Boston University ECE

EC463 Senior Design

First Prototype Testing Plan

ConnectBU

By Team 3

Team Members

Nadim El Helou nadimh@bu.edu
Hussain Albayat hussainb@bu.edu
Yousuf Baker ybaker@bu.edu
Benjamin Chan chanben@bu.edu
Damani Philip djphilip@bu.edu

Required Materials

Software:

- Front-end development
    - React.js
    - Material UI
    - Redux
    - Google Single Sign On
- Database
    - SQL
    - Amazon Web Services Relational Database Service
- Back-end development
    - Python
    - Flask

Setup

For the prototype testing, we will be screen sharing the demonstration via Zoom. We will be showing all the web application's functionalities that have been implemented so far, including but not limited to navigation across pages, Google Single Sign On, Redux state management, etc.

Pre-testing Setup Procedure

If you wish to test the prototype on your local machine, complete the following setup:

1. Go to ConnectBU's GitHub repository and clone the repository.
2. Install or update npm if you haven't already.
3. In the cloned repository, navigate to the frontend folder and install the project's node dependencies by running the following:
   ```
   $ npm install
   ```
4. Run the following command to run the web application
   ```
   $ npm run start
   ```
5. In your browser, navigate to "http://localhost:3000" to view the web app.

Testing Procedure

During this demonstration, we will be navigating through the pages of the web application.

1. Navigate through the different web pages through the buttons
2. Sign in with Google
3. Create a new account
4. Navigate pages while logged in
5. Resize browser window

Measurable Criteria

The criteria for successful running and output is as follows:

1. We are able to navigate through the different pages of the web application seamlessly and with ease.
2. The Google SSO signs the user on successfully.
3. Successfully add an entry in the database when a new account is created
4. The user remains “logged in” as they navigate through the different pages of the web application
5. Web pages retain a clean and organized layout when resizing the browser window
6. Web pages look aesthetically good, clean, and organized