Usman's Commit for 03-24

NOTES

- I think there are a bunch of things that I had to install
  - `npm install express cors dotenv`
  - These should be in the dependencies in package.json for the frontend folder
- I renamed the frontend folder to 'client' - I think I did for some backend reason but its not really needed anymore. I am fine if anyone wants to change it back to 'frontend'
- for the backend to run you need to create a file called '.env' in the backend folder. I sent what to put in that file on the discord
- add this file to gitignore as well

Usman's work for 03/24

1. Created backend folder
2. ran `npm init -y` to generate a package.json file with default settings in it
3. created `/src` folder with `index.js` in it
4. ran `npm install express cors dotenv` in backend folder
5. disabled RLS for all tables in DB since I was running into trouble with this
   1. I think we would have to re-enable this when we know we are going to have others using the app -- it was complicating things for me right now since we need to create policies in supabase that detail who can access what from the database -- so as of now everything in the database is accessible
6. I created an entry in the USERS table in the database for testing purposes
7. Within the `src` folder I created another folder called `services`
8. In the `services` folder I created a `db.js` file
9. In this file I connected the backend to supabase
10. I created a fetchUsers function that console.logs all the entries in the USERS table
11. Within the `src` folder I made a `routes` folder. I hope to use this folder more than index.js so that file can stay clean and readable
12. In `routes` I created `login.js`
13. This file works with `LoginPage.jsx` to authenticate a user into the app
    1. I was not sure how we are doing authentication just yet and I wanted to start creating routes in the backend so this was the first thing that made sense to me

14. For `login.js` to work with `LoginPage.jsx` I added a handleLogin function, I also changed the username and password field and submit button a little bit so it connects with the backend

---

How to get the backend and frontend running together

- In index.js I made a function that checks if the backend is running
- I implemented a fetchUsers function thats commented out right now
- I implemented the post route for the login page that makes the login button work

- **open a terminal and navigate to the server folder**
- **run** `npm dev`
  - this starts the backend running on port 4000

- **open another terminal and navigate to the client folder**
- **run** `npm dev`
  - the app should be up and running
  - the username and password is in the discord