**🏠 HouseHunt – Real Estate Web Application**

**📌 Project Overview**

**HouseHunt** is a full-stack real estate web application developed to simplify the property renting and booking process for both property owners and potential tenants. This platform allows users to browse listed properties, view detailed information, book rentals, and manage their profiles—all through a user-friendly interface.

**🎯 Objective**

The main goal of HouseHunt is to:

* Provide a **centralized platform** for property listings.
* Enable users to **search, view, and book** properties easily.
* Allow admins to manage listings, bookings, and users securely.
* Streamline the interaction between tenants and property owners.

**💻 Tech Stack Used**

* **Frontend**: React.js, Axios, React Router, Material UI
* **Backend**: Node.js, Express.js
* **Database**: MongoDB (Mongoose ODM)
* **Authentication**: JWT (JSON Web Tokens)
* **Cloud Storage**: Cloudinary for property images
* **Dev Tools**: Postman (API Testing), Git & GitHub (Version Control)

**🔐 Key Features**

**👤 User Features**

* Register and login securely using JWT-based authentication.
* Search properties with filters like location, price, and type.
* View property details including images, features, and owner info.
* Book a property and track the status.
* Manage personal profile and booking history.

**👑 Admin Features**

* Admin dashboard to manage all users, bookings, and properties.
* Add/edit/delete property listings.
* Upload property images using Cloudinary.
* View overall analytics of platform usage (future scope).

**🧪 Modules Breakdown**

* **Authentication Module**: Secure login/registration for users and admins.
* **Property Module**: CRUD operations for managing property data.
* **Booking Module**: Book and cancel property bookings.
* **User Module**: Fetch and manage user details.
* **Error Handling & Middleware**: Centralized error responses and async handlers.

**📽️ Project Demonstration**

A video walkthrough has been created to visually demonstrate the working of the project. It covers:

* User registration & login
* Browsing properties
* Booking process
* Admin property management
* Cloudinary image upload
* API testing through Postman

🔗 **Watch Video**: [Click here to view the project demo](https://drive.google.com/file/d/1VkkMf_VUJHGmNZ5eqs_sSvT4w3VaTNxH/view?usp=drivesdk)

This video provides clarity for evaluators and team members about the project’s workflow and implementation.

**📁 Project Structure**

projectfiles/

├── backend/

│ ├── config/

│ ├── controllers/

│ ├── middleware/

│ ├── models/

│ ├── routes/

│ ├── utils/

│ └── server.js

├── frontend/

│ ├── public/

│ └── src/

│ ├── components/

│ ├── pages/

│ ├── services/

│ └── theme.js

├── videofiles/

│ └── project-demo.mp4

└── README.md

**📌 Future Scope**

* Add payment gateway integration for rental payments.
* Implement map-based property search using Google Maps API.
* Add chat system between tenant and owner.
* Notification system for booking updates.

**👩‍💻 Developer Info**

**Name | Roles**

**| -------------- | --------------------------------------------------------------------- |**

**| \*\*Bhoja Raju\*\* | Frontend Developer, Backend Developer, Team Lead, Project Coordinator |**

**| Pavani | Frontend Developer ,** **Software Tester**

**| Padma Sree | Frontend Developer ,** **Documentation Head**

**| Raju | Backend Developer ,** **Resource Manager**

**| Yugi | Backend Developer , Research Analyst**

📄 License

This project is open source and available under the MIT License.