**https://lh7-rt.googleusercontent.com/docsz/AD_4nXf9X0cvbVR-UEYUcfm6Li7xiQL1xtS5CTpukpvA8pWCllJmxhA4wT0AmbHi3gHVWjSsANNPS-u6DzFtVn1HgqoNFYmVvQvVrvcJVFci_0sCAUMYywmQV5YpWs3j4A8RNMiSHL1M0g?key=toZvQEdoGLaMTZOJawtCyxr8**

**INTERNSHIP REPORT**

**2025**

 **Gates institute of Technology, Ananthapuram**

**Submitted by: Uppala Laxmi chaitanya**

**Rollno:** **22F21A3726**

**HouseHunt: Finding your perfect rental home**

**HOUSEHUNT: FINDING YOUR PERFECT RENTAL HOME**

# **INTRODUCTION :**

A house rent app is typically a mobile or web application designed to help users find

rental properties, apartments, or houses for rent.

**DESCRIPTION234567890-+-**

The purpose of HouseHunt is to streamline the real estate journey by providing a comprehensive, user-friendly platform that connects home buyers, sellers, renters, and real estate professionals. It aims to simplify the often complex and time-consuming process of searching for or listing a property, offering tools and resources that enhance **1.** **Empowering Users with Information:** HouseHunt provides detailed listings, market insights, and property data, helping users make informed decisions about buying, selling, or renting homes.

**2.** **Simplifying the Property Search:** With advanced search filters and features like virtual tours, HouseHunt makes it easy to find properties that match specific preferences such as location, price, and amenities.

**3.** **Facilitating Transactions:** By connecting buyers with sellers, renters with landlords, and users with real estate professionals, HouseHunt streamlines the entire transaction process, from browsing listings to negotiating deals and finalizing paperwork.

**4.** **Supporting Real Estate Professionals:** HouseHunt serves as a marketing tool for real estate agents and brokers, offering them a platform to reach potential clients and showcase properties.

**5.** **Creating a Transparent Marketplace:** By providing clear, accurate property information, HouseHunt fosters transparency, ensuring that users have access to reliable data to make confident real estate decisions. Offering a Convenient, 24/7 Platform: HouseHunt is accessible anytime, anywhere, offering user

# **TECHNICAL ARCHITECTURE**



The technical architecture of our House rent app follows a client-server model, where the frontend serves as the client and the backend acts as the server. The frontend encompasses not only the user interface and presentation but also incorporates the axios library to connect with backend easily by using RESTful Apis.

The frontend utilizes the bootstrap and material UI library to establish real-time and better UI experience for any user whether it is admin, doctor and ordinary user working on it.

On the backend side, we employ Express.js frameworks to handle the server-side logic and communication.

For data storage and retrieval, our backend relies on MongoDB. MongoDB allows for efficient and scalable storage of user data, including user profiles, for booking room, and adding room, etc. It ensures reliable and quick access to the necessary information.

Together, the frontend and backend components, along with moment, Express.js, and MongoDB, form a comprehensive technical architecture for our House rent app. This architecture enables real-time communication, efficient data exchange, and seamless integration, ensuring a smooth and immersive booking an appointment and many more experience for all users.

# **ER DIAGRAM**



**PREREQUISITE:**

Here are the key prerequisites for developing a full-stack application using Node.js, Express.js, MongoDB, React.js:

✔**Node.js and npm:**

Node.js is a powerful JavaScript runtime environment that allows you to run JavaScript code on the server-side. It provides a scalable and efficient platform for building network applications.

Install Node.js and npm on your development machine, as they are required to run JavaScript on the server-side.

Download: https://nodejs.org/en/download/

Installation instructions: <https://nodejs.org/en/download/package-manager/>

**npm init**

✔**Express.js:**

Express.js is a fast and minimalist web application framework for Node.js. It simplifies the process of creating robust APIs and web applications, offering features like routing, middleware support, and modular architecture.

Install Express.js, a web application framework for Node.js, which handles server-side routing, middleware, and API development.

Installation: Open your command prompt or terminal and run the following command:

**npm install express**

✔**MongoDB:**

MongoDB is a flexible and scalable NoSQL database that stores data in a JSON-like format. It provides high performance, horizontal scalability, and seamless integration with Node.js, making it ideal for handling large amounts of structured and unstructured data.

Set up a MongoDB database to store your application's data.

Download: https://www.mongodb.com/try/download/community

Installation instructions: https://docs.mongodb.com/manual/installation/

✔**Moment.js:**

Moment Js is a JavaScript package that makes it simple to parse, validate, manipulate, and display date/time in JavaScript. Moment. js allows you to display dates in a human-readable format based on your location. Install React.js, a JavaScript library for building user interfaces.

Follow the installation guide: [https://momentjs.com/](https://momentjs.com/%20)

✔**React.js:**

React.js is a popular JavaScript library for building user interfaces. It enables developers to create interactive and reusable UI components, making it easier to build dynamic and responsive web applications.

Install React.js, a JavaScript library for building user interfaces.

Follow the installation guide: <https://reactjs.org/docs/create-a-new-react-app.html>

✔**Antd:**

Ant Design is a React. js UI library that contains easy-to-use components that are useful for building interactive user interfaces. It is very easy to use as well as integrate. It is one of the smart options to design web applications using react.

Follow the installation guide: <https://ant.design/docs/react/introduce>

✔**HTML, CSS, and JavaScript**: Basic knowledge of HTML for creating the structure of your app, CSS for styling, and JavaScript for client-side interactivity is essential.

✔**Database Connectivity**: Use a MongoDB driver or an Object-Document Mapping (ODM) library like Mongoose to connect your Node.js server with the MongoDB database and perform CRUD (Create, Read, Update, Delete) operations. To Connect the Database with Node JS go through the below provided link:

https://www.section.io/engineering-education/nodejs- mongoosejs-mongodb/

✔**Front-end Framework**: Utilize Reactjs to build the user-facing part of the application, including entering the booking room, status of the booking, and user interfaces for the admin dashboard.

For making better UI we have also used some libraries like material UI and bootstrap.

✔**Version Control**: Use Git for version control, enabling collaboration and tracking changes throughout the development process. Platforms like GitHub or Bitbucket can host your repository.

Git: Download and installation instructions can be found at: https://git-scm.com/downloads

✔**Development Environment**: Choose a code editor or Integrated Development Environment (IDE) that suits your preferences, such as Visual Studio Code, Sublime Text, or WebStorm.

• Visual Studio Code: Download from <https://code.visualstudio.com/download>

To run the existing Video Conference App project downloaded from GitHub:

Follow below steps:

Clone the Repository:

* Open your terminal or command prompt.
* Navigate to the directory where you want to store the e-commerce app.
* Execute the following command to clone the repository:

**git clone**: <https://github.com/Laxmichaitanya28/HouseHunt-Smartbridge-project.git>

Install Dependencies:

• Navigate into the cloned repository directory:

cd house-rent

• Install the required dependencies by running the following commands:

cd frontend

npm install

cd ../backend

npm install

Start the Development Server:

• To start the development server, execute the following command:

npm start

• The house rent app will be accessible at <http://localhost:3000>

You have successfully installed and set up the online complaint registration and management app on your local machine. You can now proceed with further customization, development, and testing as needed.

**Roles and Responsibilities:**

The project has 2 types of users – Renter and Owner and the other will be Admin which takes care of all the users. The roles and responsibilities of these two types of users can be inferred from the API endpoints defined in the code. Here is a summary:

Renter/Tenant:

1. Create an account and log in to the system using their email and password.
2. They will be shown automatically all the properties in their dashboard.
3. After clicking on the Get Info, all the information of the property and owner will come and a small form will be generated in which the renter needs to send his\her details.
4. After that they can see their booking in the booking section where the status of booking will be showing “pending”. It will be changed by the owner of the property.

Admin:

1. He/she can approve the user as “owner” for the legit user to add properties in his app
2. He monitors the applicant of all doctors and approves them and then doctors are registered in the app.
3. Implement and enforce platform policies, terms of service, and privacy regulations.

Owner:

1. Gets the approval from the admin for his Owner account.
2. After approval, he/she can do all CRUD operation of the property in his/her account
3. He/she can change the status and availability of the property.

# PROJECT STRUCTURE:

+ 



The first image is of frontend part which shows all the files and folders that have been used in UI development

The second image is of the Backend part which shows all the files and folders that have been used in the backend development

# **Project Flow:**

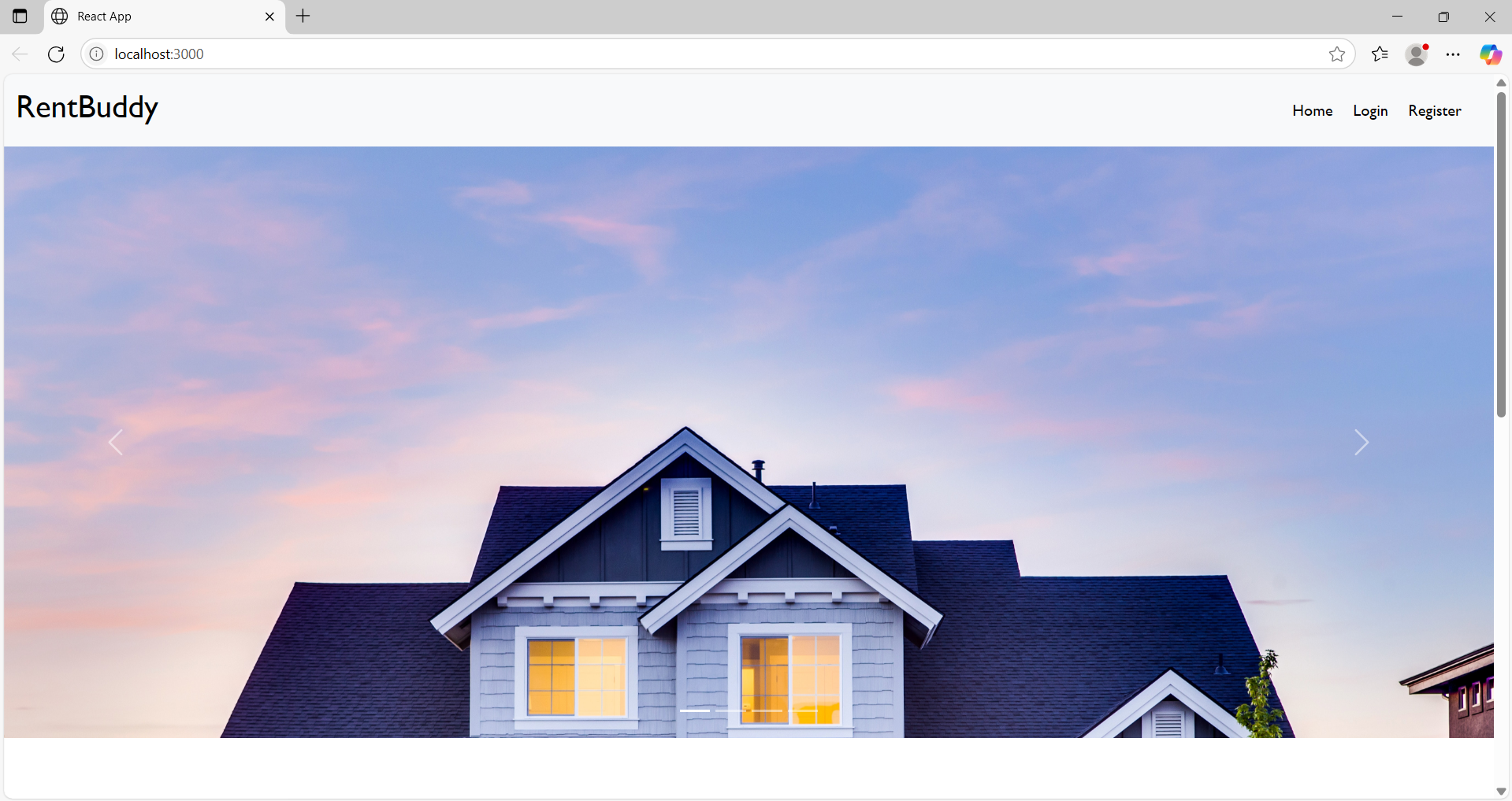
Before starting to work on this project, let’s see the demo.

**Project demo:** [**https://drive.google.com/file/d/1Sa4CYxKFB7qOaVyvE6VO\_rWWvJwXfRm-/view?usp=sharing**](https://drive.google.com/file/d/1Sa4CYxKFB7qOaVyvE6VO_rWWvJwXfRm-/view?usp=sharing)

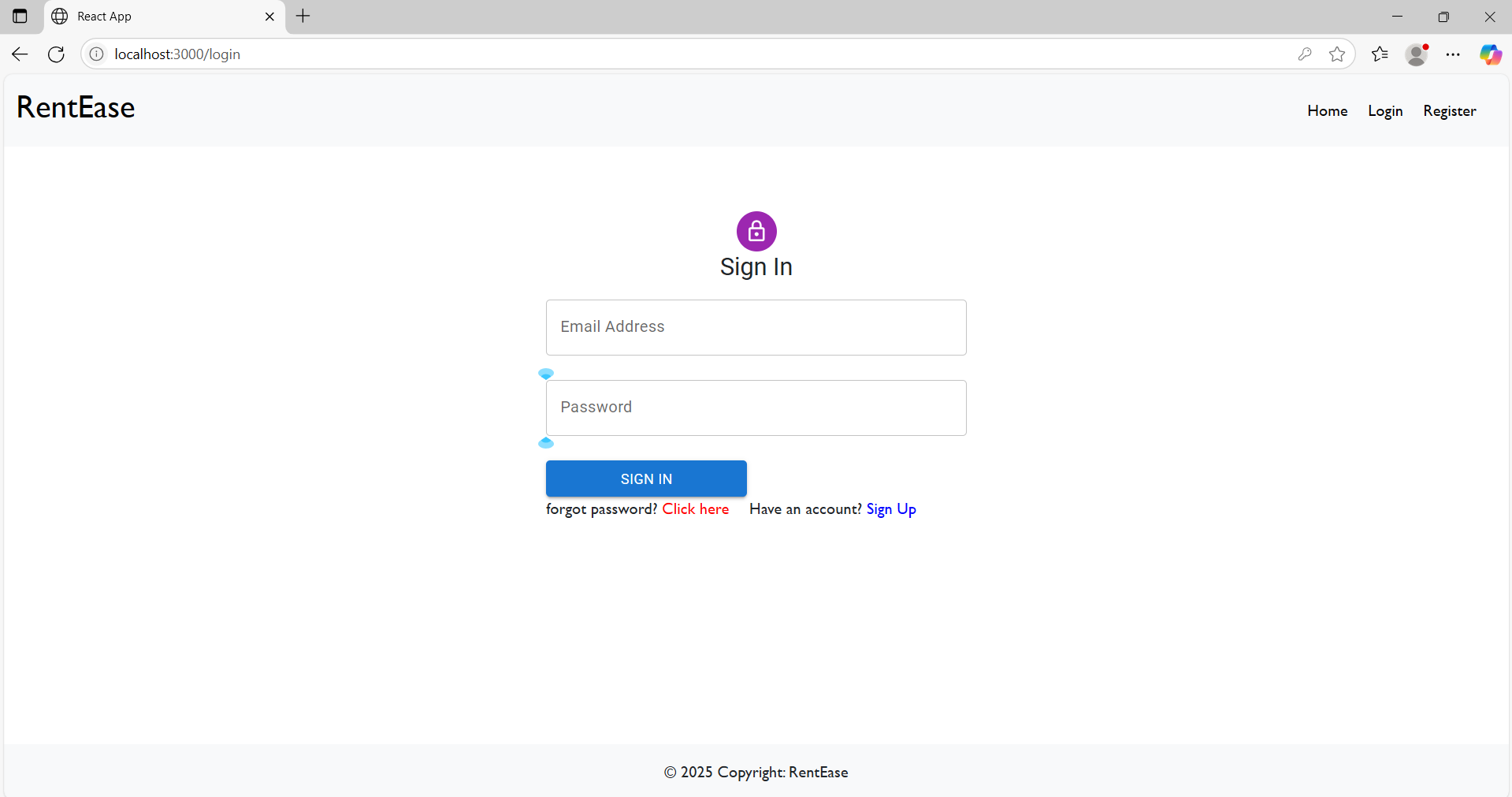
Use the code at: <https://github.com/Laxmichaitanya28/HouseHunt-Smartbridge-project.git>

or follow the videos below for a better understanding.

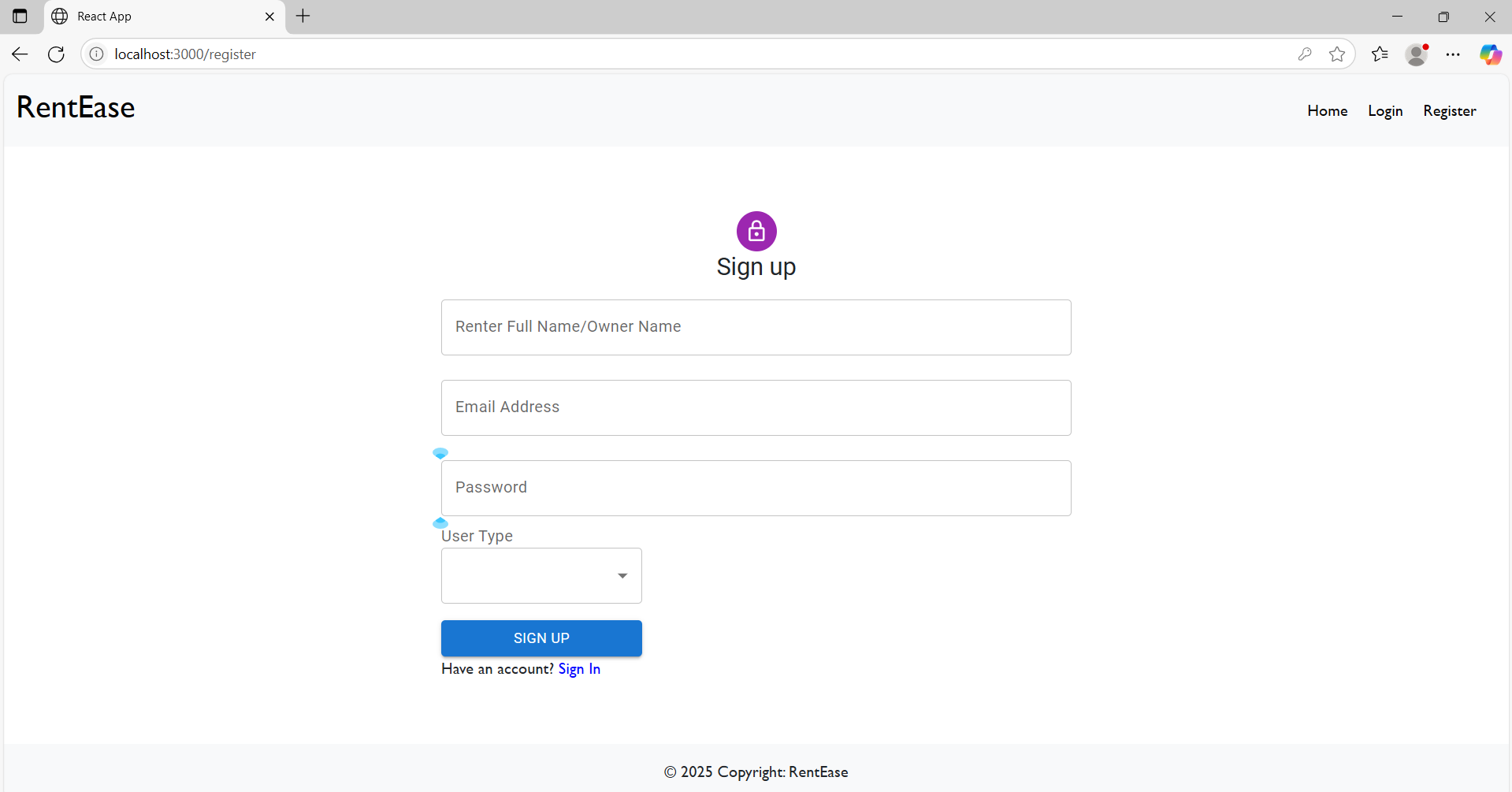
* Landing Page



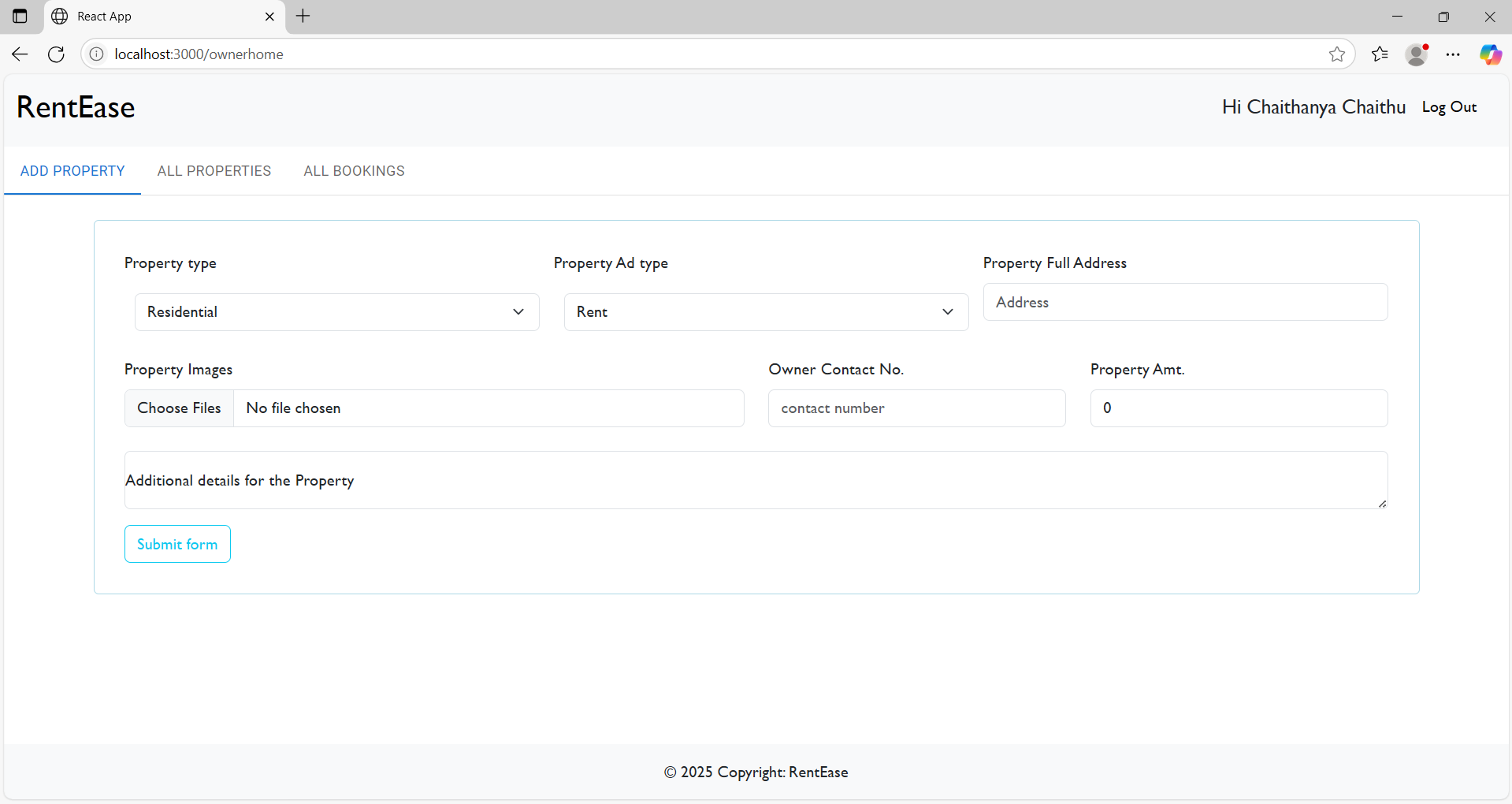
* Login Page



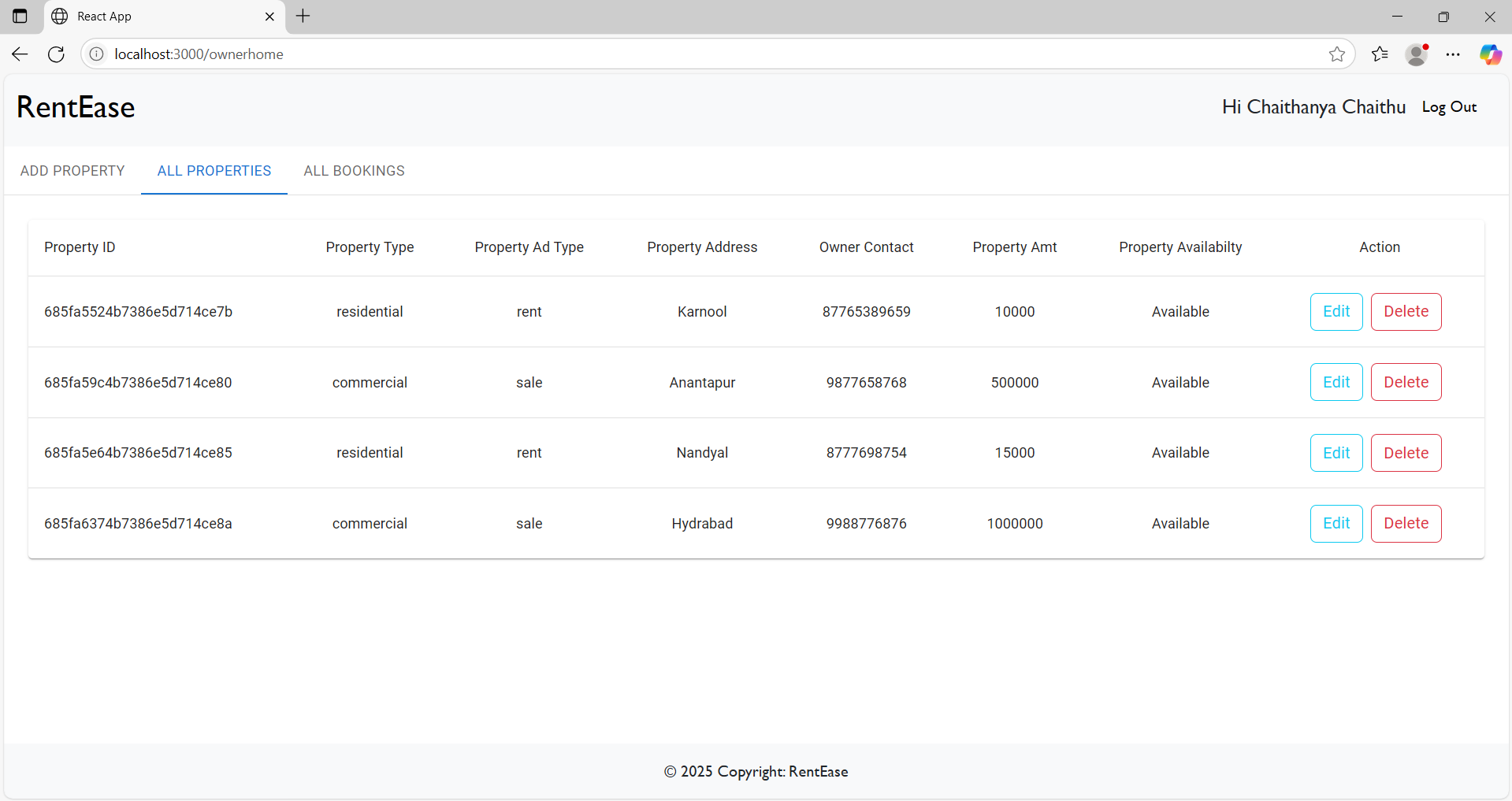
* Registration Page



* Common Dashboard For Add Properties



* Owner Dashboard



* Renter Dashboard

LTVIP2025TMID49186

For any further doubts or help, please consider the code from Github,

<https://github.com/Laxmichaitanya28/HouseHunt-Smartbridge-project.git>

The demo of the app is available at:

<https://drive.google.com/file/d/1Sa4CYxKFB7qOaVyvE6VO_rWWvJwXfRm-/view?usp=sharing>