

# GUI development rules

## Styles and components

Use of Bootstrap styles and components: [fonts](#), [alerts](#) and [modals](#)

Use of jQuery and Bootstrap plugins: [datatables](#), [columns visibility](#) in datatables, [blockUI](#) and [multiselect](#).

## Files Structure and Common files:

Each web page (jsp, xhtml) has its own JavaScript file (\*.js) that stores its specific functions - the purpose is to have the code organised by page.

[global.js](#) - global file that has all functions that are common to all pages

[default.css](#) - new styles should be added in this file

## Inclusions

[dependenciesInclusions.html](#) - The inclusions which are used in all jsp files should be added to this file.

```
1 <!--Style related inclusions-->
2 <link rel="stylesheet" type="text/css" href="css/bootstrap.css">
3 <link rel="stylesheet" type="text/css" href="css/crb_style2.css">
4 <link rel="stylesheet" type="text/css" href="css/default.css">
5 <link rel="stylesheet" type="text/css" href="css/jquery.dataTables.css">
6 <link rel="stylesheet" type="text/css" href="css/jquery-ui.css">
7 <link rel="stylesheet" type="text/css" href="css/dataTables_jui.css">
8 <link rel="shortcut icon" type="image/x-icon" href="images/favicon.ico">
9 <!--Script related inclusions-->
10 <script type="text/javascript" src="js/jquery-1.9.1.min.js"></script>
11 <script type="text/javascript" src="js/bootstrap.min.js"></script>
12 <script type="text/javascript" src="js/jquery-ui-1.10.2.js"></script>
13 <script type="text/javascript" src="js/jquery.jeditable.mini.js"></script>
14 <script type="text/javascript" src="js/jquery.dataTables.min.js"></script>
15 <script type="text/javascript" src="js/jquery.dataTables.editable.js"></script>
16 <script type="text/javascript" src="js/jquery.validate.min.js"></script>
17 <script type="text/javascript" src="js/jquery.multiselect.js" charset="utf-8"></script>
18 <script type="text/javascript" src="js/global.js"></script>
```

Don't forget to include [dependenciesInclusions.html](#) at the beginning of your jsp files between <head> tags.

```
<%@ include file="include/dependenciesInclusions.html" %>
```

Local inclusions should always be done after this line.

## **Tables Design**

If you have a table for which you want to perform some operation at the row level (delete, update), then add a column (first one) that should enclose the buttons that allow the desired operations.

If you have a table for which you want to add a new record, the button to “add” should be in the top-left side of the table.

In the global.js you have functions that allow you to create server and client-side tables :

```
TableConfigurationsClientSide(divId, data, aoColumnsFunction) ;
```

```
TableConfigurationsServerSide(divId, ajaxSource, ajaxProp, aoColumnsFunction) ;
```

```
createDataTable(tableConfigurations) ;
```

```
createDataTableWithPermissions(tableConfigurations, callbackfunction) ;
```

divId - table unique identifier

data - data that is presented in the table

ajaxSource - ajax url

ajaxProp - json property

aoColumnsFunction - function to render the columns, you must set a sName so the servlet can sort your data.

## **Data Formatting**

The response from the servlet is a Json Object which contains at least :

- The total number of entries in the table
- The number of entries displayed on the page
- The data, a Json Array of Json Objects.

If specific actions have to be performed on the dataTable (creating/editing/removing entries ...) , a boolean object with permissions for the different actions must be set.

You must implement a sorting method in the DAO for server side tables.

## **Modals**

Use of bootstrap modals

If a page require a special modal dialog, you should create it in a specific folder:

*Web Pages/include/<pagename>/<modalname.html>*

There are modals that are recurrently used, they should be saved in the folder:  
*Web Pages/include/utills/<modalname.html>*.

Examples include: confirmation modal and upload modal.

## **Error/Success Messages**

Messages are displayed using functions defined in the global.js.

Success messages resulting of operations (insert, delete, update) should be displayed in the main page.

Error messages resulting of operations (insert, delete, update) should be displayed in the page/modal where the operation is being performed

When a modal is opened, or when you click on a button that performs a certain operation, you should clean the messages that are currently being displayed.

## **Internationalisation**

The different fields displayed on the screen are handled within the javascript.

The documentation is loaded in the sessionStorage of the navigator depending on the user language.

For an easy access of the documentation, doc.js implement different utility functions.

getDoc() function return a JSONObject which contains all the documentation loaded in the sessionStorage. From this, you can access to the different docFields you need to display the labels on the screen.

See the documentation in doc.js for information on all the function you can use.

To display correctly all the label on the screen, you should add a « name » attribute to your HTML tags (corresponding to the label you want to display) and implement a Javascript function in your <pagename>.js file which will display your label using Javascript/jQuery methods.