**Product Image Orientation**

# Objective

Customize Search-result-container module to support product image orientation to render in landscape as well as in portrait mode. Currently It renders product image only in portrait mode.

Graphical user interface, website

Description automatically generated

# Prerequisite

* Need to configure attributes values of product in HQ for this feature implementation.

# Components Modified for this feature.

**Product components** need to extent to introduce business logic to segregate rendering logic based on productImageSettings. The same component is used on search-result-container view page to render the product image and in view file also applied logic to add class in li element based on image orientation.

# Modules Extended for this feature.

search-result-container module can be extended to achieve the desired feature. Please refer below command for extending the component.

*yarn msdyn365 add-component-override <theme-name> <component-name>*

This command adds component file in component folder of that theme.

# Steps to consume and test the sample code.

**Option-1:** Follow below steps to consume the attached sample code to implement and test this feature.

1. Unzip the attached code and save it to some location.
2. Copy files and folders under the same folder structure in your repository as it’s there in sample.zip.
3. Execute yarn on the root path of your repository.
4. Execute yarn build.
5. Execute yarn start.
6. Browse the configured product using pagemock using below sample url.

<https://localhost:4000/partnersite/page?mock=kit-demo>

**Note**: check for the right *partnersite* in .env file of your repository and make a change in url accordingly.

**Option-2:** If the search-result-container module is alreadyejected in the repository you want to implement this feature, you need to manually plugin the code from the attached sample files to the respective files. Please follow below steps for the same.

1. **Component extension:** (*product.component.tsx*): Check for this file under

src\themes\ customer-theme \views\components folder to make the necessary changes to this file.

**Note:** if the file does not exist on this path, execute below command to override the component. Change the highlighted text with the associated theme folder.

**CLI**: *yarn msdyn365 add-component-override customer-theme product.*

Once the component is added to the target location, add below code snippet form the sample code files.

Copy highlighted lines and call the same in ProductCard functional component of this file.

A screenshot of a computer

Description automatically generated with medium confidence

Copy highlighted code and add the same in ProductCard functional component of this file.

Text

Description automatically generated

1. **search-result-container.view changes:** Need to modify below files in search-result-container.viewfile for this feature implementation. Copy the highlighted code and call the renderProducts function in SearchResultContainerView functional component.

Text

Description automatically generated

1. **Build the repository**:

Execute yarn build from command line interface on your repository after making above code changes.

1. **Test the code changes:**

Execute yarn start.

Browse the configured product using pagemock using below sample url.

<https://localhost:4000/partnersite/page?mock=kit-demo>

**Note**: check for the right *partnersite* in .env file of your repository and make a change in url accordingly.

1. Based on the image orientation the classNames will be updated and the theme (CSS) should be updated to render the images as Landscape and Portrait.

**Note**: The sample code is tested on below package version:

* Store starter kit: 10.0.17 (i.e., @msdyn365-commerce-modules/starter-pack": "~9.27.17)
* Retail-Proxy: 10.0.17 (i.e., "@msdyn365-commerce/retail-proxy": "~9.27.8")

Additional steps to test the changes in local if attributes value of product are not set in HQ:

1. Hardcode the imageOrientation = ‘Landscape’ or imageOrientation = ‘Portrait’ before return in getImageOrientation function in search-result-container.view.tsx file. (\* please refer below screenshot)

A screenshot of a computer

Description automatically generated

1. Hardcode the imageOrientation = ‘Landscape’ or imageOrientation = ‘Portrait before return in ProductCard function in product.component.tsx file. (\* please refer below screenshot)

Text

Description automatically generated

With the above changes once, we run in local we can see the difference in the rendering of image settings for Landscape and Portrait.

For Landscape 334.663 X 215.288

Graphical user interface, text, website

Description automatically generated

For Portrait 334.663 X 334.663

Graphical user interface, Word

Description automatically generated