3813ICT Assignment: 1

Git

The layout of the git repository simply contains the project folder and a README file. Default branch is main. During the development process, commits where done frequently usually after every stable milestone was achieved so that the application could run without any errors with every commit. Examples of a ‘stable milestone’ include things like new features to the application, a new completed route, improved design of a section/page and any major bug fixes/improvements that i potentially made previously.

Data structures

There are three main data structures which were used in the program. Each structure is stored in its own separate JSON file all stored /data on the server side of the application.

The first structure is called users, which is just an object with a username and their corresponding password to go with it for every user that has an account. This data will be used for login purposes only. This structure is separated from the rest of the user’s information for security reasons. By having a separate file just for the users file, the risk of accidently revealing sensitive information is minimized as this information will remain untouched unless a user is logging in or creating an account.

The second structure is called extendedUsers. This structure is used to store information about each user. This information includes userid, username, useremail, userrole. All this information is used to create a certain profile for each user and is where all the user’s data is stored except for the user’s password.

The third structure is called groups. This structure stores all the information for the groups including the users who are in the groups, the channels/rooms for each group and the users who are in those channels. A group has a name, group users, rooms, room name, and room users.

REST API

Route: /login

Parameters: username, password

Return Value: userid, userlogin, username, useremail, userroll

Description: This route takes in a username and password when a user tries to login. In then checks if the username and password are correct by comparing the values to ‘users’ data structure. If the values are correct, it will return the users information by grabbing it from the data structure ‘extendedUsers’ as well as a true statement saying the user is logged in. It will then store this information in local storage on the front end. If the information is incorrect it will simply produce an alert and deny the user from progressing.