



# Full Stack Development with MERN

## Project Documentation: HouseHunt

---

### 1. Introduction

- **Project Title:** HouseHunt: Finding your perfect Rental Home
  - **Team Members:**
  - **Team Leader:** Ramiseti Swathi Pravallika
  - **Team Member:** Solasu Rohit Kumar
  - **Team Member:** Naga Ramya Adabala
  - **Team Member:** Sariki Hema Kanchana
  - **Team ID:** LTVIP2025TMID43868
  - **Team Size:** 4 Members
- 

### 2. Project Overview

- **Purpose:** HouseHunt is a modern rental platform that simplifies the process of renting homes. It connects tenants with landlords and allows users to explore, book, and manage rental properties online.
  - **Features:**
  - Search and browse available rental properties
  - User registration and authentication (JWT-based)
  - Admin panel for property management
  - Booking system with user history
  - Mobile-responsive UI with smooth navigation
- 

### 3. Architecture

- **Frontend:** Built with React.js using functional components and React Router for navigation. Axios is used for API calls.
  - **Backend:** Node.js with Express.js powers the backend REST API. It handles authentication, CRUD operations for users, properties, and bookings.
  - **Database:** MongoDB stores data in collections for Users, Properties, and Bookings. Mongoose is used for schema design and database interaction.
- 

### 4. Setup Instructions

- **Prerequisites:**

- Node.js >= 14.x
- MongoDB Atlas or local setup
- npm or yarn
- Git

- **Installation:**

```
git clone https://github.com/yourusername/HouseHunt.git
cd HouseHunt
```

Setup backend:

```
cd backend
npm install
touch .env
# Add MONGO_URI, JWT_SECRET, PORT
npm start
```

Setup frontend:

```
cd ../frontend
npm install
npm start
```

---

## 5. Folder Structure

```
HouseHunt/
├── frontend/
│   ├── src/
│   │   ├── components/
│   │   ├── pages/
│   │   └── App.js
├── backend/
│   ├── models/
│   ├── routes/
│   ├── controllers/
│   └── server.js
```

---

## 6. Running the Application

- **Frontend:**

```
cd frontend  
npm start
```

• **Backend:**

```
cd backend  
npm start
```

---

## 7. API Documentation

### Authentication:

- `POST /api/auth/register` - Register a new user
- `POST /api/auth/login` - Login and receive token

### Properties:

- `GET /api/properties` - List all properties
- `POST /api/properties` - Add new property (admin only)

### Bookings:

- `POST /api/bookings` - Book a property
- `GET /api/bookings` - View user's bookings

All endpoints are protected using JWT authentication.

---

## 8. Authentication

- Implemented with JWT (JSON Web Tokens)
- Secure login and registration
- Protected routes using middleware
- Token stored in browser localStorage

---

## 9. User Interface

- Responsive design using CSS Flexbox and Media Queries
- React Router used for navigation
- Key pages:
  - Home
  - Login/Register
  - Property Listings
  - Booking Summary

## 10. Testing

- Postman used for API testing
  - Manual testing for UI components across different devices
  - Edge case validation (e.g., duplicate registrations, expired tokens)
- 

## 11. Screenshots or Demo

- Demo video: [Add YouTube or Drive link here]
  - Screenshots:
    - Login Page
    - Property Listing Page
    - Booking Confirmation Page
- 

## 12. Known Issues

- No real-time availability sync
  - Booking does not validate conflicting dates
  - Limited admin dashboard features
- 

## 13. Future Enhancements

- Real-time chat between landlord and tenant
  - Integration with Razorpay/Stripe for online payments
  - Admin analytics dashboard
  - Search filters (location, price, rating)
- 

## 14. Conclusion

HouseHunt provides a scalable and user-friendly solution for managing rental properties. With enhancements, it can evolve into a complete real estate rental platform.