**PLP - USER INTERFACE DESIGN AND SPECIFICATIONS**

* The PLP Web application landing page is the Login Page. The login page will consist of options to login through:
* Google
* Facebook
* LinkedIn
* Email Id/ Password
* It will also have an option for the user to sign up.
* The signup form will consist of fields like:
* Email-Id (mandatory)
* Set Password (mandatory)
* Confirm Password (mandatory)
* Full Name (mandatory)
* Date of Birth (not mandatory)
* Gender
* Contact Number(mandatory)
* Alternate no.
* College/ Organization (mandatory)
* Profile Photo
* Terms and policies pop-up
* This page will also have a ‘Sign-In’ link which will redirect the user to the sign in page
* Once the data is stored in the database, one more functionality of “Edit Profile” will be added
* It will consist of a few fields that the user will not be able to change (like email ID). This functionality is basically for those uses who sign up through social sites. It will help capture their information.

Form validations:

* Mandatory field validations
* Email ID validations (client-side and server-side)
* Password Strength validation
* Confirm Password validation
* Contact number validation (number of digits/ numeric)

Note: These are just a few fields, more can be added as the requirements build. Most of the data gathering can be used for analytics purpose.

JavaScript and CSS:

* The JavaScript and CSS will be stored in separate files (will be an external script file and stylesheet) to speed up the browser performance
* The JavaScripts will be rendered at the end after the HTML page will be loaded.
* The CSS will be rendered before the page is loaded

Bootstrap:

* Using Bootstrap modals to show up the error messages
* If there are many error messages to be shown, we can show that using an error div so that the users get to see all the messages at a glance
* Making the UI responsive so that it is rendered on all the browsers

AJAX calls:

* AJAX POST calls to push data to the DAO and ultimately to the database
* AJAX GET calls to pull data and display it on the UI