

Dominick Founds

SWE 6623

October 30, 2022

Assignment 4: Development

## Report

### Overview:

Contained within the compressed folder titled “LabRats” is a Node.JS and React application split into with two child folders titled “backend” and “client”. The root folder contains .env files, package.json files, gitignore, and a README that displays the point of the web application, as well as some demo keys for logging in and testing the app. Please note that this app contains dependencies that might need to be redownloaded. By typing “npm i” in the terminal under the root directory, you will redownload all dependencies for the backend folder. Moreover, cd into the client folder and run “npm i” as well to redownload all dependencies for the frontend client.

### User Class:

The user class is developed as according to the UML diagram from assignment 3. Each user has the correct properties, and role based authorizations are handled in the business logic and frontend. You can find the database model for this in backend/models/userModel.js .

### Lab Class:

The lab class is developed fully and is fully functional. A user can create or join a lab by going to user settings, selecting the lab tab, and finally entering in a lab ID and password. Currently, it is beyond my technical abilities to implement a system where a lab creator can “invite” another user to join the lab. So for the sake of time I implemented this similar to how a user will “log in” but with a lab. Theoretically, it is up for the lab creator to somehow send the labID and lab password through email or something similar. There is currently no way to take care of this within the app. The lab model for the database schema is fully developed. You can find this is backend/models/labModel.js .

### Project Class:

The project class is not yet fully developed, and is prototyped using a “testData.js” file in client/src/components/projects/ directory. Moreover, the model for project data is about 95% percent developed. There are still some changes to be made in order to correctly query information and post/get/delete/edit content. The db model for projects is located in backend/models/projectModel.js .

### Task Class:

The task class is not yet fully developed, and is prototyped using the same “testData.js” file located in client/src/components/projects/. The task model for the database scheme is located is backend/models/taskModel.js .

**Registering:**

Currently, registering is fully functional.

**Logging In:**

Currently, logging in is fully functional.

**Signing Out:**

Currently, signing out is fully functional.

**Adding a lab:**

Currently, adding a lab should be fully functional. When I conduct testing on this module I will find out for sure.

**Adding a Project:**

Currently, adding a project is not functional as there are errors in post project data to the lab object which will contain a list of reference ID's for projects.

**Adding a Task:**

Currently, adding a task is not yet functional as there is no code implemented yet. I decided to take care of the UI for this module but there is nothing set to fire data to the backend.

**Displaying Labs:**

Currently, displaying the right labs from the database is functional.

**Displaying Projects:**

Currently, displaying the projects from the database is not functional, only data from 'testData.js' is being displayed.

**Displaying Tasks:**

Currently, displaying the tasks from the database is not functional, only data from 'testData.js' is being displayed.

**Current Functionality:**

In total, the current functionality of this app is that a user can log in, and if they are a Principle Investigator, they can create a lab. They will then be shown a list of their labs with corresponding ID's and Passwords to join the lab. The user then has to give the information to another user who can log in, and if they are a Graduate Research Assistant or Undergraduate Research Assistant they can pass the information and join the lab. The backend will add this user to the lab and when visiting the dashboard will be presented with all labs.

**Conclusion**

In conclusion, I feel I am about 75% to 80% finished with the project. The major modules left to complete is project and task classes and get them to be fully functional and interactable with the

database. Other tasks that need to be completed are minor things such as: “when updating a user, take care of this change within the lab model that has the user info under the users[ ] object array”.

This is the first time I have attempted to create a project of this scope, so please feel free to email me for any errors that may occur or any programming patterns that are not best practice, or optimal.