

Task Documentation

1. Task Requirements

- a. Sign up.
 - i. Create signup form.
- b. Sign in.
 - i. Create sign in form.
- c. Integration with GitHub Apis.
 - i. A page to
 - 1. List all users.
 - 2. List users with pagination.
 - 3. List users based on search values.
 - ii. A page to
 - 1. Display user details.
 - 2. Display user repos with pagination
 - 3. Logged in user add GitHub users to his favorites.
 - iii. A page to
 - 1. List favorites users.
 - 2. Remove one or many users from favorites users.
 - 3. Drag and drop favorites users.
 - iv. A page to
 - 1. Display logged in user data.
 - 2. Update his data.

2. Task Solution.

a. Tools and libraries

- i. HTML
- ii. CSS
- iii. Bootstrap
- iv. Font awesome
- v. Angular
- vi. Angular martial
- vii. Firebase database
- viii. Angular flash-messages.

b. Development

i. Guards.

1. I create guards to prevent unauthorized peoples to access any page until they login to their account.
2. Before you register try to access any page for example (users, profile, favorites etc..) it will redirect you to login page.

ii. Register.

1. A form for registration (username, email, password, gender) all of them are required and (image, date of birth) are optional.
2. By default, REGISTER Button is disabled until you fill required data.
3. Once you registered a flash message display the status of registration if success it redirects you to dashboard page (Users page) if not it displays error message with the reason.
4. Once you register it create new user in firebase user with email and password and other user data stored in firebase database in user's collection.

iii. Login

1. A form for login (email, password).
2. By default, LOGIN Button is disabled until you fill required data.
3. Once you login a flash message display the status of login if success it redirects you to dashboard page (Users page) if not it displays error message with the reason.

iv. Navbar

1. Navbar contains company name, login and registration links.
2. Once you logged in navbar replace login and registration with logout and dashboard.
3. Dashboard redirects you to user's page.

v. Users page

1. Page to list all users from this Api (<http://localhost:4200/users>).
2. Display some of user data.
3. Once click user login it redirects you to this user profile.
4. In the same page you can display users with pagination base in this link (http://localhost:4200/users?per_page=5&since=30).
5. You can search for specific user through (http://localhost:4200/users?q=ivey&per_page=1&page=2) and also you can control the number of users to display through pagination.
6. User icon and Star icon to navigate to your profile and to your favorite's users.

vi. User details page

1. A page to display user profile and display his data (name, location, company name, blog, GitHub profile etc..) based on this api (<http://localhost:4200/users/defunkt>).
2. in the top right there is Star icon to add this user to your favorites.
3. Display user repos based on api (http://localhost:4200/users/defunkt/repos?per_page=5&page=2) with pagination.

vii. Favorites users page

1. You can go to favorites page through Star icon in home page or through this link (<http://localhost:4200/favorites>).
2. Display all your favorites users (user image, id, login, GitHub, and delete button).
3. You can remove one or more user from favorites users list.

viii. User Profile page

1. You can go to profile page through USER icon in home page or through this link (<http://localhost:4200/profile>).
2. A page to display logged in user profile and display his data (username, email, password, gender, image, date of birth).
3. You can edit your data through popup modal and save user data.