

# **SISTEMAS ORIENTADOS A SERVICIOS**

## **Práctica de composición de servicios**

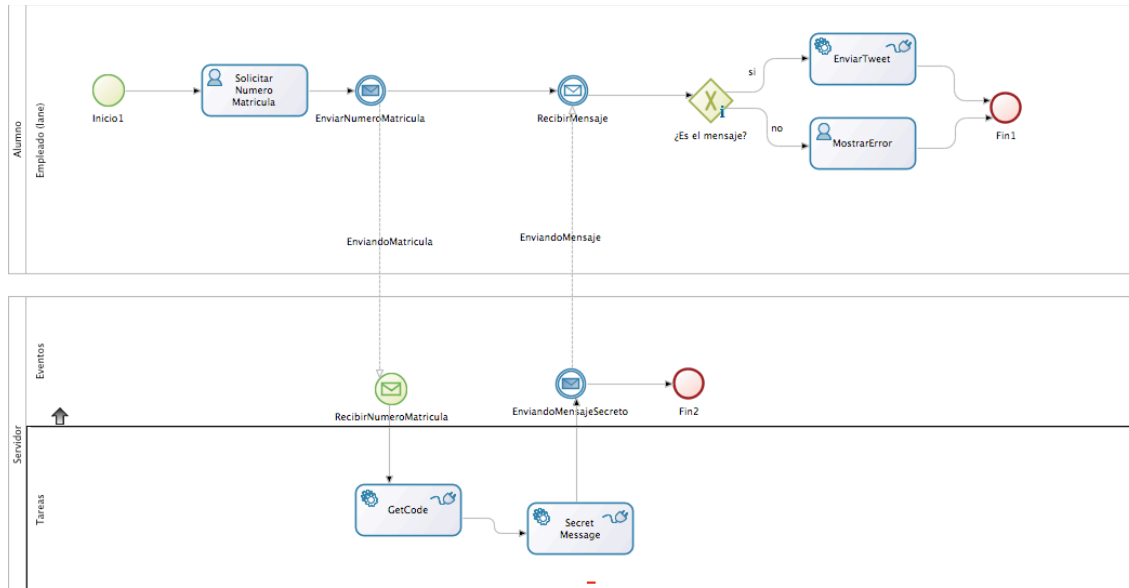
**Nombre: Angélica Guamán Albarracín, Matricula: t110212.**

## ÍNDICE

1. Coreografía implícita .....	3
2. Pool Alumno .....	3
3. Pool Servidor .....	6
4. RESULTADOS AL EJECUTAR LA APLICACIÓN .....	11

## 1. Coreografía implícita

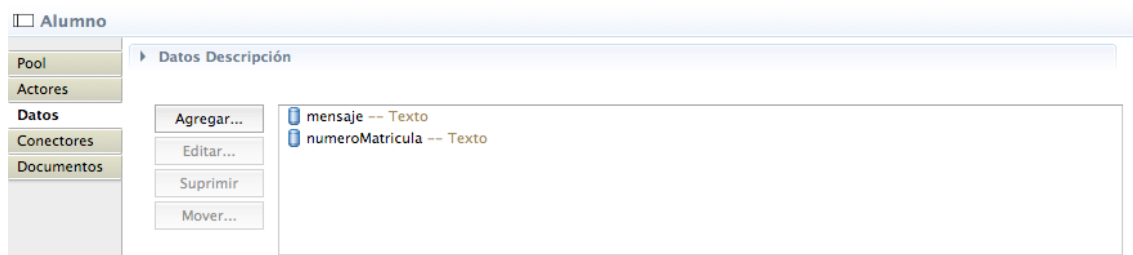
Para la creación de la coreografía utilizamos 2 pools, uno para **alumno** y otro para **servidor**, tal y como nos indica el enunciado y respetando todas las especificaciones que nos piden.



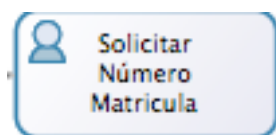
## 2. Pool Alumno

Este pool consta de una línea, el proceso Alumno es el proceso de inicio de esta práctica, primero se solicita el número de matrícula.

Creamos dos datos a nivel de pool de tipo texto, una para el número de matrícula y el otro para el mensaje descifrado.



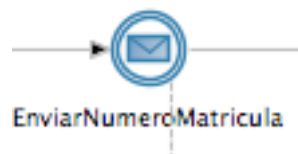
Pasamos a crear la tarea humana en el cual se desplegará un formulario pidiendo el número de matrícula.



Formulario para pedir el número de matrícula:

Numero Matricula	<input type="text"/>

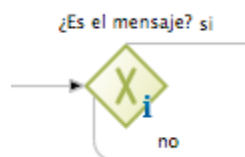
Enviamos un mensaje con el número de la matrícula al siguiente pool, Servidor, para que este proceso realice las operaciones oportunas para obtener el resultado esperado.



Una vez que el proceso Servidor realice las operaciones envía un mensaje al proceso Alumno este dato lo guardamos en la variable mensaje. La variable mensaje contiene el mensaje secreto a twittear.



Para comprobar si todo a salido con éxito utilizamos una puerta XOR que según su resultado hace una u otra operación.



Si es el mensaje debe cumplir lo siguiente:

Nombre	<input type="text" value="si"/>
Descripción	<div></div>
<input type="checkbox"/> Flujo por defecto	
Condición	<input checked="" type="radio"/> Use la expresión <input type="radio"/> Utilizar la tabla de decisión
	<div><div>mensaje!="Te has inventado el identificador"</div><div></div><div></div></div>

En el caso que se cumpla pasamos a enviar el tweet y seguidamente finaliza la tarea.



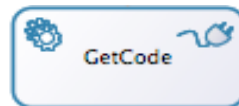
### 3. Pool Servidor

Proceso en el que vamos a llamar a los wsdl de **getCode** y **SecretMessages**. Aquí vamos a tener dos lines una para eventos y otro para tareas. En el line de tareas vamos a tener dos tareas servicio para crear la conexión SOAP con los wsdl. En este proceso tenemos tres datas, una para recibir el número de matrícula, otro para guardar el mensaje que devuelve GetCode y otra para enviar el mensaje al proceso Alumno todas de tipo Texto.

Agregar...	message -- Texto
Editar...	messageSecret -- Texto
Suprimir	numberMatricula -- Texto
Mover...	

### Tarea GetCode

Creamos el conector de SOAP.



Parámetros de la configuración:

Servicios Web SOAP 1.2 (1.0.0)

**Parametros de conexion**  
Introduzca la información de conexión del servicio web

Nombre de usuario

Contraseña

Servicio NS \*

nombre del servicio \*

Carga Guardar Prueba < Anterior Siguiente> Cancelar Finalizar

Servicios Web SOAP 1.2 (1.0.0)

### parámetros de solicitud

Introduzca la información de solicitud de servicio Web

Accion SOAP




Nombre del puerto \*

Dirección de punto final \*

Enlace \*

sobre \*

[Cambiar editor](#)

   [< Anterior](#) [Siguiente >](#) [Cancelar](#) [Finalizar](#)

Servicios Web SOAP 1.2 (1.0.0)




### Configuración de respuesta

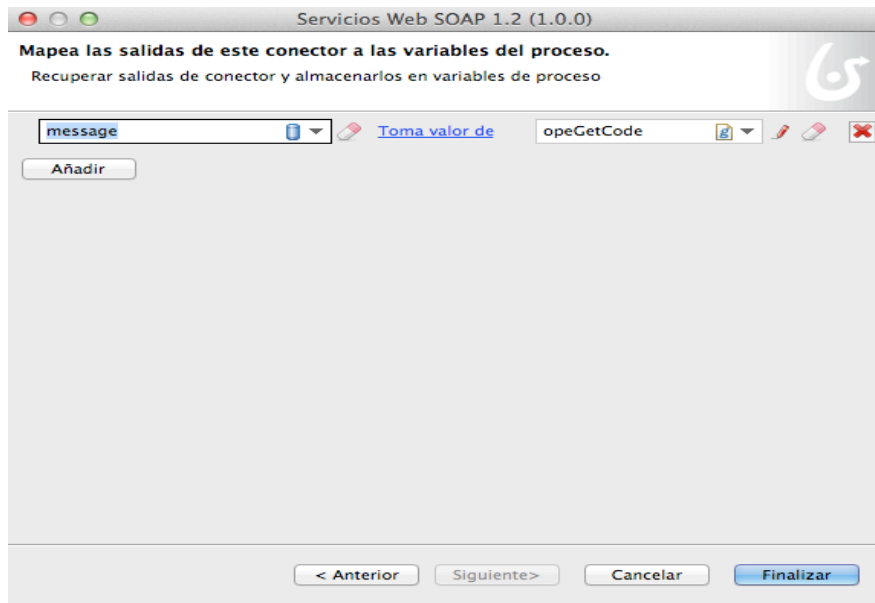
Entre las propiedades de respuesta

Returns envelope ☐ [Cambiar editor](#)

Revisar cuerpo ☒ [Cambiar editor](#)

Imprimir información de depuración ☐ [Cambiar editor](#)

   [< Anterior](#) [Siguiente >](#) [Cancelar](#) [Finalizar](#)

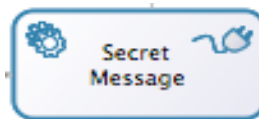


opeGetCode es un Script que contiene lo siguiente:

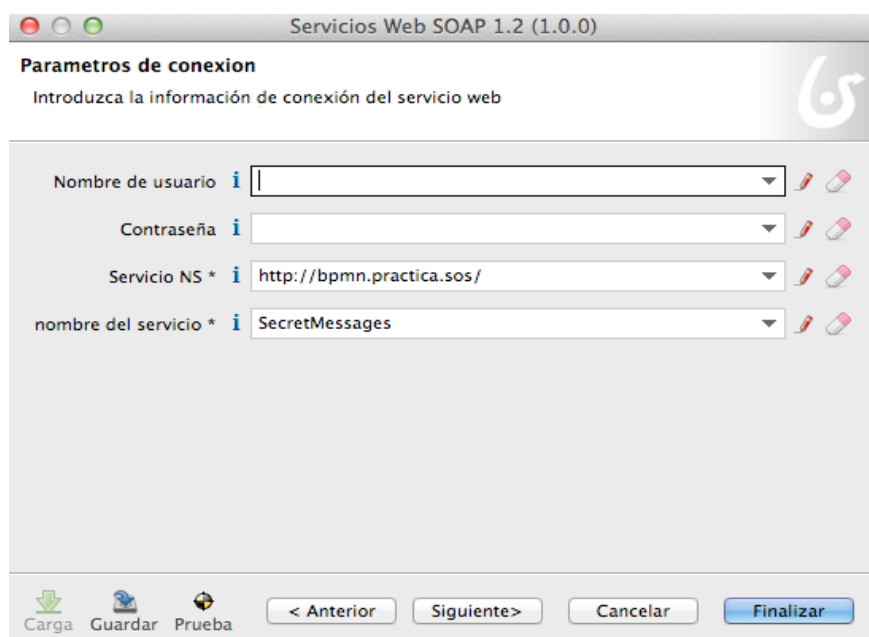
```
BonitaXML.evaluateXPathOnVariable(responseDocumentBody,
"/getCodeResponse/return/text()")
```

## ***Tarea Secret Message***

Creamos el conector de SOAP:



Parámetros de la configuración:





Servicios Web SOAP 1.2 (1.0.0)

### parámetros de solicitud

Introduzca la información de solicitud de servicio Web

Accion SOAP

Nombre del puerto \*




Dirección de punto final \*

Enlace \*

sobre \* 

```
<?xml version="1.0" encoding="UTF-8"?>
<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/">
  <S:Header/>
  <S:Body>
    <ns2:getSecretMessage xmlns:ns2="http://bpmn.practica.sos/">
      <id> ${messageSecret}</id>
    </ns2:getSecretMessage>
  </S:Body>
</S:Envelope>
```

[Cambiar editor](#)

Servicios Web SOAP 1.2 (1.0.0)

### Configuración de respuesta

Entre las propiedades de respuesta

Returns envelope ☐ [Cambiar editor](#)

Revisar cuerpo ☒ [Cambiar editor](#)

Imprimir información de depuración ☐ [Cambiar editor](#)

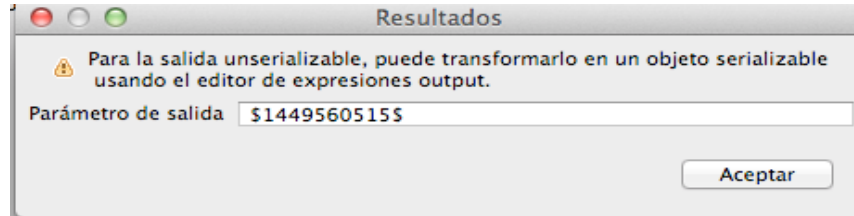


opeSecretMessage es un Script que contiene lo siguiente:

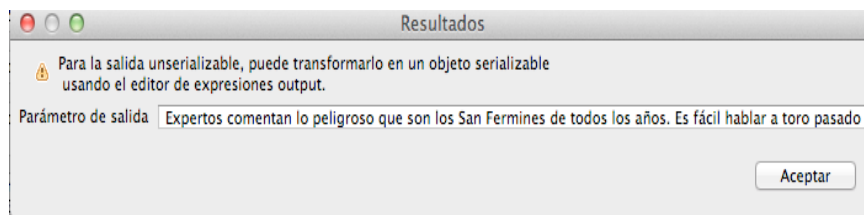
```
BonitaXML.evaluateXPathOnVariable(responseDocumentBody,  
"/getMessageResponse/return/text()")
```

## 4. RESULTADOS AL EJECUTAR LA APLICACIÓN

Resultado del Getcode:



Resultado del SecretMessages:



Formulario al pedir el número de matricula:

Solicitar Numero Matricula

Numero Matricula

Después de ejecutar la aplicación en conjunto, nos publica el tweet.

