Journal:

For the purpose of exploration 1, I have dived down into a CSS framework that Is based on the Material Design concept adopted by the current Google’s mobile operating system, Android. Material design boasts a “flat” design across the board and provides an intuitive user interface that is also responsive to the user’s interaction.

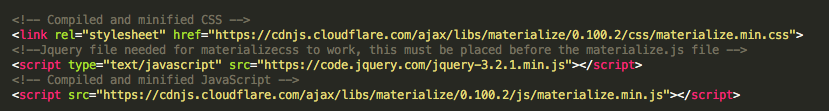
The framework I’ll be working on will be the MaterializeCSS (<http://www.materializecss.com>). MaterializeCSS can be installed and implemented on webpages either through NPM or including a script tag for the CDN link provided by MaterializeCSS. This framework also requires the heavy use of Javascript and Jquery for the animation to run smoothly. Installation step could be found on the following link, <http://materializecss.com/getting-started.html>

In this exploration, I have chosen to explore how by using MaterializeCSS to create a responsive and intuitive UI for the user with its forms and preloader element found in their framework.

The following will be a tutorial on implementing both forms and preloader into your website: \

Getting Started:

1. To include MaterializeCSS, I would recommend using the CDN to obtain the needed CSS and Js file from their server to improve realiability and caching
2. Include the following code into your head tag of the webpage:



1. The code above needs to be written in said order, as the MaterializeCSS javascript is dependent on Jquery to run.

Forms:

1. To include a form into your website, you will be required to introduce a div with the class “input-field” as so:



1. Create as many field you require into your html code to gather the needed information for your web app:



1. If you would like to validate the data provided by the user, you could use a “validate” class as so:



1. Finally, if you would like to disable a specific field for certain condition, MaterializeCss could disable the field by adding “disabled” in the tag of an input:



Preloader:

1. To include a preloader animation, first you would have to create a div container to contain the item you would want to hide with the animation.
2. Then you would then include the following code inside the container:



1. As shown at the above codes, the div tag with the ID “preloader” contained a class called active. This is the way to turn on and off the preloader animation.
2. You could use a javascript function to simply remove the active class from the div to prevent it from executing anymore:



I did however encountered issues while testing and exploring said elements when the user interface didn’t execute as indicated. After reading through the Getting-Started page of the documentation, I have found out that the MaterializeCSS javascript file needs to be declared after declaring the jQuery javascript library. There is a slight dependency issue with the javascript file as the framework is still in its beta phase.

There is still a lot of elements found in this framework that could create a uniform UI for the user while maintaining the industry standard of clean and functional design that can be incoperated with the Android operating system side-by-side.

Examples of my work could be found here:

<http://cs4830.weixianlow.me/exploration1/forms.html>

<http://cs4830.weixianlow.me/exploration1/preloader.html>

Source code could be found here:

<https://github.com/weixianlow/CS4830_Exploration1_Fall2017>