

Full Stack Development with MERN

Project Documentation

1. Introduction

- **Project Title:** House-Hunt: Finding Your Perfect Rent Home
- **Team Members:** Nellore Tejashwini (TL,Developer)

2. Project Overview

- **Purpose:** To simplify the rental journey for both renters and landlords by offering a centralized, verified, and user-friendly platform featuring rich search filters, direct communication, and streamlined management tools.
- **Features:** Verified properties with owner authentication
- Advanced filters: locality, budget, furnishing, availability
- Direct renter-landlord chat without brokers
- Dashboards for tracking rental status, payments, and communication
- Digital lease agreement and secure payment options

3. Architecture

- **Frontend:** Built with **React.js**, using component-based architecture. React Router handles navigation, and Axios is used for API calls. State management implemented via Context API/Redux.
- **Backend:** Powered by **Node.js** with **Express.js** framework. RESTful APIs handle CRUD operations for listings, users, and transactions. Middleware includes token authentication and error handling.

Database: MongoDB (with Mongoose ODM), storing:

- Users (renter/landlord role, profile, verification)
- Listings (property info, status, filters)
- Messages (chat records)
- Rental Transactions (agreements, payment links)

4. Setup Instructions

- **Prerequisites:** Node.js v18+
- MongoDB (local or Atlas)
- Git
- dotenv
- Recommended: Postman for backend testing
- **Installation:**
- Clone the repository:
- bash
- git clone https://github.com/your-repo/rentRight.git
- Setup backend:
- bash
- cd server
- npm install
- touch .env
- # Add environment variables (DB_URI, JWT_SECRET, etc.)

- Setup frontend:
- bash
- cd client
- npm install
-

5. Folder Structure

- **Client:**

```
client/
|
├─ public/
├─ src/
|   ├─ components/
|   ├─ pages/
|   ├─ context/
|   ├─ api/
|   └─ App.js
```

-
- **Server;**

```
server/
|
├─ controllers/
├─ models/
├─ routes/
├─ middleware/
├─ config/
└─ index.js
```

6. Running the Application

- Provide commands to start the frontend and backend servers locally.
 - **Frontend:** npm start in the client directory.
 - **Backend:** npm start in the server directory.

7. API Documentation

Endpoint	Method	Description	Auth Required
/api/auth/register	POST	Register renter/landlord	No
/api/auth/login	POST	Login user	No
/api/listings	GET	View all available listings	No
/api/listings/new	POST	Add new property listing	Yes
/api/chat/send	POST	Send a message	Yes
/api/rent/pay	POST	Initiate rental payment	Yes

8. Authentication

- Implemented via **JWT tokens**
- Auth Middleware protects private routes
- Tokens stored in localStorage and verified with `Authorization` headers
- Role-based access for landlords and renters

9. User Interface

Key Screens:

- Signup/Login
- Home feed with listings
- Filter/Search panel
- Listing details with "Verified" badge
- Chat window
- Landlord dashboard
- Renter dashboard with wishlist and application history

10. Testing

- **Manual Testing:** Frontend tested in Chrome/Firefox; backend tested with Postman.
- **Automated Testing:** Included basic unit tests (Jest) for key routes and model validation.

11. Screenshots or Demo

HOUSE HUNT - OWNER PANEL

Hi, John (Owner) | [Log Out](#)

[MY PROPERTIES](#) [ADD NEW PROPERTY](#) [BOOKING REQUESTS](#)

My Properties


Image	Property ID	Address	Type	Status	Actions
-------	-------------	---------	------	--------	---------

HOUSE HUNT - TENANT PANEL


Hi, Sarah (Tenant) | [Log Out](#)

[BROWSE PROPERTIES](#) [MY BOOKINGS](#) [FAVORITES](#)


Available Properties



Luxury Villa
123 Main St, Miami
\$2,500/month
[View Details](#)



Modern Apartment
456 Park Ave, New York
\$1,800/month
[View Details](#)



Cozy Studio
789 Beach Rd, California
\$1,200/month
[View Details](#)

HOUSE HUNT

Hi Rohan | [Log Out](#)

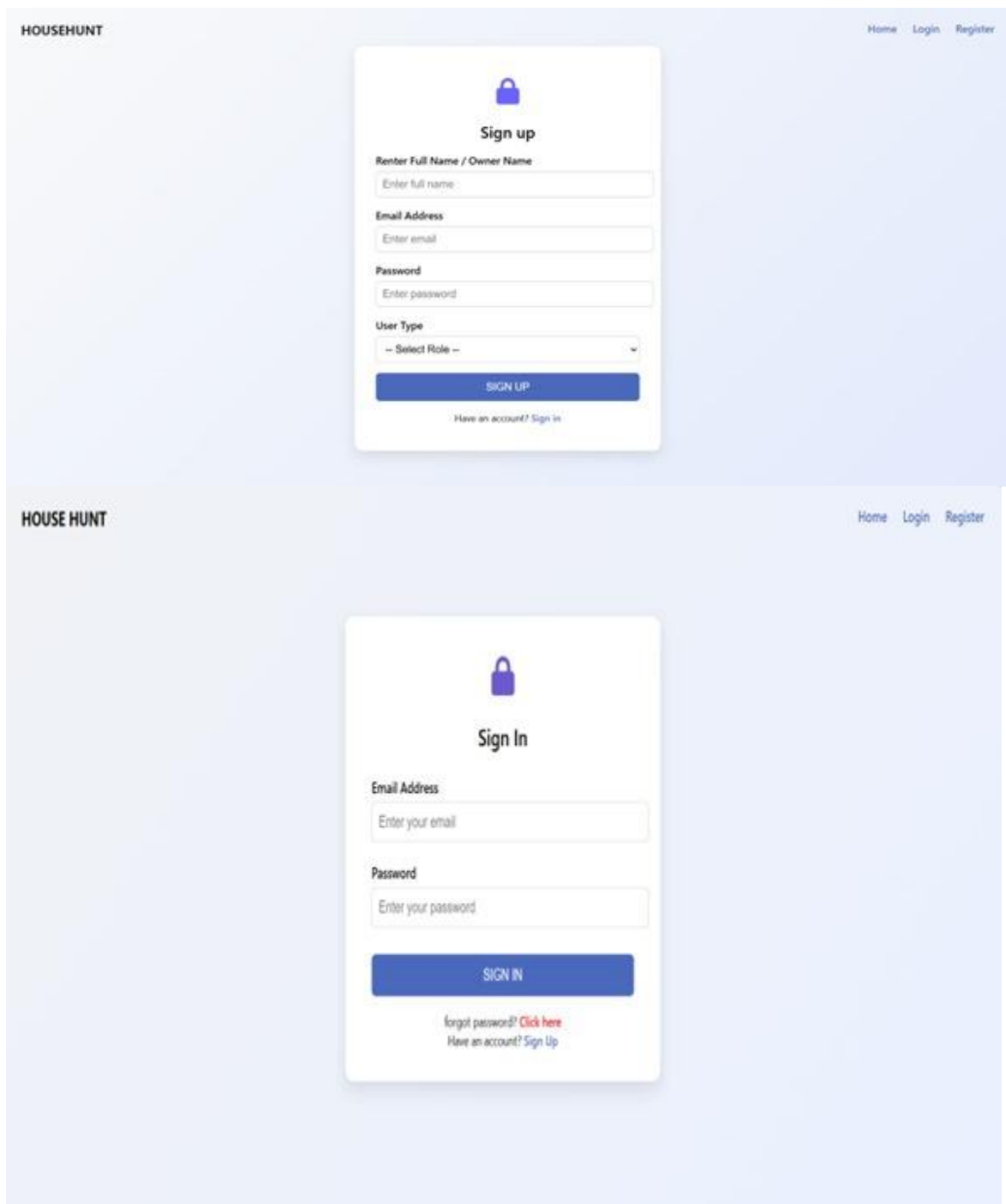
[ALL PROPERTIES](#) [BOOKING HISTORY](#)

Filter By:

All Ad Types ▼

All Types ▼

No Properties available at the moment.



12. Known Issues

- Payments require manual verification (Stripe sandbox setup pending)
- Mobile responsiveness for landlord dashboard needs improvement
- Error messages need standardization across API and UI

13. Future Enhancements

- Virtual property tours using 360° view
- Auto-matching based on user preferences using AI
- Mobile app using React Native
- Integration with verified government IDs for user verification
- Notification system (email/SMS alerts for new listings)