A6 Forward Engineering User Interface Descriptions

For this forward engineering, we decided to use bootstrap to help us create a user interface. Ultimately what we are going for was a simplistic and easily navigable interface that allows the user to create an account or log in to their account, to view different modules and materials that the site has to offer and upload new modules or materials to the site. From our reverse engineering analysis, we created the following pages to accomplish this goal.

1. Homepage – index.html
   1. With the index page, we just wanted to introduce the users to watch the site has to offer and essentially give them a background on what we are trying to achieve. There is also a small contact form if they have any other further questions at the bottom of the homepage.
2. Sign in Page – signin.html
   1. With this sign-in page, we were hoping to create a simplistic design that only asked to use us for an email and password. We didn’t see the need to create other variations of verification such as Google, GitHub, etc. This is because we wanted to use our system to validate users.
3. Registration Page – signup.html
   1. Again, we only wanted to take in information that was pertinent to the system and the user. Therefore, we asked for the username email password. We decided to keep it simple with this regard so that there wasn’t any unnecessary information in our database is.
4. Account Page - account.html
   1. Once the user is either logged in or has not registered, they are taking me to this account preferences page. In the upper right-hand corner, you may have noticed that the sign in, sign up options have changed into a logout option. This is because if we had a back and we would verify that the user is logged in and are presenting the correct options too. However, with this page it is quite simple we want to use her to have the ability to change any information as they deem necessary.
5. My Modules - myModules.html
   1. The Moe modules page presents to the user a list of modules where they are an author. This allows them quick and easy access to either edit or deletes these pages. Additionally, we wanted to keep the design simple and so we have links that allow the user to quickly navigate to that edit and delete option.
6. Search Modules
   1. The search option doesn’t have a page of its own rather what would happen is if we had a back and once the user entered something into the main search bar, this parameter would be taken in by the backend and the user would be redirected to the filtered version of the browser page.
7. Add Module Page - addModule.html
   1. The ad modules page gives the user the option of creating a brand-new module. We are taking information that we deem necessary and store them in our database. We don’t have a back and set up which means that once the user clicks to create or submit a module, we use JavaScript to display a simple message that says it has been saved. Otherwise, if they are working on a module and decide not to go through with it they also have that delete option.
8. Edit Module Page
   1. If we were developed this with a second, the edit module page would be an instance of the ad modules page. This is because we would essentially reload the ad modules page however we would pass in the specific information to that module and give the user the ability to edit the content of the module.
9. View Module Page - browse.html
   1. Is the user does not have a specific module that they are looking for or have a module to upload, they can easily browse our view modules page which allows him to see all the modules currently loaded on our database. They have a variety of filtering options from author to topic.