

Full Stack Development with MERN

Project Documentation

1. Introduction

Project Title: HouseHunt: Finding Your Perfect Rental Home

Team ID: LTVIP2025TMID54596

Team Members:

- Lobhisetti Rajesh - Full Stack Developer
- Koushik Eslavath- Frontend Developer
- Kottakota Mounika - Backend Developer
- Karnati Jyothi Swaroop Reddy - UI/UX Designer

2. Project Overview

Purpose:

HouseHunt is designed to simplify the rental property search process by offering a user-friendly platform where

Core Features:

- User registration and login
- Property listing by owners
- Search/filter by location, price, and type
- Property booking request system
- Admin dashboard for property/user management
- Fully responsive user interface

3. Architecture

Frontend:

- Built using React.js
- Routing handled by React Router
- State management via Context API- API integration with Axios

Backend:

- Developed with Node.js and Express.js
- RESTful APIs for business logic and routing- Middleware for authentication and error handling

Database:

- MongoDB with Mongoose

4. Setup Instructions

Prerequisites:

- Node.js
- MongoDB (local or cloud)
- npm or yarn
- Git
- .env with MONGO_URI and JWT_SECRET

Installation:

1. Clone the repo:

```
git clone [your-repo-link] 2.  
Install frontend dependencies:  
  cd client  
  npm install  
3. Install backend dependencies:  
  cd ../server  
  npm install  
4. Configure environment variables  
5. Start MongoDB server
```

5. Folder Structure

```
/client  
■ ■ ■ public/  
■ ■ ■ src/  
■   ■ ■ ■ components/  
■   ■ ■ ■ pages/  
■   ■ ■ ■ App.js  
■   ■ ■ ■ index.js
```

```
/server  
■ ■ ■ controllers/  
■ ■ ■ models/  
■ ■ ■ routes/  
■ ■ ■ middleware/  
■ ■ ■ config/  
■ ■ ■ index.js
```

6. Running the Application

Frontend:
cd client
npm start

Backend:
cd server
npm start

7. API Documentation

User Registration: POST /api/users/register
Login: POST /api/users/login
Add Property: POST /api/properties/add (Auth Required)
Get Properties: GET /api/properties

8. Authentication

- JWT-based authentication
- Passwords hashed using bcryptjs
- Token verification with middleware
- Authenticated users can post/book properties

9. User Interface

Key Screens:

- Home Page
- Login/Register Page
- Property Listings
- Add Property
- Admin Dashboard

10. Testing

Tools:

- Postman for API testing
- Jest for backend tests

Strategy:

- Functional endpoint testing
- UI and booking flow validation

11. Screenshots or Demo

Deployed Site: [Insert URL]

Include screenshots of:

- Home Page
- Property Details
- Admin Panel

12. Known Issues

- Filter options may not work on older browsers
- Admin panel lacks sorting
- No 2FA or OTP

13. Future Enhancements

- Payment gateway integration
- User-owner chat
- Map-based property search
- Notifications
- React Native mobile app