Test Plans - Milestone 5 - 102-2 Sai, Jeff, Andrew, Colin

Features that we want to test:

- Login/Authentication
- Uploading Schedule
- Adding friends
- Viewing friends' schedules

- **Login and Authentication**

To test login and authentication, we need to test to make sure a new user is able to sign up with an email/username and a valid password. Once they sign in, their information needs to be stored in the backend database to ensure that they can sign back in using the same login information when they come back next time. Their username and password needs to matched when they login the next time as well. Here are the test cases listed below:

- New user needs to be created
  - This same user should be able to logout and log back in successfully
- If no email/username is found in the database, the user will be notified that no user was found with that specific username or email so they need to enter with a different username/email
- The username and password for a certain user must match when attempting to login, otherwise the user will be notified that the password was incorrect
- All authentication criteria must be met when creating a new user
  - If username is already taken, the user will be prompted to use a different username
  - An invalid username will prompt the user to sign up with a different username
  - An invalid password will fail authentication and prompt user to create a different password
  - A valid password needs to have a special character and a number and at least be 8 characters long

- **Upload Schedule**

In order to test the upload schedule feature, we need to ensure that a user is able to upload their schedule and have whatever items are on the uploaded schedule populate their calendar. Once the schedule has been successfully uploaded, the data needs to be written to the backend database so that the information can populate the calendar and weekly views of their schedule. We need to asses the following test cases:

- A schedule needs to be added to the user's profile.

- The schedule should be in the proper file format, and contain valid start and end times for each event included.
- The tentative plan for valid file formats to upload are .pdf and possibly .png - however this is subject to change based on the difficulty of getting the application to read the schedule from an image.
- The criteria for valid start and end times are: the end time/date must be after the start time, and they shouldn’t overlap with other items on the uploaded schedule, or the events currently in the application database for that user.
- If events do overlap with their timing, there should be some visual display or warning message to the user.
- In order to test that schedules are uploaded, we will need to upload several valid schedules to ensure the data is migrated correctly to the database, then the user view. We will also need to attempt to upload invalid schedules to make sure that the application properly catches incorrect files/invalid event times or dates.

- The schedule needs to populate the calendar and/or weekly views for that user.
  - Additionally, the events on the uploaded schedule will need to populate the users schedule when their friends view their schedule.
  - In order to test this feature, we will therefore need to test from the user profile that is uploading the feature, as well as one of the users on their friend list.

- **Adding Friends**

To test adding friends, we need to make sure that we can find users based on their user id. This means that the webpage for adding friends needs to accurately communicate with the database to update both party’s information to include the other as a friend. The user should be able to input information into the search bar, the page should confirm the user id, and then requests should be sent to the database to update information. The database should then update and the both parties should see each other as friends.

- A friend is successfully added
  - Before the request is sent, a message should be prompted to the user confirming the userid inputted and display the name of the person being added
  - A message should be prompted declaring the successful addition
  - Both party’s should see each other in their own friends list
- The user id was not valid
  - The webpage should prompt the user for a different user id, since the one they inputted did not match any currently stored in the database

Viewing Friends' Schedules

Viewing friend's schedules will start from the logged in user's friends page. They should be able to click on the name of one of their friends from the list and be brought to a page with that friend's schedule as well as a second calendar view of the times the user and their friend are both free.

- If the friend chosen from the friends list has uploaded a schedule, the user should be brought to a new page which should have 2 schedules on it: 1 to show the other user's schedule, and 1 to show the times they are both free
    - The correct friend's schedule should be shown
    - The correct free times should be shown in the second schedule
    - The page header should say "<correct friend's name>'s Schedule>"
    - The page should have a back button that successfully brings the user back to the friends list page and allow them to choose another friend.
- If the friend has not yet uploaded a schedule, a modal should pop up saying that friend has not yet added a schedule, and the user should be brought back to their friends list.