ArcGIS Web AppBuilder Configuration Notes

1. ArcGIS Web Appbuilder Themes:

ArcGIS Web AppBuilder Themes Has 4 Major Components

1) Layout: Layout defines the overall structure of the theme. It defines the placeholder and position of the widgets in the app; also the off-panel and on-panel widgets that are part of the theme. A theme can have multiple layouts, which are defined in the client ➜ stempapp ➜ themes ➜ FoldableTheme ➜ layouts folder. To modify any layout, open the config.json file for the specific layout and change the content and widget positions in there. Show in section 2

2) Panel: Panel is a container of theme where widget contents are displayed. Designing of a panel requires background in dojo and JavaScript programming concepts.
3) Widgets: Widgets that are part of theme and the one that are always displayed when that theme is selected.
4) Style: Style is the color schemes that theme makes it available for the user. Overall, colors of different panel, elements of the html page are all part of style. Shown in section 4.

2. Changing position of Off-Panel Widget for a specific theme :

off-panel widget are the one which gets displayed on directly top of the map, and do not require any container or popup box to open. Examples of off-panel widget is scalebar, ZoomIn-ZoomOut Tool, Home Button, Coordinate Location, Search Widget.
In this example we are going to change the location of Search Widget from top left corner to top right corner of **Foldable Theme (Default Layout).**

- To do this go to client ➜ stempapp ➜ themes ➜ FoldableTheme ➜ layouts ➜ default
- Open config.json file.
- In the widgets array, find the object text that say "uri": "widgets/Search/Widget"
  This tells the layout builder to point to a particular widget from the widgets folder
- It has position description just below
  "position": {
       "left": 55,
       "top": 5

> }
> This tells builder to put Search Widget on the top 5 points and left 55 points on the map.
> To change the Position to the top right, change the text to
> "position": {
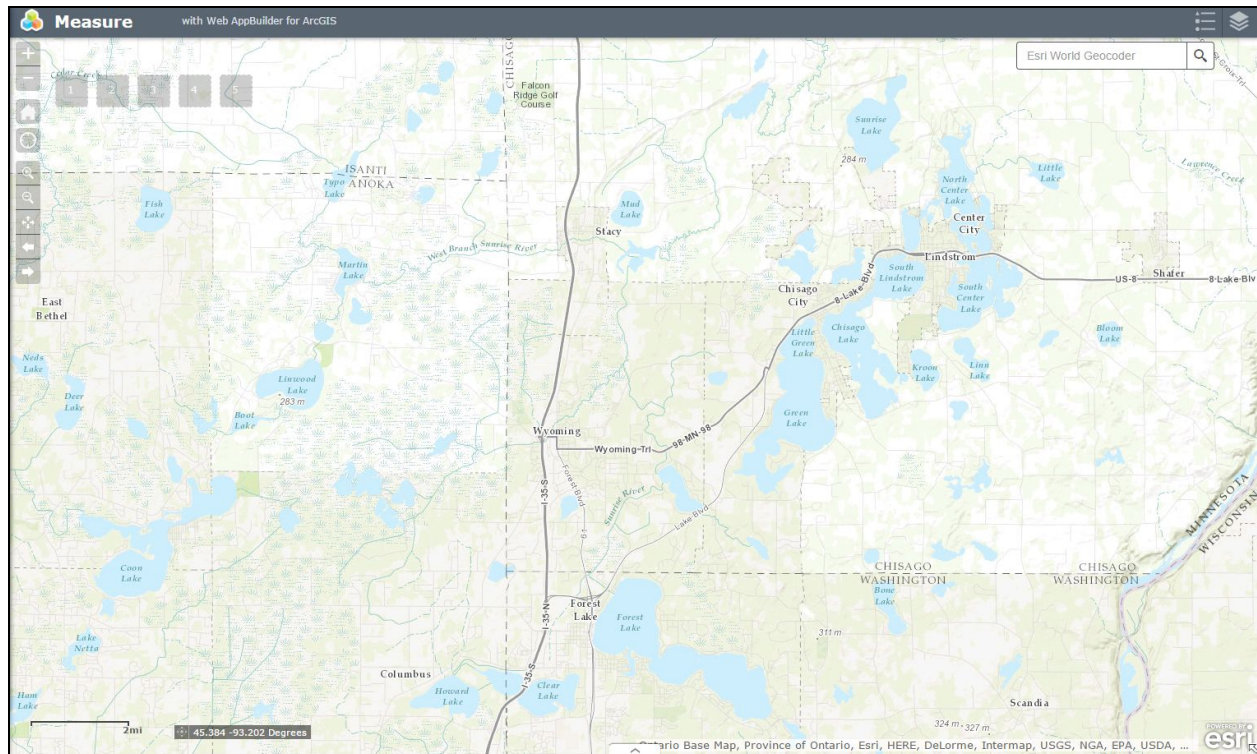> "right": 55,
> "top": 5
> }

Save the file, and restart the ArcGIS App Builder, and go to any App, and selected Foldable Theme, with Default layout.

1. Config.json with default location

2. Change the position from "left":55 to "right":55

3. Search Widget Moved to Top Right.

Following the same process, we change the position of any other off-panel widget.

Note: The above process is applied to Default layout of Foldable theme, to make changes in the other layout for the theme , go to the   FoldableTheme ➜ layouts
and select the layout folder that you want to apply change for, open config.json file , and follow the same process.

3. Adding New Placeholders for the Widget Container :

In the section, we are going to add a new Widget Placeholder in the Foldable Theme, Default Layout.
Default Layout, in the foldable theme has 5 widget placeholders.
 To add new 6th widget placeholder:
   - Go to client ➜ stempapp ➜ themes ➜ FoldableTheme ➜ layouts ➜ default
   - Open the config.json file. Highlighted text below in the config.json file shows one widget placeholder and its position on the screen. In this config.json file, there are 5 placeholder tags listed like that.

```
1   {
2     "widgetOnScreen": {
3       "widgets": [{
4         "uri": "themes/FoldableTheme/widgets/HeaderController/Widget",
5         "position": {
6           "left": 0,
7           "top": 0,
8           "right": 0,
9           "height": 40,
10          "relativeTo": "browser"
11        },
12        "version": "1.4"
13      }, {
14        "uri": "widgets/Scalebar/Widget",
15        "position": {
16          "left": 25,
17          "bottom": 25
18        },
19        "version": "1.4"
20      }, {
21        "uri": "widgets/Search/Widget",
22        "position": {
23          "right": 55,
24          "top": 5
25        },
26        "version": "1.4"
27      }, {
28        "uri": "widgets/Coordinate/Widget",
29        "position": {
30          "left": 200,
31          "bottom": 20
32        },
33        "version": "1.4"
34      }, {
35        "position": {
36          "left": 55,
37          "top": 45
38        }
39      }, {
40        "position": {
41          "left": 105,
42          "top": 45
43        }
44      }, {
45        "position": {
46          "left": 155,
47          "top": 45
48        }
49      }, {
50        "position": {
51          "left": 205,
```
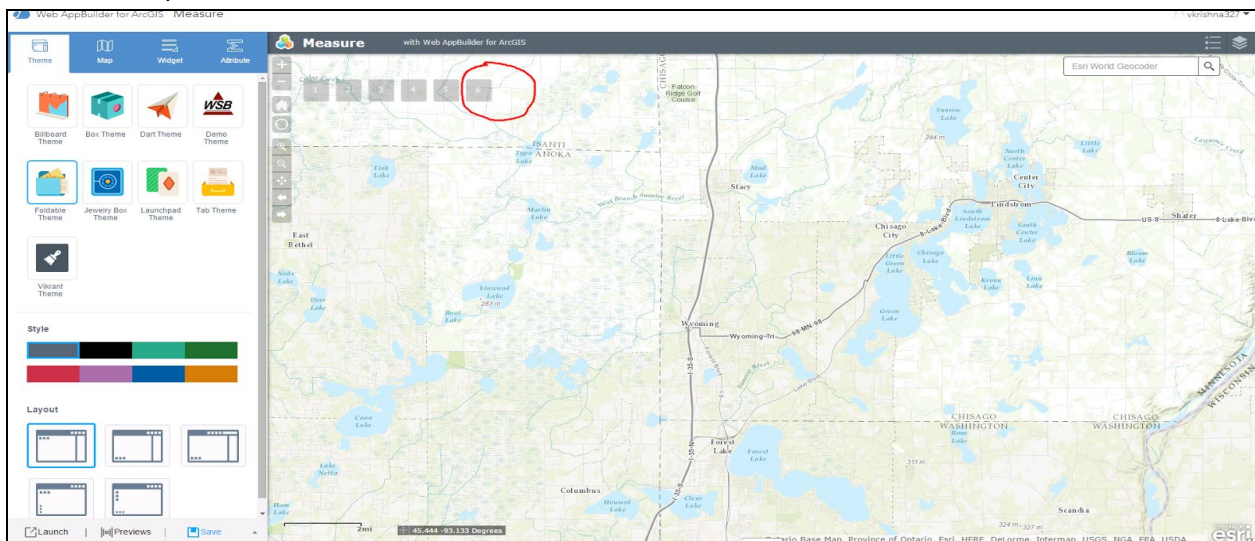
- To add another widget placeholder to the layout, copy and paste the selected text, and change the position of values to define the location where you want it to show.
- Example: Added a new placeholder tag below the last placeholder tag and defined its position to be 305; which is 50 points left of 5th placeholder.


```
      }
    }, {
      "position": {
        "left": 255,
        "top": 45
      }
    },
    {
      "position": {
        "left": 305,
        "top": 45
      }
    }
```

- New placeholder on the theme looks below

In similar way, we can add any number of widget placeholder to any theme layout.

Note : Careful when added placeholder tag, keep the formatting correct or it will cause error.

4. Adding New Color Scheme to Theme Layout:

Although, Most of the theme that are part of web app builder comes with good number of style colors; In some cases user might want to add some custom color schemes. In this section we are going to create custom color style to Foldable theme.

To add new style color to theme:

- Go to the client ➜ stempapp ➜ themes ➜ FoldableTheme
- Open manifest.json file. Look for Style section, defined as json object array.

- Add the color object definition , with color code value to this object array.

```
{
  "name": "MyCustomColor",
  "description": "this is my custom color style",
  "styleColor": "#650010"
}
```

```
"styles": [
    {
        "name": "default",
        "description": "this is default style",
        "styleColor": "#586679"
    },
    {
        "name": "black",
        "description": "this is black style",
        "styleColor": "#000000"
    },
    {
        "name": "green",
        "description": "this is green style",
        "styleColor": "#2aaa8a"
    },
    {
        "name": "cyan",
        "description": "this is cyan style",
        "styleColor": "#267030"
    },
    {
        "name": "red",
        "description": "this is red style",
        "styleColor": "#c93048"
    },
    {
        "name": "purple",
        "description": "this is purple style",
        "styleColor": "#aa6fad"
    },
    {
        "name": "blue",
        "description": "this is blue style",
        "styleColor": "#005fa2"
    },
    {
        "name": "yellow",
        "description": "this is yellow style",
        "styleColor": "#d07d0e"
    },
    {
        "name": "MyCustomColor",
        "description": "this is my custom color style",
        "styleColor": "#650010"
    }
],
```

- In the next step, go to client ➜ stempapp ➜ themes ➜ FoldableTheme ➜ Style .
  Create a folder called **"MyCustomColor"**   (same name as defined in manifest.json).
  Create **style.css**   file inside this folder, and finally paste the content below in this css.

```
.jimu-main-background{
 background-color: #650010;
}
.jimu-widget-header-controller .jimu-drop-menu{
 background-color: #650010;
}
.jimu-header-more-popup {
 background-color: #435063;
}
.jimu-header-more-popup .icon-node {
 background-color: #650010;
}
.jimu-header-more-popup .close {
 background-color: #435063;
}
```

```css
.jimu-header-more-popup .close-inner {
  background-color: #650010;
}
.jimu-on-screen-widget-panel>.jimu-panel-title,
.jimu-foldable-panel>.jimu-panel-title,
.jimu-title-panel>.title{
  color: #fff;
  background-color:#650010;
}
.jimu-widget-header-controller .jimu-drop-menu .menu-item{
  border-top: 1px solid rgba(255, 255, 255, 0.2);
}
.jimu-widget-header-controller .jimu-drop-menu .menu-item:hover{
  background-color: rgba(255, 255, 255, 0.2);
}
.esriPopup .titlePane {
  background-color: #650010;
}
```

Note: color value #650010 in style.css is the color of this specific style scheme. This will change based on your color scheme.