**Name: ContextDropdown**

**BWG Name:** ContextDropdown.bwg

**Group:** Bizagi Widgets

**Description:** This is a dropdown combo which can be used against any data, it does not require to be persisted or even controlled by an entity, the process to calculate the results are performed in a rule and the dropdown will display whatever the rule return as a JSON. This depends on the rule returning the json in the following format:

“[ { \”id\”: <id>, \”value\”: “<value>” }, …. ]

**Type:** Forms widget

**BAS parameters:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Tab** | **Name** | **Data Type** | **Description** |
| Basic | Display Text | Text | Text to show as display in the label, If not provided this will use the display Attrib based on the xpath selected (Data Source) |
| Basic | Data Source | Rule | This is the rule that should return a JSON string with the content to be displayed in the dropdown. |
| Basic | Visible | Boolean / Boolean Business Rule | Parameter that defines the visibility of the field |
| Basic | Editable | Boolean / Boolean Business Rule | Parameter that defines if the button is enabled or not |
| Basic | Format | Selection | Format configuration of the field |
| Basic | Data Rule | Business Rule | Business rule to be executed on click |

**Example**

**Rule Example:**

var systems = "";

var s = CHelper.GetValueAsCollection( <mMMG.kmMbrSs.xSystemPlans>);

for (var x = 0; x < s.Count; x++) {

var ss = s[x];

if (systems.Length > 0) {

systems += ", ";

}

var system = "{ \"id\": " + ss.getXPath("id") + ", \"value\": \"" + ss.getXPath("kpSystemPlan.sName") + "\" }";

systems += system;

}

systems = "[" + systems + "]";

CHelper.trace("Cross", "systems: " + systems);