**The Banner and Icons**

***Figure 1 (top):*** *The Banner and Icons at 1366px screen width.*

***Figure 2 (left):*** *The Banner and Icons at 768 px and below.*

This is the code for the Banner and Icons:

<div class="row cells12 **no-margin**">

<div class="cell colspan8 **padding10**" id="**banner**">

<a href="/index.html">

<img width="500px" src="/images/logo/psa-logo.png"></img>

</a>

</div>

<div class="cell colspan4" id="**icon\_con**">

<div style="letter-spacing:2px;">

<p id="**icons**">

<a href=""><span style="font-size:1.7em" class="**fg-hover-darkBlue fg-blue mif-facebook"></**span></a>

<a href=""><span style="font-size:1.7em" class="**fg-hover-darkBlue fg-blue mif-twitter"></**span></a>

<a href=""><span style="font-size:1.7em" class="**fg-hover-darkBlue fg-blue mif-mail**"></span></a>

</p>

</div>

</div>

</div>

**no-margin –** removes margins on the content of the tag where the it was used.

**padding10 –** applies padding of 10px on all sides. You may change the padding size by changing the last two digits (e.g.*padding50*). All padding sizes only works when the value is five (5) or anything divisible by ten from ten (10) to one-hundred (100).

**banner –** only used at 768px screen width and below in the CSS where the banner is centered.

**icon\_con –** used in the CSS to adjust the position of the icons.

**icons –** also used in the CSS to position the icons inside the icon’s container.

**fg-hover-darkBlue –** used to change the color of the icons when hovered. In this case dark blue.

**fg-blue –** used to change the default color of the icons. In this case blue.

**mif-facebook** **–** calls the facebook icon made by Metro UI. **mif-twitter** and **mif-mail** calls twitter and a mail icon. (for more icons go to /docs/font.html)

**Navigation Bar:**

The **navigation bar**’s code is located on the */includes/header.html.*

1. **The F-Menu.**

***Figure 1 (far top)****: Navigation bar at 1366px.*

***Figure 2 (top):*** *Navigation bar at 954px.*

***Figure 3 (left):*** *Navigation bar at 510px.*

The **navigation bar** is created with this sample code:

<ul class="**f-menu bg-darkCobalt**">

<li id="**navList2**">

<a href="#"> <!--Any href value will work-->

<span class="**fg-white**">Statistics</span>

</a>

</li>

</ul>

**f-menu –** the type of menu. I tampered with the **f-menu** in */css/metro.css* so it’s a quite different than the original CSS. (*for types of menu* *refer to Metro UI’s documents located in docs/menu.html*)

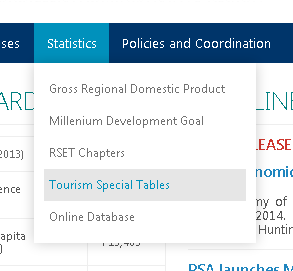
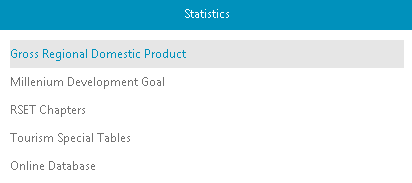
**bg-darkCobalt –** the background color of the **f-menu**. (*for colors in Metro UI’s documents located docs/color.html*)

**navList2 –** the width of each item inside the menu has fixed these are used to adjust the width on the menu like **Statistics** or **Policies and Coordination.** The function is the same as **navList1** to **navList5**. You can adjust the width in */css/metro.css.*

**fg-white –** used to change the color of fonts.

Note: You can change the height of **f-menu** in the CSS under ***.f-menu.*** When changed it will not center the item’s text vertically so you’ll also have to edit the padding in the CSS under ***.f-menu > li > a*** where the first value in padding is for the top and bottom while the second value is for the left and right.

1. **The D-Menu**.



***Figure 4 (left):*** *Dropdown menu at 1366px.*

***Figure 5 (top):*** *Dropdown menu at 510px.*

This **dropdown menu** is created with this code:

<li id="navList2">

<a href="#"> <!--Any href value will work-->

<span class="fg-white">Statistics</span>

</a>

<ul class="**d-menu**" data-role="**dropdown**" id="**nav2**">

<div class="padding10">

<li>

<a href="/statistics/rset/rset.html">RSET Chapters</a>

</li>

</div>

</ul>

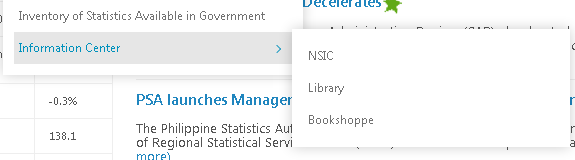
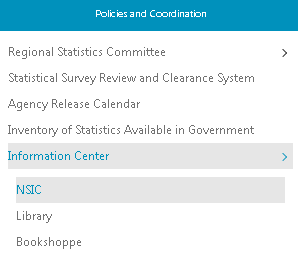
</li>

**d-menu –** used to create the dropdown menu inside the **f-menu.**

**dropdown –** always used for as the value of **d-menu’s** data-role attribute**.**

**nav2 –** used to adjust the size of the dropdown menu. When the screen width reaches 510px and below the dropdown menu’s size becomes 100%.

1. **Nested D-Menu**



***Figures 6 & 7:*** *Dropdown in 1366px (left), dropdown in 510px (right).*

This **nested dropdown** **menu** is created with this code:

<ul class="d-menu" data-role="dropdown" id="nav2">

<div class="padding10">

<li>

<a href="#" class="**dropdown-toggle**">Information Center</a>

<ul class="d-menu" data-role="dropdown" id="**dropnest**">

<div class="**padding10**">

<li><a href="/policies\_and\_coordination/nsic/nsic.html">NSIC</a></li>

<li><a href="#">Library</a></li>

<li><a href="#">Bookshoppe</a></li>

</div>

</ul>

</li>

</div>

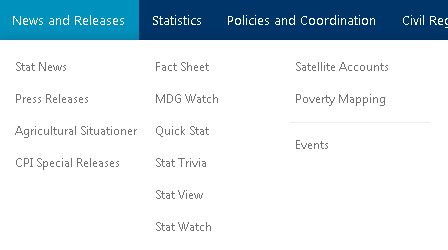
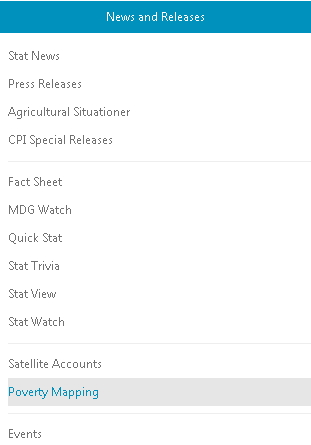
</ul>

**dropdown-toggle –** this class is used to put an arrow facing right ( > ) on an item inside the **d-menu** to make it look like another dropdown menu**.**

**dropnest –** I added this id to fix the position of the nested dropdown menu at *510*px width.

***Note:*** The **nested** **d-menu** in this example doesn’t have the **nav2** id.

1. **News and Releases Dropdown Menu**



***Figure 8 (top):*** *The News and Releases dropdown in 1366px.*

***Figure 9 (left):*** *The News and Releases dropdown in 510px.*

The news and releases’ dropdown menu has two different codes. ***Figure 8*** is separated by columns with the <div> tag while ***figure 9*** is divided by <hr> tags. When the browser window is in 1366px in width, the dropdown menu that will showup is the one in ***figure 8*** unless its width is reduced to 510px where the dropdown will be “replaced” by the other code, the dropdown shown in ***figure 9*** and vice versa.

The sample code like in ***figure 8***:

<div class="row cells3 padding10" id="**navi1a**">

<div class="cell">

<li><a href="/news\_and\_releases/statnews.html">Stat News</a></li>

<li><a href="/news\_and\_releases/pressreleases.html">Press Releases</a></li>

</div>

<div class="cell">

<li><a href="/news\_and\_releases/mdg-watch.html">MDG Watch</a></li>

<li><a href="/news\_and\_releases/statwatch.html">Stat Watch</a></li>

</div>

<div class="cell">

<li><a href="/news\_and\_releases/sat\_acc/sa\_index.html">Satellite Accounts</a></li>

<li><a href="/news\_and\_releases/pov\_map/pm\_index.html">Poverty Mapping</a></li>

<hr class="**thin bg-grayLighter**">

<li><a href="/events/events-index.html">Events</a></li>

</div>

</div>

The sample code in like in ***figure 9***:

<div class="padding10" id="**navi1b**">

<li><a href="/news\_and\_releases/statnews.html">Stat News</a></li>

<li><a href="/news\_and\_releases/pressreleases.html">Press Releases</a></li>

<li><a href="/news\_and\_releases/cpi\_sr/cpisr\_index.html">CPI Special Releases</a></li>

<hr class="**thin bg-grayLighter**">

<li><a href="/news\_and\_releases/factSheet.html">Fact Sheet</a></li>

<li><a href="/news\_and\_releases/stattrivia.html">Stat Trivia</a></li>

<li><a href="/news\_and\_releases/statview.html">Stat View</a></li>

<li><a href="/news\_and\_releases/statwatch.html">Stat Watch</a></li>

<hr class="**thin bg-grayLighter**">

<li><a href="/news\_and\_releases/sat\_acc/sa\_index.html">Satellite Accounts</a></li>

<li><a href="/news\_and\_releases/pov\_map/pm\_index.html">Poverty Mapping</a></li>

<hr class="**thin bg-grayLighter**">

<li><a href="/events/events-index.html">Events</a></li>

</div>

**navi1a & navi1b –** I used these ids to change their display values. These are what I used to control when the other code will show up and hide.

**thin –** reduces the height of <hr>.

**bg-grayLighter –** changes the color of <hr>.

The CSS code for showing and hiding:

/\*The default CSS that works once the browser window’s width reaches 651px\*/

#navi1a {

display: block; /\*Shows the contents of navi1a\*/

}

#navi1b {

display: none; /\*Hides the contents of navi2b\*/

}

/\*The CSS inside this will trigger if and only if the width reaches 510px below\*/

@media only screen and (max-width: 510px) {

#navi1a {

display: none; /\*Hides the contents of navi1a\*/

}

#navi1b {

display: block; /\*Shows the contents of navi2b\*/

}

}

1. **The Search Bar**



***Figure 10 (top):*** *The default search bar.*

***Figure 11 (left):*** *Search button when screen width is between 954px and 511px. Opens search bar when clicked.*

***Figure 12 (right):*** *The search bar when the magnifying glass icon is clicked.The cross icon is used to hide the search bar.*

The search bar’s responsiveness works almost just like the dropdown menu of news and releases that also has showing and hiding. I also used jQuery for the animation of fade in and fade out in ***Figures 11 and 12.***

The code on ***Figure 10***:

<li id="**navList6**">

<form>

<div class="**input-control text** **place-right**" style="margin-right: 10px;height:20px ;">

<input style="height:65%; border-radius: 10px;" type="text" placeholder="Search...">

<button class="**button**" style="height:65%; border-radius: 25px;">

<span class="**mif-search mif-1x**"></span>

</button>

</div>

</form>

</li>

**navList6** – used to adjust the size and position of the search bar in the CSS code. It is also used to show and hide the contents of its tag.

**input-control text** – used to create a textfield for texts.

**place-right** – places the contents on the right. Works the same as “float:right”

**button** – creates a button

**mif-search** – calls the search icon of Metro UI. (for more icons go to /docs/font.html)

**mif-1x** – resizes the icon called.

The code on ***Figures 11 & 12:***

<li id="**navList6b**">

<a id="**show**" href="#">

<span class="fg-white mif-search **mif-lg**"></span>

</a>

<a id="**hide**" style="display:none;z-index:600" href="#">

<span class="fg-white mif-cross mif-lg"></span>

</a>

<div id="**popup**" style="display:none;z-index:500;position:absolute;right:45px;width:300px">

<form>

<div class="input-control text place-right" style="margin-right: 10px;height:20px;">

<input style="height:65%; border-radius: 10px;" type="text" placeholder="Search...">

<button class="button " style="height:65%; border-radius: 10px;">

<span class="mif-search mif-1x"></span>

</button>

</div>

</form>

</div>

</li>

**navList6b** – used in the CSS code to resize and adjust the position of the search bar.

**show/hide** – used in the jQuery code to fade in and fade out the search bar.

**mif-lg** – also used to adjust the size of the icon.

**popup** - used in the jQuery code to fade in and fade out the search bar.

**jQuery** used: (*see at /includes/header.html*)

</script>

$(document).ready(function(){

$("#**show**").click(function(){ /\* Upon clicking the tag with the id **show**: \*/

$("#**gray**").fadeIn(); /\* A transparent gray bar will cover the menu. \*/

$("#**popup**").fadeIn(); /\* The search bar with the tag popup will appear. \*/

$("#**hide**").fadeIn(0); /\* The <a> with the id **hide** will replace the <a> \*/

$("#**show**").fadeOut(0); /\* with the id **show** since it will fade out**.** \*/

}); /\* the 0 inside the parenthesis disables the slow animation \*/

$("#**hide**").click(function(){ /\* The reverse version of the one above. \*/

$("#**gray**").fadeOut();

$("#**popup**").fadeOut();

$("#**hide**").fadeOut(0);

$("#**show**").fadeIn(0);

});

});

</script>

Note: You need to enter the code below first on the page’s <head> in order to use jQuery.

<script src="/js/jquery-2.1.3.min.js"></script>

**gray -** it has the same shape and position of the navigation bar but it is blank and a little transparent. This helps disable the other links in the menu so they can’t be accessible while the search bar is showing.

**half-opacity** – gives the contents half transparency.

<div id="**gray**" class="row **half-opacity**">

<ul class="f-menu bg-gray">

</ul>

</div>

1. **Google Custom Search Engine**

The code for Search Bar after Google Custom Search Engine have been implemented:

<form id="searchbox\_015504511590388255650:zjupxrz0wp8" action="http://www.google.com/cse">

<div class="input-control text place-right" style="margin-right: 10px;height:20px ;">

<input value="015504511590388255650:zjupxrz0wp8" name="cx" type="hidden"/>

<input value="FORID:0" name="cof" type="hidden">

<input id="q" name="q" style="height:65%; border-radius: 10px;" placeholder="Search..." type="text"/>

<button class="button" style="height:65%; border-radius: 25px;"><span class="mif-search mif-1x"></span></button>

</div>

</form>

This javascript code is very important when using a Google CSE:

<script>

$(document).ready(function() {

var cx = '011032539808336368714:zrjx3xmtobu';

var gcse = document.createElement('script'); gcse.type = 'text/javascript'; gcse.async = true;

gcse.src = (document.location.protocol == 'https:' ? 'https:' : 'http:') + '//www.google.com/cse/cse.js?cx=' + cx;

var s = document.getElementsByTagName('script')[0]; s.parentNode.insertBefore(gcse, s);

})();

</script>

searchbox\_015504511590388255650:zjupxrz0wp8 – this is used to the as the id of the **<form>** to implement the search engine.

<http://www.google.com/cse> - the value of the attribute **action** of the **<form>** should lead to here in order for the google to host the search results.

015504511590388255650:zjupxrz0wp8 – this code is the very important part for the google search engine to work. Its a unique code to filter what should show on the results.

Note: The JavaScript code can be obtained when you created the custom search engine at <http://www.google.com/cse>.

As for the HTML code. I just found it through searching.