HouseHunt: Finding Your Perfect Rental Home

A house rent app is typically a mobile or web application designed to help users find rental properties, apartments, or houses for rent. These apps often offer features to make the process of searching for and renting a property more convenient and efficient. Here are some common features you might find in a house rent app:

Property Listings: The app provides a database of available rental properties, complete with detailed descriptions, photos, location, rent amount, and other relevant information.

Search Filters: Users can apply various filters to narrow down their search results based on criteria such as location, rent range, property type (apartment, house, room, etc.), number of bedrooms, amenities, and more.

Contact Landlords/Property Managers: The app might provide a way for users to contact the property owners or managers directly through the app, often through messaging or email.

Scenario-based Case Study:

Scenario: Renting an Apartment

User Registration: Alice, who is looking for a new apartment, downloads your house rent app and registers as a Renter. She provides her email and creates a password.

Browsing Properties: Upon logging in, Alice is greeted with a dashboard showcasing available rental properties. She can see listings with detailed descriptions, photos, and rental information.

She applies filters to narrow down her search, specifying her desired location, rent range, and the number of bedrooms.

Property Inquiry: Alice finds an apartment she likes and clicks on it to get more information. She sees the property details and owner's contact information.

Interested in renting, Alice fills out a small form with her details and sends it to the owner.

Booking Confirmation: The owner receives Alice's inquiry and reviews her details. Satisfied, the owner approves Alice's booking request.

Alice receives a notification that her booking is confirmed, and the status in her dashboard changes to "pending owner confirmation."

Admin Approval (Background Process): In the background, the admin reviews new owner registrations and approves legitimate users who want to add properties to the app.

Owner Management: Bob, a property owner, signs up for an Owner account on the app and submits a request for approval.

The admin verifies Bob's credentials and approves his Owner account.

Property Management: With his Owner account approved, Bob can now add, edit, or delete properties in his account.

He updates the status and availability of his properties based on their occupancy.

Platform Governance: Meanwhile, the admin ensures that all users adhere to the platform's policies, terms of service, and privacy regulations.

The admin monitors activities to maintain a safe and trustworthy environment for all users.

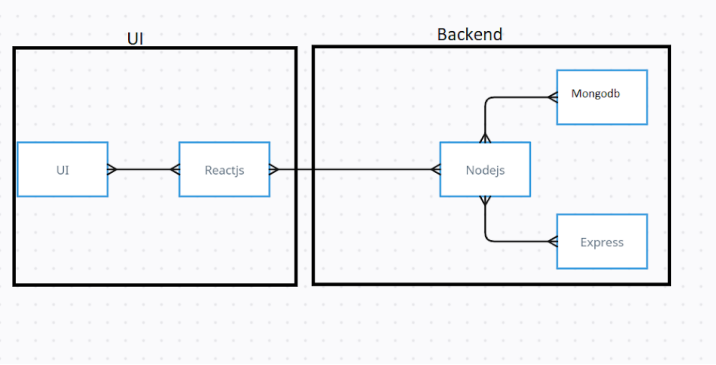
Transaction and Lease Agreement: Once Alice's booking is confirmed, she and the owner negotiate the terms of the lease agreement through the app's messaging system.

They finalize the rental contract and payment details within the app, ensuring transparency and security.

Move-in Process: Alice successfully moves into her new apartment, marking the completion of the rental process facilitated by the house rent app.

This scenario highlights the main functionalities of your MERN-based house rent app, including user registration, property browsing, inquiry and booking process, admin approval, owner management, platform governance, and the overall rental transaction.

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.

Pre-requisites

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

Node.js and npm:

npm init

Express.js:

npm install express

MongoDB

Moment.js

React.js

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

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

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

Application Flow

Roles and Responsibilities:

The project has 2 type of user – Renter and Owner and other will be Admin which takes care to all the user. 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/Tenent:

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 small form will generate in which the renter needs to send his\her details.
4. After that they can see their booking in booking section where the status of booking will be showing “pending”. It will be change by owner of the property.

Admin:

5. He/she can approve the user as “owner” for the legit user to add properties in his app

6. He monitors the applicant of all doctors and approve them and then doctors are registered in the app.

7. Implement and enforce platform policies, terms of service, and privacy regulations.

Owner:

8. Gets the approval from the admin for his Owner account.

9. After approval, he/she can do all CRUD operation of the property in his/her account

10. He/she can change the status and availability of the property.