Tabla de contenido

[Paso 1: Obtener datos del expediente en curso 2](#_Toc403637281)

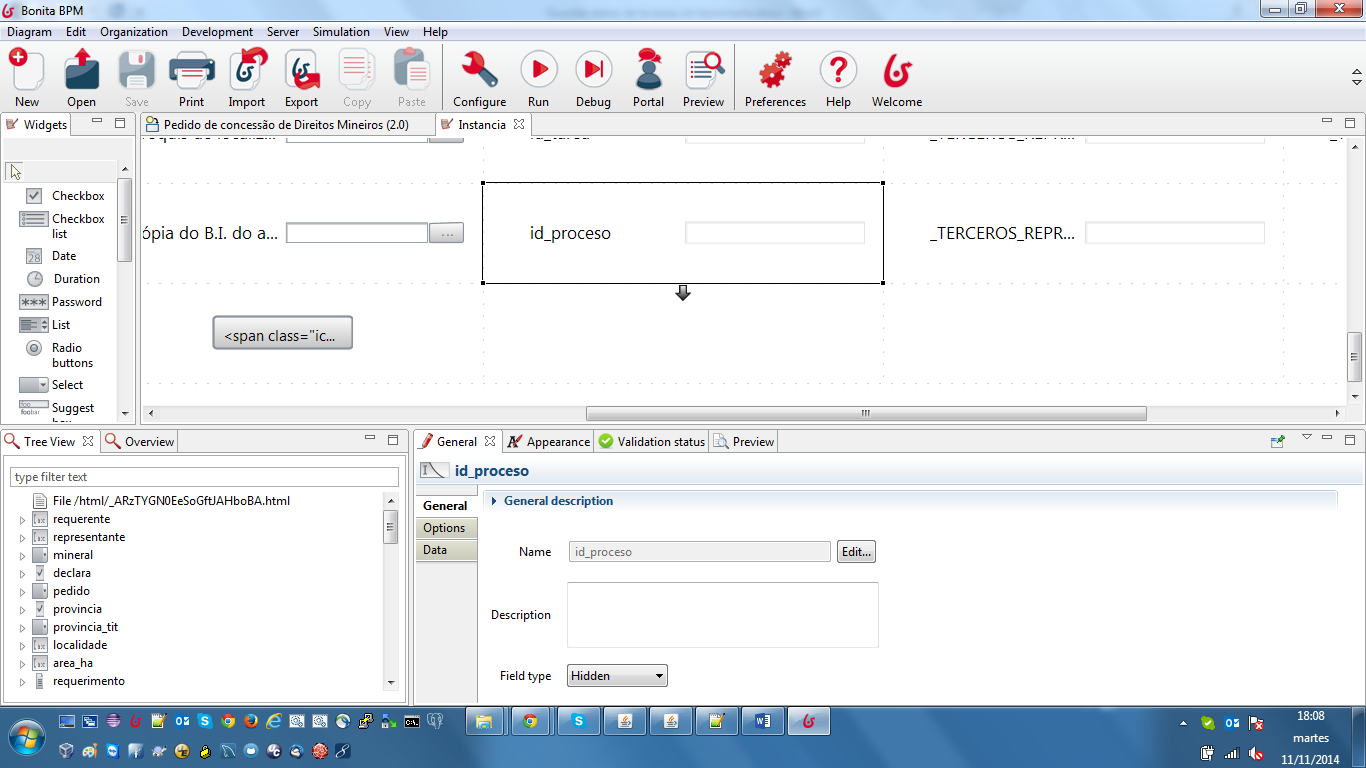
[Paso 2: Programar botón de guardado 5](#_Toc403637282)

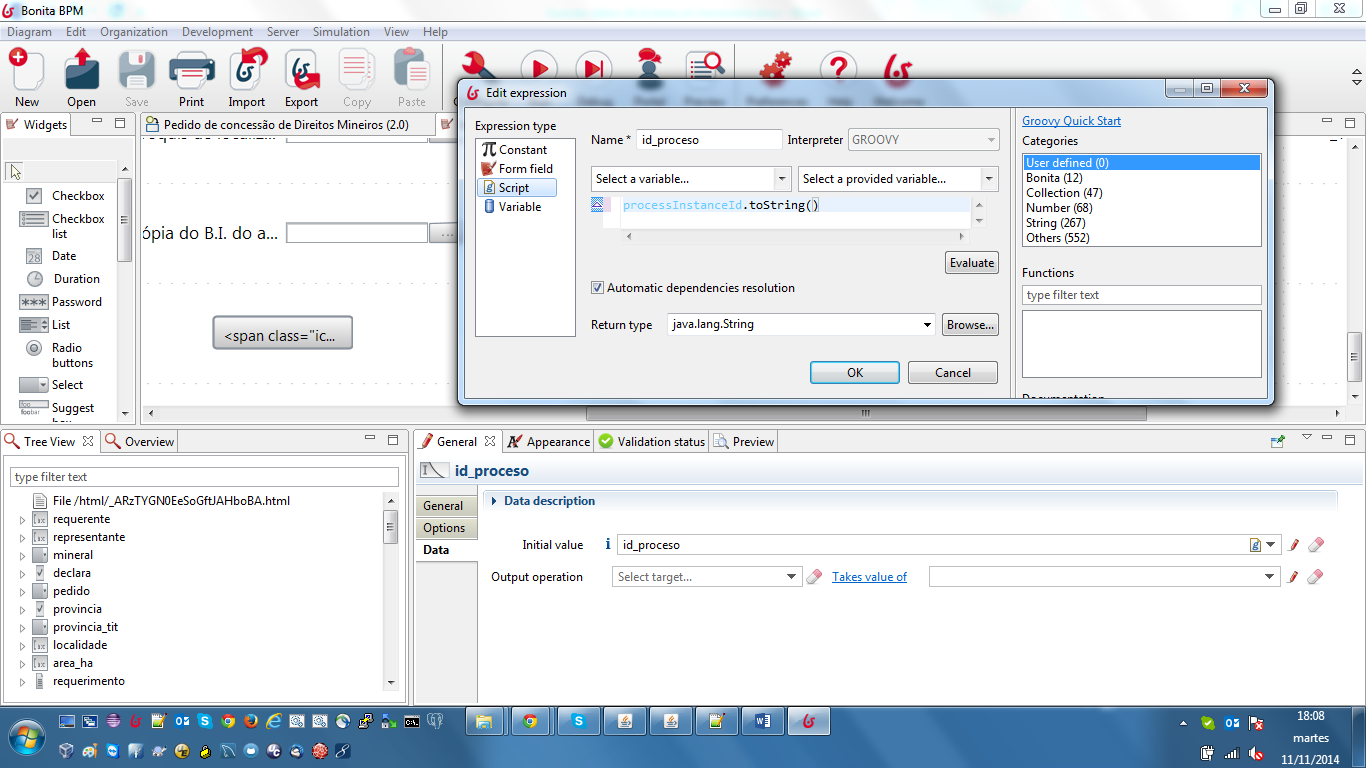
[ANEXO 7](#_Toc403637283)

## Paso 1: Obtener datos del expediente en curso

Puede que se necesiten varios campos ocultos, dependiendo de las necesidades:

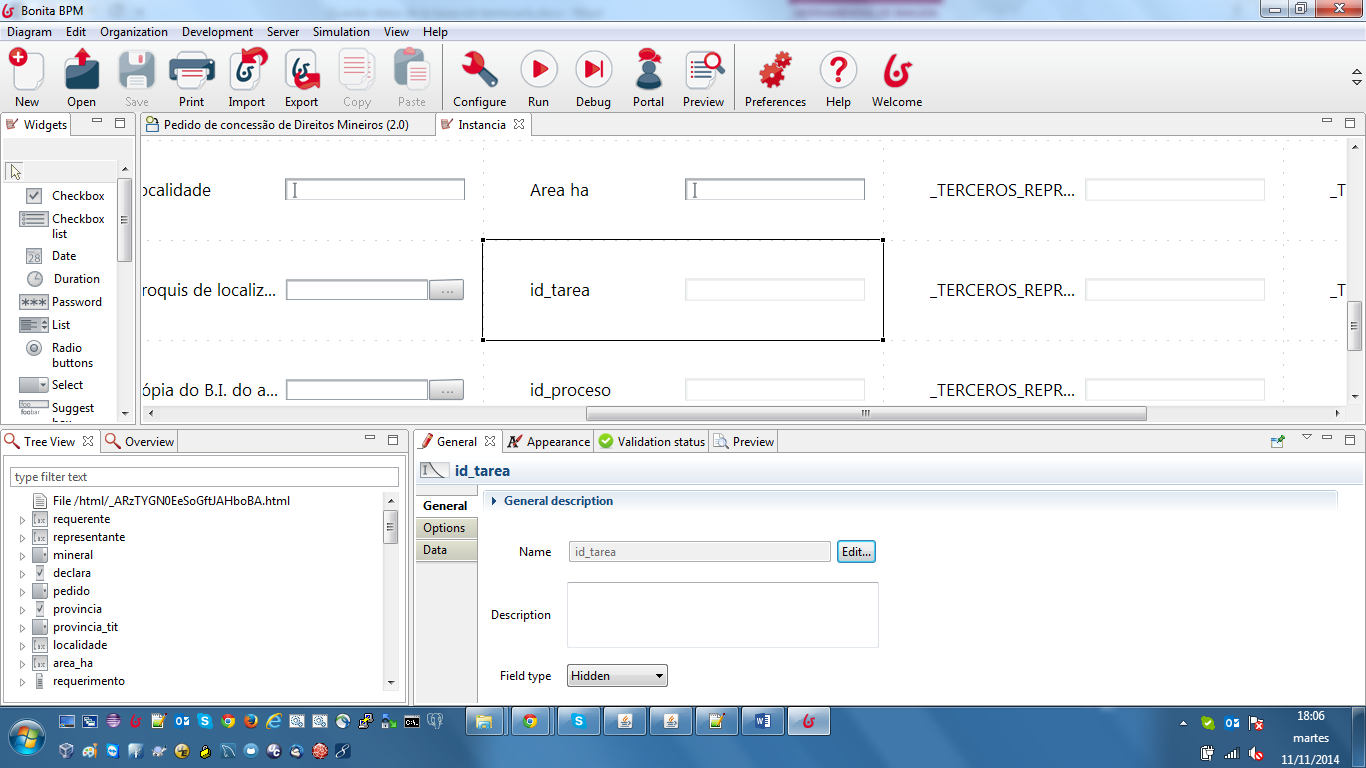
* uno que nos informe del id del expediente.

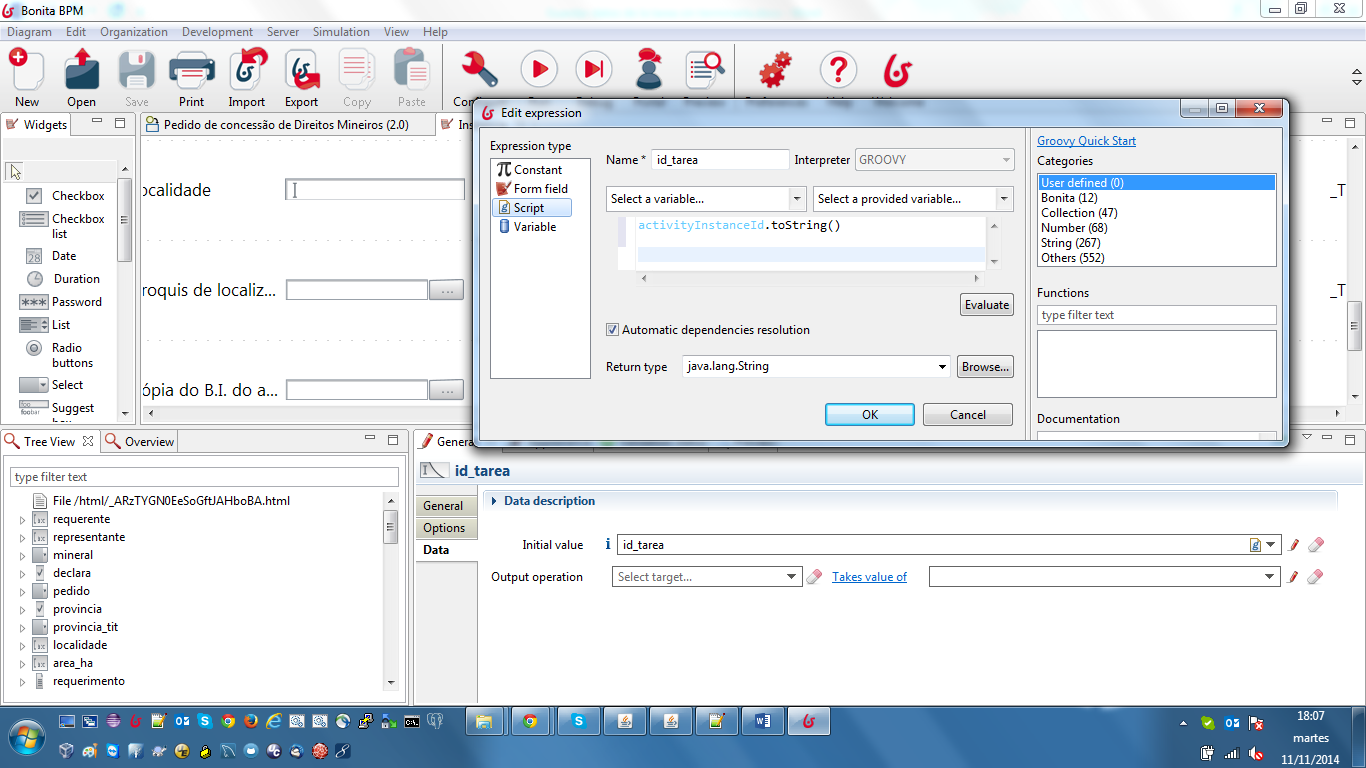




processInstanceId.toString()

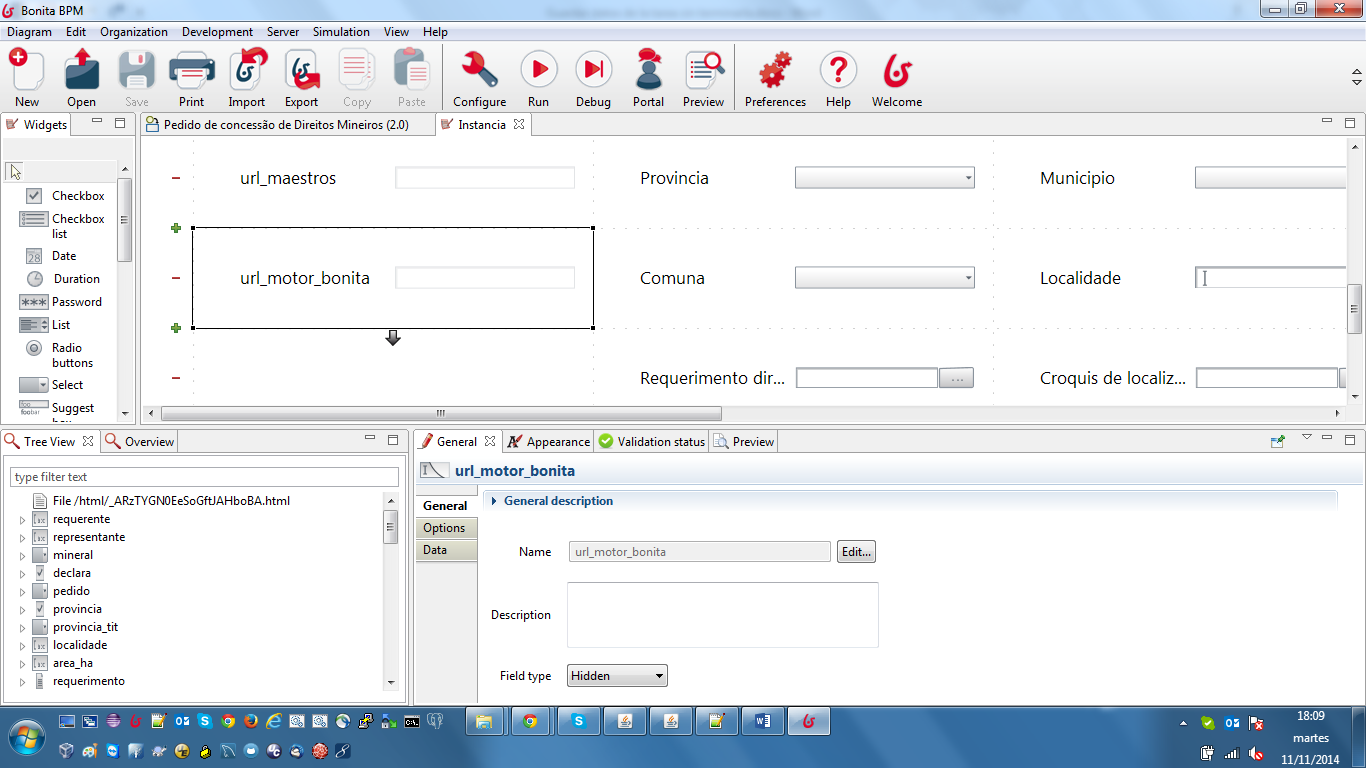
* Otro con el id de la tarea en curso



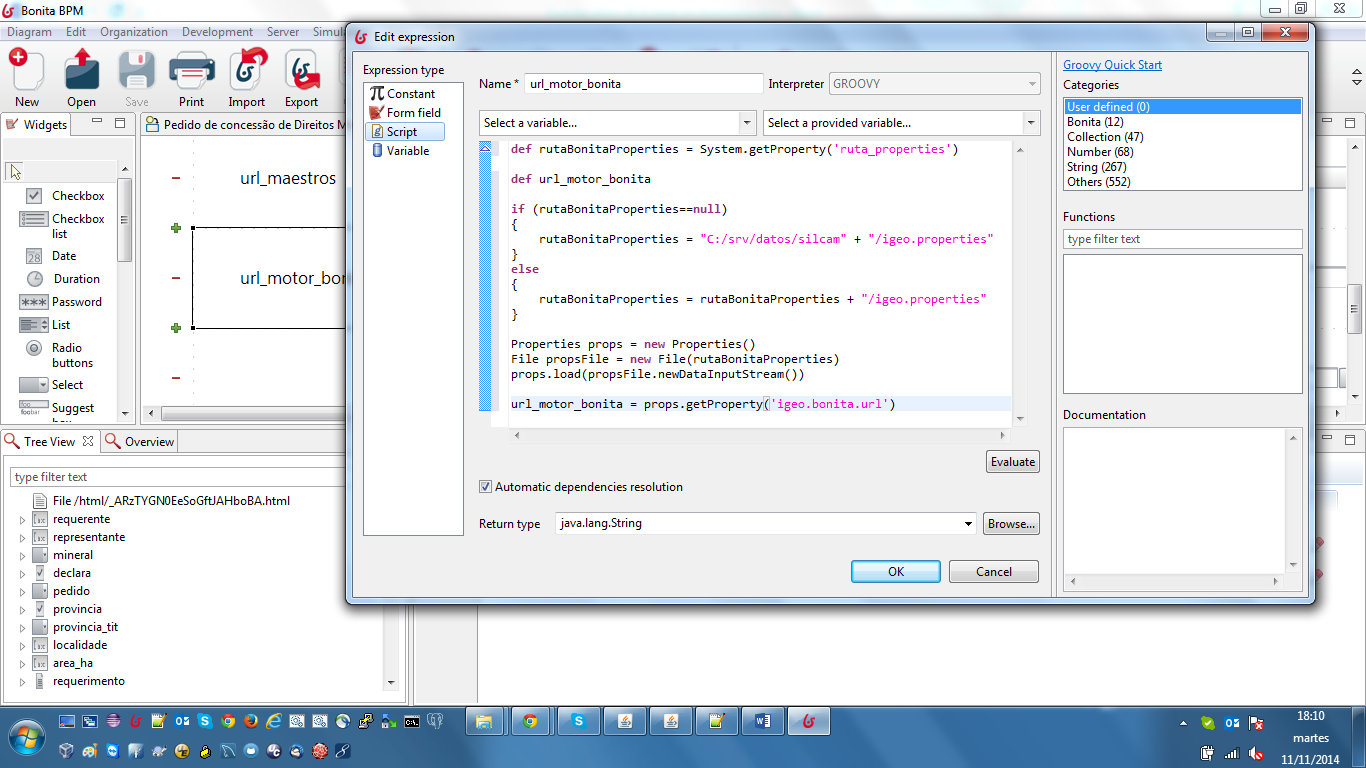


activityInstanceId.toString()

* otro que nos indique la url del motor de bonita en el entorno en el que estamos.



Leemos de un properties la clave que nos indica la url:



Código:

**def** rutaBonitaProperties = System.getProperty('ruta\_properties')

**def** url\_motor\_bonita

rutaBonitaProperties = rutaBonitaProperties + "/igeo.properties"

Properties props = **new** Properties()

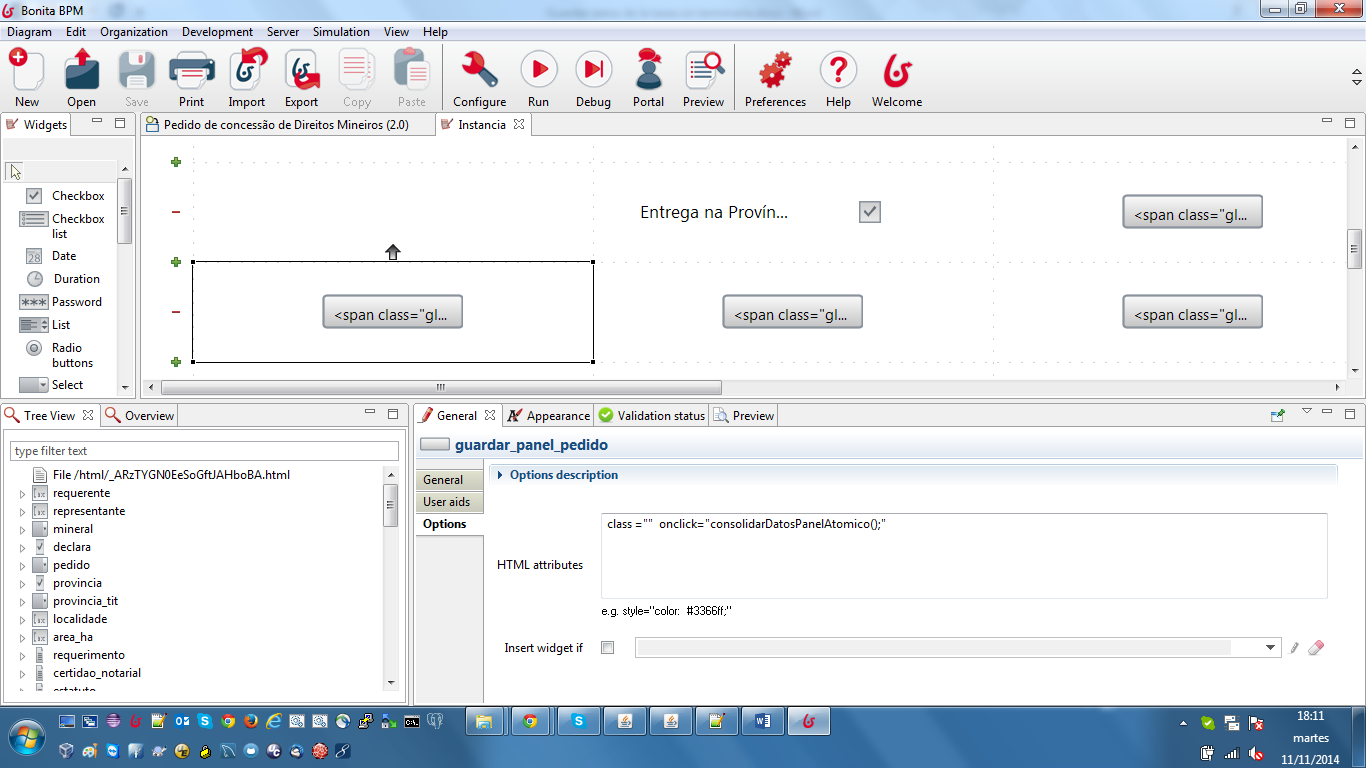
File propsFile = **new** File(rutaBonitaProperties)

props.load(propsFile.newDataInputStream())

url\_motor\_bonita = props.getProperty('igeo.bonita.url')

## Paso 2: Programar botón de guardado

El botón de guardar al final lo que hace es llamar a una función javascript en su evento onClick, esta solución se puede llevar a paneles o cualquier componente html que lance eventos javascript.



Una vez tenemos esto en el layout mediante javascript usamos las librerias javascript creadas para silcam-bonita

Dependencias javascript necesarias:

<script src=*"silcam/vendor/alertify.js/lib/alertify.js"*></script>

<script src=*"silcam/js/silcam-validate.js"*></script>

<script src=*"silcam/js/silcam-utils.js"*></script>

<script src=*"silcam/js/silcam-widget.js"*></script>

<script src=*"silcam/js/silcam-message.js"*></script>

<script src=*"silcam/js/silcam-rest.js"*></script>

Implementamos la llamada al método javascript:

//FUNCION QUE GESTIONA LOS BOTONES DE GUARDAR

**function** consolidarDatosPanelAtomico()

{

id\_tarea = $('#id\_tarea').find('.bonita\_form\_field').val();

url\_motor\_bonita = $('#url\_motor\_bonita').find('.bonita\_form\_field').val();

panel = $('#tabs li.active a').attr('href');

**var** lista\_campos = listaCamposPanel(panel);

//Esta función recibe como parámetro una lista con las variables

//del formulario que queremos actualizar.

//debe coincidir el nombre del campo con el nombre de la variable.

//No es atómica, actualiza las variables de una en una,

//de modo que puede actualizar el resto de variables aunque una falle.

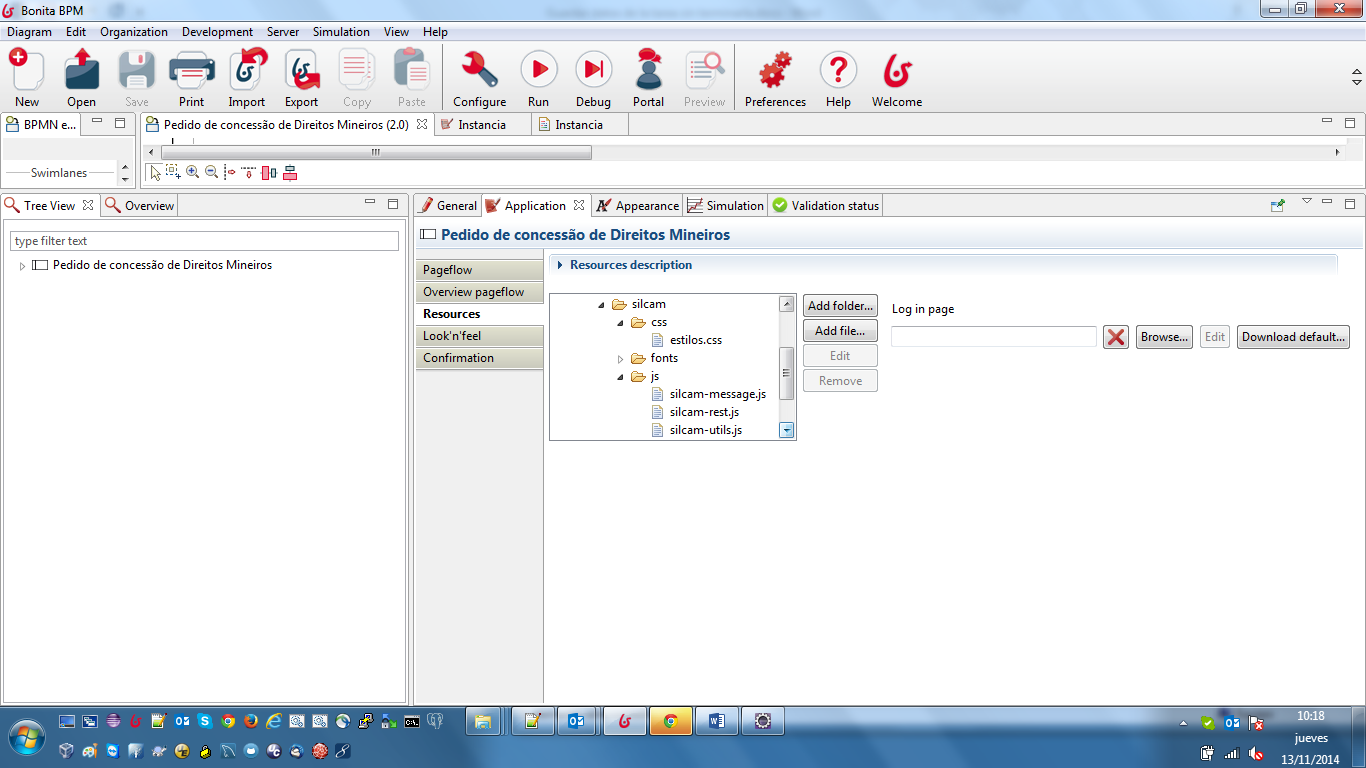
saveListData(url\_motor\_bonita, id\_tarea, lista\_campos);

}

Para que todo funcione **correctamente con este Método, el nombre del campo del formulario y el de la variable de bonita deben coincidir**. Donde usamos panel puede ser el id del DIV que contiene todos los campos que queramos guardar al dar al botón.

Hasta aquí es todo lo que un programador de formularios de bonita debería conocer.

El proceso debe tener incluido en sus recursos, los jars y javascript necesarios.



## ANEXO

También se pueden usar otras funciones javascript ya creadas en los ficheros de utilidades para funcionar pasando 2 listas o un mapa que represente el mapeo de campos de formulario con variables del proceso.

//FUNCION QUE GESTIONA LOS BOTONES DE GUARDAR

**function** consolidarDatosConcretos()

{

id\_proceso = $('#id\_proceso').find('.bonita\_form\_field').val();

url\_motor\_bonita = $('#url\_motor\_bonita').find('.bonita\_form\_field').val();

//Esta función recibe como parámetros la lista de variables que queremos actualizar.

//debe coincidir el nombre del campo con el nombre de la variable.

//No es atómica, actualiza las variables de una en una,

//de modo que puede actualizar el resto de variables aunque una falle.

actualizaVariables(url\_motor\_bonita, id\_proceso, arguments);

}

//FUNCION QUE GESTIONA LOS BOTONES DE GUARDAR

**function** consolidarDatosIndicandoCamposFormulario(lista\_variables, lista\_campos)

{

id\_proceso = $('#id\_proceso').find('.bonita\_form\_field').val();

url\_motor\_bonita = $('#url\_motor\_bonita').find('.bonita\_form\_field').val();

//Esta función recibe como parámetros la lista de variables que queremos actualizar

// y la lista de campos del formulario que contiene el valor correspondiente

//a cada variable de la lista anterior.

//De este modo podemos realizar el mapeo independientemente del nombre

//de cada campo del formulario.

//No es atómica, actualiza las variables de una en una,

//de modo que puede actualizar el resto de variables aunque una falle.

actualizaVariablesIndicandoCampos(url\_motor\_bonita, id\_proceso, lista\_variables, lista\_campos);

}

//FUNCION QUE GESTIONA LOS BOTONES DE GUARDAR

**function** consolidarDatosPanel()

{

id\_proceso = $('#id\_proceso').find('.bonita\_form\_field').val();

url\_motor\_bonita = $('#url\_motor\_bonita').find('.bonita\_form\_field').val();

panel = $('#tabs li.active a').attr('href');

//Esta función recibe como parámetro el id del div que contiene todos los campos

//del formulario que queremos actualizar.

//debe coincidir el nombre del campo con el nombre de la variable.

//No es atómica, actualiza las variables de una en una,

//de modo que puede actualizar el resto de variables aunque una falle.

guardarPanel(url\_motor\_bonita, id\_proceso, panel);

}