Online complaint registration and management system (using MERN)

**[Screenshots of installation and execution process]**

**Requirements:**

**Software Requirements**

* **Node js:**

Node.js is a popular, open-source, cross-platform runtime environment that allows you to run JavaScript code outside of a browser, particularly for building backend services.

Download and install the latest version of Node.js from the official website: [Node.js download](https://nodejs.org/).

After installation, for verification we had to run the following command



For verifying the npm modules the below command will be executed



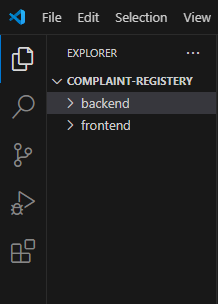
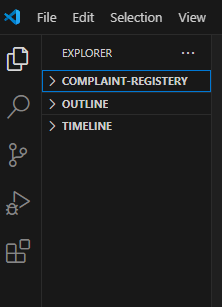
* **Visual studio code editor:**

**VS Code** (Visual Studio Code) is a powerful, lightweight code editor developed by Microsoft. It is widely used for web development, including **Node.js**, **React**, and many other technologies.

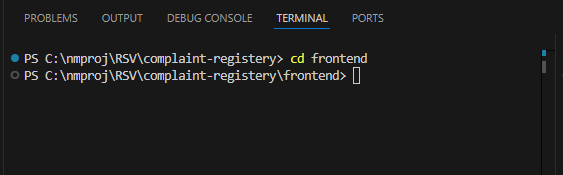
Download and install VS Code from the official website: [VS Code Download](https://code.visualstudio.com/).

**Project configuration:**

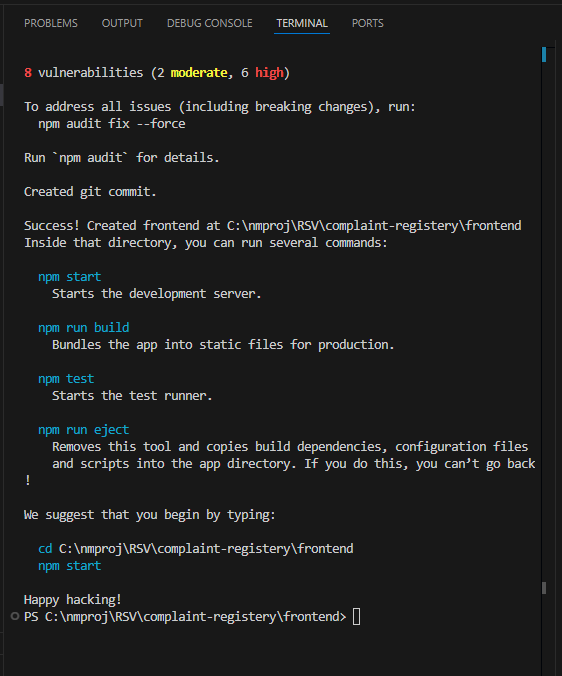
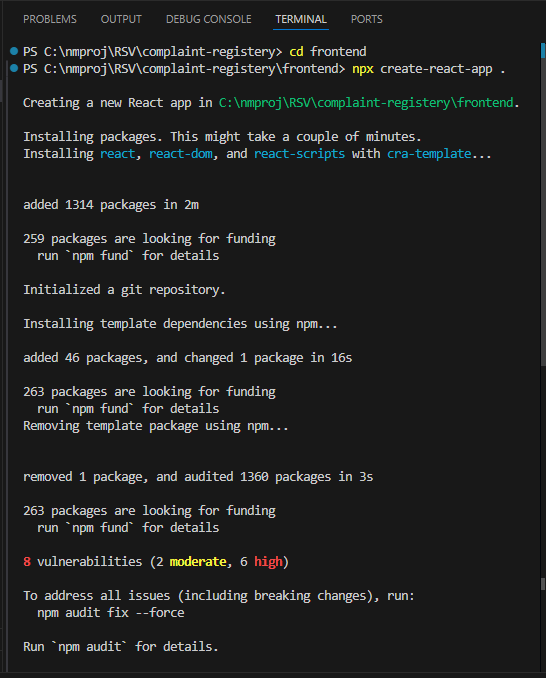
Open the **VS Code** editor and create a project folder named **Complaint Registry**. Inside this folder, create two subfolders: **frontend** and **backend**.



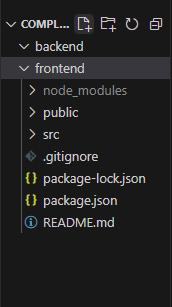
After creating the folder open the frontend folder in the terminal by giving the below command



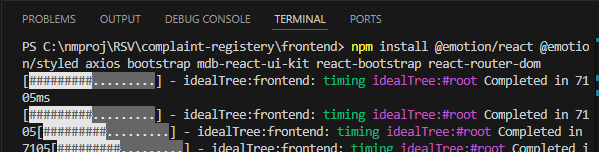
Now, to install the React module inside the frontend folder, using the following command:



The react packages are installed in the frontend folder



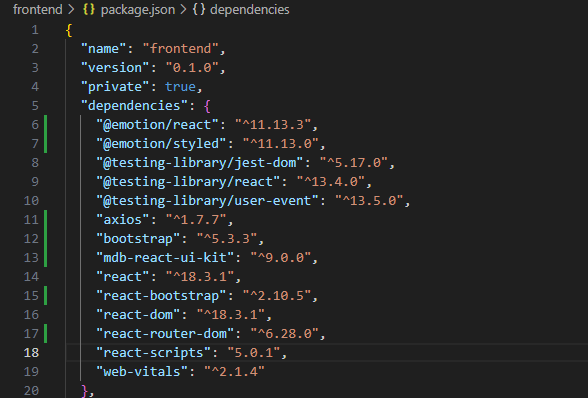
Now we need to install needed packages to build and execute our project



Packages such as @emotion/react axios react-bootstrap etc., modules are installed using the “**npm install**” command.

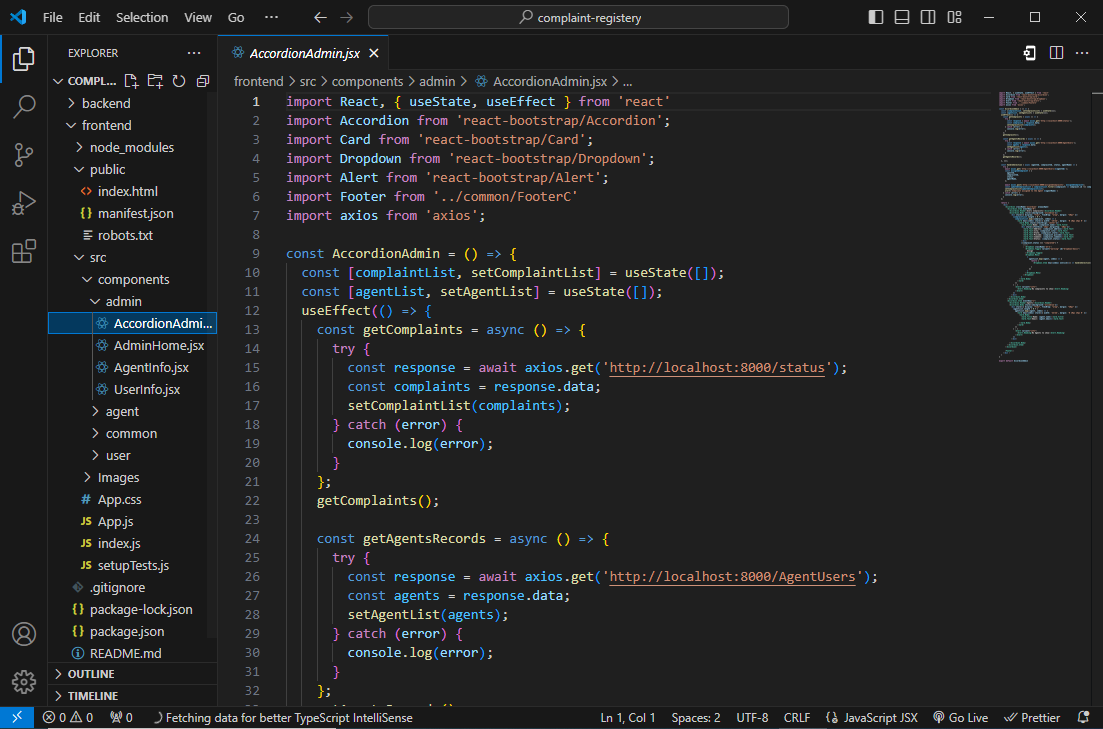
Once these packages are installed we can see these packages in the dependencies column of package.json file.

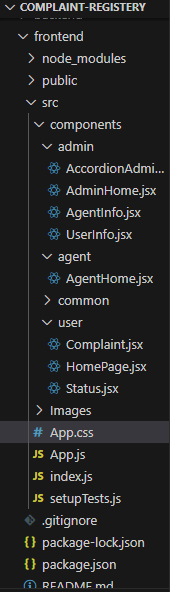
The picture of package.json file of frontend folder



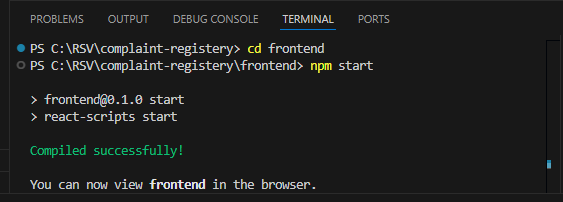
The package.json file is a crucial part of a **Node.js** or **JavaScript** project, especially when using frameworks like **React**. It is automatically generated when you create a new project with tools like **create-react-app**, and it contains metadata and configuration details for the project.

Now we need to inserted the code in our desired files and addition folders are created in the frontend folder

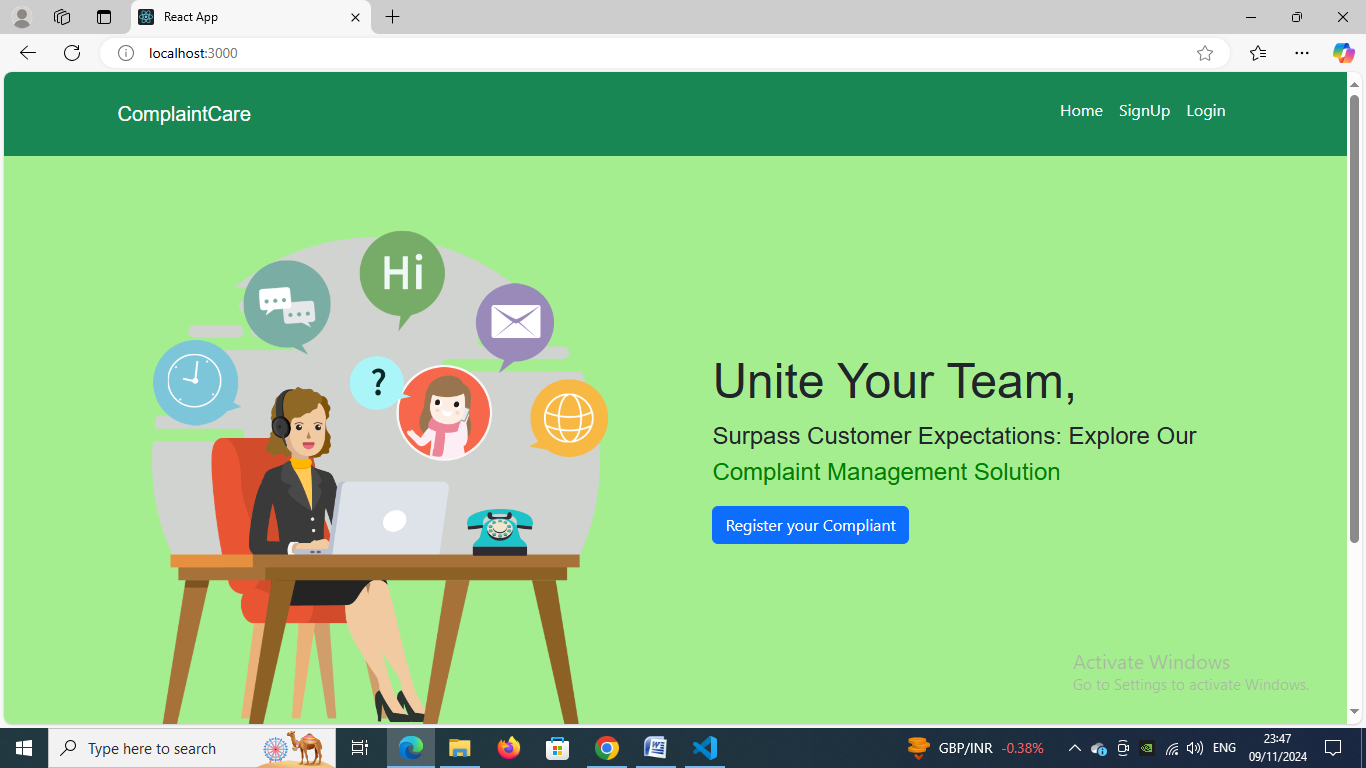




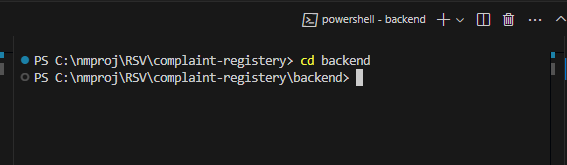
Now, we can use the following command to view the output:



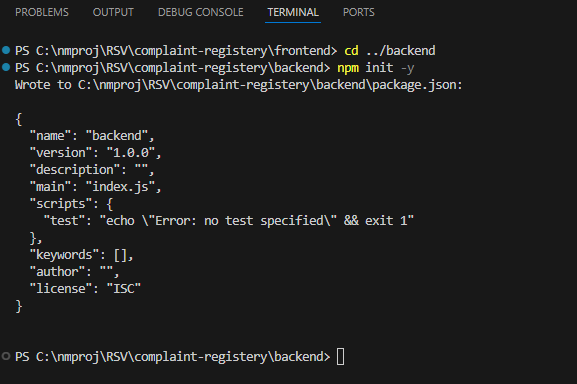
The output of the frontend was shown in the browser



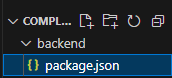
Next, to set up the backend, navigate to the backend folder using the terminal by opening it and ensuring you are in the correct directory before proceeding with the necessary backend setup.



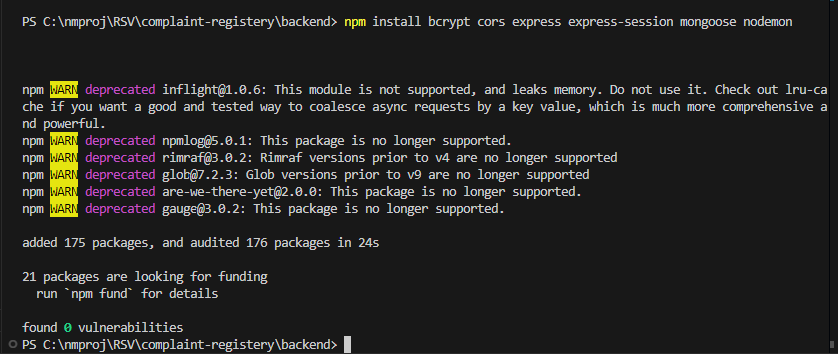
By using the above command we got navigated to backend folder. For backend setup we need to install the needed modules



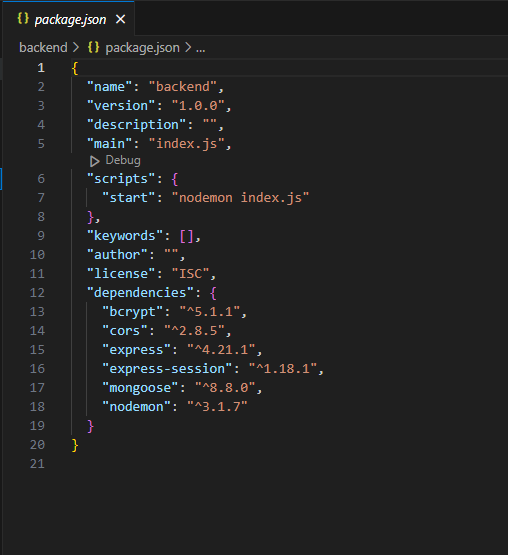
The above command will install the package json file



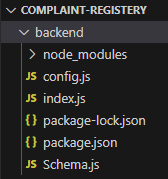
Now we need to install node modules and needed packages



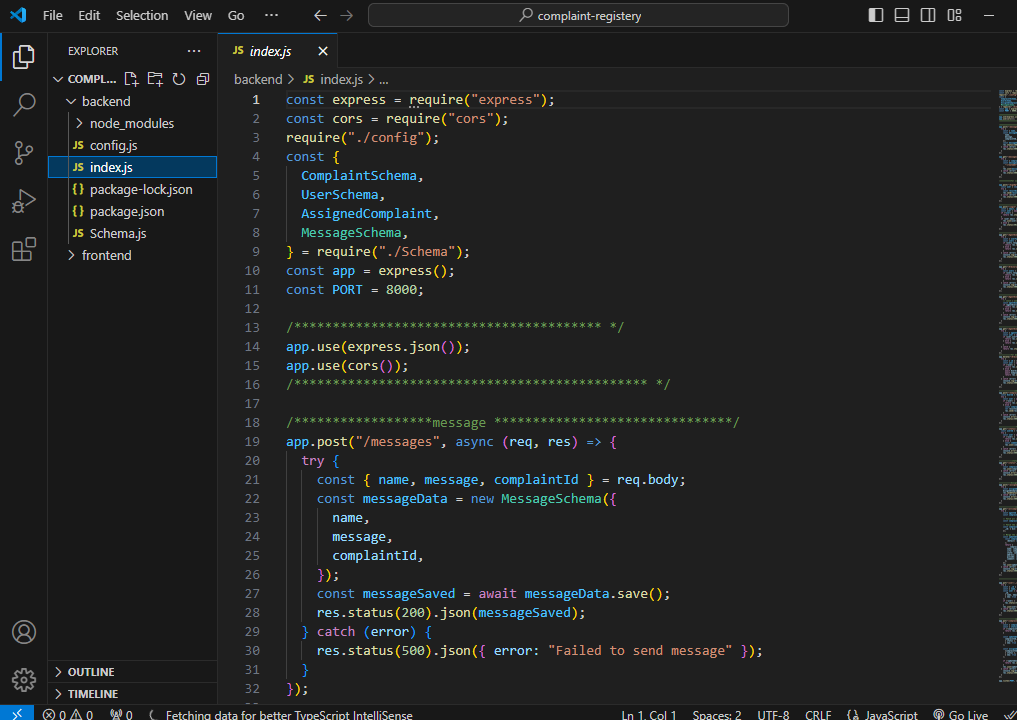
The above command will install the bcrypt, cors, express, express-session etc., packages in the backend.



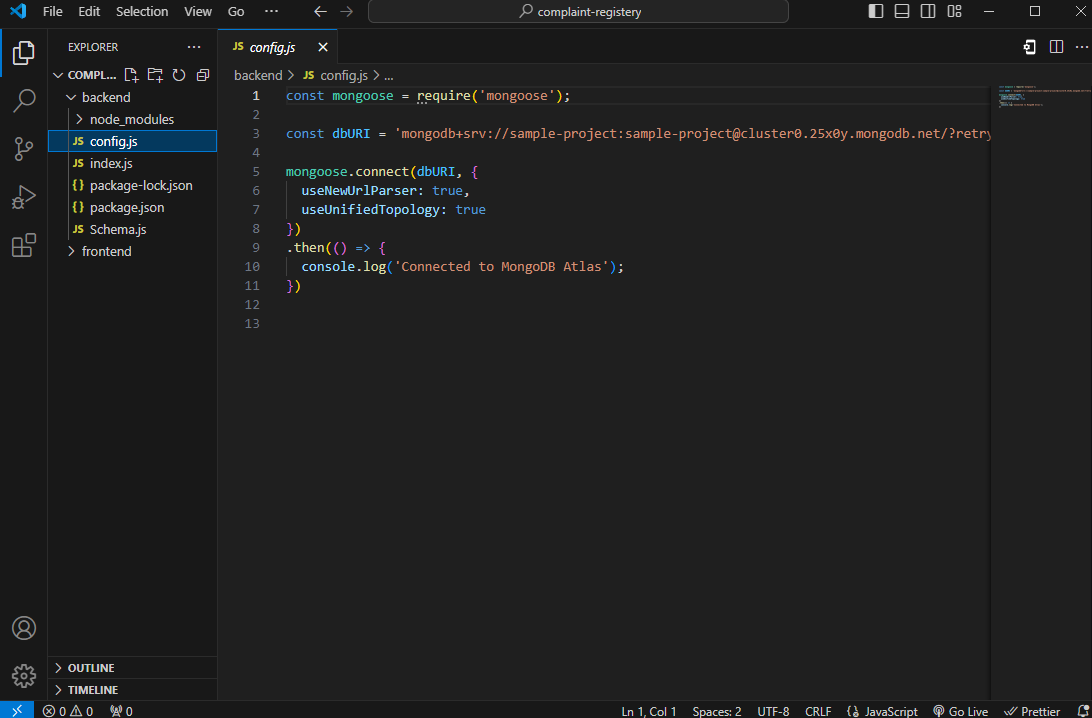
Add the extra needed files , mongo db configuration file etc



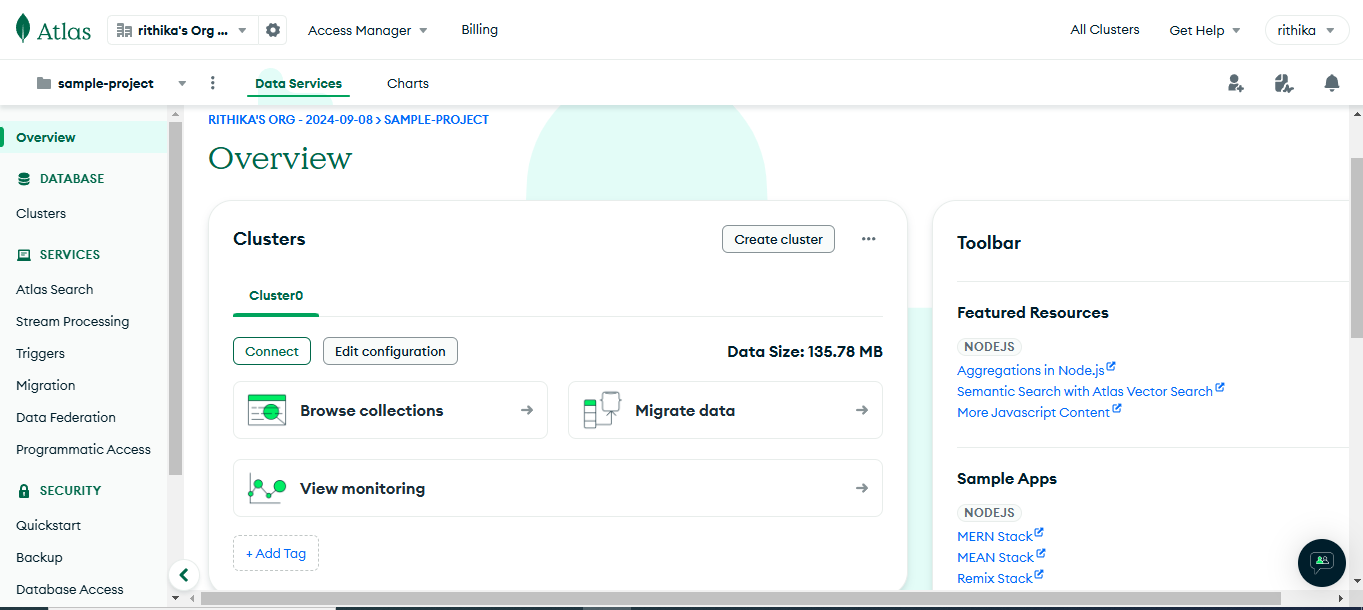
The code was written in the appropriate files



the config.js file was created to connect the mongodb

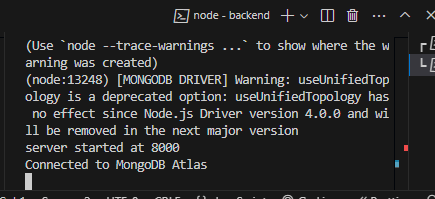
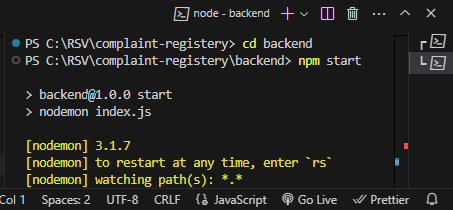


Create a database in the mongodb atlas



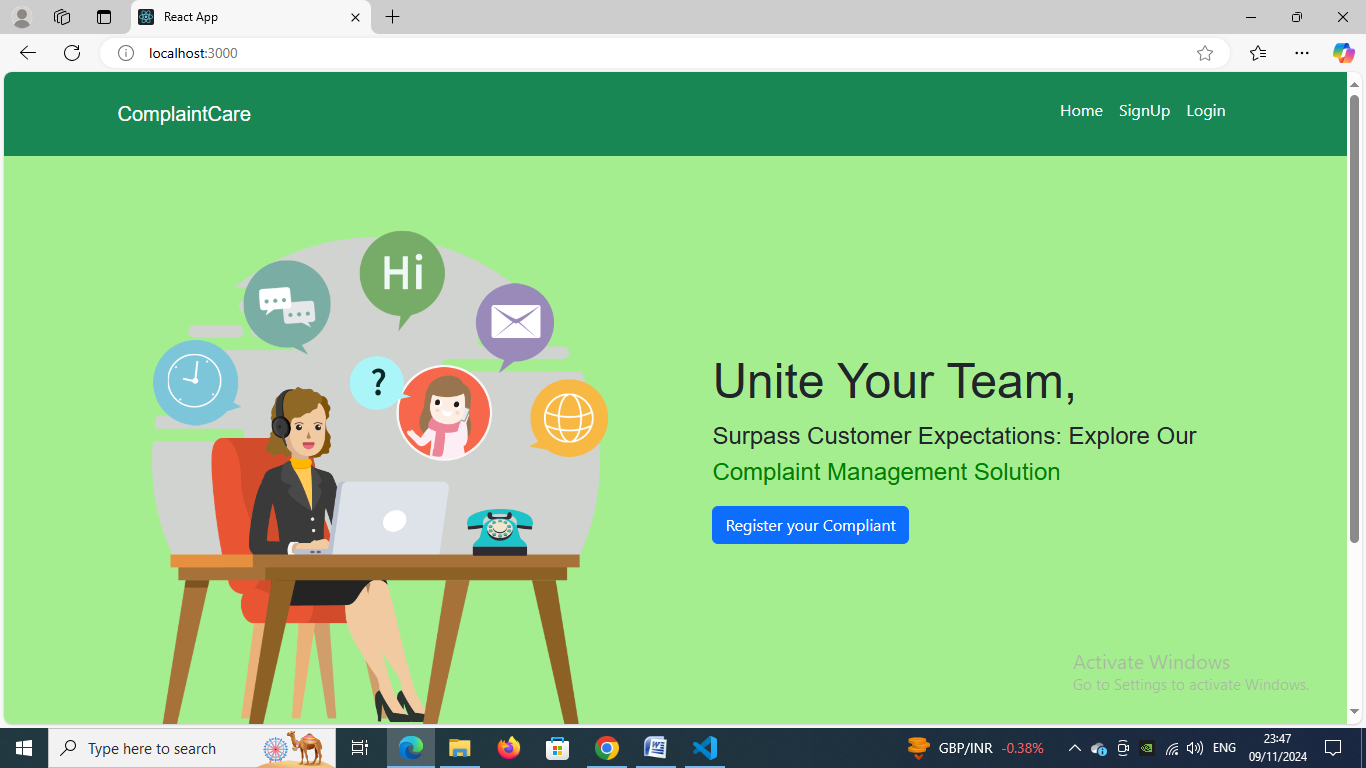
And give the generated string in the config js file.

Now to execute the backend modules, we need to run the below command

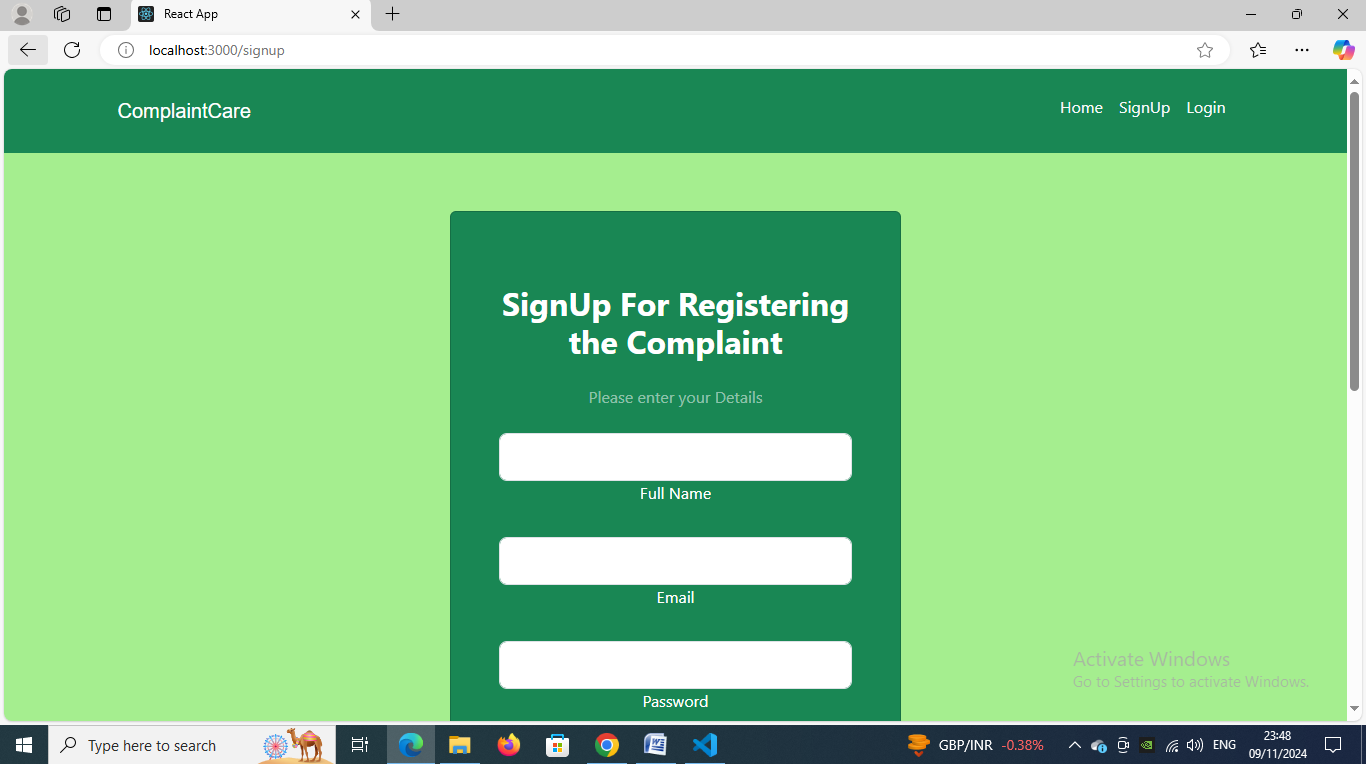


Now the backend folder was successfully executed and mongodb was connected.

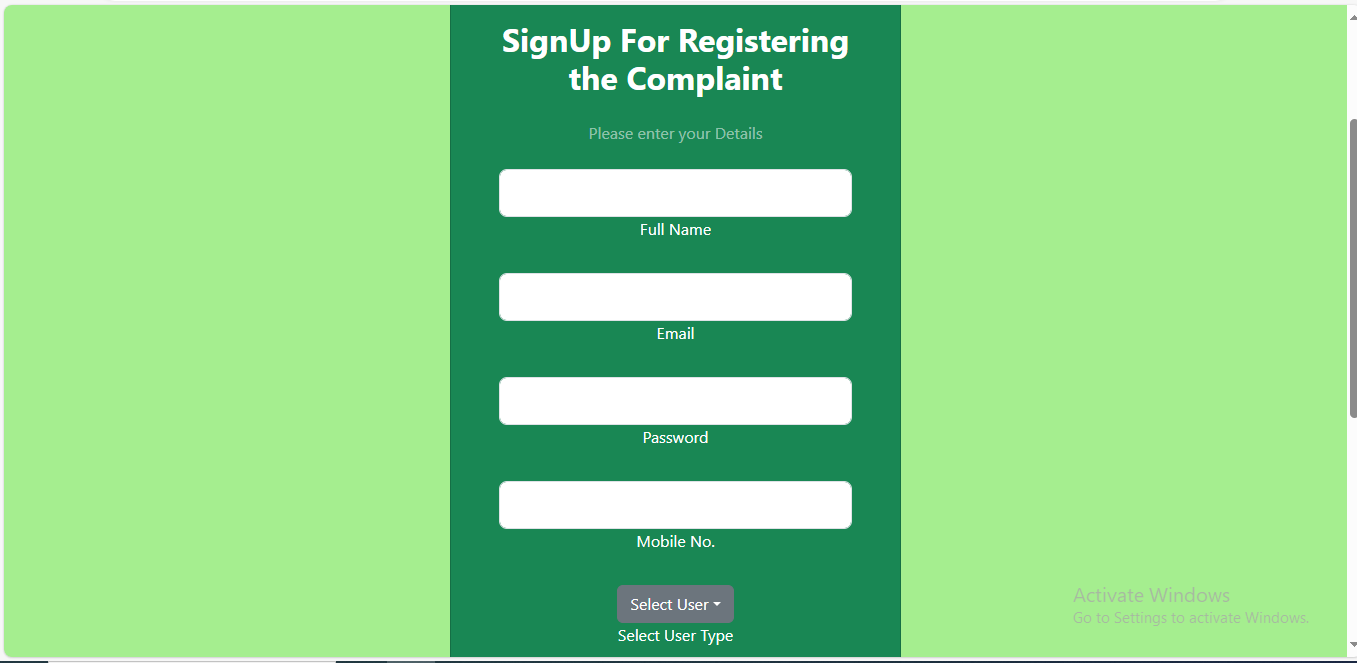
The output can be seen in the browser



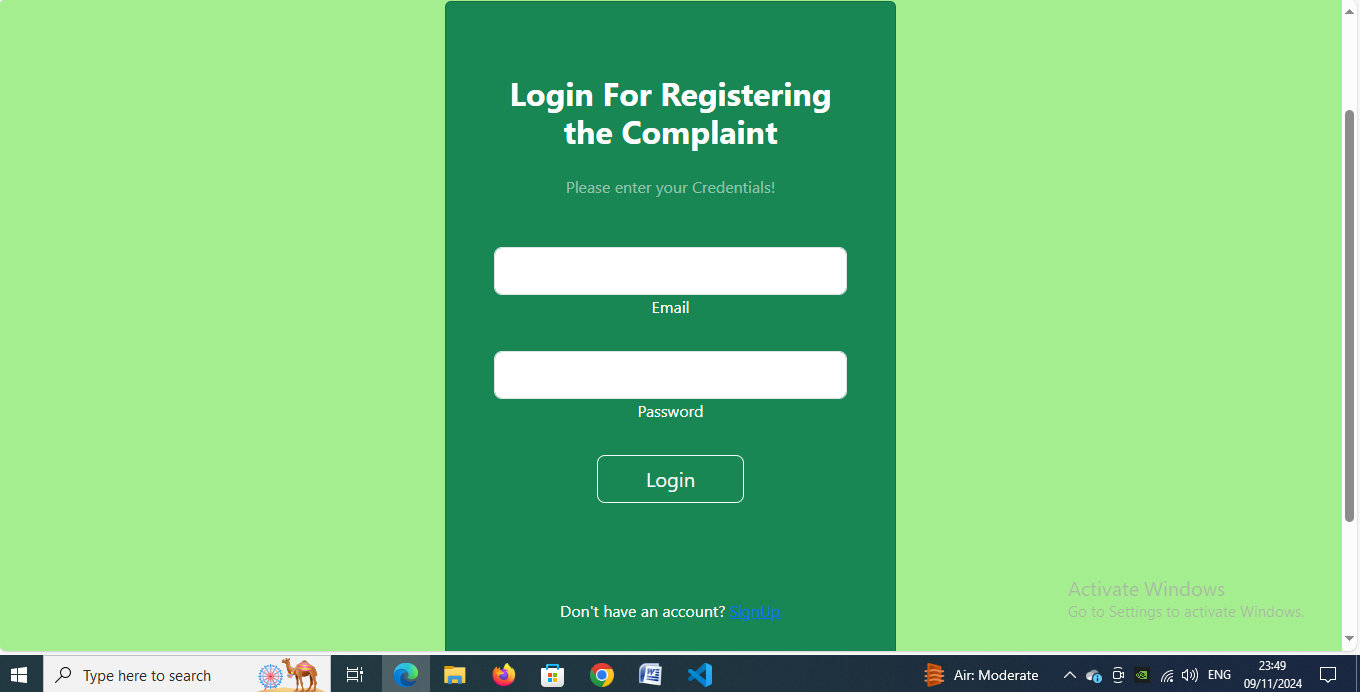
Home page



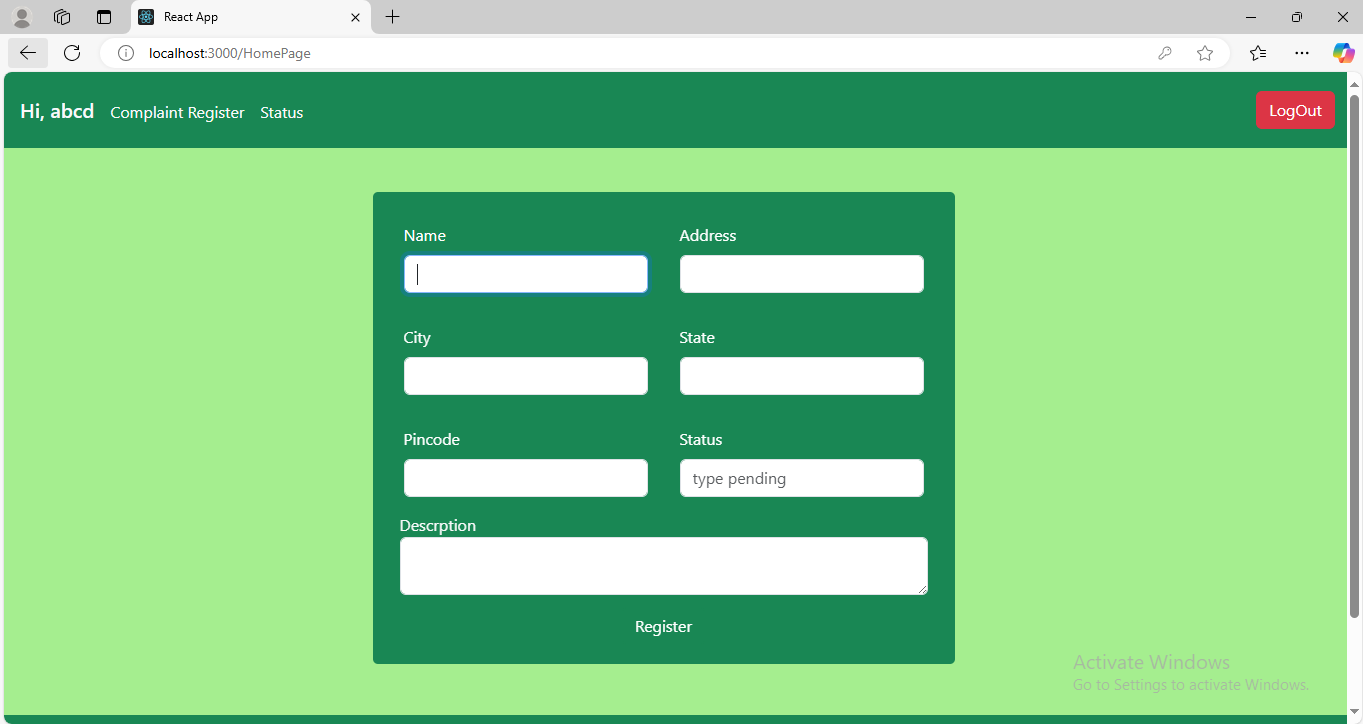
Sign up page



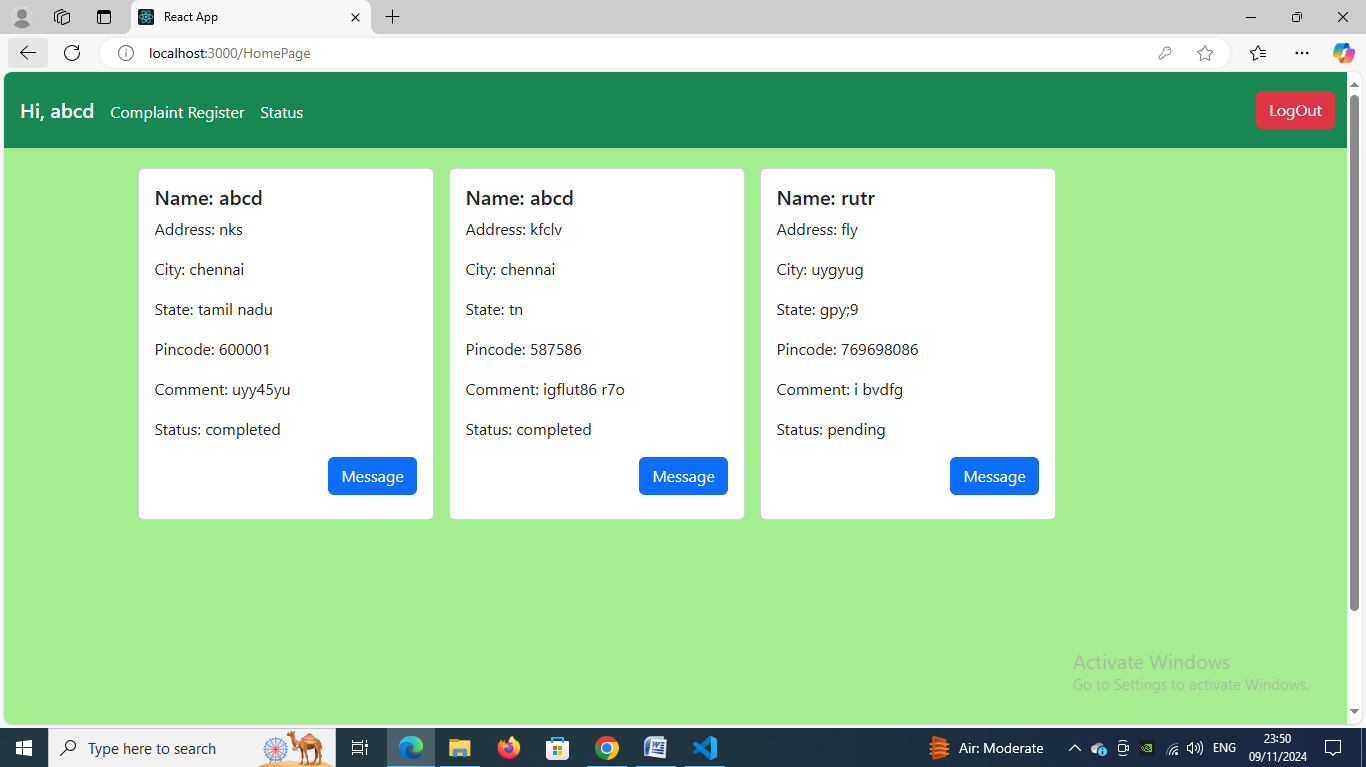
Login page



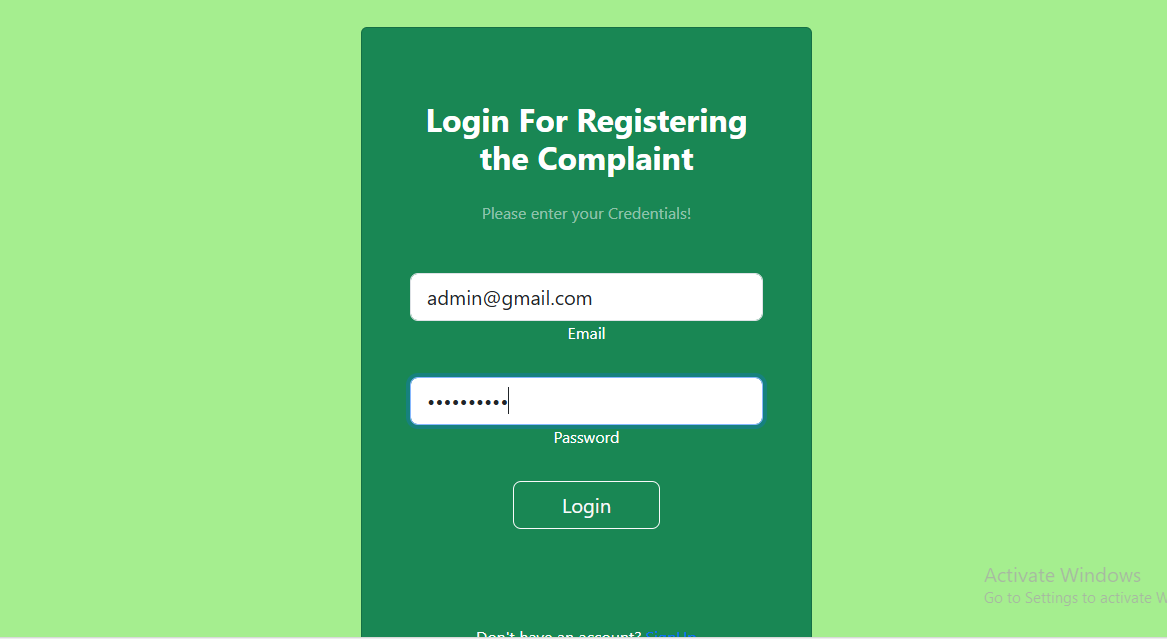
Complaint registration page



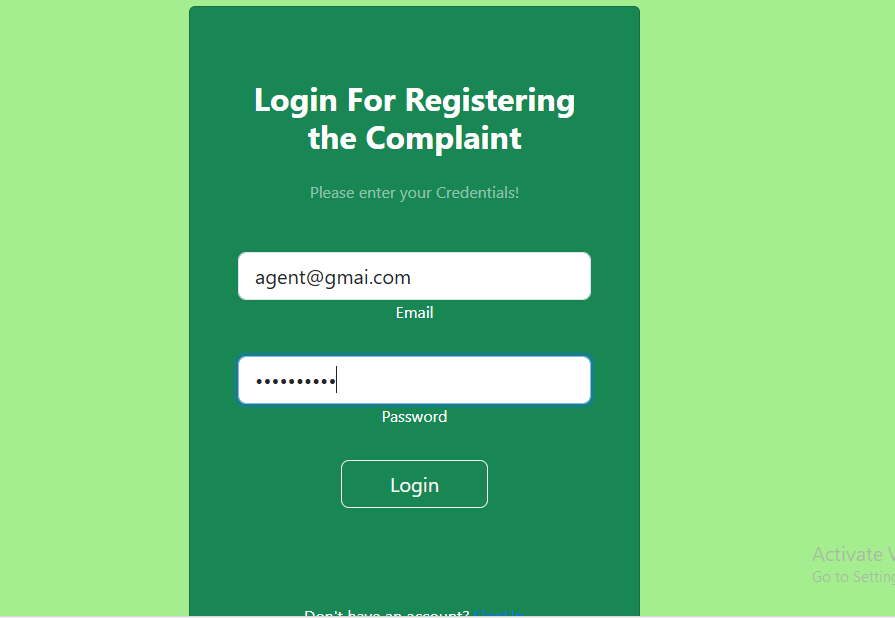
Status page



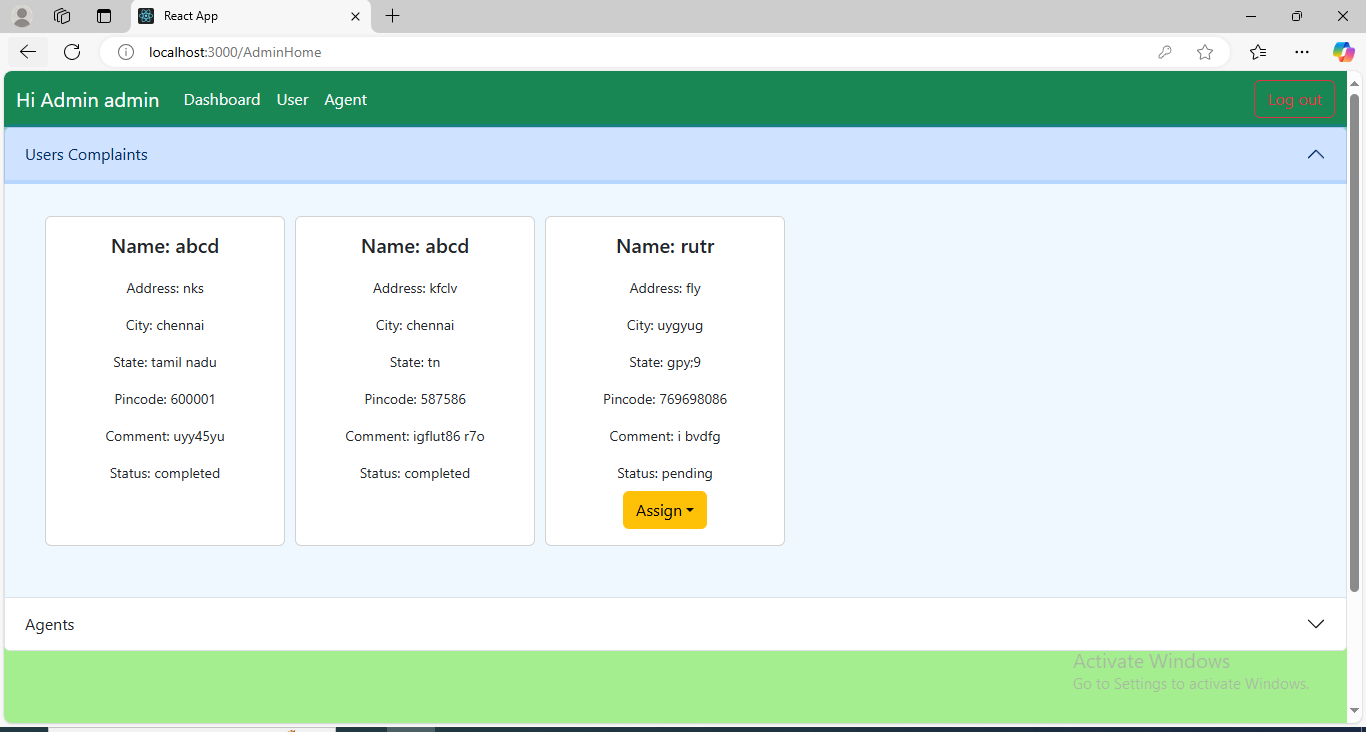
Admin login



Agent login



Admin dashboard



Agent dashboard

