Bi-Weekly Report (4/14/21~4/27/21)

Accomplishments

- Configured backend environment so that user data can be fetched via API
- Implemented optional question with a accordion package, which it will be optional for user to fill out
- Research on react native life cycle concept as well as fetching API calls with javascript.
- Scholarship and College Survey check for user's previous submissions
- Scholarship and College Surveys are now editable post submission with previous results visible
- Rough Lazy Loader for Scholarship Table to have limited view at a time
- College Major question research and spreadsheet of answers per major filled out
- Backend update user profile methods.
- Scraped all college data from niche.com.
- Sorting the scholarship listing by title, amount, deadline, and similarity score.
- Tested the device unique ID (UUID) with secure store API on the frontend.
- Modified the UI effects on various screens and fixed the frontend header titles not correctly showing.
- Implemented Error page for sign up and forgot password
- Researched Database encryption

Upcoming Goal

- Implement recent user view history for each scholarship and college detail.
- View history feature where users can check all the scholarship and college details that they have viewed in chronological order.

- Designed a major questionnaire User Experience (UI) and range slider for some of the major recommendation questions.
- Improve Lazy Loader and eliminate related bugs
- Incorporate pop up modal for long press initiated bookmarking
- Add Canada and Canadian Provinces to Multi-selector for regions
- College and Major Recommendations
- Major Information stored into mongodb
- Integrate the frontend with backend authorization system
- Bookmark, history (recent viewed), and survey APIs for the backend
- Test the memory usage on the frontend to detect any memory leaks.
- Attempt to implement a database encryption

Issue & Barriers

- Since lack of knowledge and inexperience on backend and API fetching, it was challenging to post and fetch data via API.
- An unexpected error had occurred , which some icons had not been displayed, amount of time had consumed to solve the errors.
- One of the backend dependencies is not supported on Windows, much trial and error and research for the windows user in attempting to set up the backend locally before this was discovered.
- When sorting the amount (scholarship recommendation), JavaScript built-in sorting function automatically converts the item into Unicode encoding. As a result, it was sorted correctly in Unicode encoding format, but it is incorrect for our application. Therefore, a merge sort algorithm was utilized.
- There might be an issue of memory leak on the frontend when utilizing the navigation component. However, a possible solution is here. It needs to be tested. The same situation happened to other developers as well, here.

- The sign up email does not work when the user uses NYIT emails. Therefore, a personal email address is strongly recommended.