VirtualShelf Component Overview

Dataset

This component creates a virtual product shelf using row objects from the **rowArray** dataset. Each row object has the following configurable properties

* **id** {String}: row product id name
* **label** {String}: row product label
* **description** {String}: row product description
* **image** {String}: comma separated list of row product images. The image displayed on the shelf will be the first image from the list.
* **price** {String}: row product price (should include the currency sign)
* **price\_image** {String}: row product price tag image.
* **item\_count**: row product item count. This will dictate how many copies of the image to display on the shelf. Each image copy is placed adjacent to one another.
* **width**: row product image width.
* **height**: row product image height.

For **price** and **price\_image** properties, **price\_image** will take precedence over **price**. This means if a valid **price\_image** is provided it will display the image and hide the price label.

Parameters

The VirtualShelf component supports 4 question types via the **compQuestionType** {String} parameter.

* **Single Choice**: select only 1 item.
* **Multiple** **Choice**: select multiple items.
* **Multiple** **Soft** **Cap**: select up to X items, where X is determined by **compCapValueQtyMax** {Numeric} parameter. Will display an ‘Item Remaining’ counter in the upper left.
* **Multiple** **Hard** **Cap**: select X items, where X is determined by **compCapValueQtyMax** {Numeric} parameter. Will display an ‘Item Remaining’ counter in the upper left.

The **shelfWidth** {Numeric} parameter dictates how many items will ultimately fit on 1 shelf row. Using **shelfItemHorizontalGap** and **shelfItemVerticalGap** {Numeric} parameters you can control the spacing of row items.

You can import a custom shelf image by setting **shelfDispType** {String} parameter to ‘**import**’ and pointing **shelfBckgrndImgImport** {String} parameter to a valid image URL.

If you set **shelfDispType** {String} parameter to ‘**vector**’ you can customize the look of the default shelf using **shelfBckgrndColor** {Color}, **shelfBorderStyle** {String}, **shelfBorderColor** {Color}, and **shelfBorderWidth** {Numeric} parameters. You can also customize the 3D appearance of the shelf using **shelfLeftRightDepthSkew** {Numeric} and **shelfDepthHeight** {Numeric} parameters.

The row product items are automatically horizontally centered on the shelf. Using the **itemViewStackCnt** {Numeric} parameter you can set how many item copies you want stacked behind one another. Depending on the row product position, each row product stack will be slightly angled to give it more a realistic depth. If the product images are rectangular, you can use **itemViewShowBoxShadow** {Boolean} parameter to add a shadow and accentuate the depth.

Each row product can optionally display the following

* Price: use **itemViewDisplayPrice** {Boolean} parameter to toggle price label/image display. The price label/image is placed underneath the product image, adjacent to a quantity dropdown (if one is displayed).
* Quantity dropdown: use **itemViewDisplayQty** {Boolean} parameter to toggle quantity dropdown display. When set true the user must select a quantity greater than 0 to select. The dropdown is placed underneath the product image, adjacent to a price label or image (if one is displayed).
* Image Zoom button: use **itemViewAllowZoom** {Boolean} parameter to toggle zoom feature. Additional image views can be viewed during a zoomed view (assuming they are provided) along with row description text. The zoomed image view size width can be adjusted using the **zoomViewWidth** {Numeric} parameter. The height is automatically set to the browser viewport height (less a little padding). The zoom view will automatically rescale and center on browser resize. A zoom button will only display when the quantity dropdown is not displayed (**itemViewDisplayQty** set to false).
  + If a zoom button is created, it will be placed underneath the product image and price/quantity dropdown combo.
  + If a zoom button is not created, the user can click on the shelf image to activate image zoom.

Using **cartViewDisplay** {Boolean} parameter, you can choose to display a ‘view cart’ button which when clicked will display a cart breakdown. The cart breakdown container width will be match the **shelfWidth** parameter value.

Data record

The VirtualShelf component records 3 types of data – user selection, time (in millisecs) to make selection, and quantity selected. The timer is reset everytime an item is selected or unselected.

Public Methods

Call **bindExitCallback** method and pass it a callback function which fires when the ‘Checkout’ button is selected in the cart breakdown view and when mandatory conditions are met.

Call **getDimenResp** method to return a data object containing row response information. Each selected row returns the following

* Row element id (**id**).
* Quantity selected (**qty**)
* Time, in milliseconds, it took to make the selection (**time\_ms**).