React Project Setup Manual

This manual guides you through the steps to set up and run our React project on Ubuntu, including the installation of necessary software, project cloning, and execution instructions.

Prerequisites

1. **Ubuntu** : Ensure Ubuntu is installed on your machine. If not, you can download it from the official Ubuntu website and follow their installation instructions.
2. **PostgreSQL**: A relational database system used by our project for data management.
3. **Node.js**: A JavaScript runtime required to run our project's backend and frontend.

- ## Installation Steps

Installing Ubuntu

If you haven't installed Ubuntu, please visit the official Ubuntu website (https://ubuntu.com/download) and download the latest version. Follow the installation instructions provided on the site.

Installing PostgreSQL on Ubuntu

1. Open a terminal in Ubuntu.
2. Update your system's package list with **sudo apt update**.
3. Install PostgreSQL using **sudo apt install postgresql postgresql-contrib**.

Installing Node.js on Ubuntu

1. Open a terminal in Ubuntu.
2. Install Node.js using Ubuntu's package manager with the command: **sudo apt install nodejs**.
3. Install npm, Node.js's package manager, with **sudo apt install npm**.

Cloning the Project

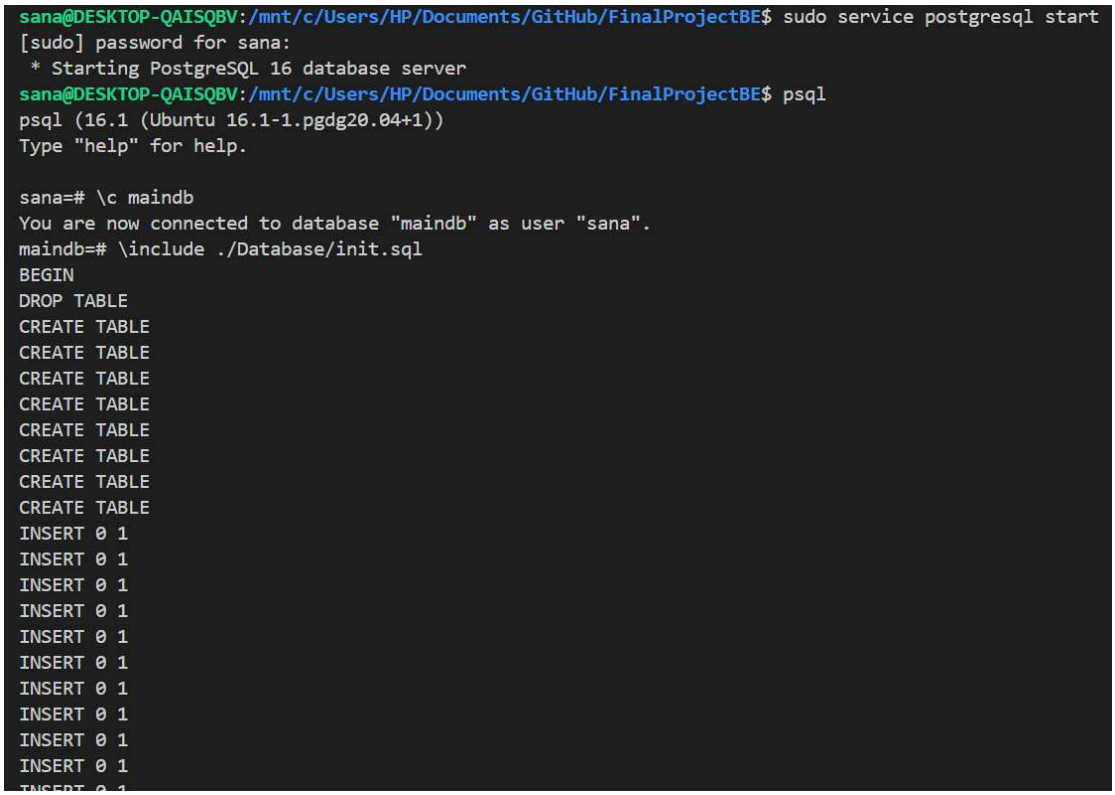
1. Clone the project repository from the gitlab to your computer.
2. Navigate to the project directory. You will find three folders: **Admin**, **FinalProject**, and **FinalProjectBE**.

Project Setup

Setting Up the Backend (FinalProjectBE)

1. Open Visual Studio Code and open the **FinalProjectBE** folder.
2. Open a new terminal in Visual Studio Code (ensure it's using WSL by checking the terminal's title bar).
3. Run **"npm install"** to install the project dependencies.
4. Open another terminal window and start PostgreSQL with **"sudo service postgresql start"**. Enter your Ubuntu password when prompted.
5. After starting PostgreSQL, access the PostgreSQL prompt by running **"psql"** and this will open the database , connect to the project database by **"\c maindb"** and also **"\include ./Database/init.sql"** to access to the data that already In the database.

That what you need to see:



6. Return to the first terminal and start the server with "**node server.js**".

Setting Up the Frontend (FinalProject)

1. In Visual Studio Code, open the **FinalProject** folder in a new window.
2. Open a terminal and run **"npm install"** to install the necessary libraries.
3. Once the installation is complete, start the project with **"npm start**". This will open the website in your default browser. Ensure the **FinalProjectBE** server is running simultaneously for the frontend to function properly.
4. (to stop running the FinalProject server you can type "^c"(ctrl c) in the terminal).

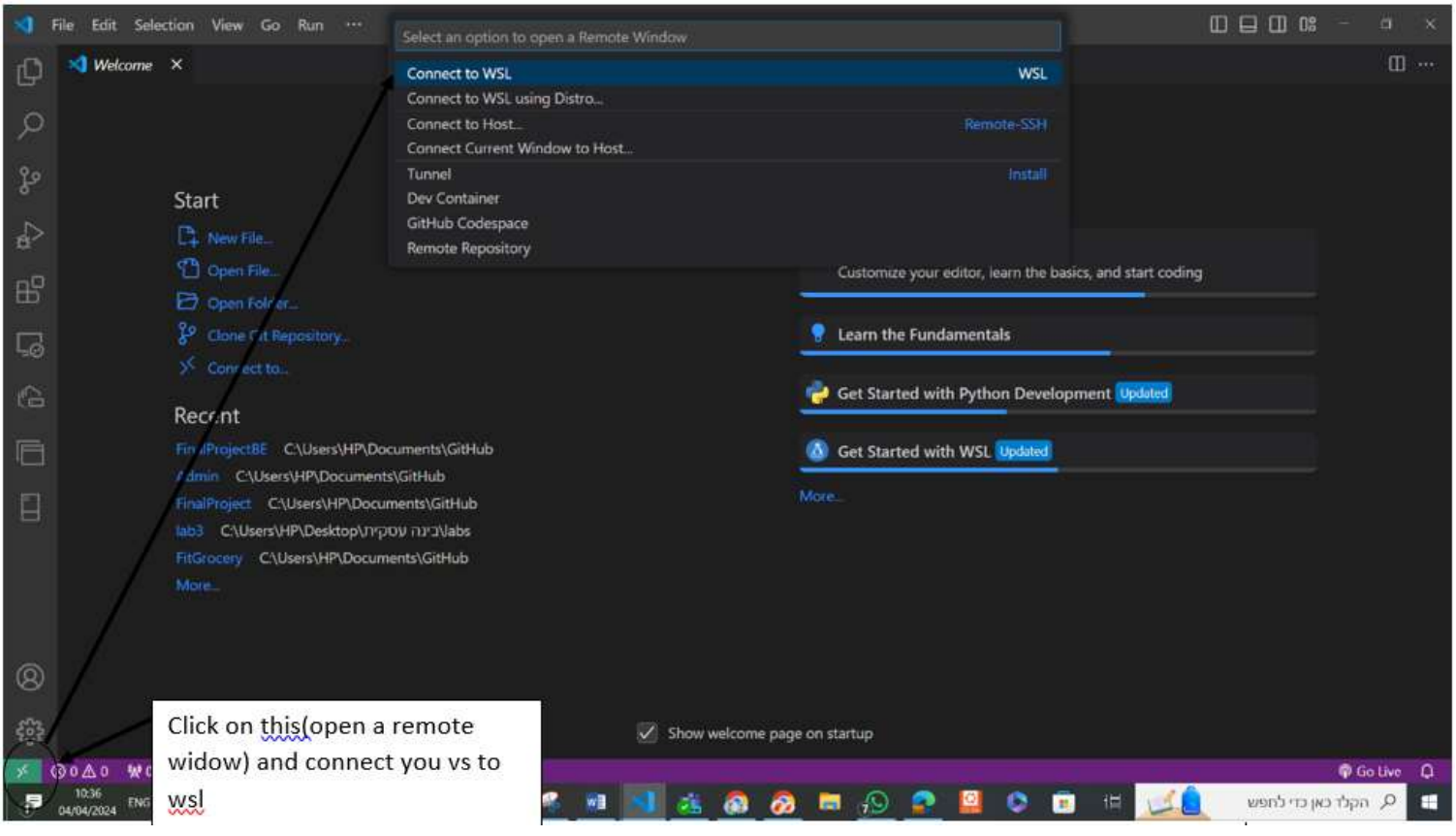
Setting Up the Admin Panel (Admin)

1. Ensure that the **FinalProject** server is not running, as both **Admin** and **FinalProject** use the same server port (3000).
2. Open the **Admin** folder in Visual Studio Code in a new window.
3. Follow the same steps as the **FinalProject** setup by running "**npm install**" followed by **"npm start**" in the terminal.

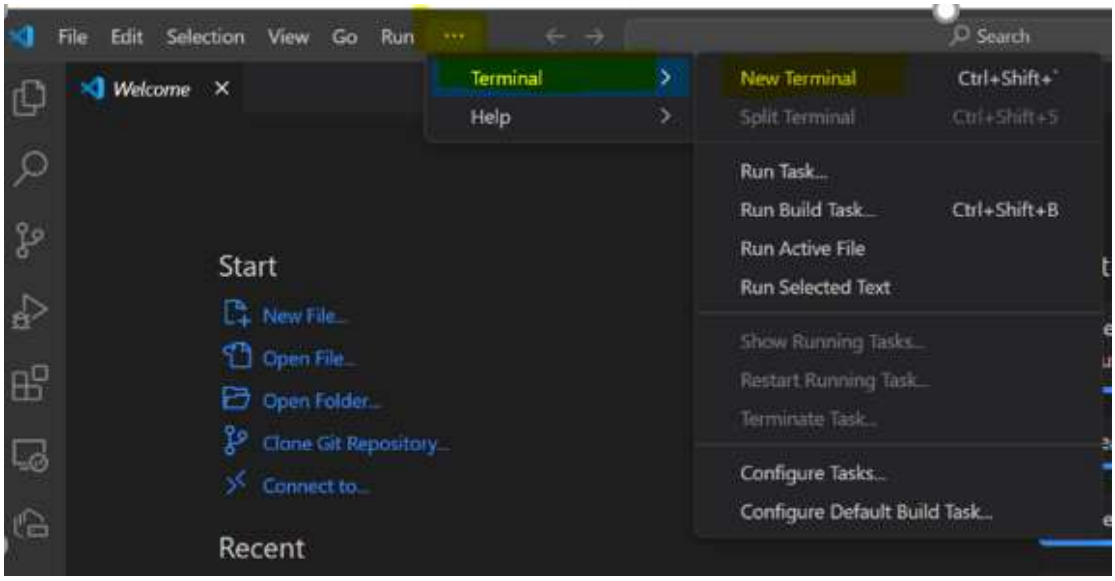
Notes

- Ensure that the backend server (**FinalProjectBE**) is always running when you are working with the frontend projects (**FinalProject** and **Admin**).
- You can switch between the **FinalProject** and **Admin** applications by stopping one (**Ctrl + C** in the terminal where **npm start** was executed) and starting the other.

Some screen shot that can helping you :



Open new tirmenal:



Or :