**HOUSE HOUNT: Project Report**

**1. INTRODUCTION**

**1.1 Project Overview:**

HouseHunt is a MERN-based web platform designed to simplify the rental housing process.It allows renters to browse, filter, and request bookings for verified properties.  
Property owners can list, update, and manage their rentals through an intuitive dashboard.  
Admins oversee the platform by approving owners and managing user roles.  
The app supports real-time updates, booking status, and notifications.  
Key technologies include React.js, Node.js, Express.js, and MongoDB.  
HouseHunt ensures a secure, responsive, and user-friendly experience for all users.

**1.2 Purpose**

The purpose of HouseHunt is to create a streamlined and transparent platform for renting properties. It connects renters, owners, and admins through real-time listings, booking, and verification features. The platform simplifies property search, enhances communication, and promotes trust in the rental process.

**2. IDEATION PHASE**

**2.1 Problem Statement**

Renters face difficulty finding verified properties, and owners lack a reliable platform to manage listings. HouseHunt addresses this by offering a secure, real-time, and user-friendly rental management system.

**2.2 Empathy Map Canvas**

**User: Sarah,** a WebDeveloper searching for home.

* **Think & Feel**: Wants a smooth, transparent rental process with reliable listings that match their needs and budget.
* **See**: Modern UI with filtered property listings, high-quality images, virtual tours, and booking statuses.
* **Say & Do**: Expresses interest in budget-friendly rentals, uses filters, submits inquiry forms, and tracks bookings.
* **Hear**: Gets recommendations from peers and notifications from the platform about listings and updates.
* **Pain**: Experiences delays, scams, lack of clarity, outdated listings, and poor landlord communication.
* **Gain**: Enjoys a fast, verified, and user-friendly experience with real-time booking updates and trusted data.

**2.3 Brainstorming**

Key ideas brainstormed:

1. **Personalized Property Feed** – Tailors listings to user’s budget, location, and preferences for faster discovery.
2. **Virtual Tour Scheduler** – Lets users tour properties remotely, saving time and travel.
3. **Instant Booking with Pre-Verified Profile** – Enables seamless booking without repeated document uploads.
4. **In-App Chat with Owners** – Facilitates quick communication and negotiation with property owners.
5. **Neighborhood Explorer** – Provides safety, schools, and transport info to help users choose the right area.

**3. REQUIREMENT ANALYSIS**

**3.1 Customer Journey Map**

1. Learns about House Hunt through ads or referrals while seeking rental options.
2. Installs the app and quickly registers using a simple onboarding flow.
3. Uses filters to search for properties that match their preferences.
4. Explores listings with photos, virtual tours, and neighborhood insights.
5. Fills out an inquiry form and contacts the property owner via in-app chat.
6. Receives booking confirmation and prepares for move-in.
7. Accesses rental documents and checklists within the app.
8. Leaves a property review and earns rewards for sharing feedback.

**3.2 Solution Requirement**

**Functional Requirements:**

* User Registration & Login
* Profile Management
* Property Search & Filters
* View Property Details
* Save Favorites / Wishlist
* Booking Inquiry& Booking History
* Chat with Property Owners
* Notifications & Alerts
* Document Upload &View Rental Agreement
* Review & Rating System
* Logout Functionality

**Non-Functional Requirements:**

* Performance
* **Security**
* Fast API Performance
* **Availability**
* Usability

**3.3 Data Flow Diagram**

**Level 1 DFD:**

* User interacts with Frontend (React.js)
* Frontend sends API requests to Backend (Express.js)
* Backend processes logic and accesses MongoDB
* MongoDB responds with requested data or confirmation
* Backend sends response to Frontend for display

**3.4 Technology Stack**

1. **Frontend:** React.js, Axios, Bootstrap, Material UI, Ant Design
2. **Backend:** Node.js, Express.js, JWT, Multer, Bcrypt.js
3. **Database:** MongoDB with Mongoose
4. **Additional Tools: Nodemon, dotenv, Git/GitHub, VS Code**

**4. PROJECT DESIGN**

**4.1 Problem**

Renters face difficulty finding verified rental properties due to scattered listings, lack of transparency, and poor communication with owners.

* 1. **Solution**

House Hunt offers a centralized, user-friendly platform with verified listings, virtual tours, booking tracking, and real-time communication.

**4.3 FIT**

It directly solves user pain points by streamlining the rental process, increasing trust, and providing end-to-end rental support in one app.

**5. PROJECT PLANNING & SCHEDULING**

**5.1 Project Planning**

Project divided into 4 sprints:

1. **Sprint 1:** Set up project structure, database, and backend authentication.

2. **Sprint 2:** Build core frontend UI and integrate property search and listings.

3. **Sprint 3:** Implement booking system, admin panel, and owner dashboard.

4. **Sprint 4:** Test, optimize, deploy the app, and prepare demo and documentation.

**6. FUNCTIONAL AND PERFORMANCE TESTING**

**6.1 Performance Testing**

* **Tools Used:** Postman, JMeter, Lighthouse, Loader.io, and k6 for API, load, and UI performance testing.
* **Test Cases:** Tested API load, login response, booking requests, page load time, and database stress.
* **Outcome:** Achieved <2s API response, 90+ UI score, stable performance under 100 users, and no data loss

**7. RESULTS**

**7.1 Output Screenshots**

* Registration/Login Page
* DashBoard
* User panel for homes
* Admin Panel with Reports
* Confirmation of home

**8. ADVANTAGES & DISADVANTAGES**

**✅ Advantages:**

* User-Friendly Interface
* Advanced Search Filters
* 360° Virtual Tours & Images
* Role-Based Functionality
* Real-Time Booking & Status Tracking
* Verified Owner System
* Scalable and Modern Tech Stack (MERN)
* Notifications & Alerts

**❌ Disadvantages:**

* No Mobile App Version
* Basic Admin Panel
* Limited Security Details
* No Payment Integration
* Owner Verification is Manual

**9. CONCLUSION**

The **House Hunt App**, built using the MERN stack, provides a comprehensive and user-friendly solution for renters, property owners, and administrators to manage rental properties efficiently. With features like advanced search filters, virtual property tours, real-time booking status, and secure user authentication, the platform simplifies the rental process end-to-end. By leveraging modern technologies and a scalable architecture, House Hunt ensures accessibility, transparency, and ease of use, making it a reliable platform for the online house rental market.

**10. FUTURE SCOPE**

* Develop a mobile app version for Android and iOS to enhance accessibility.
* Integrate secure online payment and rent transaction features.
* Implement AI-based property recommendations and smart notifications.
* Add multilingual support and chatbot assistance for better user engagement.

**11. APPENDIX**

|  |  |  |
| --- | --- | --- |
| **S.No.** | **Item** | **Details** |
| 1 | Project Title | |  | | --- | |  |  |  | | --- | | House Hunt – Finding Your Perfect Rental Home | |
| 2 | Team ID | LTVIP2025TMID54764 |
| 3 | Team Members | K.Sravani  K.Sahitya  D.sudheer  G.Rushda |
| 4 | Technology Stack | MongoDB, Express.js, React.js, Node.js (MERN), Bootstrap, Material UI, Axios, JWT |
| 5 | |  | | --- | |  |  |  | | --- | | Repository Link | | [GitHub-HouseHunt](https://github.com/sravani-4415/HouseHunt) |
| 6 | Project Demo | [Video Demo](https://drive.google.com/file/d/1C73sk9dvyc8cXugtmDV_b1FxCiUQuwPV/view) |
| 7 | Development Tools  &  Testing Tools | Postman (API), JMeter (Load), Lighthouse (Frontend Performance),Visual Studio Code, GitHub, Postman |