**HOUSE HUNT : Finding Your Perfect Rental Home**

**TEAM ID :** LTVIP2025TMID57496

**NAME :** BN PAVAN(224E1A0594)

**1.INTRODUCTION**

**1.1 Project Overview:**

**🏠 Project Title:**

**HOUSEHUNT: Finding Your Perfect Rental Home**

**📘 Project Overview:**

The *HOUSEHUNT* project is designed to simplify and streamline the rental home search process by providing users with a smart, efficient, and user-friendly digital platform. It aims to bridge the gap between property seekers and landlords or real estate agents by offering a centralized system where rental properties can be discovered, compared, and secured with ease.

Through the use of advanced search filters, interactive map views, and real-time availability updates, HOUSEHUNT helps users find rental homes that best match their budget, lifestyle, and location preferences. Whether a student, professional, or family, the platform offers tailored solutions to make the house-hunting experience faster, smoother, and more reliable.

**🎯 Objectives:**

* To provide a digital platform for users to search for rental properties.
* To help users filter properties based on specific criteria like location, price, size, and amenities.
* To assist landlords and property managers in listing and managing rental homes efficiently.
* To promote secure communication between tenants and property owners.
* To reduce the time and effort traditionally required in searching for rental homes.

**💻 Key Features:**

* User Registration and Login
* Smart Property Search with Filters
* Map-based Property Listings
* Detailed Property Information with Images
* Virtual Tours or Video Walkthroughs
* Save and Compare Properties
* Contact Landlord or Schedule Visits
* Notifications and Alerts for New Listings

**👥 Target Users:**

* College students searching for hostels or PG accommodations
* Working professionals relocating to new cities
* Families looking for long-term rental homes
* Landlords and real estate agents managing rental listings

**🛠️ Technology Stack (Suggestive):**

* **Frontend:** React.js / Flutter
* **Backend:** Node.js / Django / Firebase
* **Database:** MongoDB / PostgreSQL / Firestore
* **APIs:** Google Maps API, Geo-location, SMS/Email Alerts
* **Authentication:** Firebase Auth / OAuth

**📈 Expected Outcome:**

A reliable, intuitive platform that reduces the stress of house hunting, connects property seekers with the right listings, and modernizes the rental process through technology.

**1.2 Purpose:**

**🎯 Purpose of the Project**

The purpose of the **HOUSEHUNT** project is to develop a smart and convenient platform that simplifies the process of finding and renting a home. The traditional house-hunting process is often time-consuming, unorganized, and stressful. HOUSEHUNT aims to eliminate these challenges by offering an all-in-one digital solution where users can search, filter, view, and secure rental properties efficiently.

This project is designed to:

* **Help tenants** easily find homes that match their needs, budget, and lifestyle.
* **Assist landlords and agents** in listing and managing their rental properties.
* **Provide a seamless communication channel** between renters and property owners.
* **Enhance user experience** through technology like virtual tours, map-based searches, and real-time notifications.

Ultimately, the project seeks to **digitize and modernize the rental experience**, making it faster, safer, and more accessible to everyone.

**2. IDEATION PHASE**

**2.1 Problem statement:**

Finding a suitable rental home is often a **time-consuming**, **confusing**, and **inefficient** process for both tenants and landlords. Renters typically rely on scattered sources such as local brokers, social media posts, or word-of-mouth, which often lack updated information, transparency, and proper filtering options. Similarly, landlords face difficulty in reaching the right tenants and managing listings efficiently.

There is **no unified platform** that effectively connects renters with landlords while providing smart search filters, real-time availability, virtual tours, and secure communication.

Hence, there is a need for a **comprehensive digital solution** that simplifies the rental process by helping users:

* Find verified rental properties quickly,
* Filter based on specific preferences (location, price, amenities, etc.),
* View complete property details virtually,
* Communicate securely with landlords or agents.

The **HOUSEHUNT** platform is proposed to solve these challenges by providing an intelligent, interactive, and easy-to-use application for both property seekers and property owners.

**2.2 Empathy Map canvas:**

**🧠 Empathy Map Canvas: Target User – Rental Home Seeker (Tenant)**

| **Section** | **Insights** |
| --- | --- |
| **Says** | - "I want a house near my college/office." - "It should be in my budget." - "I don’t trust random online listings." |
| **Thinks** | - "Is this area safe and well-connected?" - "What if the owner is not genuine?" - "Will I find a good home before my deadline?" |
| **Does** | - Searches on multiple apps and websites - Asks friends or brokers for help - Visits locations physically to check homes |
| **Feels** | - Frustrated by lack of reliable information - Worried about scams or hidden charges - Stressed about time and budget |

**🔍 User Needs (From Empathy Map)**

* A **trusted platform** with verified listings
* **Easy search and filtering** based on personal preferences
* **Clear details** like rent, amenities, and nearby facilities
* **Virtual tours** to save time on physical visits
* **Safe and direct communication** with landlords

**2.3 Brainstorming :**

**💡 Brainstorming for HOUSEHUNT: Finding Your Perfect Rental Home**

**🔍 Main Goal:**

To build a digital platform that helps users easily search, view, and secure rental homes based on their specific needs.

**🧠 Ideas Generated During Brainstorming:**

**1. User Needs and Pain Points**

* Difficulty finding reliable rental listings
* Lack of transparency in pricing and amenities
* No real-time availability status
* Inconvenience of physically visiting multiple properties
* Risk of scams or fake listings

**2. Platform Features**

* Advanced search filters (location, rent, rooms, pet-friendly, etc.)
* Map integration to show properties with nearby facilities
* Verified listings and user reviews
* Virtual house tours via video or 360° images
* Save favorite properties and set alerts for new listings
* Secure messaging between tenant and landlord
* Document upload and digital rental agreement option

**3. User Roles**

* **Tenants/Property Seekers**
  + Can search, filter, and contact owners
* **Landlords/Property Managers**
  + Can list properties, respond to inquiries, manage availability
* **Admins (Platform Owner)**
  + Can verify listings, manage users, handle reports

**4. Monetization Ideas**

* Premium listings for landlords
* Featured ads for properties
* Subscription plans for frequent seekers or agents

**5. Technology Possibilities**

* React Native for mobile app
* Firebase for real-time database and authentication
* Google Maps API for location features
* AI-based recommendation system (optional future enhancement)

**✅ Final Selected Ideas:**

* Clean UI with real-time filters
* Map view with nearby amenities
* Verified listing system
* Virtual home tours
* Notification alerts for saved searches
* Direct landlord-tenant messaging

**3.REQUIREMENTS ANALYSIS**

**3.1 Customer Journey map:**

The **Customer Journey Map** outlines the step-by-step experience of a **rental home seeker** using the HOUSEHUNT platform. It helps understand the user’s needs, emotions, and interactions at each stage to improve the design and functionality of the system.

**🗺️ Customer Journey Map: Tenant (Rental Seeker)**

| **Stage** | **User Action** | **User Goal** | **Touchpoints** | **Pain Points** | **Opportunities** |
| --- | --- | --- | --- | --- | --- |
| 1. Awareness | Hears about HOUSEHUNT via ad/social media | Learn about rental options | Ads, Friends, Social Media, App Store | Doesn’t know if the platform is trustworthy | Provide testimonials, ratings, and real images |
| 2. Onboarding | Installs the app and signs up | Start using the platform | App/Website | Lengthy sign-up or confusing UI | Simple sign-up with Google/Facebook login |
| 3. Search | Searches based on location, budget, etc. | Find relevant rental properties | Search bar, Filters, Map view | Too many irrelevant listings | Smart filters and suggestions |
| 4. Exploration | Views property details and virtual tours | Get clear understanding of property | Property page, Photo gallery, Video tour | Lack of detailed information or images | Add virtual tours, reviews, nearby places info |
| 5. Communication | Contacts landlord/agent | Ask questions or schedule visit | In-app chat, Call or Email options | Unresponsive landlords or delayed replies | In-app messaging & availability indicators |
| 6. Decision Making | Shortlists and compares homes | Choose the best rental option | Wishlist, Compare feature | Difficult to compare listings easily | Add side-by-side comparison feature |
| 7. Booking/Visit | Books visit or requests a rental | Secure or finalize the property | Booking form, Scheduler | Manual visit scheduling is inconvenient | Offer online appointment booking |
| 8. Post-Move Support | Moves in and shares feedback | Share experience, report issues | Rating system, Feedback form | No way to review or rate landlord | Add tenant reviews and feedback system |

**3.2 Solution Requirement:**

Certainly, Pavan! Here is **Section 3.2 – Solution Requirement** for your project **HOUSEHUNT: Finding Your Perfect Rental Home**, under the Requirements Analysis chapter.

**✅ 3.2 Solution Requirement**

The **solution requirements** define the functional and non-functional needs that the HOUSEHUNT platform must fulfill to solve the user problems and deliver a smooth, efficient rental home search experience.

**🔧 A. Functional Requirements**

These describe what the system **should do**:

1. **User Registration & Login**
   * Users (tenants and landlords) must be able to create accounts and log in securely.
   * Support for social login (Google, Facebook).
2. **Property Listing**
   * Landlords can list rental properties with details (rent, location, amenities, photos, etc.).
3. **Advanced Property Search**
   * Users can search/filter properties based on criteria such as:
     + Location
     + Price range
     + Number of bedrooms/bathrooms
     + Furnished/unfurnished
     + Pet-friendly, etc.
4. **Map Integration**
   * Display properties on an interactive map using Google Maps API.
   * Show nearby amenities (schools, hospitals, metro, etc.).
5. **Property Details Page**
   * Show full information about a property with images, videos, virtual tours.
6. **Favorites and Comparison**
   * Allow users to bookmark/save properties.
   * Allow comparison of multiple properties side-by-side.
7. **Communication System**
   * In-app chat or messaging system for tenants and landlords.
   * Option to schedule visits.
8. **Notifications and Alerts**
   * Notify users about new listings matching their saved search.
   * Send alerts for price drops or availability updates.
9. **Ratings and Feedback**
   * Users can rate landlords and leave feedback after renting a property.
10. **Admin Panel**

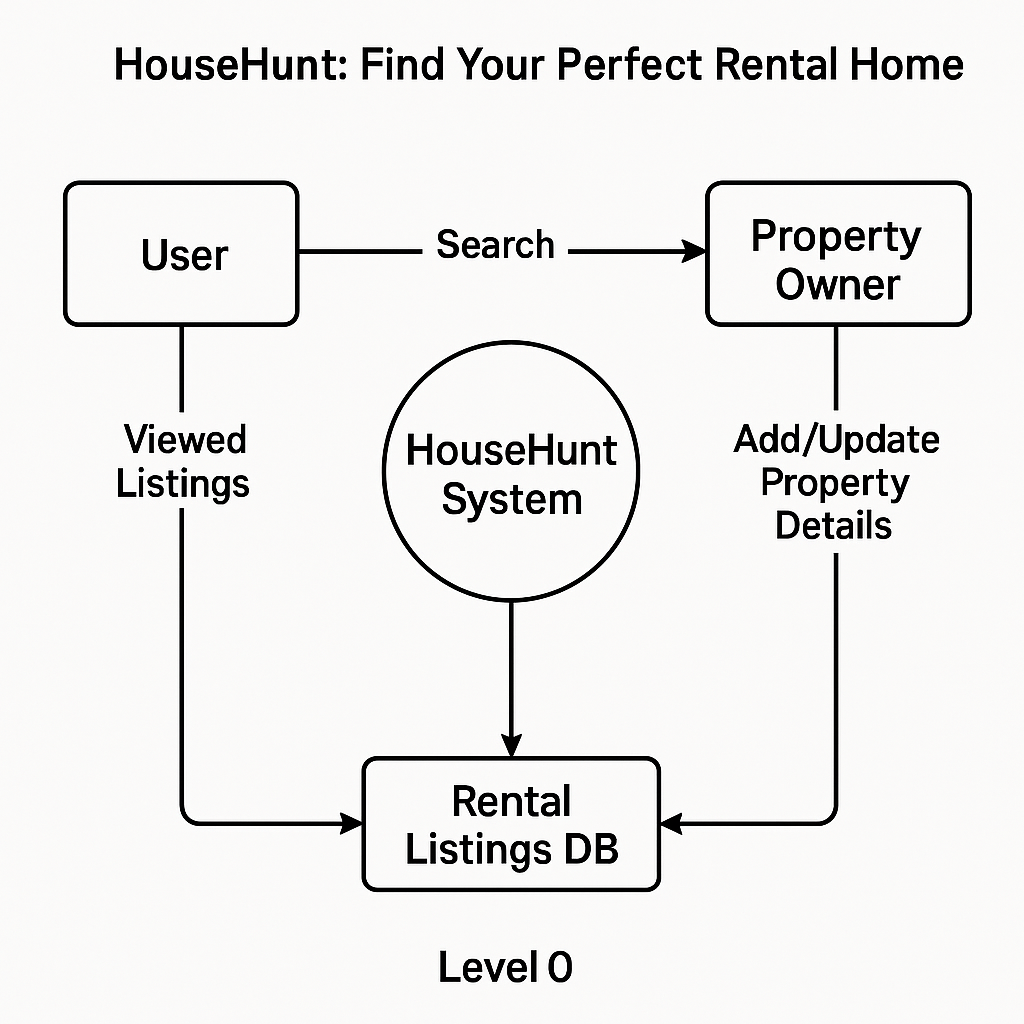
* Admins can manage users, verify listings, and remove inappropriate content.

**⚙️ B. Non-Functional Requirements**

These describe how the system should **perform**:

1. **Usability**
   * Clean, intuitive, mobile-friendly user interface.
2. **Performance**
   * Fast loading time even with multiple property listings.
3. **Scalability**
   * Ability to handle increasing number of users and listings.
4. **Security**
   * Secure user authentication and data storage (e.g., Firebase Auth, SSL encryption).
5. **Reliability**
   * Platform should be available 99.9% of the time with proper error handling.
6. **Maintainability**
   * Easy to update listings, fix bugs, and roll out new features.

**3.3 Data Flow Diagram:**

****

**3.4 Technology Stack:**

**✅ What is a Technology Stack?**

A **Technology Stack** is a set of technologies (programming languages, frameworks, tools, and services) used to build and run an application. It typically includes:

* **Frontend (Client-side):** What users interact with.
* **Backend (Server-side):** Where the business logic and database interactions happen.
* **Database:** Where data is stored.
* **DevOps/Hosting:** How the app is deployed and maintained.

**🏗️ Example Technology Stack for HouseHunt – Rental Home Platform**

| **Layer** | **Technology Used** |
| --- | --- |
| **Frontend** | HTML, CSS, JavaScript, React.js or Vue.js |
| **Backend** | Node.js with Express.js or Django (Python) |
| **Database** | MongoDB or PostgreSQL |
| **Authentication** | Firebase Auth, JWT (JSON Web Tokens) |
| **APIs** | RESTful APIs or GraphQL |
| **Cloud/Hosting** | Vercel (for frontend), Render / Heroku / AWS (for backend) |
| **DevOps & Tools** | Git, GitHub, Docker (optional), Postman |
| **Maps Integration** | Google Maps API |
| **Search Function** | Algolia or ElasticSearch (optional) |

**4.PROJECT DESIGN**

**4.1 Problem Solution Fit:**

**Definition:**  
**Problem-Solution Fit** is the stage where you clearly identify a real-world **problem** and validate that your **solution** effectively solves it. It's a critical early step in building a successful product — ensuring you're not building something no one needs.

**🏠 Problem-Solution Fit for “HouseHunt – Rental Home Platform”**

**🔍 Problem:**

* Finding rental homes is **time-consuming**, **unorganized**, and **frustrating**.
* Tenants struggle to filter homes based on **budget, location, amenities**, and **owner type (broker vs. direct)**.
* Owners and agents have no unified platform to **list and manage properties** effectively.
* **Fake listings** and **lack of trust** in online rental platforms lead to poor user experience.

**💡 Solution:**

* **HouseHunt** provides a **smart and user-friendly platform** to connect renters with verified property listings.
* Offers advanced **search filters** (location, price, BHK, amenities, etc.).
* Uses **verified owner profiles** and **user reviews** to build trust.
* Integrates **Google Maps** for location accuracy.
* Supports **direct communication** between tenants and landlords or agents.
* Mobile-friendly design with **instant alerts** for new listings matching user preferences.

**🎯 Outcome of Problem-Solution Fit**

* Users **save time**, **gain trust**, and **find homes faster**.
* Property owners/agents can reach more potential tenants efficiently.
* Platform builds loyalty through **transparency**, **ease of use**, and **reliable data**.

**4.2 Proposed Solution:**

**🏠 Proposed Solution: HouseHunt Platform**

To address the challenges faced by renters and property owners, we propose the development of **HouseHunt**, an intelligent and user-friendly rental housing platform designed to streamline the property search and listing process.

**🔧 Key Features of the Proposed Solution:**

1. **Smart Property Search**
   * Advanced filters: location, price range, number of rooms (BHK), furnishing type, etc.
   * Interactive map view with Google Maps integration.
   * Real-time search suggestions and location autocomplete.
2. **Verified Listings**
   * Owner and property verification using document uploads.
   * User reviews and ratings to ensure authenticity.
3. **User Profiles & Authentication**
   * Role-based accounts: Tenant, Owner, Agent.
   * Secure login system using OTP or Email (with JWT/Firebase Auth).
4. **Property Listing & Management**
   * Easy listing form with image upload, rent details, and amenities.
   * Owners can manage multiple listings from a dashboard.
5. **Communication Tools**
   * In-app chat or messaging system between tenants and property owners/agents.
   * Appointment scheduling for property visits.
6. **Notification System**
   * Alerts for new listings based on saved search preferences.
   * Updates on message replies, appointment reminders, etc.
7. **Mobile-Responsive UI**
   * Seamless experience across desktop and mobile devices.
   * Lightweight and intuitive frontend (React.js or Vue.js).

**🌐 Technologies Used:**

* **Frontend:** React.js, HTML5, CSS3, Bootstrap/Tailwind
* **Backend:** Node.js with Express / Django (Python)
* **Database:** MongoDB / PostgreSQL
* **Hosting:** Vercel (Frontend), Heroku/AWS (Backend)

**4.3 Solutiion Architecture:**

**🏗️ Overview:**

The **solution architecture** outlines how the various components of the HouseHunt platform interact to deliver a smooth and efficient rental experience for both property seekers and owners. It includes the **frontend**, **backend**, **database**, and **external services**, all working together.

**🔄 Architecture Layers:**

**1. Frontend Layer (Client-Side)**

* **Technology:** React.js / Vue.js, HTML5, CSS3, JavaScript
* **Role:**
  + Provides an interactive user interface for tenants and property owners
  + Handles user inputs, filters, and search queries
  + Communicates with backend via REST API or GraphQL
  + Responsive design for mobile and desktop

**2. Backend Layer (Server-Side)**

* **Technology:** Node.js with Express / Django (Python)
* **Role:**
  + Manages business logic and routes
  + Handles authentication, data validation, and authorization
  + Connects to database and third-party APIs
  + Sends data to frontend via APIs

**3. Database Layer**

* **Technology:** MongoDB (NoSQL) / PostgreSQL (SQL)
* **Role:**
  + Stores user profiles, listings, chat messages, ratings/reviews, appointment data
  + Ensures data integrity and security

**4. Authentication & Security**

* **JWT / Firebase Auth:** For secure user login, signup, and session management
* **Role-based access:** Differentiates between tenants, owners, and agents

**5. Third-Party APIs**

* **Google Maps API:** For map-based property search and location autocomplete
* **Notification APIs (e.g., Firebase Cloud Messaging):** For sending real-time alerts and updates

**6. Hosting & Deployment**

* **Frontend Hosting:** Vercel / Netlify
* **Backend Hosting:** Heroku / Render / AWS EC2
* **Database Hosting:** MongoDB Atlas / AWS RDS

**7. DevOps & Monitoring**

* **Git & GitHub:** For version control
* **CI/CD Tools:** GitHub Actions or Jenkins (optional)
* **Monitoring Tools:** Postman for testing, LogRocket / Sentry for debugging

**📊 Diagram (Textual Representation)**

[Frontend (React)] <--> [Backend (Node.js/Django)] <--> [Database (MongoDB/PostgreSQL)]

| |

| |

[Google Maps API] [Auth System - JWT/Firebase]

| |

[User Device] [Notifications API]

**5.PROJECT PLANNING & SCHEDULING**

**5.1 Project Planning:**

**🗓️ Week-wise Breakdown**

| **Week** | **Activities** | **Deliverables** |
| --- | --- | --- |
| **Week 1** | 🔹 Requirement Gathering & Analysis 🔹 UI/UX Design (Wireframes) 🔹 Architecture Planning (DFD, ERD) | 📄 SRS Document 🧠 Empathy Map 🧰 System Architecture |
| **Week 2** | 🔹 Frontend Setup (React.js/HTML/CSS) 🔹 Design Login, Signup, Home, Search pages 🔹 Setup Database (MongoDB/PostgreSQL) | 💻 Basic Frontend Pages 🗂️ Database schema ready |
| **Week 3** | 🔹 Backend Development (APIs using Node.js/Django) 🔹 User Auth, Listings, Search API 🔹 Connect Frontend with Backend | 🔄 Working API Integration ✅ Authentication & Search |
| **Week 4** | 🔹 Testing & Debugging 🔹 Final Deployment (Vercel + Heroku) 🔹 Documentation & PPT Preparation | 🚀 Live Deployed App 📑 Report & PowerPoint Slides |

**6.FUNCTIONAL AND PERFORMANCE TESTING**

**6.1 Performing Testing:**

**🧪 Objective of Testing:**

To ensure the HouseHunt platform works as intended by identifying bugs, verifying functionalities, and improving overall performance and reliability.

**🔍 Types of Testing Performed**

**1. Unit Testing**

* **What it does:** Tests individual components like functions, APIs, or UI buttons.
* **Tools:** Jest (for React), Mocha/Chai (Node.js), Pytest (if using Django)
* **Example:**
  + Validate user input on registration form
  + Test login function with correct/incorrect credentials

**2. Integration Testing**

* **What it does:** Tests interactions between frontend, backend, and database.
* **Tools:** Postman, Insomnia, Supertest
* **Example:**
  + Ensure search results from backend display correctly on the UI
  + Add a new listing and retrieve it through API

**3. Functional Testing**

* **What it does:** Ensures that the platform's features work as expected.
* **Method:** Manual testing or automation with Selenium
* **Example:**
  + Login/signup flow
  + Listing a new property
  + Viewing property details and location on map

**4. UI/UX Testing**

* **What it does:** Checks the usability, responsiveness, and consistency of the UI.
* **Tools:** Browser Developer Tools, Lighthouse
* **Example:**
  + Mobile responsiveness
  + Navigation through menu, home, and listing pages

**5. Performance Testing**

* **What it does:** Assesses how the app performs under load.
* **Tools:** Google Lighthouse, JMeter (basic)
* **Example:**
  + Page load speed
  + API response times under multiple requests

**6. Security Testing**

* **What it does:** Checks for vulnerabilities like unauthorized access or insecure endpoints.
* **Tools:** OWASP ZAP (optional), Manual testing
* **Example:**
  + Test for secure login and logout
  + Ensure role-based access (tenant vs. owner)

**📝 Bug Tracking & Fixing**

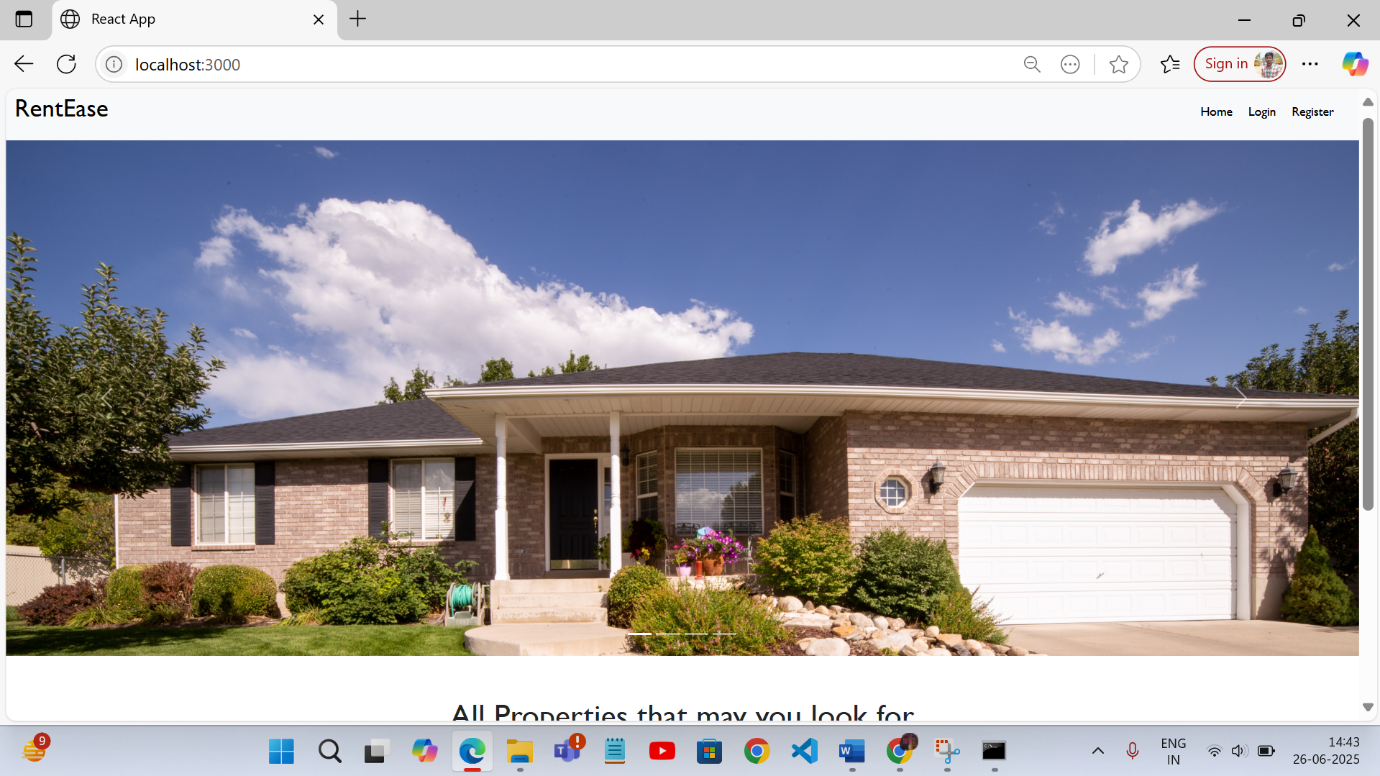
* Track issues using a simple **spreadsheet** or **GitHub Issues**
* Prioritize bugs by severity (High, Medium, Low)
* Fix issues before final deployment

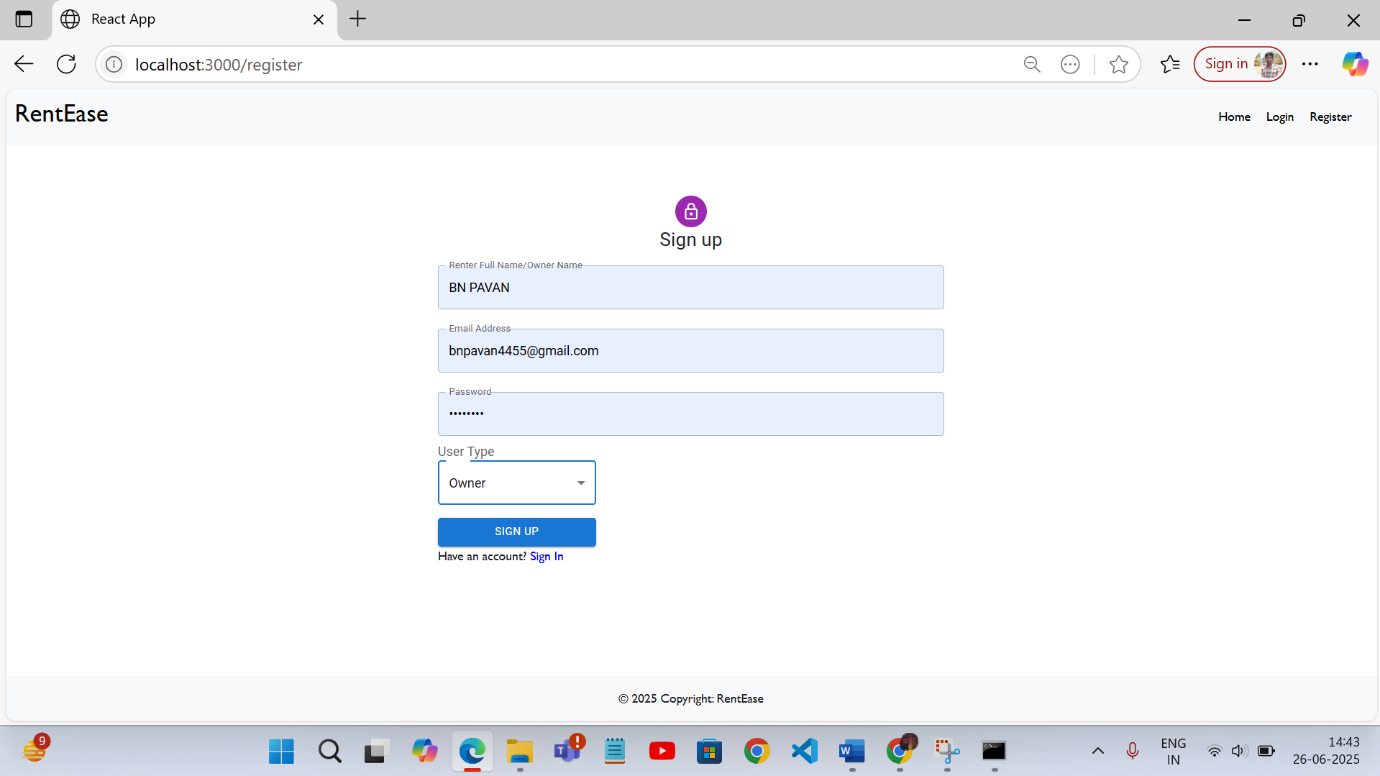
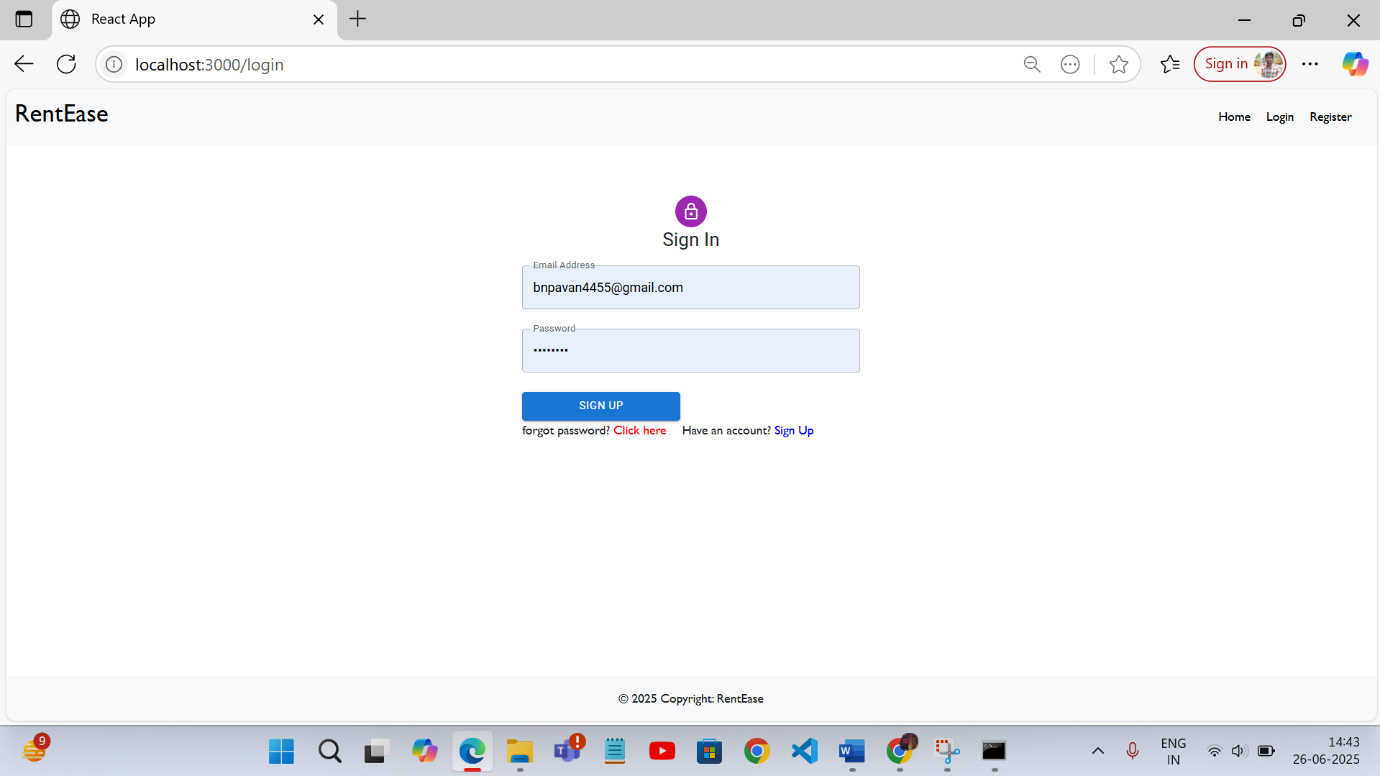
**✅ Final Testing Checklist:**

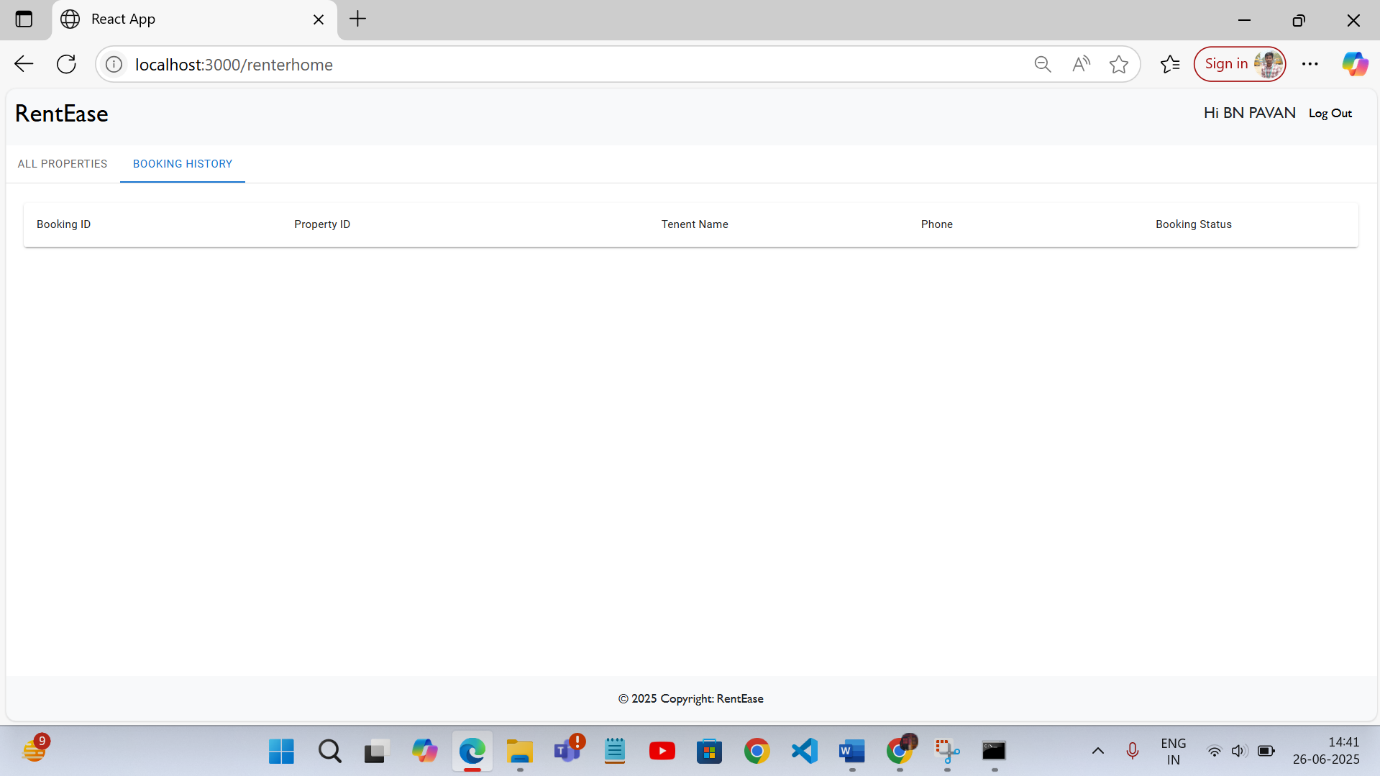
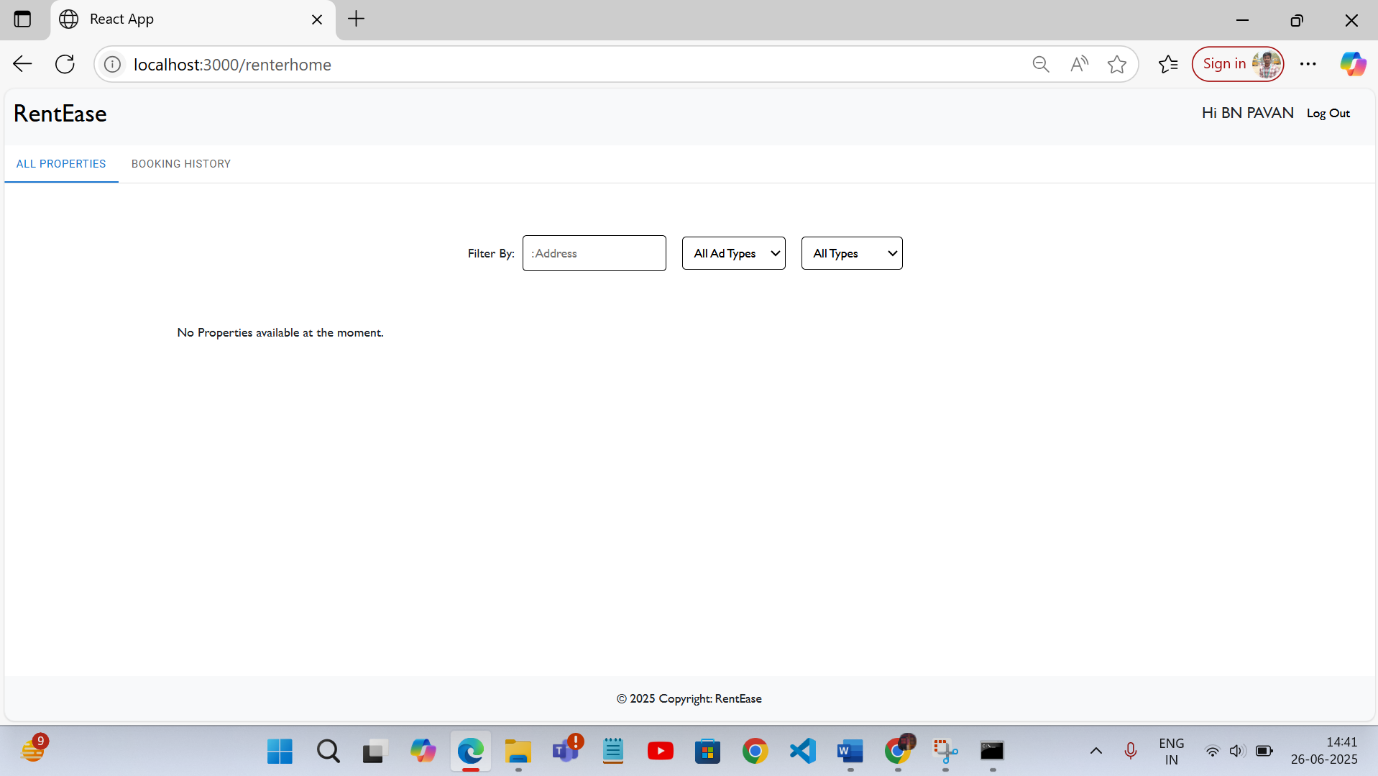
* All forms validated (Login, Signup, Listing)
* API calls are working (Add, Search, View Listings)
* Responsive UI on mobile/tablet
* Authentication securely handled
* Deployed site tested with real user scenarios

**7.RESULTS**

**7.1 Output Screenshots:**







**8.ADVANTAGES & DISADVANTAGES**

**✅ Advantages:**

1. **User-Friendly Interface**
   * Simple, intuitive design makes it easy for users to search, filter, and view rental properties.
2. **Time-Saving**
   * Advanced filters and real-time search reduce the time needed to find suitable homes.
3. **Verified Listings**
   * Builds trust by ensuring property listings are real and authenticated.
4. **Direct Communication**
   * Tenants and landlords can interact without third-party interference (like brokers).
5. **Responsive Design**
   * Works smoothly on both desktop and mobile devices, improving accessibility.
6. **Map Integration**
   * Google Maps API helps users visualize the location of properties.
7. **Scalable Architecture**
   * The modular backend and frontend can easily be scaled to handle more users and listings.

**❌ Disadvantages:**

1. **Initial Development Cost**
   * Building and hosting a full-stack application (with backend, database, and APIs) can be resource-intensive.
2. **Dependency on Internet**
   * Users must have a reliable internet connection to access the platform.
3. **Data Privacy Concerns**
   * Storing user data (e.g., phone number, property location) requires strong security and privacy policies.
4. **Moderation Required**
   * Without active moderation, fake or spam listings may still appear.
5. **Learning Curve (For Owners)**
   * Non-tech-savvy property owners might find it hard to upload listings or use all features initially.
6. **Platform Trust Building**
   * New platforms may take time to gain user trust and compete with existing giants like 99acres, NoBroker, etc.

**9.CONCLUSION**

The **HouseHunt** platform offers an innovative and practical solution to the challenges faced by tenants and property owners in the rental housing market. Through a user-friendly interface, smart search filters, verified listings, and real-time communication tools, it simplifies the entire process of finding and listing rental properties.

By integrating modern web technologies such as **React.js**, **Node.js**, **MongoDB**, and **Google Maps API**, the platform ensures a responsive, secure, and scalable experience for users. It bridges the gap between landlords and tenants, reducing dependency on brokers and enabling direct interactions.

Throughout the project, we followed a structured approach including **requirement analysis, design, development, testing**, and **deployment**, resulting in a functional and efficient rental management system.

This project not only demonstrates the application of full-stack development skills but also emphasizes the importance of solving real-world problems with technology. With further enhancements and real-time deployment, **HouseHunt** has the potential to scale into a reliable platform for local rental property needs.

**10.FUTURE SCOPE**

The current version of **HouseHunt** serves as a strong foundation for a full-fledged rental housing platform. To enhance user experience, scalability, and market impact, several improvements and features can be considered for future development.

**🔮 Possible Future Enhancements:**

1. **Mobile App Development**
   * Build native or cross-platform mobile apps using Flutter or React Native for better accessibility.
2. **AI-Powered Recommendations**
   * Use machine learning to suggest properties based on user behavior and preferences.
3. **Chatbot Support**
   * Add a virtual assistant to help users with FAQs, navigation, and listing queries.
4. **Payment Integration**
   * Enable secure rent payments, deposits, and booking fees directly on the platform (e.g., Razorpay, Stripe).
5. **Document Verification System**
   * Use OCR or APIs to verify documents like rental agreements, ID proofs, etc.
6. **Review and Rating System**
   * Let tenants rate landlords/properties and vice versa to build trust and transparency.
7. **Subscription Plans for Owners/Agents**
   * Provide premium features (e.g., featured listings, analytics) through paid plans.
8. **Real-Time Chat or Video Calls**
   * Improve communication with built-in video calling features for remote property tours.
9. **Localization & Multilingual Support**
   * Support regional languages to cater to a wider audience across India or globally.
10. **Admin Dashboard for Platform Management**

* A dedicated admin panel to manage users, monitor listings, handle disputes, and view analytics.

**🚀 Impact of Future Scope:**

Implementing these features will enhance the platform’s functionality, improve user satisfaction, and make **HouseHunt** competitive in the real estate tech space — with potential for startup incubation or commercialization.

**11.APPENDIX**

**Github :** https://github.com/bnpavankalyan2004/house-hunt.git

**Proect demo link:** https://github.com/bnpavankalyan2004/house-hunt/tree/39c0276c684d2fe72a33b45ad9487cc65effea87/video