, verifica los mensajes de la consola.

The screenshot shows the Camunda Modeler interface. In the center, a BPMN diagram for 'payment.bpmn' is displayed, featuring a 'Payment Retrieval Requested' start event, a 'Charge' task, and a 'Payment Received' end event. A red box highlights a 'Deployment failed' dialog box with the message 'See the log for further details.' To the right, the 'Properties Panel' for the 'payment-retrival' task is visible, showing fields for Id, Name, Version Tag, and a checked 'Executable' checkbox. At the bottom, the console displays an error message: 'ENGINE-09005 Could not parse BPMN process. Errors: * One of the attributes 'class', 'delegateExpression', 'type', or 'expression' is mandatory on serviceTask. | resource payment.bpmn | line 8 | column 71 [deploy-error]'. A blue callout box with the text 'Click para abrir la consola' points to a 'Log' button in the bottom right corner, which is also highlighted with a red box. The bottom status bar shows 'XML Camunda Platform' and the version '4.11.1'.

Camunda Modeler

File Edit Window Help

payment.bpmn x

Deployment failed
See the log for further details.

Payment Retrieval Requested

Charge

Payment Received

payment-retrival

General Variables Listeners Extensio >

General

Id
payment-retrival x

This maps to the process definition key.

Name
Payment Retrieval

Version Tag

☒ Executable

External Task Configuration

Task Priority

ENGINE-09005 Could not parse BPMN process. Errors:
* One of the attributes 'class', 'delegateExpression', 'type', or 'expression' is mandatory on serviceTask.
| resource payment.bpmn | line 8 | column 71 [deploy-error]

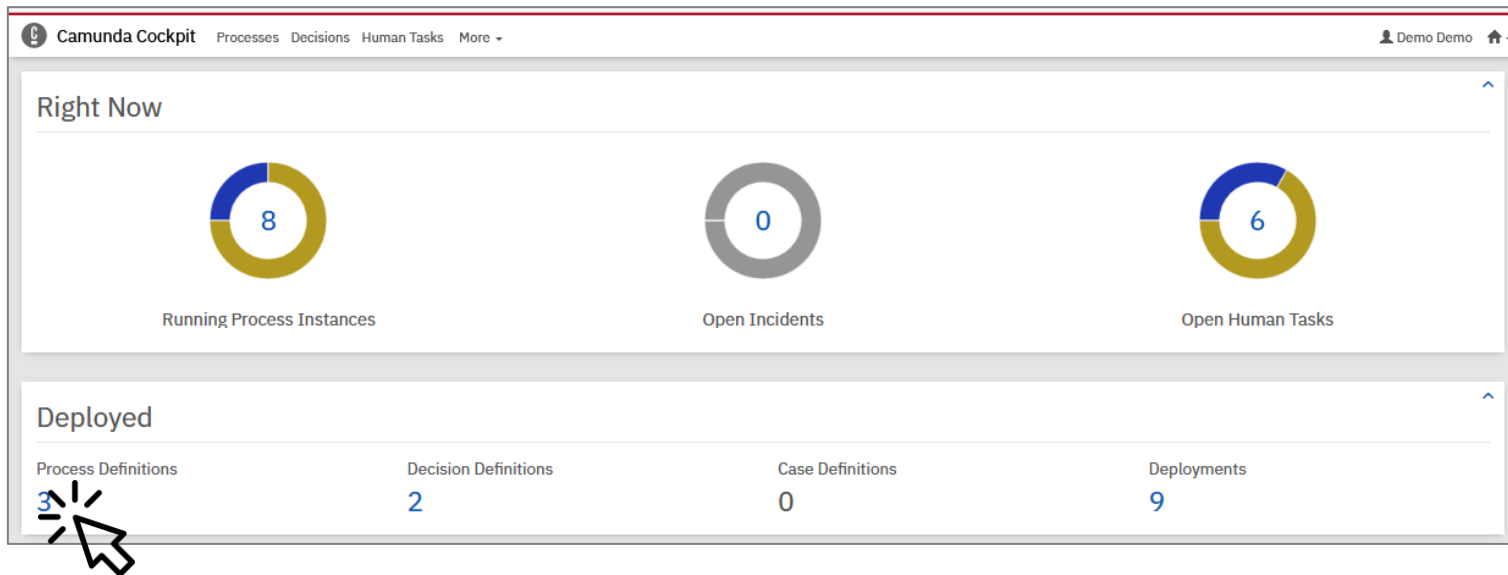
Click para abrir la consola

Log

XML Camunda Platform 4.11.1

2. Verifica el despliegue en Camunda Cockpit

Ir a <http://localhost:8080/camunda/app/cockpit/default/> y loguearse con **demo / demo**



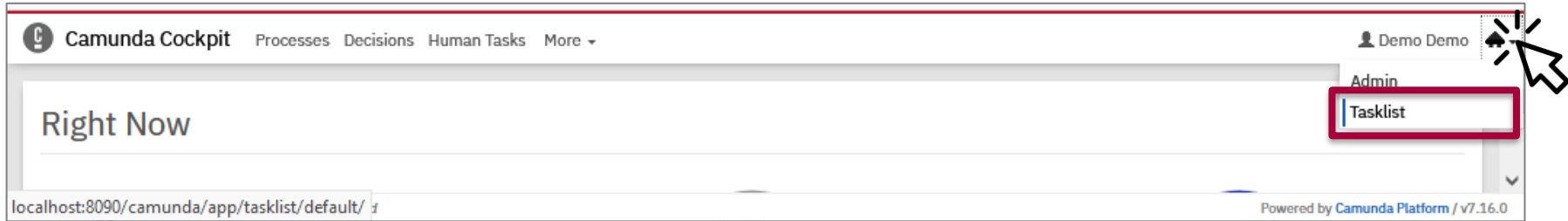
3 process definitions deployed

State	Incidents	Running Instances	Name	Tenant ID
✓	0	6	Invoice Receipt	
✓	0	0	Payment Retrieval	
✓	0	2	Review Invoice	

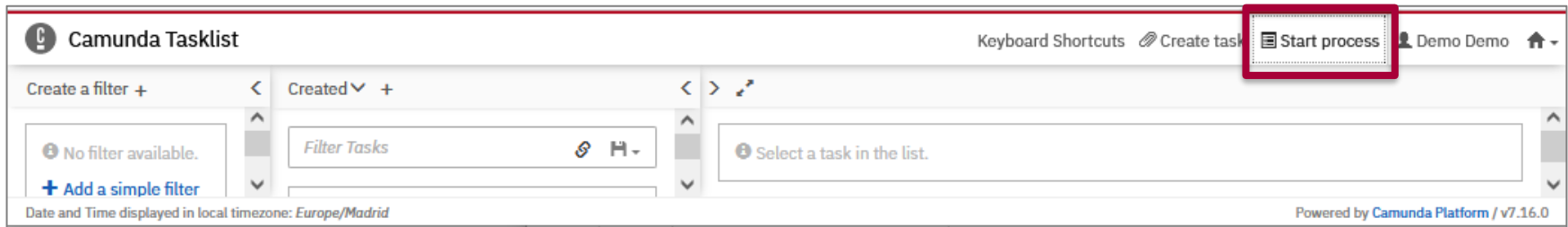
The 'Payment Retrieval' row is highlighted with a red border.

3. Inicia un nuevo proceso (I)

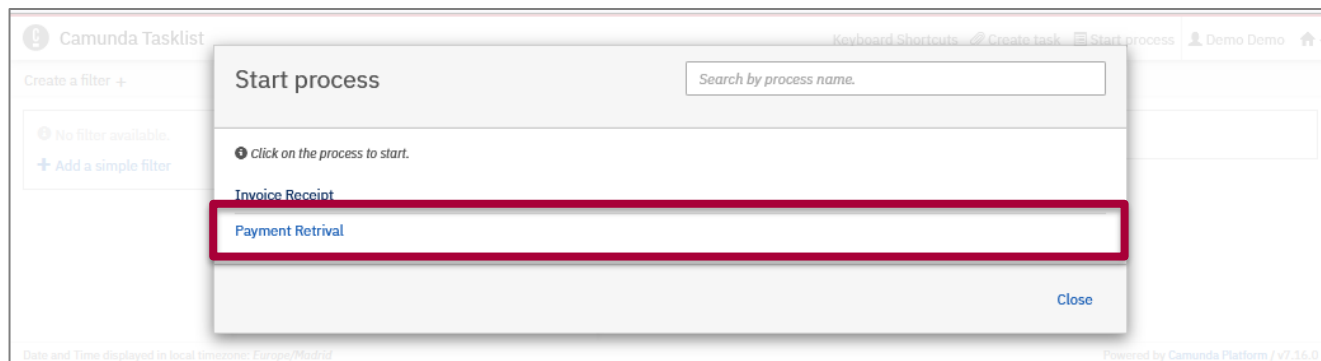
Accede a **Camunda Tasklist** con el usuario **demo/demo**



Selecciona “Start Process”

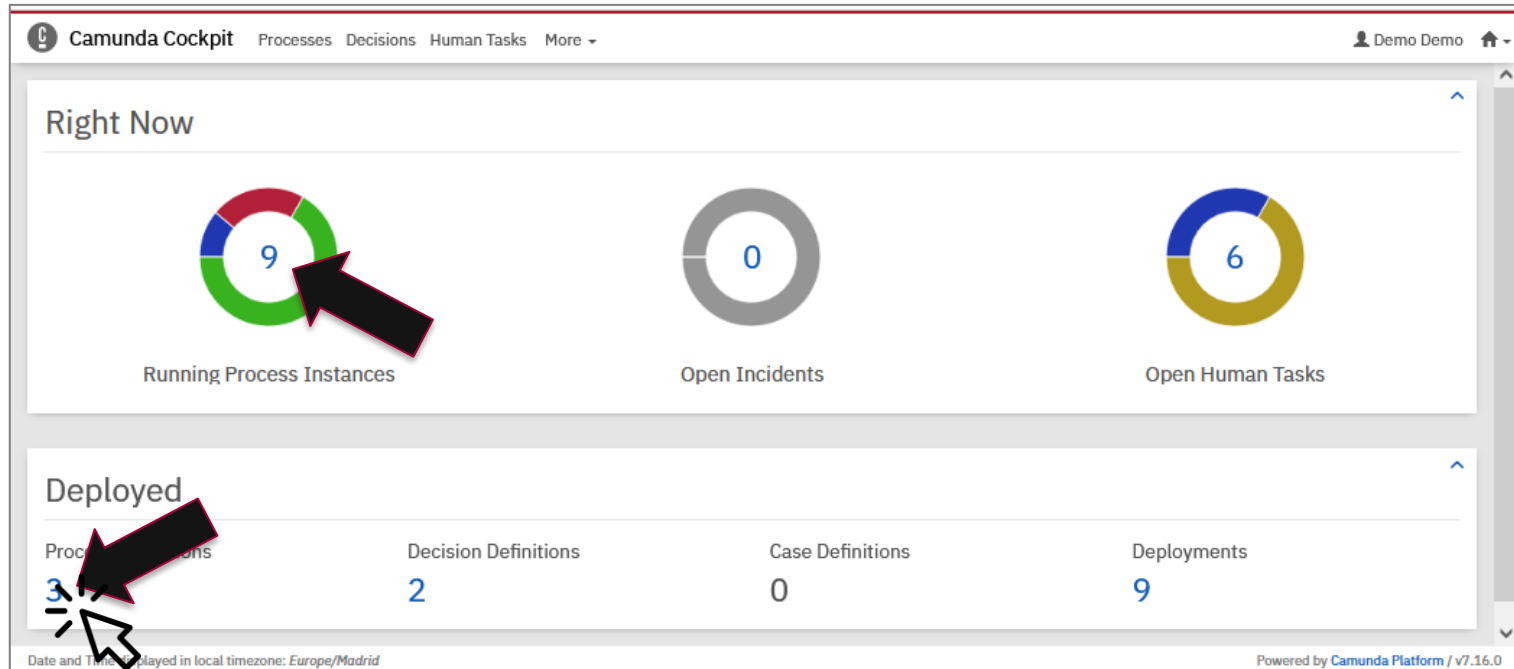


Seleccionar el proceso a desplegar, en este caso “Payment Retrieval”



3. Inicia un nuevo proceso (II)

Vuelve a “*Camunda Cockpit*” > *Processes*



Dashboard » Processes

3 process definitions deployed

List Previews

State	Incidents	Running Instances	Name	Tenant ID
✓	0	6	Invoice Receipt	
✓	0	1	Payment Retrieval	
✓	0	2	Review Invoice	

3. Inicia un nuevo proceso (III)

Camunda Cockpit Processes Decisions Human Tasks More ▾

Demo Demo Home ▾

Dashboard » Processes » Payment Retrieval : Runtime

Definition Version:
1

Version Tag:
null

Definition ID:
payment-retrieval:6:67a17473-5425-...

Definition Key:
payment-retrieval

Definition Name:
Payment Retrieval

History Time To Live:
null

Tenant ID:
null

Deployment ID:
679c1d41-5425-11ec-b06b-049226...

Instances Running:

- current version: 1
- all versions: 1

```
graph LR; A((Payment Retrieval Requested)) --> B[Charge Credit Card]; B --> C((Payment Received));
```

Activity Instance Statistics: on ●

Process Instances Incidents Called Process Definitions Job Definitions

Add criteria

1 🔗 📄 ▾

State	ID	Start Time ▾	Business Key
✔	d19460b0-5428-11ec-b06b-049226be0810	2021-12-03T12:04:43	

Date and Time displayed in local timezone: Europe/Madrid

Powered by Camunda Platform / v7.16.0

3. Inicia un nuevo proceso (IV)

¿Qué ha pasado en Eclipse?

The screenshot shows the Eclipse IDE with a Java project named 'charge-card-worker'. The main class is 'ChargeCardWorker.java'. The code defines a static logger and a main method that subscribes to an external task topic and logs the amount of a credit card. The console output shows a log message: 'INFO: Charging credit card with an amount of 'null'€ for the item 'null'...'.

```
1 package org.camunda.bpm.getstarted.chargecard;
2
3 import java.util.logging.Logger;
4
5 public class ChargeCardWorker {
6     private final static Logger LOGGER = Logger.getLogger(ChargeCardWorker.class.getName());
7
8     public static void main(String[] args) {
9         ExternalTaskClient client = ExternalTaskClient.create()
10             .baseUrl("http://localhost:8090/engine-rest")
11             .asyncResponseTimeout(10000) // long polling timeout
12             .build();
13
14         // subscribe to an external task topic as specified in the process
15         client.subscribe("charge-card")
16             .lockDuration(1000) // the default lock duration is 20 seconds, but you can override this
17             .handler((externalTask, externalTaskService) -> {
18                 // Put your business logic here
19
20                 // Get a process variable
21                 String item = (String) externalTask.getVariable("item");
22                 Long amount = (Long) externalTask.getVariable("amount");
23
24                 LOGGER.info("Charging credit card with an amount of '" + amount + "€ for the item '" + item + "'...");
25
26                 try {
27                     Desktop.getDesktop().browse(new URI("https://docs.camunda.org/get-started/quick-start/complete"));
28                 } catch (Exception e) {
29                     e.printStackTrace();
30                 }
31             });
32     }
33 }
```

No hemos asignado las variables!!!

ChargeCardWorker [Java Application] C:\Users\bestrada\Desktop\eclipse GPS2021-22\eclipse\plugins\org.eclipse.jdt.launcher\org.eclipse.jdt.launcher.exe

```
438 [main] INFO org.camunda.bpm.client - TASK/CLIENT-01026 Discovered data format provider: org.camunda.bpm.client.variable.impl.format.serializable.SerializableFormatProvider
439 [main] INFO org.camunda.bpm.client - TASK/CLIENT-01025 Discovered data format: org.camunda.bpm.client.variable.impl.format.xml.DomXmlDataFormat
439 [main] INFO org.camunda.bpm.client - TASK/CLIENT-01026 Discovered data format: org.camunda.bpm.client.variable.impl.format.xml.DomXmlDataFormat
440 [main] INFO org.camunda.bpm.client - TASK/CLIENT-01026 Discovered data format: org.camunda.bpm.client.variable.impl.format.xml.DomXmlDataFormat
460 [main] INFO org.camunda.bpm.client - TASK/CLIENT-01025 Discovered data format: org.camunda.bpm.client.variable.impl.format.xml.DomXmlDataFormat
INFO: Charging credit card with an amount of 'null'€ for the item 'null'...
```

4. Inicia un nuevo proceso (I)

- Ir a **Camunda Tasklist** con el usuario **demo/demo**
- Seleccionar “**Start Process**”
- Seleccionar el proceso “**Payment Retrieval**”
- Definir las variables que necesitamos
Name: *item* **Name:** *amount*
Type: *String* **Type:** *Long*
Value: *my_item* **Value:** *1200*

Camunda Tasklist

Keyboard Shortcuts Create task Start process Demo Demo

Create a filter +

No filter available.
+ Add a simple filter

Start process

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

Business Key

	Name	Type	Value
Add a variable... +			
Remove x	<input type="text" value="item"/>	<input type="text" value="String"/>	<input type="text" value="my_item"/>
Remove x	<input type="text" value="amount"/>	<input type="text" value="Long"/>	<input type="text" value="1250"/>

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

4. Inicia un nuevo proceso (II)

The screenshot displays the Camunda Cockpit interface. The top navigation bar includes 'Processes', 'Decisions', 'Human Tasks', and 'More'. The main content area shows the 'Payment Retrieval' process instance with ID '311dfa46-5439-11ec-80ee-049226be0810' in a 'Runtime' state. The left sidebar lists various instance details: Instance ID, Business Key (null), Definition Version (1), Definition ID, Definition Key (payment-retrieval), Definition Name (Payment Retrieval), Tenant ID (null), Deployment ID, and Super Process Instance ID (null). The central BPMN diagram illustrates a flow from 'Payment Retrieval Requested' to a 'Charge Credit Card' task (marked with a blue '1') and finally to 'Payment Received'. Below the diagram, a table titled 'Variables' lists the current state of process variables.

Name	Type	Value	Scope	Actions
amount	Long	1250	Payment Retrieval	[Edit] [Delete]
item	String	my_item	Payment Retrieval	[Edit] [Delete]

The screenshot shows the Eclipse IDE console with the following output:

```
ChargeCardWorker [Java Application] C:\Users\bestrada\Desktop\eclipse GPS2021-22\eclipse\plugins\org.eclipse.justj.openjdk.hot  
dic 03, 2021 1:53:13 P. M. org.camunda.bpm.getstarted.chargecard.ChargeCardWorker lambda$0  
INFO: Charging credit card with an amount of '1250'€ for the item 'my_item'...
```



TAREAS DE USUARIO

Pasos para modificar el proceso agregando una tarea de usuario

Los pasos que se describen a continuación están basados en el tutorial disponible en:

<https://docs.camunda.org/get-started/quick-start/user-task/>

1. Agrega una tarea de usuario
2. Configura la tarea de usuario
3. Configura un formulario básico en la tarea de usuario
4. Despliega el proceso (Diferencias con formulario externo)
5. Inicia una nueva instancia del proceso
6. Verifica que se está ejecutando el proceso (y la tarea para demo)
7. Continúa con la ejecución del proceso

1. Agrega una tarea de usuario (I)

Crea una actividad y nómbrala como “*Approve Payment*”

The screenshot displays the Camunda BPMN editor interface. On the left, a toolbar contains various BPMN symbols. A blue callout box points to the 'Activity Shape' icon, with the text: "Select **activity shape** and drag into position between **Start Event** and 'Charge Credit Card' Service Task".

The main workspace shows a BPMN diagram with the following elements:

- Start Event:** A circle labeled "Payment Retrieval Requested".
- Activity Shape:** An empty rounded rectangle positioned between the start event and the service task.
- Service Task:** A rounded rectangle with a gear icon, labeled "Charge Credit Card".
- End Event:** A circle labeled "Payment Received".

Arrows indicate the flow from the start event to the activity shape, and from the activity shape to the service task, and finally to the end event.

On the right, the **Properties Panel** is open for the selected activity shape. It shows the following configuration:

- payment-retrival** (Title)
- General** (Tab selected)
- Id:** payment-retrival
- Name:** Payment Retrieval
- Version Tag:** (Empty field)
- Executable:** ☒
- External Task Configuration:**
 - Task Priority:** (Empty field)
- Job Configuration:**
 - Job Priority:** (Empty field)
- Candidate Starter Configuration:**
 - Candidate Starter Groups:** (Empty field)
 - Specify more than one group as a comma separated list.

1. Agrega una tarea de usuario (II)

Cambia el tipo de actividad a tarea de usuario “*User Task*”

The screenshot shows a BPMN editor interface with a workflow diagram and a properties panel.

Workflow Diagram: The diagram shows a process flow starting with an event labeled "Payment Retrieval Requested", followed by an empty rectangular task box (highlighted with a blue dashed border), then a task labeled "Charge Credit Card", and finally an event labeled "Payment Received".

Task Type Selection: A context menu is open over the empty task box, listing various task types. The "User Task" option, represented by a person icon, is currently selected and highlighted.

Properties Panel: The "Properties Panel" is open on the right, showing the configuration for the selected "userTask".

- General:** The "Id" field is set to "userTask". The "Name" field is empty.
- Asynchronous Continuations:** Both "Asynchronous Before" and "Asynchronous After" checkboxes are unchecked.
- Documentation:** The "Element Documentation" field is empty.

2. Configura la tarea de usuario

The screenshot displays the Camunda BPMN Designer interface for editing a process named 'payment.bpmn'. The main workspace shows a BPMN diagram with the following elements:

- Start Event: Payment Retrieval Requested
- Task: Approve Payment (highlighted with a blue dashed border)
- Exclusive Gateway
- Task: Charge Credit Card
- End Event: Payment Received

The Properties Panel on the right is open for the 'Approve Payment' task, showing the 'General' tab. The 'Assignee' field is highlighted with a red box and contains the value 'demo'. A blue callout box with the text 'Set demo as Assignee' has an arrow pointing to this field.

Activity_0rvyfnd

General

Id: Activity_0rvyfnd

Name: Approve Payment

Details

Assignee: demo

Candidate Users

Candidate Groups

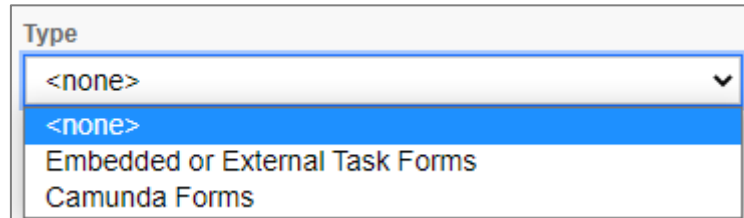
Due Date

The due date as an EL expression (e.g. \${someDate}) or an ISO date (e.g. 2015-06-26T09:54:00)

XML Camunda Platform Log 4.11.1

3. Configura un formulario básico en la tarea de usuario

Hay varias formas de implementar un formulario en Camunda:



- None
- Embedded or External Task Forms
- Camunda Forms

Más información sobre formularios:

<https://docs.camunda.org/get-started/quick-start/user-task/>

3. Configura un formulario básico en la tarea de usuario (I)

The screenshot displays the Camunda BPMN Editor interface for a process named "payment.bpmn". The main workspace shows a BPMN diagram with the following elements:

- Start Event:** A circle labeled "Payment Retrieval Requested".
- Task:** A rounded rectangle labeled "Approve Payment" with a user icon, highlighted with a dashed blue border.
- Intermediate Event:** A circle labeled "Payment Received".
- Callout Boxes:**
 - "Selecciona Forms tab" points to the "Forms" tab in the Properties Panel.
 - "Selecciona el tipo de Form" points to the "Type" dropdown in the Properties Panel.
 - "Agrega los Form Fields" points to the "Form Fields" list in the Properties Panel.

The **Properties Panel** on the right is expanded for the "Approve Payment" task, showing the following configuration:

- Activity_Orvyfnd:** The "Forms" tab is selected.
- Forms:**
 - Type:** A dropdown menu currently showing "<none>".
 - Form Fields:** An empty list area for defining form fields.

The bottom status bar indicates the editor is running on "Camunda Platform" version "4.11.1".

3. Configura un formulario básico en la tarea de usuario (II)

The screenshot shows the Camunda Modeler interface. The BPMN diagram on the left includes a start event 'Payment Retrieval Requested', a task 'Approve Payment' (highlighted with a dashed blue border), an intermediate event, a task 'Charge Credit Card', and an end event 'Payment Received'. The 'Properties Panel' on the right is open for the 'Approve Payment' task, showing the 'Forms' tab. In the 'Form Fields' list, the 'item' field is selected. A red box highlights the 'item' field in the list, and a red box highlights the 'Form Field' configuration section below it. A red box also highlights the 'Campo 1:' list in the center, with arrows pointing from its items to the corresponding fields in the 'Form Field' section.

Campo 1:

- ID: item
- Type: string
- Label: Item

Form Field Configuration:

- ID (process variable name): item
- Type: string
- Label: Item:
- Default Value:
- Validation:

3. Configura un formulario básico en la tarea de usuario (III)

The screenshot shows the Camunda Modeler interface with a BPMN diagram titled 'payment.bpmn'. The diagram includes a start event 'Payment Retrieval Requested', a task 'Approve Payment' (highlighted with a dashed blue border), an intermediate event, a task 'Charge Credit Card', and an end event 'Payment Received'. The 'Approve Payment' task is selected, and the 'Properties Panel' on the right is open to the 'Forms' tab. In the 'Form Fields' list, 'item' and 'amount' are shown, with 'amount' selected. A red box highlights the 'amount' field in the list, and a red box highlights the 'Form Field' configuration section below. A red box also highlights the 'Form Fields' list header. Arrows point from the 'Form Field' configuration to the 'amount' field in the list.

Activity_Orvyfnd

General Forms Listeners Input/Output Extension

Forms

Type
<none>

Form Fields

- item
- amount

Form Field

ID (process variable name)
amount

Type
long

Label
Amount:

Default Value

Validation

Campo 2:

- ID: amount
- Type: long
- Label: Amount

3. Configura un formulario básico en la tarea de usuario (IV)

The screenshot shows the Camunda Modeler interface. On the left, a toolbar contains various BPMN symbols. The main canvas displays a BPMN diagram for 'payment.bpmn'. The process starts with a start event 'Payment Retrieval Requested', followed by a task 'Approve Payment' (highlighted with a blue dashed border), then an intermediate event, a task 'Charge Credit Card', and finally an end event 'Payment Received'. On the right, the 'Properties Panel' is open for the 'Approve Payment' task. The 'Forms' tab is selected, showing a list of form fields: 'item', 'amount', and 'approved'. The 'approved' field is selected and highlighted in blue. Below this, the 'Form Field' configuration is visible, showing the ID 'approved', Type 'boolean', and Label 'Approved?'. A red box highlights the 'Campo 3:' configuration details, with arrows pointing to the corresponding fields in the Properties Panel.

Campo 3:

- ID: approved
- Type: boolean
- Label: Approved?

Activity_0rvyfnf

General Forms Listeners Input/Output Extension

Forms

Type
<none>

Form Fields

item
amount
approved

Form Field

ID (process variable name)
approved

Type
boolean

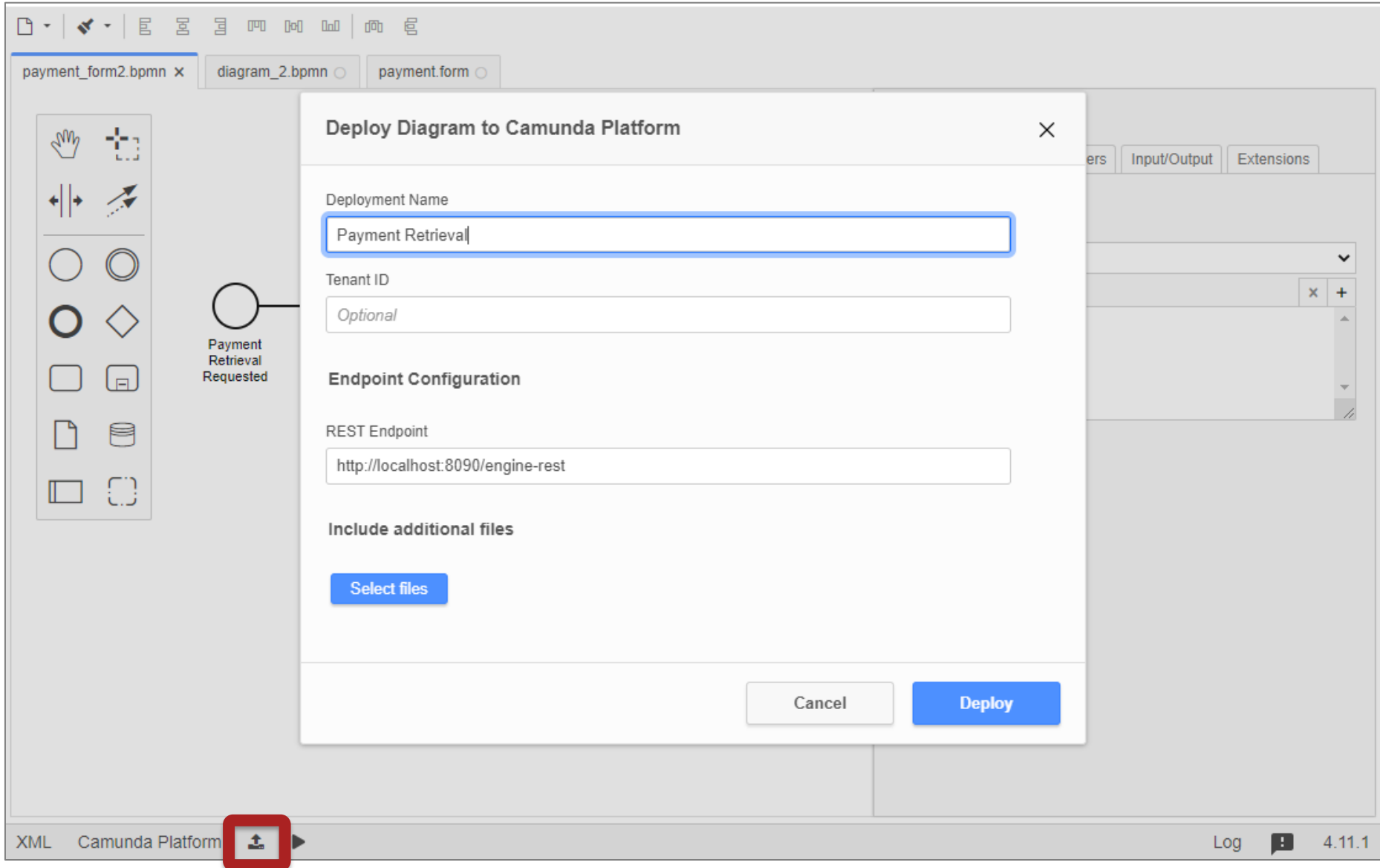
Label
Approved?

Default Value

Validation

XML Camunda Platform Log 4.11.1

4. Despliega el proceso



Diferencias para utilizar un formulario externo (I)

Selecciona el tipo de form “*Embedded or External Task Forms*”

Define el identificador del formulario: “*camunda-forms:deployment:payment.form*”

The screenshot displays the Camunda Modeler interface. On the left, the 'File' menu is open, and the 'Form (Camunda Platform or Cloud)' option is selected. The main workspace shows a BPMN diagram with two tasks: 'Approve Payment' and 'Charge Credit Card', connected by a flow. The 'Approve Payment' task is highlighted with a dashed blue border. On the right, the 'Properties Panel' is open, showing the 'Forms' tab for the selected task. The 'Type' is set to 'Embedded or External Task Forms' and the 'Form Key' is set to 'camunda-forms:deployment:payment.form'. The 'Form Fields' section is empty.

Más información sobre formularios:
<https://docs.camunda.org/get-started/quick-start/user-task/>

Diferencias para utilizar un formulario externo (II)

Crea un formulario en Camunda Modeler y guárdalo como “*payment.form*”

payment_form2.bpmn x payment.form o

FORM ELEMENTS LIBRARY

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

Amount

Item

☐ Approved?

ABC FORM payment.form

General

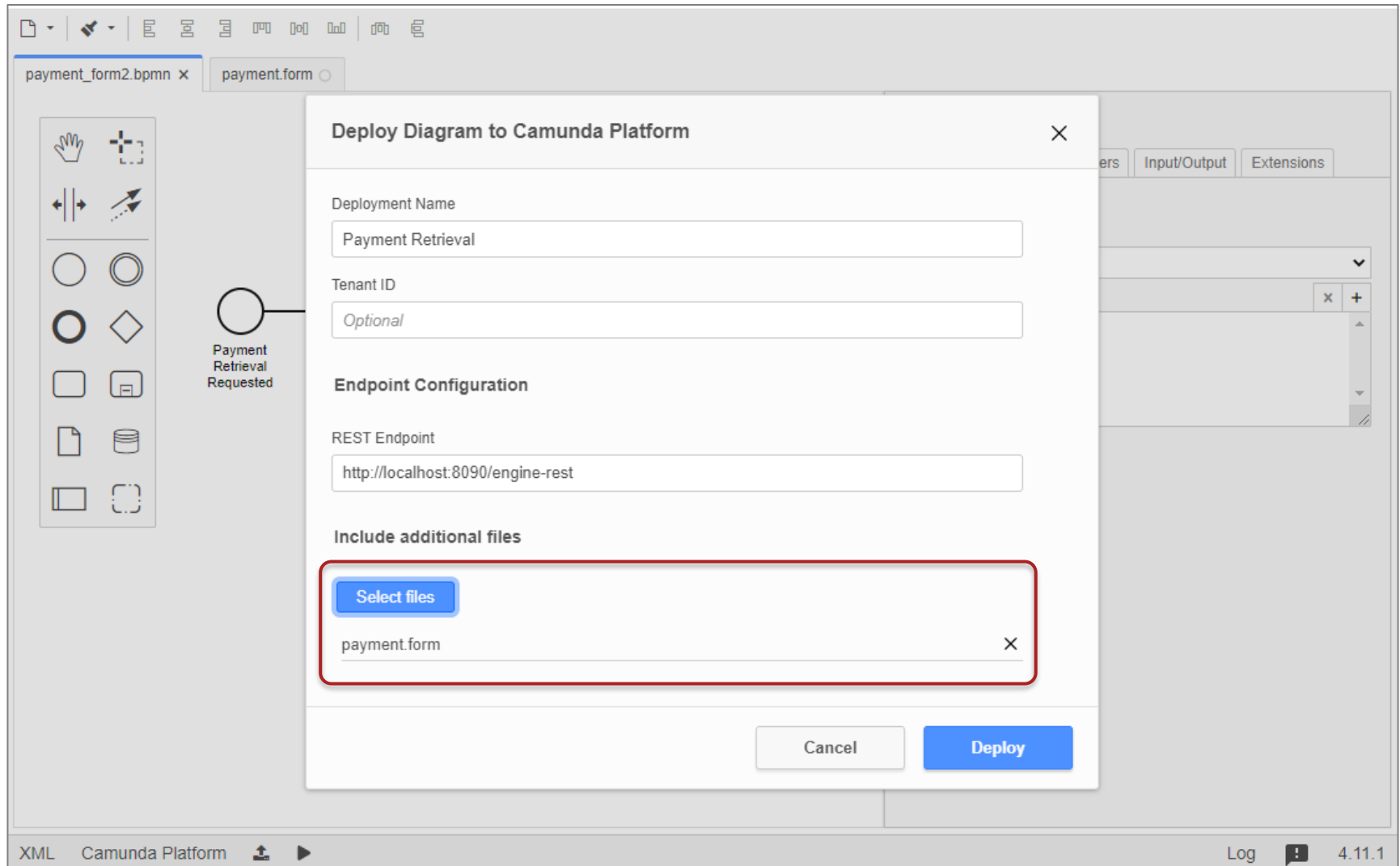
Id

payment.form

JSON Camunda Platform 7.16 0 errors Log 4.11.1

Diferencias para utilizar un formulario externo (III)

Despliega el proceso, agregando el archivo “*payment.form*”



5. Inicia una nueva instancia del proceso

- Ir a **Camunda Cockpit** y verificar que tenemos el nuevo proceso.
- Ir a **Camunda Tasklist** e iniciar una nueva instancia del proceso “**Payment Retrieval**”.

Camunda Tasklist

Create a filter +

All Tasks (6)

Keyboard Shortcuts Create task Start process Demo Demo

Start process

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

Business Key

Add a variable +

	Name	Type	Value
Remo... x	<input type="text" value="item"/>	<input type="text" value="String"/>	<input type="text" value="GPS_Item_2021"/>
Remo... x	<input type="text" value="amount"/>	<input type="text" value="Long"/>	<input type="text" value="3125"/>

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

Define los valores de las variables **item** y **amount**

5. Inicia una nueva instancia del proceso

El proceso inicia y el usuario *demo* tiene una tarea asignada

The screenshot displays the Camunda Tasklist interface. At the top, the header includes the Camunda logo, the title 'Camunda Tasklist', and navigation links for 'Keyboard Shortcuts', 'Create task', 'Start process', and the user 'Demo Demo'. Below the header, there is a filter section with 'Create a filter +' and a dropdown menu set to 'Created'. The main list shows 'All Tasks (6)' with a 'Filter Tasks' input field. The tasks are grouped into four identical entries, each containing a main task name, a sub-task 'Invoice Receipt', and a due date 'Due in a day, Created 6 days ago'. The tasks are: 'Assign Reviewer', 'Review Invoice', 'Prepare Bank Transfer', and 'Approve Invoice'. Each task is assigned to 'Demo Demo' and has a value of '50'. On the right side, there is a box with the text 'Select a task in the list.' At the bottom right, there are two notifications: 'The process has been started.' and 'You are assigned to the following tasks in the same process : Payment Retrieval'. The footer shows 'Date and Time displayed in local timezone: Europe/Madrid' and 'Powered by Camunda Platform / v7.16.0'.

Task Name	Sub-task	Due Date	Assigned To	Value
Assign Reviewer	Invoice Receipt	Due in a day, Created 6 days ago	Demo Demo	50
Review Invoice	Invoice Receipt	Due in a day, Created 6 days ago	Demo Demo	50
Prepare Bank Transfer	Invoice Receipt	Due in a day, Created 6 days ago	Demo Demo	50
Approve Invoice	Invoice Receipt	Due in a day, Created 6 days ago	Demo Demo	50

Notifications:

- The process has been started.
- You are assigned to the following tasks in the same process : Payment Retrieval

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

6. Verifica que se está ejecutando el proceso

Verifica la ejecución del proceso en Camunda Cockpit

Camunda Cockpit Processes Decisions Human Tasks More ▾

Dashboard » Processes » Payment Retrieval : Runtime

Definition Version: 2 ▾

Version Tag: null

Definition ID: payment-retrieval:2:a80d6111-54...

Definition Key: payment-retrieval

Definition Name: Payment Retrieval

History Time To Live: null

Tenant ID: null

Deployment ID: a808f43f-5462-11ec-887f-0492...

Instances Running:

- current version: 1
- all versions: 1

Activity Instance Statistics: on ●

Payment Retrieval Requested → Approve Payment (1) → Charge Credit Card → Payment Received

Process Instances Incidents Called Process Definitions Job Definitions

Add criteria 1 🔍 📄

State	ID	Start Time ▾	Business Key
✔	73762f13-58db-11ec-887f-049226be0810	2021-12-09T11:33:29	

Date and Time displayed in local timezone: Europe/Madrid

Powered by Camunda Platform / v7.16.0

Clic para ver los detalles de la instancia

6. Verifica que se está ejecutando el proceso (II)

Verifica la las variables asignadas

Camunda Cockpit Processes Decisions Human Tasks More ▾

Dashboard » Processes » Payment Retrieval : 73762f13-58db-11ec-887f-049226be0810 : Runtime

Information Filter

Instance ID:
73762f13-58db-11ec-887f-0492...

Business Key:
null

Definition Version:
2

Definition ID:
[payment-retrival:2:a80d6111-54...](#)

Definition Key:
payment-retrival

Definition Name:
Payment Retrieval

Tenant ID:
null

Deployment ID:
[a808f43f-5462-11ec-887f-0492...](#)

Super Process Instance ID:
null

Payment Retrieval Requested → Approve Payment → Charge Credit Card → Payment Received

Variables Incidents Called Process Instances User Tasks Jobs External Tasks

Add criteria 2 ⚙ H ▾

Name ^	Type —	Value	Scope	Actions
amount	Long	3125	Payment Retrieval	
item	String	GPS_Item_2021	Payment Retrieval	

Date and Time displayed in local timezone: Europe/Madrid

Powered by Camunda Platform / v7.16.0

6. Verifica que se está ejecutando el proceso (III)

Accede a Camunda Tasklist y verifica que la tarea ha sido asignada al usuario **demo**

The screenshot displays the Camunda Tasklist interface. On the left, a sidebar shows a filter for 'All Tasks (12)'. The main task list on the left contains three tasks: 'Approve Payment' (Payment Retrieval, Created 3 minutes ago, assigned to Demo Demo, priority 50), 'Assign Reviewer' (Review Invoice, Created 6 days ago, assigned to Demo Demo, priority 50), and 'Prepare Bank Transfer' (Invoice Receipt, Due in a day, Created 6 days ago, assigned to Demo Demo, priority 50). The right panel shows the detailed view for the 'Approve Payment' task. It includes a title 'Approve Payment', a subtitle 'Payment Retrieval', and action buttons: 'Set follow-up date', 'Set due date', 'Add groups', and 'Demo Demo'. Below these are tabs for 'Form', 'History', 'Diagram', and 'Description'. The 'Form' tab is active, showing fields for 'Item:' (GPS_Item_2021), 'Amount:' (3125), and 'Approved?' (checkbox). At the bottom right of the form are 'Save' and 'Complete' buttons. The footer indicates the date and time are displayed in the local timezone of Europe/Madrid and that the interface is powered by Camunda Platform v7.16.0.

Camunda Tasklist

Keyboard Shortcuts Create task Start process Demo Demo

Create a filter +

All Tasks (12)

Filter Tasks 12

Approve Payment

Payment Retrieval Demo Demo

Created 3 minutes ago 50

Assign Reviewer

Review Invoice Demo Demo

Created 6 days ago 50

Prepare Bank Transfer

Invoice Receipt Demo Demo

Due in a day, Created 6 days ago 50

Approve Invoice

Invoice Receipt Demo Demo

Due in a day, Created 6 days ago 50

Approve Payment

Payment Retrieval

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

Form History Diagram Description

Item:

GPS_Item_2021

Amount:

3125

Approved?

Save Complete

Date and Time displayed in local timezone: Europe/Madrid

Powered by Camunda Platform / v7.16.0

7. Continua con la ejecución del proceso (I)

Aprueba o no, el formulario asignado al usuario **demo**

Click en “**Complete**”

The screenshot displays the Camunda Tasklist web application. The interface is divided into three main sections: a top navigation bar, a left sidebar for task filtering, and a main content area for task details.

Top Navigation Bar: Includes the Camunda logo, the title "Camunda Tasklist", and several utility links: "Keyboard Shortcuts", "Create task", "Start process", and a user profile section for "Demo Demo".

Left Sidebar: Contains a "Create a filter +" button and a list of tasks. The tasks are filtered by "Created" date. The visible tasks are:

- Approve Payment** (Payment Retrival, Created 3 minutes ago, Demo Demo, 50)
- Assign Reviewer** (Review Invoice, Created 6 days ago, Demo Demo, 50)
- Prepare Bank Transfer** (Invoice Receipt, Due in a day, Created 6 days ago, 50)
- Approve Invoice** (Invoice Receipt, Due in a day, Created 6 days ago, 50)

Main Content Area: Displays the details for the selected "Approve Payment" task. The title is "Approve Payment" with a sub-header "Payment Retrival". Below the title are several action links: "Set follow-up date", "Set due date", "Add groups", and a user selection dropdown for "Demo Demo". There are four tabs: "Form" (selected), "History", "Diagram", and "Description".

Form Fields:

- Item:** A text input field containing "GPS_Item_2021".
- Amount:** A text input field containing "3125".
- Approved?:** A checkbox that is checked.

Bottom Right: Two buttons, "Save" and "Complete", are visible. The "Complete" button is highlighted in red.

Footer: The bottom of the page shows "Date and Time displayed in local timezone: Europe/Madrid" on the left and "Powered by Camunda Platform / v7.16.0" on the right.

7. Continua con la ejecución del proceso (II)

Se procesa la tarea externa

Camunda Cockpit Processes Decisions Human Tasks More ▾

Demo Demo Home ▾

Dashboard » Processes » Payment Retrieval : Runtime

Definition Version: 2 ▾

Version Tag: null

Definition ID: payment-retrieval:2:a80d6111-54...

Definition Key: payment-retrieval

Definition Name: Payment Retrieval

History Time To Live: null

Tenant ID: null

Deployment ID: a808f43f-5462-11ec-887f-0492...

Instances Running: 1
• current version: 1
• all versions: 1

Activity Instance Statistics: on ●

Payment Retrieval Requested → Approve Payment → Charge Credit Card (1) → Payment Received

Process Instances Incidents Called Process Definitions Job Definitions

Add criteria 1 🔍 H ▾

State	ID	Start Time ▾	Business Key
✓	73762f13-58db-11ec-887f-049226be0810	2021-12-09T11:33:29	

Date and Time displayed in local timezone: Europe/Madrid

Powered by Camunda Platform / v7.16.0

7. Continua con la ejecución del proceso (III)

Verifica el valor de las variables, antes de que se ejecute la tarea externa

Camunda Cockpit Processes Decisions Human Tasks More ▾

Dashboard » Processes » Payment Retrival : 73762f13-58db-11ec-887f-049226be0810 : Runtime

Information Filter

Instance ID:
73762f13-58db-11ec-887f-049226b...

Business Key:
null

Definition Version:
2

Definition ID:
payment-retrival:2:a80d6111-5462-...

Definition Key:
payment-retrival

Definition Name:
Payment Retrival

Tenant ID:
null

Deployment ID:
a808f43f-5462-11ec-887f-049226b...

Super Process Instance ID:
null

Payment Retrieval Requested → Approve Payment → Charge Credit Card (1) → Payment Received

Variables Incidents Called Process Instances User Tasks Jobs External Tasks

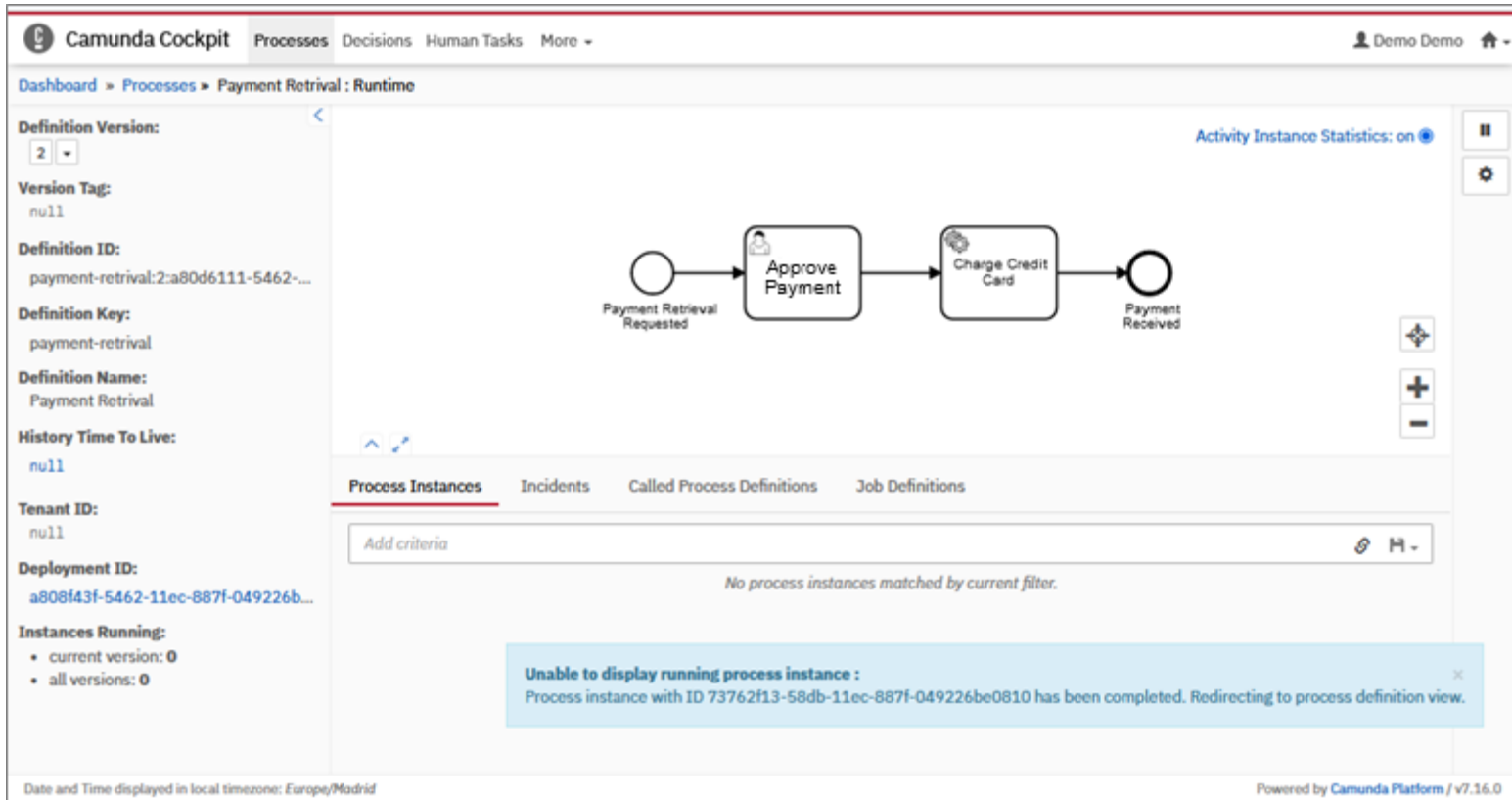
Add criteria 3 ⚙️ 📄

Name ^	Type —	Value	Scope	Actions
amount	Long	3125	Payment Retrival	✎️ 🗑️
approved	Boolean	true	Payment Retrival	✎️ 🗑️
item	String	GPS_Item_2021	Payment Retrival	✎️ 🗑️

localhost:8090/camunda/app/cockpit/default/#/ Powered by Camunda Platform / v7.16.0

7. Continua con la ejecución del proceso (IV)

Si el proceso finaliza, las variables ya no están disponibles.



Camunda Cockpit Processes Decisions Human Tasks More

Dashboard » Processes » Payment Retrieval : Runtime

Definition Version: 2

Version Tag: null

Definition ID: payment-retrieval:2:a80d6111-5462-...

Definition Key: payment-retrieval

Definition Name: Payment Retrieval

History Time To Live: null

Tenant ID: null

Deployment ID: a808f43f-5462-11ec-887f-049226b...

Instances Running: 0

- current version: 0
- all versions: 0

Activity Instance Statistics: on

Payment Retrieval Requested → Approve Payment → Charge Credit Card → Payment Received

Process Instances Incidents Called Process Definitions Job Definitions

Add criteria

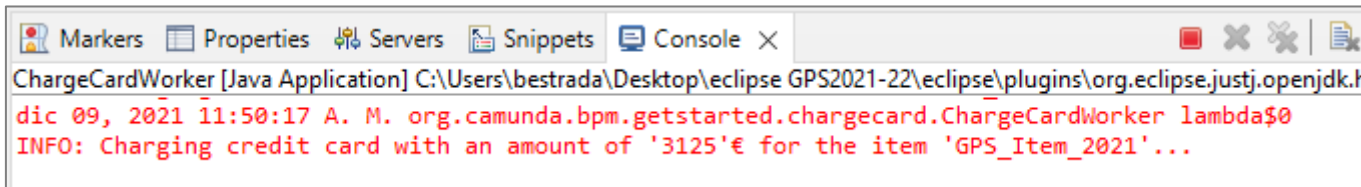
No process instances matched by current filter.

Unable to display running process instance : Process instance with ID 73762f13-58db-11ec-887f-049226be0810 has been completed. Redirecting to process definition view.

Date and Time displayed in local timezone: Europe/Madrid

Powered by Camunda Platform / v7.16.0

En la consola de Eclipse se muestra el resultado de la ejecución de la tarea.



```
Markers Properties Servers Snippets Console X
ChargeCardWorker [Java Application] C:\Users\bestrada\Desktop\eclipse GPS2021-22\eclipse\plugins\org.eclipse.justj.openjdk.h
dic 09, 2021 11:50:17 A. M. org.camunda.bpm.getstarted.chargecard.ChargeCardWorker lambda$0
INFO: Charging credit card with an amount of '3125'€ for the item 'GPS_Item_2021'...
```

GATEWAYS

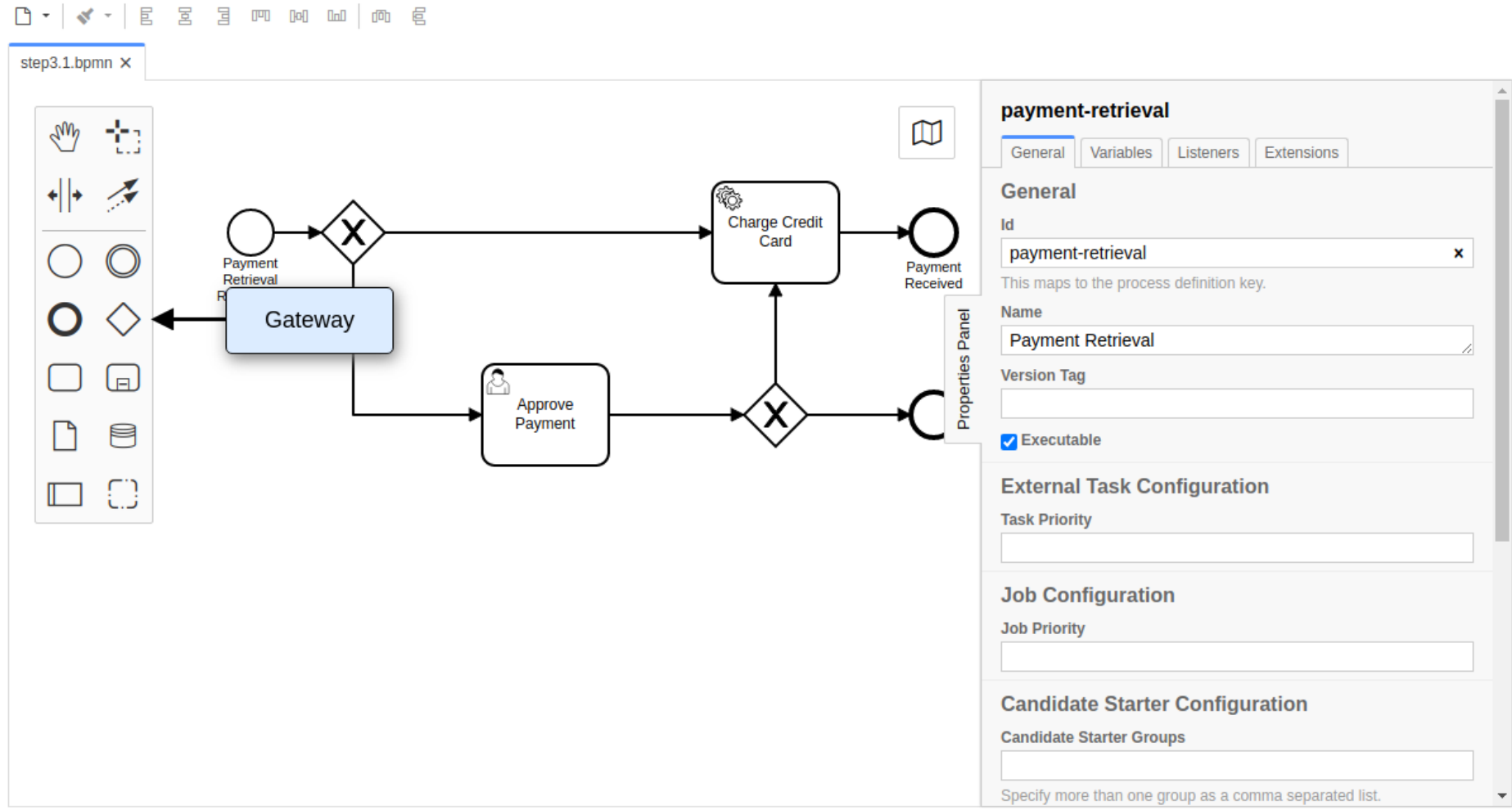
Pasos para hacer dinámico el proceso agregando compuertas

Los pasos que se describen a continuación están basados en el tutorial disponible en:

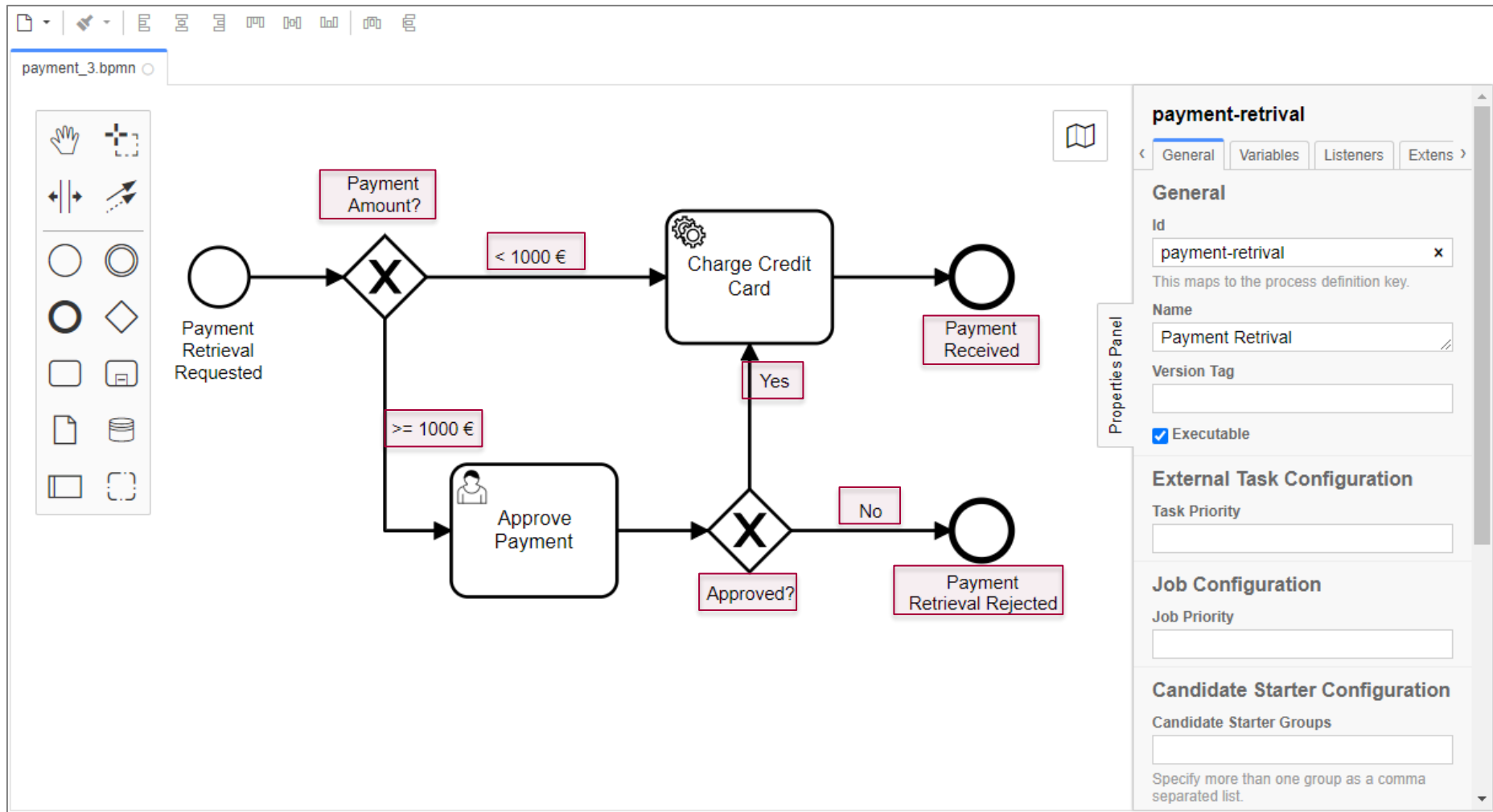
<https://docs.camunda.org/get-started/quick-start/gateway/>

1. Agrega dos compuertas al proceso
2. Asigna nombres a los nuevos elementos del proceso
3. Configura las compuertas
4. Despliega el Proceso
5. Inicia una nueva instancia del proceso
6. Verifica el Tasklist

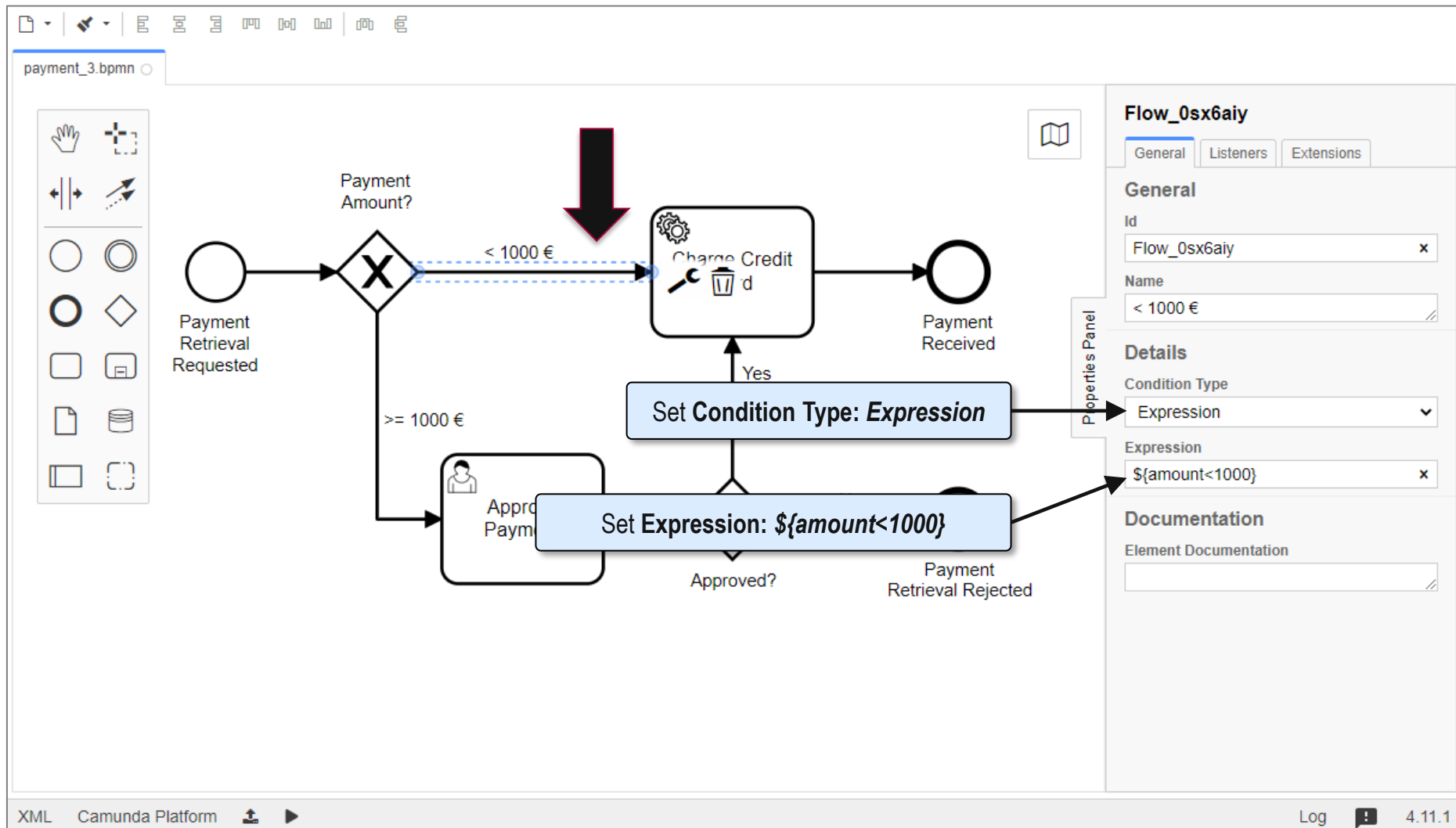
1. Agrega dos compuertas al proceso



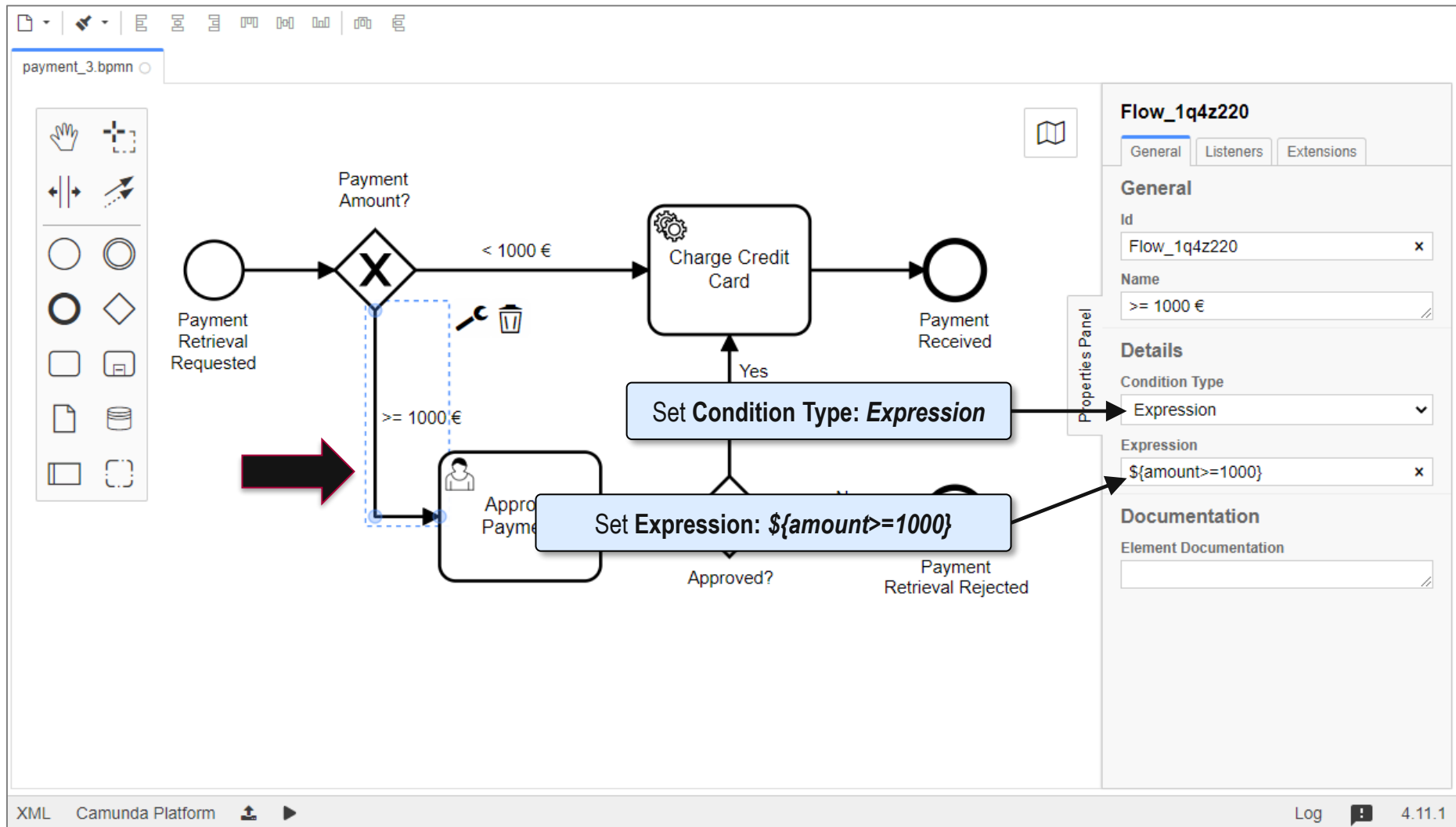
2. Asigna nombres a los nuevos elementos del proceso



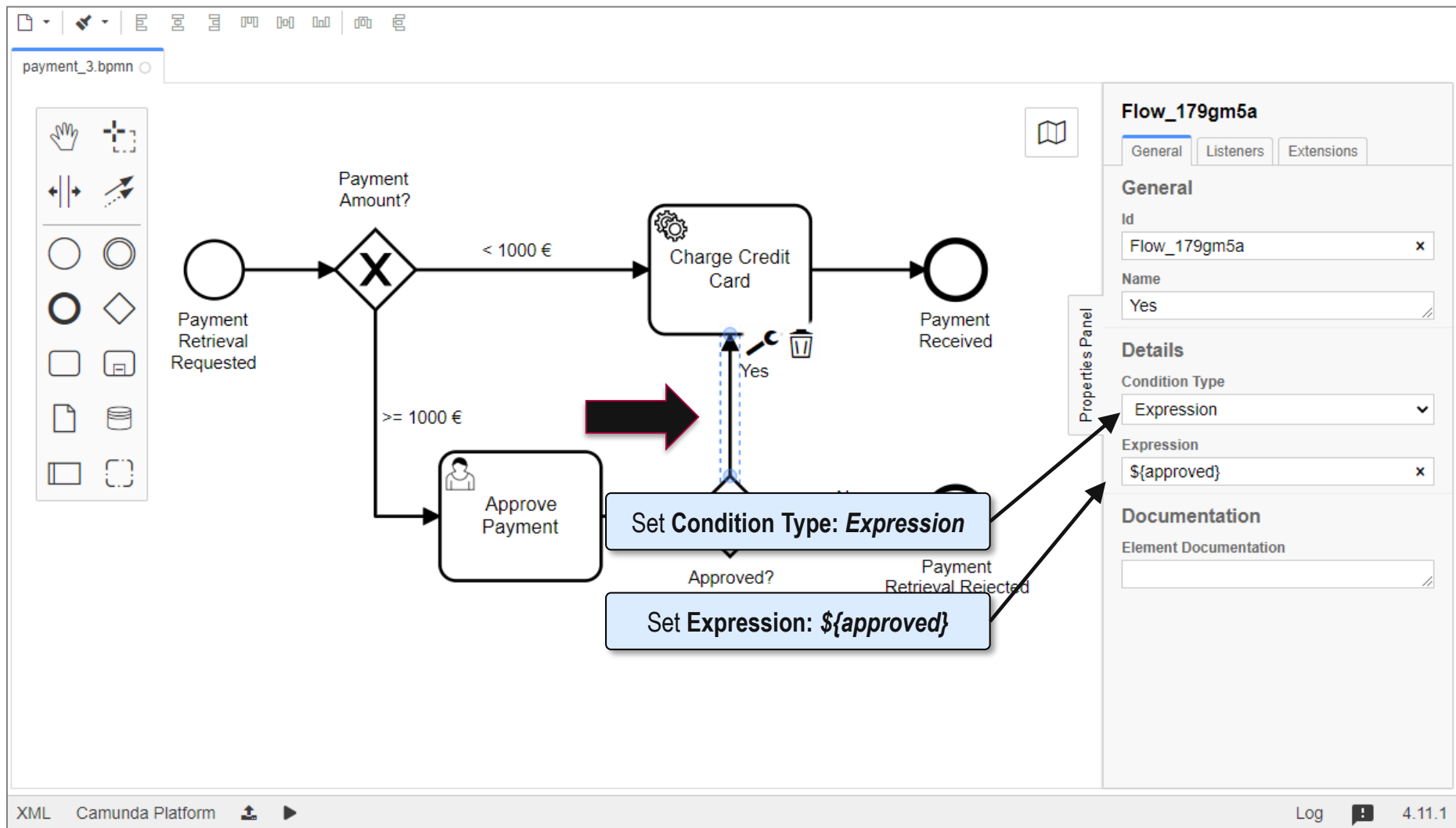
3. Configura las compuertas (I)



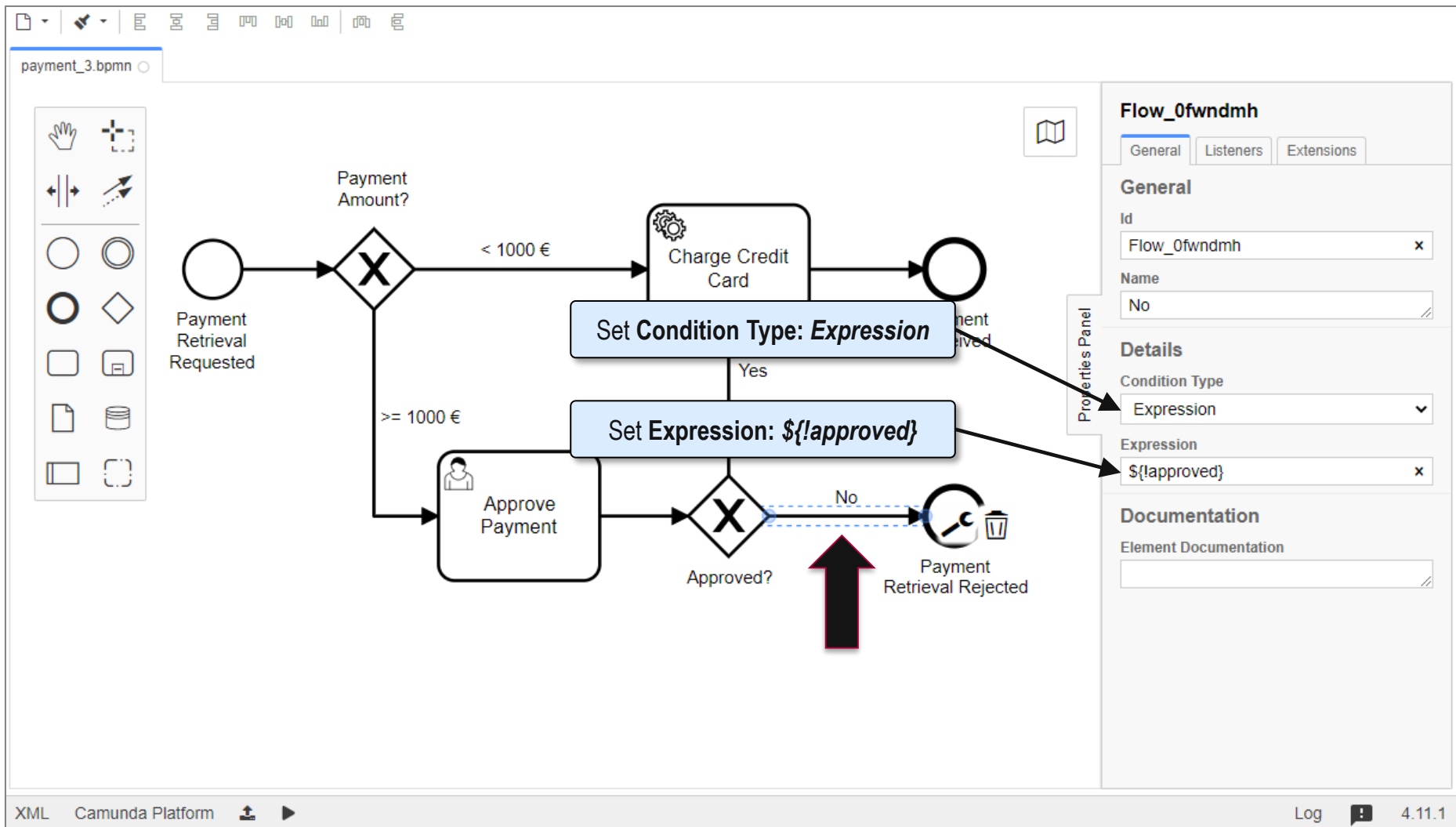
3. Configura las compuertas (II)



3. Configura las compuertas (III)



3. Configura las compuertas (IV)



4. Despliega el Proceso

The screenshot displays the Camunda Platform interface. On the left, a BPMN diagram titled "payment_3.bpmn" is open, showing a "Payment Retrieval Requested" event. A red box highlights the "Deploy" button in the bottom-left corner of the interface.

In the center, a modal dialog titled "Deploy Diagram to Camunda Platform" is open. The dialog contains the following fields and sections:

- Deployment Name:** A text field containing "Payment Retrieval".
- Tenant ID:** A text field with the placeholder "Optional".
- Endpoint Configuration:**
 - REST Endpoint:** A text field containing "http://localhost:8090/engine-rest".
- Include additional files:** A section with a "Select files" button.

At the bottom of the dialog are "Cancel" and "Deploy" buttons.

On the right side of the interface, the "payment-retrival" configuration panel is visible, showing tabs for "General", "Variables", "Listeners", and "Extens". The "General" tab is active, displaying fields for "Id" (payment-retrival), "Name" (Payment Retrieval), and "Version Tag". It also includes checkboxes for "Executable" (checked) and sections for "External Task Configuration", "Job Configuration", and "Candidate Starter Configuration".

5. Inicia una nueva instancia del proceso

- Ir a **Camunda Toolkit** e iniciar una nueva instancia del proceso “*Payment Retrieval*”.
- Utilice un valor mayor que 1000 en la variable “*amount*” para ver la tarea de usuario “Approve Payment”

Start process

Click on the process to start.

Invoice Receipt

Payment Retrieval

Close

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_01"/>
Remo... x	<input type="text" value="amount"/>	<input type="text" value="Long"/>	<input type="text" value="1247"/>

Back

Close

Start

6. Verifica el Tasklist (I)

The screenshot displays the Camunda Tasklist application. The top navigation bar includes the Camunda logo, the text 'Camunda Tasklist', and links for 'Keyboard Shortcuts', 'Create task', 'Start process', and a user profile 'Demo Demo'. Below this, a filter bar shows 'Create a filter +' and a dropdown menu currently set to 'Created'. A sidebar on the left lists filters, with 'All Tasks (7)' highlighted by a red box. The main task list shows three tasks: 'Approve Payment' (highlighted with a red box), 'Assign Reviewer', and 'Prepare Bank Transfer'. Each task entry includes the task name, a sub-task name, the assignee 'Demo Demo', and a priority '50'. The 'Approve Payment' task details are shown on the right, including a 'Payment Retrieval' link, action buttons like 'Set follow-...', 'Set due date', and 'Add groups', and tabs for 'Form', 'History', 'Diagram', and 'Description'. The 'Form' tab is active, showing fields for 'Item:' (containing 'Item_GPS_01') and 'Amount:' (containing '1247'). There is an 'Approved?' checkbox which is currently unchecked. At the bottom right of the form are 'Save' and 'Complete' buttons.

Camunda Tasklist

Keyboard Shortcuts Create task Start process Demo Demo

Create a filter +

All Tasks (7)

Created +

Filter Tasks 7

Approve Payment

Payment Retrieval Demo Demo
Created 2 minutes ago 50

Assign Reviewer

Review Invoice Demo Demo
Created 6 days ago 50

Prepare Bank Transfer

Invoice Receipt Demo Demo
Due in a day, Created 6 days ago 50

Approve Invoice

Invoice Receipt Demo Demo
Due in a day, Created 6 days ago 50

Approve Payment

Payment Retrieval

Set follow-... Set due date Add groups Demo Demo

Form History Diagram Description

Item:

Item_GPS_01

Amount:

1247

Approved?

☐

Save Complete

Date and Time displayed in local timezone: Europe/Madrid

Powered by Camunda Platform / v7.16.0

6. Verifica el Tasklist (II) (Diagrama)

Camunda Tasklist Keyboard Shortcuts Create task Start process Demo Demo

Create a filter + < Created ▾ + > > ↗ Add Comment +

All Tasks (7) *Filter Tasks* 7 ⚙️ 🏠

- Approve Payment**
Payment Retrieval Demo Demo
Created 4 minutes ago 50
- Assign Reviewer**
Review Invoice Demo Demo
Created 6 days ago 50
- Prepare Bank Transfer**
Invoice Receipt
Due in a day, Created 6 days ago 50
- Approve Invoice**
Invoice Receipt
Due in a day, Created 6 days ago 50

Approve Payment
Payment Retrieval

Set follow-up ... Set due date Add groups Demo Demo ×

Form History **Diagram** Description

```
graph LR; Start(( )) --> G1{X  
Payment Amount?}; G1 -- "< 1000 €" --> T1[Charge Credit Card]; T1 --> E1((Payment Received)); G1 -- ">= 1000 €" --> T2[Approve Payment]; T2 --> G2{X  
Approved?}; G2 -- "Yes" --> T1; G2 -- "No" --> E2((Payment Retrieval Rejected));
```

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

6. Verifica el Tasklist (III)

Completa la tarea **aprobando el pago**.

The screenshot displays the Camunda Tasklist interface. On the left, a sidebar shows a list of tasks: 'Approve Payment' (Payment Retrival, Demo Demo, Created 6 minutes ago, 50), 'Assign Reviewer' (Review Invoice, Demo Demo, Created 6 days ago, 50), 'Prepare Bank Transfer' (Invoice Receipt, Due in a day, Created 6 days ago, 50), and 'Approve Invoice' (Invoice Receipt, Due in a day, Created 6 days ago, 50). The main panel shows the details for the 'Approve Payment' task. It includes a 'Payment Retrival' link, buttons for 'Set follow-up ...', 'Set due date', 'Add groups', and 'Demo Demo'. Below these are tabs for 'Form', 'History', 'Diagram', and 'Description'. The 'Form' tab is active, showing fields for 'Item:' (Item_GPS_01) and 'Amount:' (1247). At the bottom, there is a checkbox labeled 'Approved?' which is checked, and a red arrow points to it. To the right of the checkbox are 'Save' and 'Complete' buttons. The footer indicates the date and time are displayed in the local timezone: Europe/Madrid, and the interface is powered by Camunda Platform / v7.16.0.

Camunda Tasklist

Keyboard Shortcuts Create task Start process Demo Demo

Create a filter + < Created + > Add Comment +

All Tasks (7) Filter Tasks 7

Approve Payment

Payment Retrival Demo Demo

Created 6 minutes ago 50

Assign Reviewer

Review Invoice Demo Demo

Created 6 days ago 50

Prepare Bank Transfer

Invoice Receipt

Due in a day, Created 6 days ago 50

Approve Invoice

Invoice Receipt

Due in a day, Created 6 days ago 50

Approve Payment

Payment Retrival

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

Form History Diagram Description

Item:

Item_GPS_01

Amount:

1247

Approved?

☒

Save Complete

Date and Time displayed in local timezone: Europe/Madrid

Powered by Camunda Platform / v7.16.0

6. Verifica el Tasklist (III)

Iniciar un nuevo proceso y completa la tarea sin aprobar el pago.

Camunda Tasklist

Keyboard Shortcuts Create task Start process Demo Demo

Create a filter + < Created + > > > Add Comment +

All Tasks (7) Filter Tasks 7 7 7

Approve Payment
Payment Retrieval Demo Demo
Created 7 minutes ago 50

Assign Reviewer
Review Invoice Demo Demo
Created 6 days ago 50

Prepare Bank Transfer
Invoice Receipt 50
Due in a day, Created 6 days ago

Approve Invoice
Invoice Receipt 50
Due in a day, Created 6 days ago

Approve Payment
Payment Retrieval
Set follow-up ... Set due date Add groups Demo Demo x

Form History Diagram Description

Item:
Item_GPS_01

Amount:
1247

Approved?
☐

Save Complete

- El proceso acaba sin pasar por la tarea de Servicio.
- Verificar la consola de Eclipse. Los datos de esta ejecución no aparecen.

EJEMPLO EXTRA

¿Qué hacer si tenemos más de una
tarea de servicio?

1. Modifica el modelo desde Camunda Modeler y configura la tarea.

The screenshot displays the Camunda Modeler interface with a BPMN diagram titled "payment_4.bpmn". The diagram includes a start event "Payment Retrieval Requested", a task "Charge User Data" (highlighted with a blue dashed box and an annotation "1. Agrega una nueva Tarea de Servicio"), an exclusive gateway "Payment amount?", a task "Charge Credit Card", an "Approve Payment" task, another exclusive gateway "Approved?", and an end event "Payment Retrieval Rejected".

The Properties Panel on the right is open for the selected task "Activity_0dl0wb2". The "General" tab is active, showing the task name "Charge User Data". The "Details" section is highlighted with a red box, showing the "Implementation" set to "External" and the "Topic" set to "charge-userdata". The "External Task Configuration" section is also visible.

Three numbered annotations guide the user:

- 1. Agrega una nueva Tarea de Servicio (points to the "Charge User Data" task)
- 2. Configura como tarea **External** y asigna un **topic** (charge-userdata) (points to the "Details" section in the Properties Panel)
- 3. Después de configurar, desplegar el proceso (points to the deployment icon in the bottom toolbar)

The bottom toolbar shows the "XML" view, "Camunda Platform" logo, a deployment icon (highlighted with a red box), and a play button. The status bar at the bottom right indicates "Log", a warning icon, and version "4.11.1".

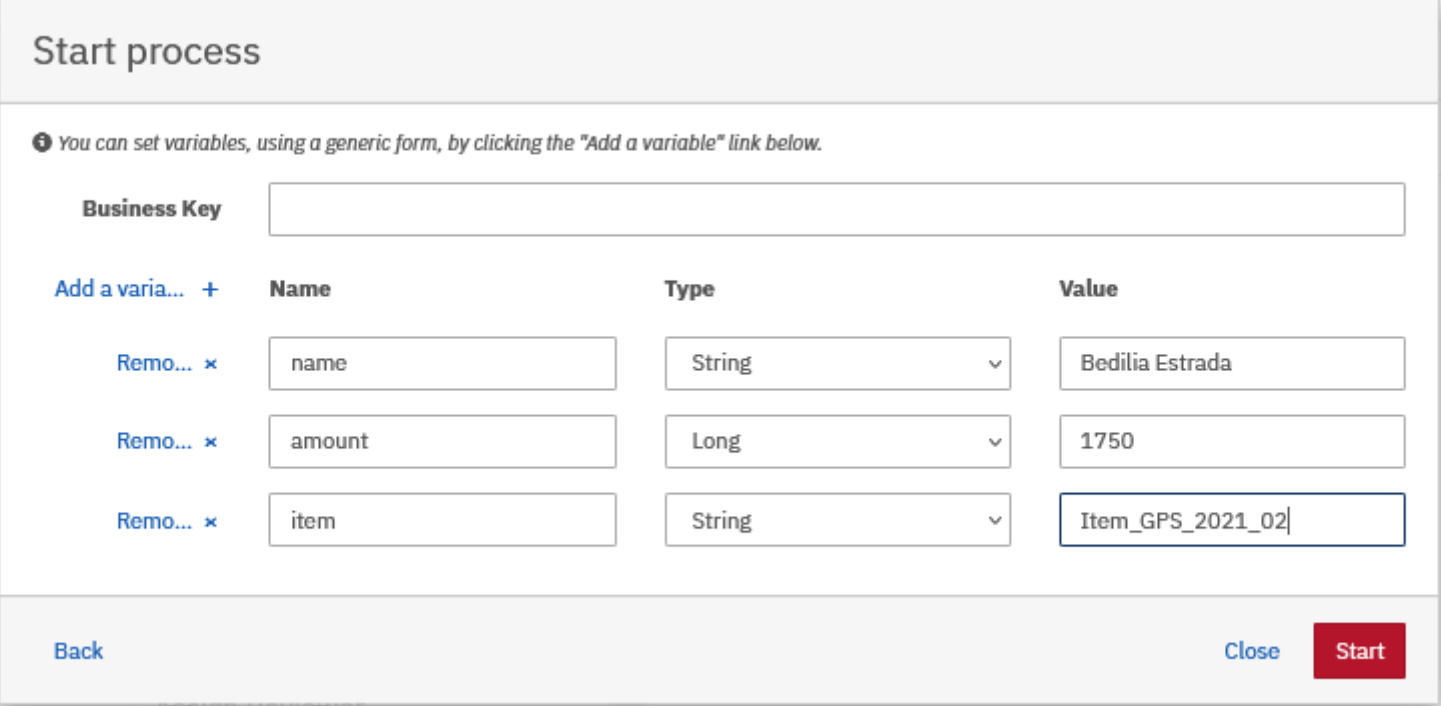
2. Agrega un nuevo “*client.subscribe*” a la clase Java

```
ChargeCardWorker.java X
18 // subscribe to an external task topic as specified in the process
19 client.subscribe("charge-card")
20     .lockDuration(1000) // the default lock duration is 20 seconds, but you can override this
21     .handler((externalTask, externalTaskService) -> {
22         // Put your business logic here
23
24         // Get a process variable
25         String item = (String) externalTask.getVariable("item");
26         Long amount = (Long) externalTask.getVariable("amount");
27
28         LOGGER.info("Charging credit card with an amount of '" + amount + "'€ for the item '" + item +
29
30         // Complete the task
31         externalTaskService.complete(externalTask);
32     })
33     .open();
34 /**Inicia el código para la nueva tarea de servicio*****
35 client.subscribe("charge-userdata")
36     .lockDuration(1000) // the default lock duration is 20 seconds, but you can override this
37     .handler((externalTask, externalTaskService) -> {
38         // Put your business logic here
39
40         // Get a process variable
41         String item = (String) externalTask.getVariable("item");
42         Long amount = (Long) externalTask.getVariable("amount");
43         String name = (String) externalTask.getVariable("name");
44
45         LOGGER.info("User data::\n- Name: '" + name + "'\n- Amount: '" + amount + "'\n- Item: '" + item + "'
46
47         // Complete the task
48         externalTaskService.complete(externalTask);
49     })
50     .open();
51 /**Termina el código para la nueva tarea de servicio*****
```

Run as...

3. Ejecuta el proceso (I)

Incluye las tres variables



The image shows a screenshot of the Camunda Tasklist interface. A modal dialog titled "Start process" is open, allowing a user to configure and start a process. The dialog includes a "Business Key" field, an "Add a variable" link, and a table for defining variables. The table has three columns: "Name", "Type", and "Value". Three variables are listed: "name" (String, "Bedilia Estrada"), "amount" (Long, "1750"), and "item" (String, "Item_GPS_2021_02"). At the bottom of the dialog are "Back", "Close", and "Start" buttons.

Camunda Tasklist

Create a filter +

All Tasks (6)

Keyboard Shortcuts Create task Start process Demo Demo

Start process

i 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... ✕	<input type="text" value="name"/>	<input type="text" value="String"/>	<input type="text" value="Bedilia Estrada"/>
Remo... ✕	<input type="text" value="amount"/>	<input type="text" value="Long"/>	<input type="text" value="1750"/>
Remo... ✕	<input type="text" value="item"/>	<input type="text" value="String"/>	<input type="text" value="Item_GPS_2021_02"/>

Back Close Start

3. Ejecuta el proceso (II)

Camunda Cockpit

ProcessesDecisionsHuman TasksMore ▾

Demo Demo

Dashboard » Processes » Payment Retrieval : Runtime

Definition Version:
4 ▾

Version Tag:
null

Definition ID:
payment-retrival:4:aac6f2c2-58e6-1...

Definition Key:
payment-retrival

Definition Name:
Payment Retrieval

History Time To Live:
null

Tenant ID:
null

Deployment ID:
aab9ac50-58e6-11ec-887f-049226b...

Instances Running:

- current version: 1
- all versions: 1

Activity Instance Statistics: on ●

```
graph LR; Start((Payment Retrieval Requested)) --> ChargeUser[Charge User Data]; ChargeUser --> Amount{Payment Amount?}; Amount -- "< 1000 €" --> ChargeCard[Charge Credit Card]; Amount -- ">= 1000 €" --> Approve[Approve Payment]; ChargeCard --> End1((Payment Received)); Approve --> Approved{Approved?}; Approved -- Yes --> ChargeCard; Approved -- No --> End2((Payment Retrieval Rejected));
```

Process InstancesIncidentsCalled Process DefinitionsJob Definitions

Add criteria

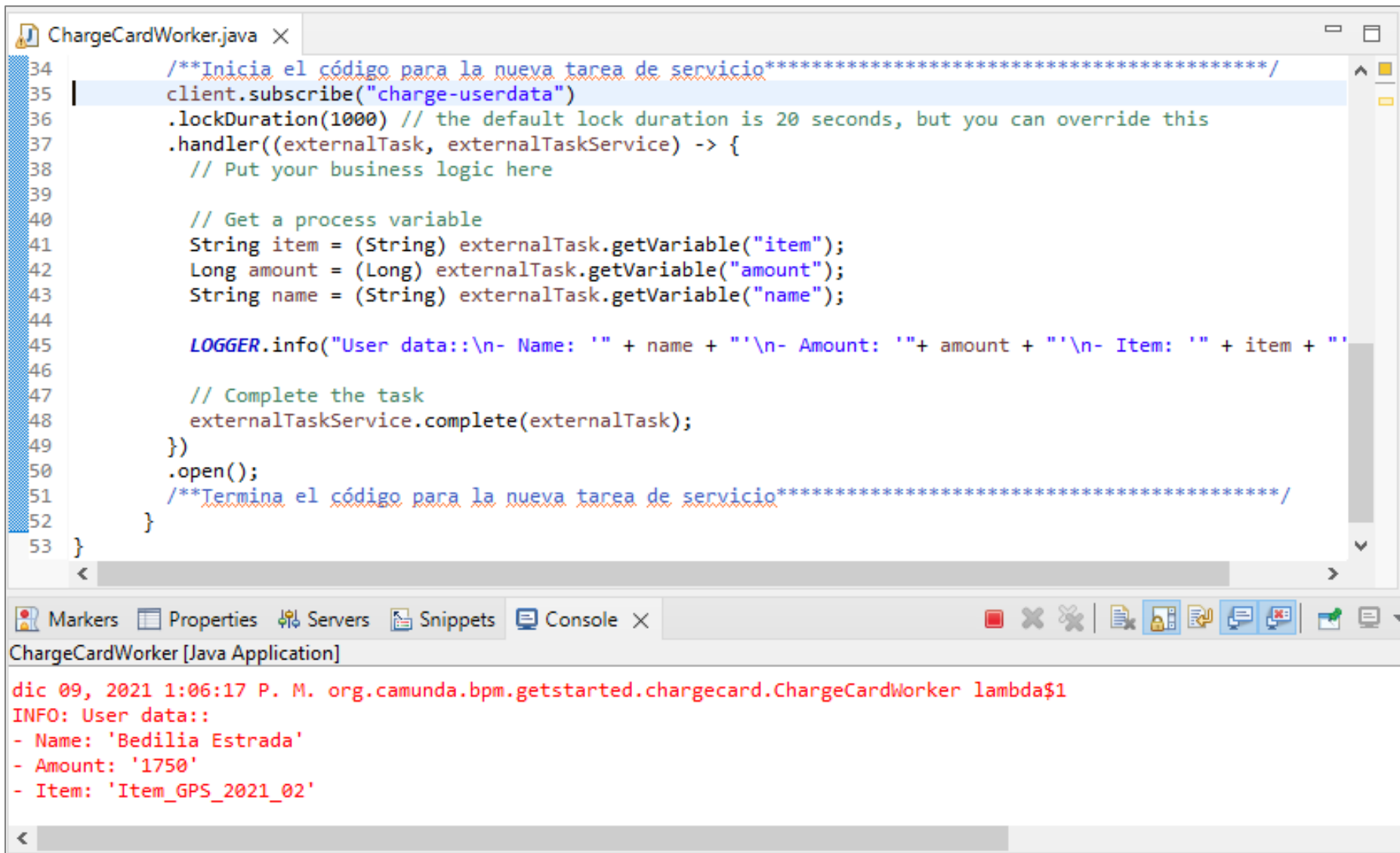
1 🔗 📄

State	ID	Start Time ▾	Business Key
✔	6150e10a-58e8-11ec-887f-049226be0810	2021-12-09T13:06:02	

Date and Time displayed in local timezone: Europe/Madrid

Powered by Camunda Platform / v7.16.0

4. Verifica que se ejecuta como tarea externa



The screenshot shows an IDE window titled 'ChargeCardWorker.java'. The code is a Java class implementing a task worker. It subscribes to a service named 'charge-userdata' with a lock duration of 1000 milliseconds. The handler method retrieves variables 'item', 'amount', and 'name' from the external task, logs them, and then completes the task. The console output at the bottom shows the execution of the task, logging the user data with the following values: Name: 'Bedilia Estrada', Amount: '1750', and Item: 'Item_GPS_2021_02'.

```
34  /**Inicia el código para la nueva tarea de servicio*****  
35  client.subscribe("charge-userdata")  
36  .lockDuration(1000) // the default lock duration is 20 seconds, but you can override this  
37  .handler((externalTask, externalTaskService) -> {  
38      // Put your business logic here  
39  
40      // Get a process variable  
41      String item = (String) externalTask.getVariable("item");  
42      Long amount = (Long) externalTask.getVariable("amount");  
43      String name = (String) externalTask.getVariable("name");  
44  
45      LOGGER.info("User data::\n- Name: '" + name + "'\n- Amount: '" + amount + "'\n- Item: '" + item + "'");  
46  
47      // Complete the task  
48      externalTaskService.complete(externalTask);  
49  })  
50  .open();  
51  /**Termina el código para la nueva tarea de servicio*****  
52  }  
53  }
```

Markers Properties Servers Snippets Console X

ChargeCardWorker [Java Application]

dic 09, 2021 1:06:17 P. M. org.camunda.bpm.getstarted.chargecard.ChargeCardWorker lambda\$1
INFO: User data::
- Name: 'Bedilia Estrada'
- Amount: '1750'
- Item: 'Item_GPS_2021_02'

RESUMEN

Resumen

Instalación de Camunda (Run y Modeler)

Modelado de un proceso simple

Implementar un External Task Worker

Despliegue

Tareas de usuario

Gateways

Ejemplo extra

