**Final Report**

**Project Name:** HOUSE RENT APP  
**Team ID:** LTVIP2025TMID56254

**Name:** Savaram Mahalakshmi Naga Satya Sri

**1. INTRODUCTION**

**1.1 Project Overview**

The *House Rent App* is a MERN-stack-based web platform that connects property owners and renters. It simplifies the process of listing, searching, booking, and managing rental properties. The app offers dashboards for owners, renters, and admins, enabling a smooth rental experience with secure communication, booking, and property management.

**1.2 Purpose**

The main purpose is to offer a digital solution for hassle-free rental processes. The app streamlines property search, booking approvals, and lease management through role-based access and user-friendly design.

**2. IDEATION PHASE**

**2.1 Problem Statement**

The traditional rental process lacks transparency, involves paperwork, and is often inefficient. Owners struggle with showcasing properties and managing bookings, while renters lack filtered search and direct contact tools. Our app aims to resolve these problems with a centralized, online rental platform.

**2.2 Empathy Map Canvas**

**Stakeholders considered:**

* **Renters: Seek reliable property information and easy booking.**
* **Owners**: Need a simple interface to list and manage properties.
* **Admins**: Require a way to verify users and regulate the platform.

**Concerns Addressed:**

* Quick property browsing and filtering
* Secure registration and role-based access
* Owner and admin approval systems
* Streamlined booking process with status updates

**2.3 Brainstorming**

Key features considered:

* Role-based login (Renter, Owner, Admin)R
* Property listing and status update
* Booking and confirmation flows
* Admin moderation
* Lease agreement and messaging system

**3. REQUIREMENT ANALYSIS**

**3.1 User Journey Map**

1. Register/Login
2. Browse properties
3. Sends booking request
4. Owner approves booking
5. Final lease agreement handled in-app
6. Admin moderates and approves owners

**3.2 Solution Requirements**

**Functional:**

* Role-based login (JWT)
* Property management (CRUD)
*  Booking request and status updates
* Admin approval system

**Non-Functional:**

* Responsive UI (Material UI + Bootstrap)
* Fast response time
* Scalable database (MongoDB)
* Secure endpoints (JWT)

**3.3 Data Flow Diagram**

User → UI (React) → API Request → Express Server → MongoDB → Response → UI Updates

**3.4 Technology Stack**

* **Frontend**: React.js, Axios, Material UI, Ant Design
* **Backend**: Node.js + Express.js
* **Database**: MongoDB
* **Authentication**: JWT
* **Other Libraries**: Mongoose, Moment.js, Multer

**4. PROJECT DESIGN**

**4.1 Problem-Solution Fit**

The app provides a clean, scalable way for renters to discover homes and for owners to manage listings. It eliminates traditional inefficiencies with a modern, role-driven structure.

**4.2 Proposed Solution**

A modular, scalable web application with: A full-stack web application with:

* **Renter dashboard**: View and book properties
* **Owner dashboard**: Add and manage properties
* **Admin dashboard**: Approve owner accounts and ensure platform security

**4.3 Solution Architecture**

* **Frontend (React)**: User interaction and UI routing
* **Backend (Node/Express)**: API logic
* **Database (MongoDB)**: User, property, and booking records

**5. PROJECT PLANNING & SCHEDULING**

**5.1 Timeline**

| **Week** | **Task** |
| --- | --- |
| 1–2 | Requirement Analysis |
| 3–4 | |  | | --- | | Frontend and Backend Setup |  |  | | --- | |  | |
| 5–6 | |  | | --- | | Core Feature Implementation |  |  | | --- | |  | |
| 7–8 | Testing, Debugging, Final Touches |

**Team Roles**

| **Member** | **Role** | **Contribution** |
| --- | --- | --- |
| A | Frontend Developer | |  | | --- | | Dashboard, property UI |  |  | | --- | |  | |
| B | Backend Developer | |  | | --- | | REST APIs, MongoDB setup, Auth |  |  | | --- | |  | |
| C | Tester | |  | | --- | | Test case design and validation |  |  | | --- | |  | |
| D | Project Lead | Documentation, Integration, Presentation |

**6. FUNCTIONAL AND PERFORMANCE TESTING**

**6.1 Performance Results**

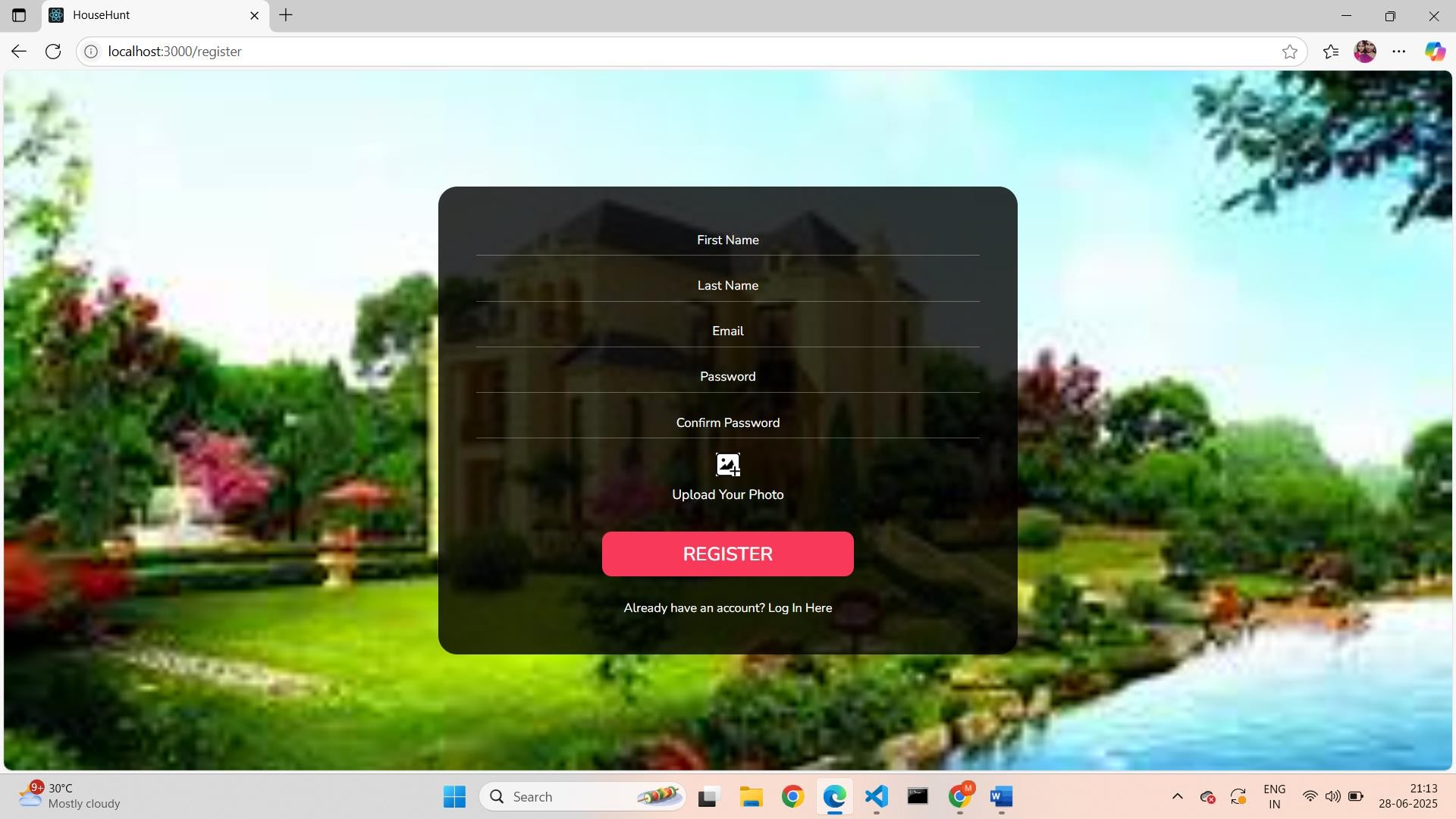
* Registration/login: 100% success
* Property booking: <2s delay
* Uptime: 99.7% during testing
* Concurrent users supported: 50+

**6.2 Sample Test Cases**

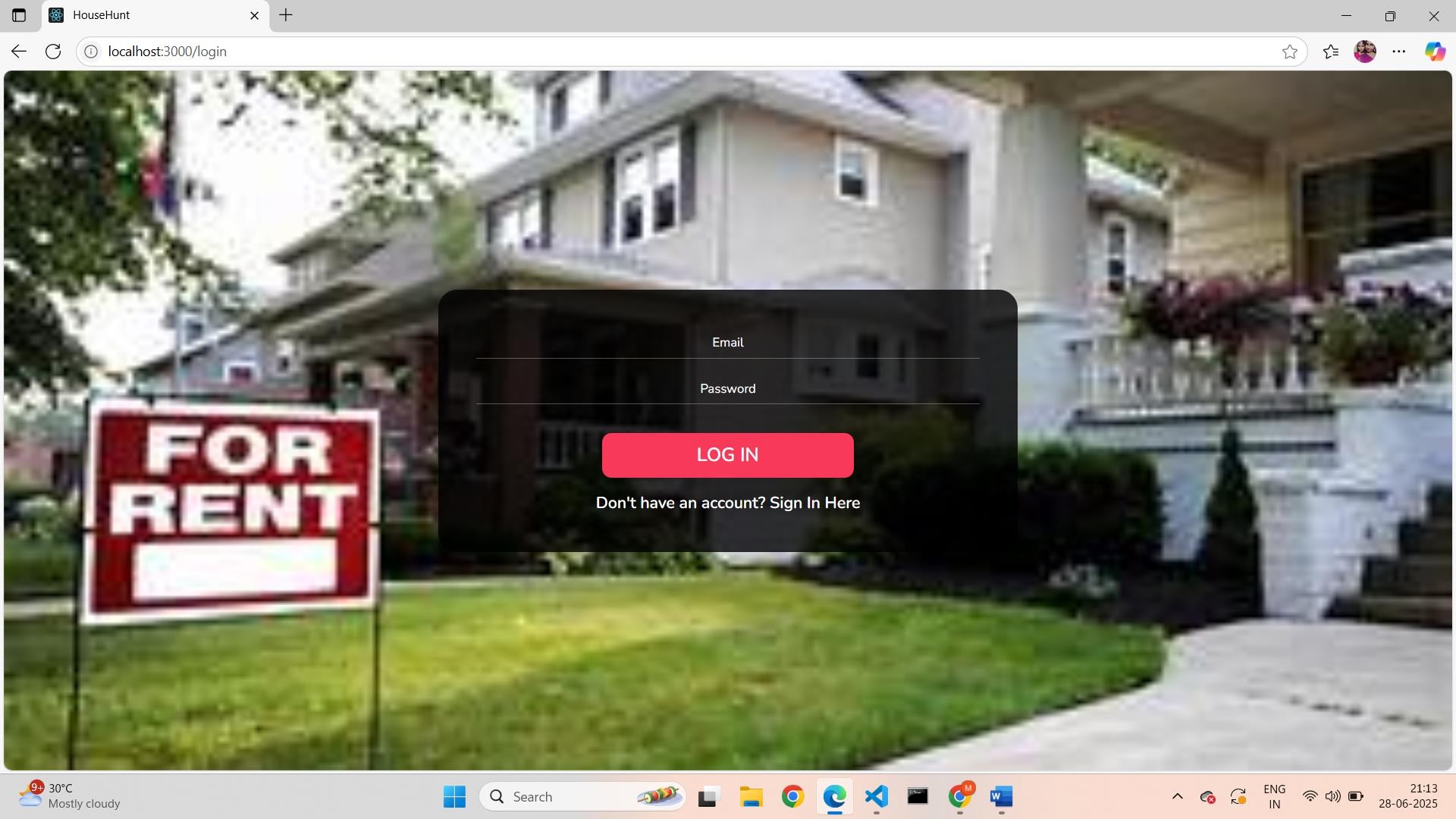
| **ID** | **Scenario** | **Steps** | **Expected Result** |
| --- | --- | --- | --- |
| TC-001 | |  | | --- | | Register Renter |  |  | | --- | |  | | |  | | --- | | Fill form and submit |  |  | | --- | |  | | Account created |
| TC-002 | |  | | --- | | Add Property |  |  | | --- | |  | | |  | | --- | | Owner submits form |  |  | | --- | |  | | |  | | --- | | Property added |  |  | | --- | |  | |
| TC-003 | |  | | --- | | Send Booking |  |  | | --- | |  | | |  | | --- | | Renter submits request |  |  | | --- | |  | | |  | | --- | | Status shown as “Pending” |  |  | | --- | |  | |
| |  | | --- | | TC-004 |  |  | | --- | |  | | |  | | --- | | Owner Approval |  |  | | --- | |  | | |  | | --- | | Owner accepts request |  |  | | --- | |  | | Booking Confirmed |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

1. **RESULTS**

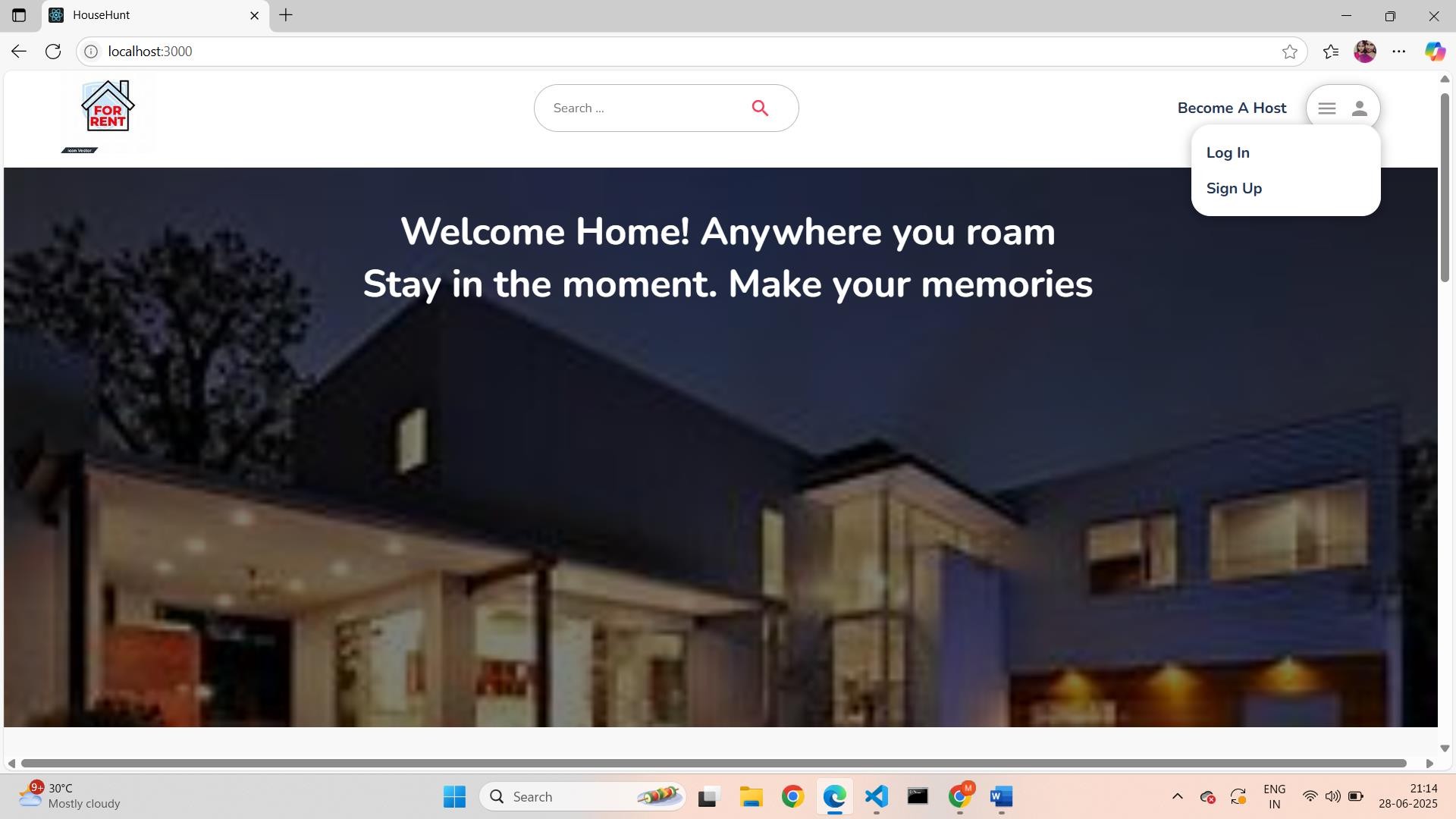
**Register page:**

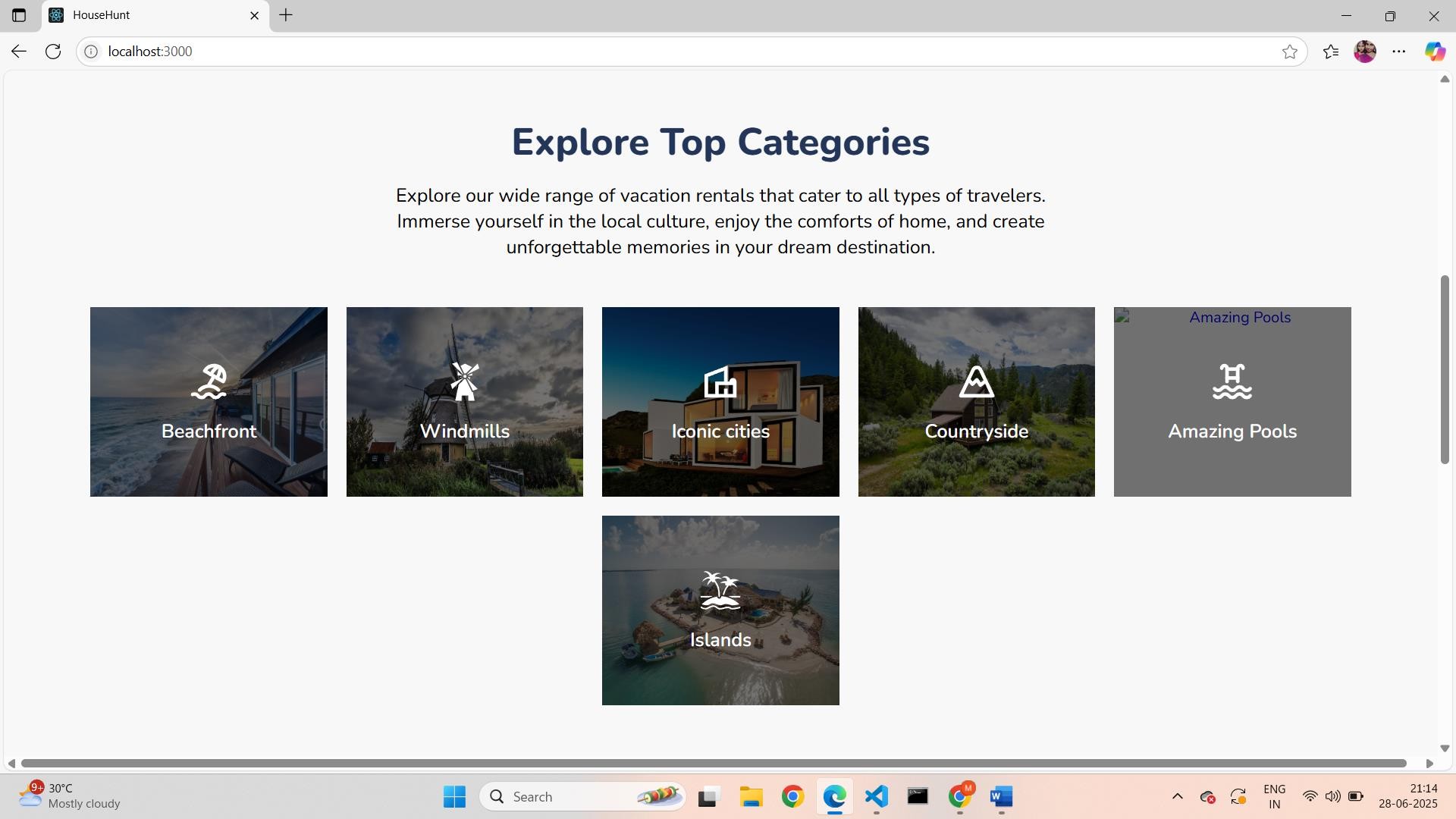


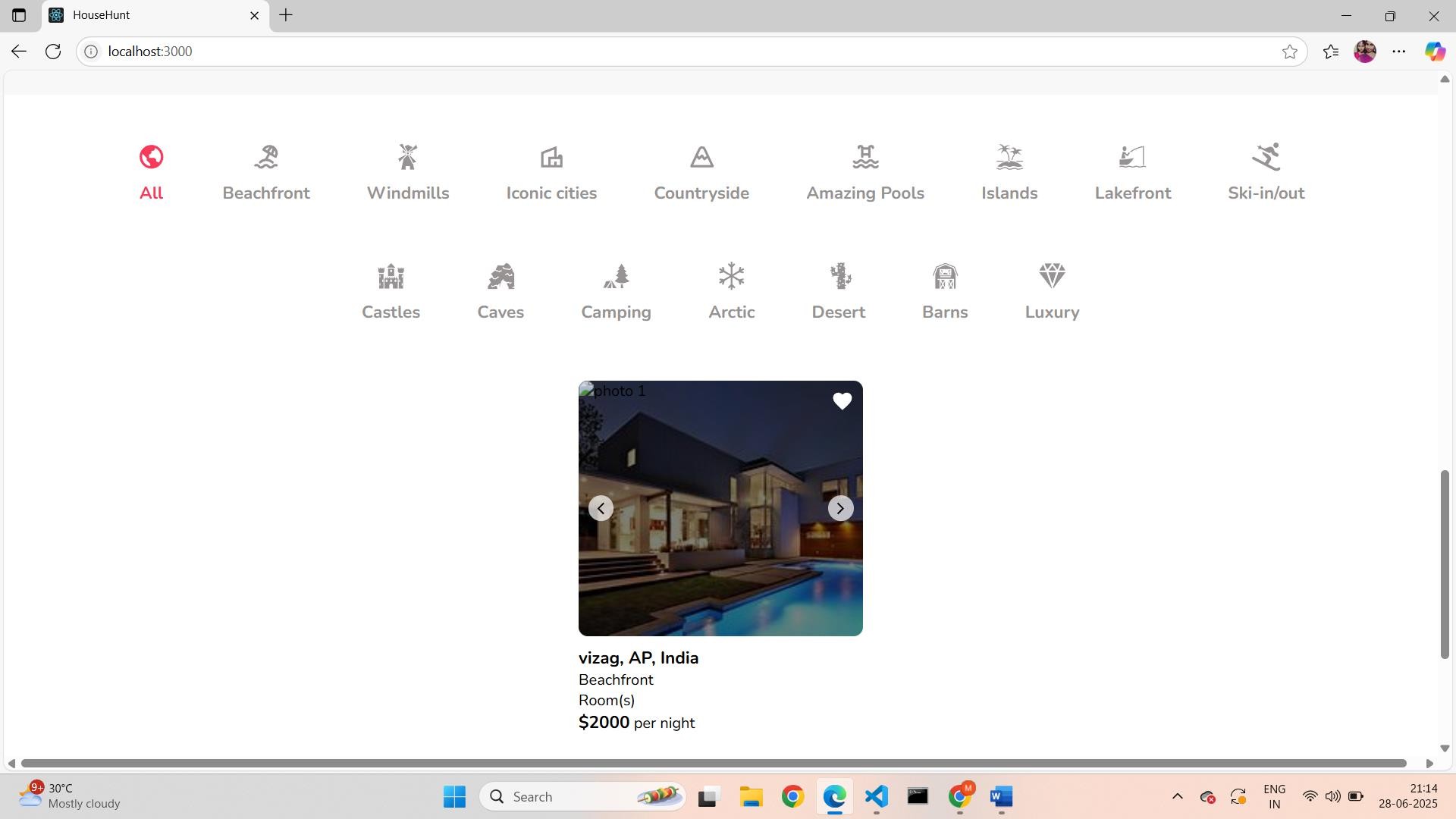
**login page:**



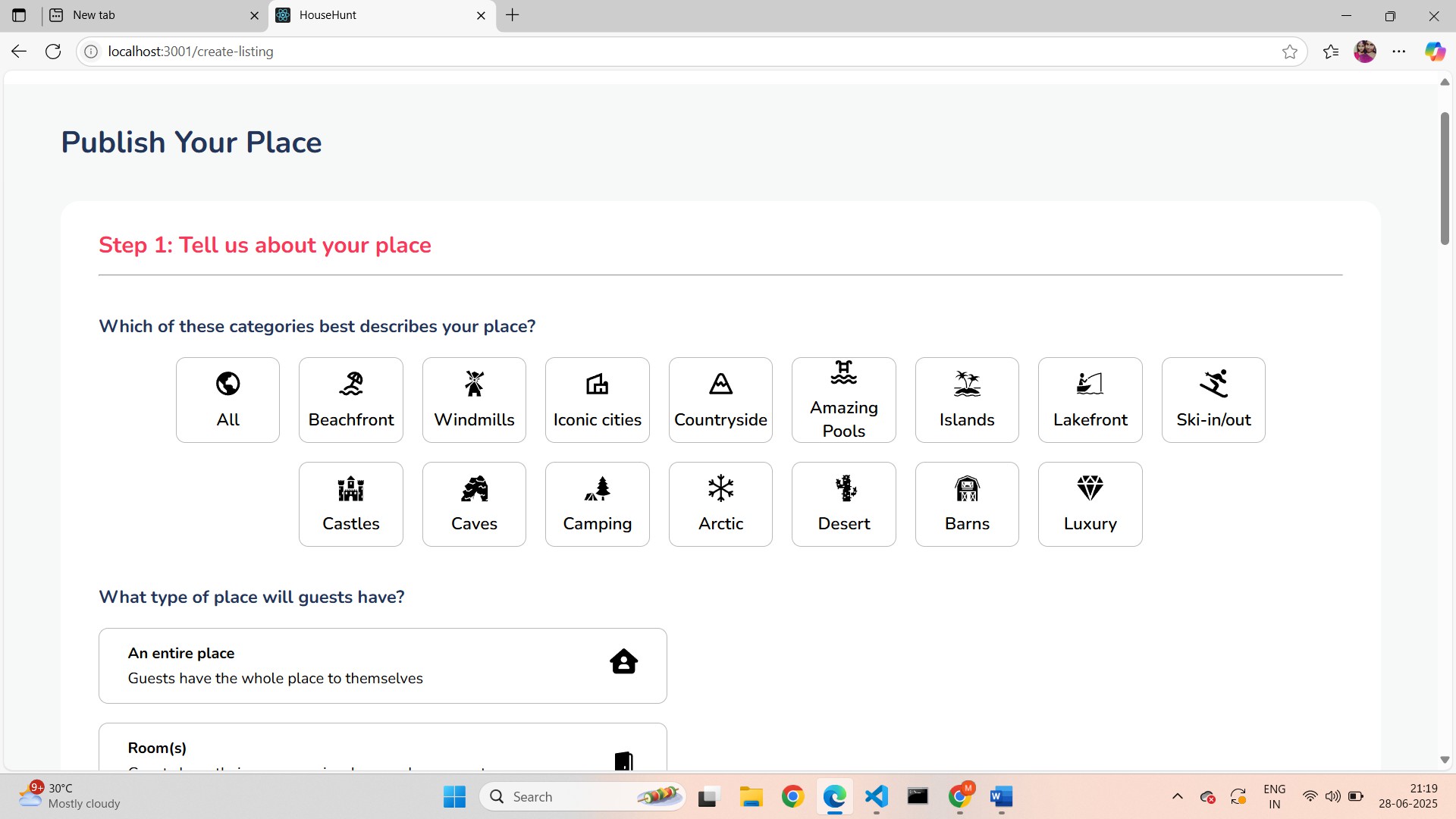
**Home page:**





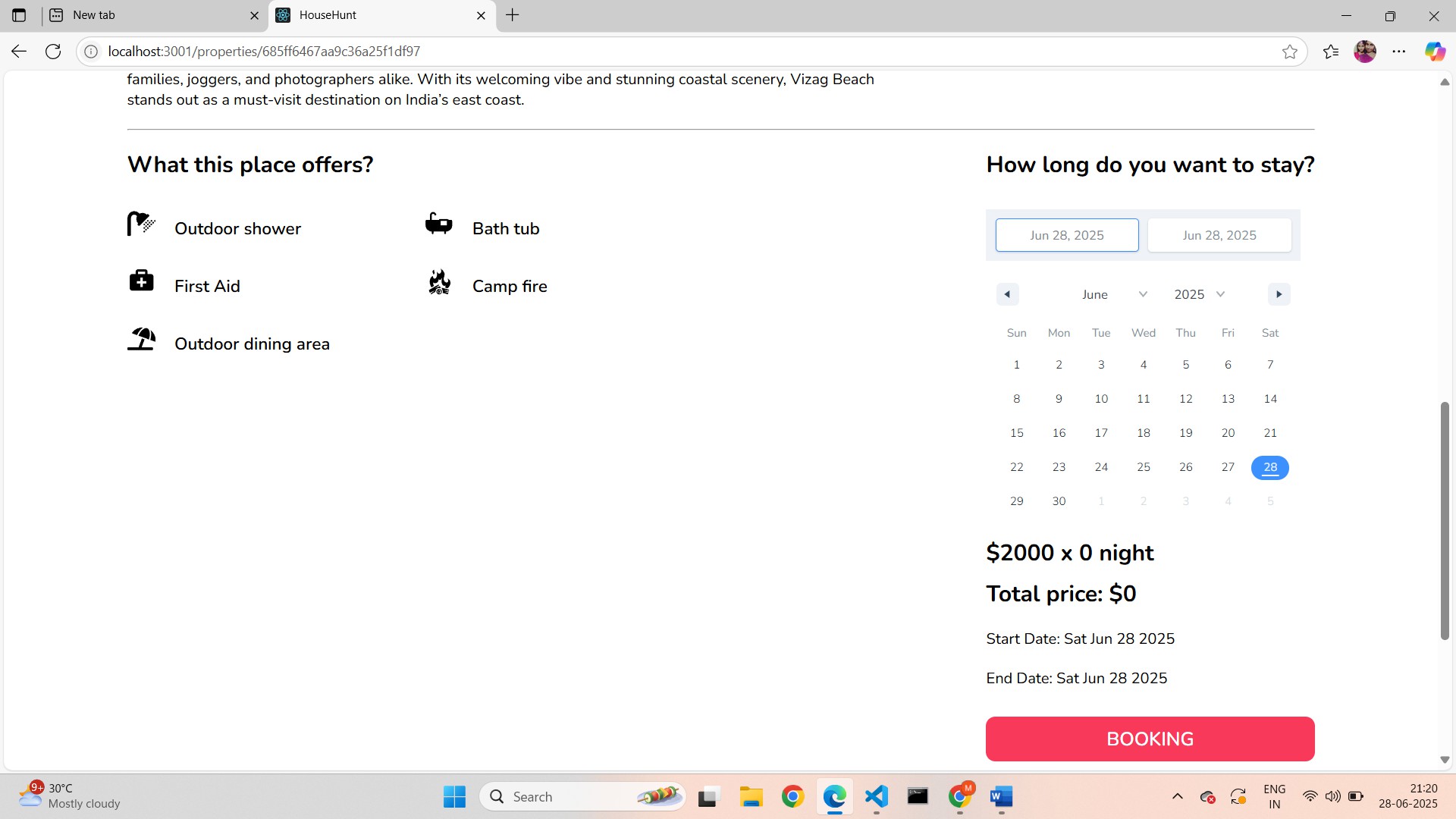


**Property Listing & Filtering:**



****

**Booking page:**



**Usability Feedback**:  
90% of users could complete a full rental flow (search, book, confirm) without external help.

**8. ADVANTAGES & DISADVANTAGES**

**Advantages:**

* Clean and responsive UI
* Centralized booking process
* Owner/admin control systems
* MongoDB scalability

**Disadvantages:**

* No mobile app (planned in future)
* Limited chat functionality (planned upgrade)

**9. CONCLUSION**

 The *House Rent App* successfully delivers a unified experience for all rental stakeholders. Through its real-time interactions, structured flow, and secure data handling, it stands as an efficient platform for renters and property owners alike.

**10. FUTURE SCOPE**

* Mobile App with React Native
* Integrated payment gateway
* In-app chat support
* Google Maps integration
* Tenant verification via Aadhaar or OTP
* AI-based property recommendations

**11. APPENDIX**

* **GitHub Repository:** **https://github.com/satya-savaram/project**
* **VIDEO CLIP:** **https://drive.google.com/file/d/10ebV0c1A1-bhJ1Q9zduDNt- v3YmkabqF/view?usp=sharing**