



**INSTITUTE FOR ADVANCED COMPUTING AND
SOFTWARE DEVELOPMENT, AKURDI, PUNE**

“StayHub - Find & Book Your Perfect Stay”

PG-DAC Feb 2025

Submitted By:

Group No: 70

Roll No.	Name of Student
252126	Aman Patel
252156	Jaiendra Rajput

Mr. Prithviraj Shinde
Project Guide

Mr. Prashant Deshpande
Centre Coordinator

ABSTRACT

Stayhub is a comprehensive platform designed to streamline accommodation and Paying Guest booking management. It allows users to efficiently search, book, and manage their Paying Guest while providing hosts and businesses with an integrated platform for managing their properties. The platform uses modern technologies like React and Redux for a responsive front-end, and Spring Boot for a robust backend API.

Stayhub is a robust and scalable application designed to streamline the booking process and facilitate seamless hospitality operations. The project aims to address the challenges faced by hosts and travelers in managing property listings and online bookings through an integrated platform built with React for the front-end and Spring Boot for the backend.

ACKNOWLEDGEMENT

A project usually falls short of its expectation unless aided and guided by the right persons at the right time. We avail this opportunity to express our deep sense of gratitude towards **Mr. Prashant Deshpande** (Centre coordinator, IACSD Akurdi), **Mr. Narendra Pawar** (Course Coordinator, IACSD Akurdi) and **Mr. Prithviraj Shinde** (Our Project Guide and Faculty Member, IACSD Akurdi).

We are deeply indebted and grateful to them for their guidance, encouragement and deep concern for our project. Without their critical evaluation and suggestions at every stage of the project, this project could never have reached its present form. Last but not the least we thank the entire faculty and the staff members of institute for advanced computing and Software development, akurdi, pune for their support.

Aman patel (250241220024)
Jaiendra Rajput (250241220080)

Table of Contents

Sr.No	Description	Page No.
1	Introduction	1
2	SRS	6
3	Diagrams	12
3.1	ER Diagram	15
3.2	Use Case Diagram	16
3.3	Data Flow Diagram	17
3.4	Activity Diagram	18
3.5	Class Diagram	21
3.6	Sequence Diagram	23
4	Database Design	24
5	Snapshots	27
6	Conclusion	28
7	References	29

1. INTRODUCTION

StayHub is a cutting-edge online platform designed to simplify and enhance the process of finding and booking PG accommodations. Utilizing modern web technologies and best practices in software development, StayHub aims to create a seamless and efficient marketplace where tenants can discover, compare, and book rental stays with ease, while property owners can effectively list and manage their accommodations.

Technologies Used:

- **React:** A powerful front-end library used to build a dynamic and responsive user interface.
- **Java EE:** Provides a robust and scalable architecture for enterprise-level applications.
- **Spring Boot:** Facilitates the creation of standalone, production-grade Spring-based applications, simplifying backend development.
- **MySQL:** A reliable and widely-used relational database management system to handle data storage and retrieval.

Core Features:

- **User Roles and Authentication:**
 - **User:** Can register and log in to access personalized features. Once logged in, users can search for PG accommodations across various locations using advanced filters such as minimum and maximum rent and location-based search. They can explore top-rated stays and book accommodations seamlessly.
 - **Owner:** Allows property owners to register, log in, and manage their PG listings. Owners can edit PG details, update availability, and track bookings, ensuring their properties remain up to date for potential tenants.

- **JWT Token Authentication:**

Ensures secure user authentication and authorization. JSON Web Tokens (JWT) are used to manage sessions and validate user access to various features of the platform.

- **PG Discovery:**

Once logged in, users can explore top-rated PG accommodations based on their preferred location. Advanced filters, including minimum and maximum rent, help users find the most suitable stays.

- **Booking Functionality:**

Users can seamlessly search for PGs, view detailed listings, and proceed with the booking process in a user-friendly manner.

How It Works:

1. **Registration and Login:**

Users and property owners can create accounts and log in using a secure authentication system powered by Spring Boot Security.

2. **PG Discovery:**

Users can search for PG accommodations based on location using filters like minimum and maximum rent. A map is available for visual reference (non- functional, using API).

3. **Booking Management:**

Users can explore PG details, including pictures, rent per month, amenities, services, and food menu. They can select check-in and check-out dates and proceed with reservations.

4. **Property Management:**

Owners can add, edit, or delete PG listings. They can also manage booking requests by approving or rejecting them.

Access Level:

- Limited access based on assigned permissions.
Can view and interact with specific features (e.g., order placement, product browsing) but cannot alter system settings.
No access to sensitive system configurations, financial data, or user management functionalities.

Role-Based Access Control (RBAC)

The system implements Role-Based Access Control (RBAC) using **Spring Security**. This ensures that users only have access to the functionalities they need based on their roles. For instance:

- Admins have unrestricted access to all resources and can manage both the system and its users.
- Users have limited access, often restricted to viewing and interacting with inventory items, placing orders, and managing their profiles.

Security Considerations

Both roles are secured through robust authentication mechanisms provided by Spring Security.

This includes:

- **Authentication:** Verifying the identity of users before granting access to the system.
- **Authorization:** Ensuring that users can only access the resources permitted by their role.
- **Session Management:** Secure handling of user sessions to prevent unauthorized access.

1.1 Purpose

The purpose of the Stayhub system is to provide a comprehensive and efficient platform for managing and booking accommodations in a hospitality environment. The system is designed to streamline various operations related to property management, booking processing, and guest interactions, ultimately enhancing the overall productivity and profitability of the business and improving the guest experience.

1.2 Scope

The scope of the **Stayhub** project encompasses the design, development, deployment, and maintenance of a comprehensive accommodation and booking platform that integrates modern technologies to meet the needs of property owners, guests, and hospitality businesses. This scope is defined by the functionalities the system will provide, the technologies used, and the stakeholders involved..

1.3 Objective of StayHub

The objectives of the **Stayhub** project outline the primary goals that the system aims to achieve, providing clear guidance on its design, implementation, and operational focus. These objectives ensure that the system effectively supports the needs of its users while driving business efficiency and growth.

1. Efficient Room Management
2. Streamlined Booking Processes
3. Enhanced User Experience
4. Robust Security and Access Control
5. Scalable and Flexible Architecture
6. Comprehensive Reporting and Analytics
7. Cost Optimization
8. Enhanced Guest Experience
9. Compliance and Audit Readiness

1.4 Functionalities Provided by StayHub

1. User Management

- User Registration and Login:
 - Guests and hosts can create accounts, log in, and manage their profiles.
 - Secure authentication and authorization using Spring Security.
- Role-Based Access Control:
 - Different user roles (e.g., Guest, Host, Admin) will have specific permissions.
 - Admins can create and manage these roles and assign them to users.
- Profile Management:
 - Users can update their personal information, such as name, email, and contact details.

2. Guest Management

- Guest Profiles:
 - Maintain detailed profiles for each guest, including booking history, preferences, and contact information .
- Loyalty Programs and Discounts:
 - Implementation of loyalty programs or discounts to reward frequent customers and encourage repeat business

3. Security and Compliance

- Data Encryption:
 - Encryption of sensitive data, such as payment information and personal details, to protect against unauthorized access

4. Guest Experience Enhancement

- Search and Filtering:
 - Advanced search and filtering options for guests to find properties based on various criteria such as location, dates, price, or amenities.
- Wishlist and Saved Properties:
 - Features allowing guests to save properties for future bookings.
- Responsive Design:
 - A user interface that adapts to different devices, ensuring a seamless experience on desktops, tablets, and smartphones.

SOFTWARE REQUIREMENT SPECIFICATION

The functional requirements for **Stayhub** outline the specific features and capabilities that the system must provide to meet the needs of its users. These requirements are essential for guiding the development process and ensuring that the final product aligns with the business objectives.

2.1.1 Login Functionality

1. User Login

- Objective: Provide users with seamless access to StayHub's accommodation booking platform.
- Process:
 - Users enter their email and password on the login page.
 - Authentication is secured using Spring Boot Security and JWT tokens.
 - Upon successful login, users are redirected to the landing page, where they can browse available PG listings and manage their profiles.

2. Owner Login

- Objective: Enable property owners to manage PG listings and track bookings.
- Process:
 - Owners log in using registered credentials.
 - JWT-based authentication ensures secure access to the owner dashboard.
 - Upon login, owners can add, update, and remove PG listings, manage pricing, and view booking requests.

2.1.2 User Experience

1. Exploring & Viewing PG Listings

- Objective: Help users find accommodations tailored to their needs.
- Process:
 - Users can search PGs by location and apply filters like price range, amenities, and availability.
 - Each PG listing displays images, rent per month, facilities, and food menu details.
 - A map interface shows PG locations, improving navigation and user experience.

2. Booking a PG

- Objective: Provide a simple and efficient reservation system.
- Process:
 - Users select a PG, specify check-in/check-out dates, and proceed with the booking.
 - Clicking "Reserve Now" redirects users to a payment confirmation page.
 - After successful payment, a "Show Booking" button appears, allowing users to view their confirmed reservations.

3. User Dashboard & Profile Management

- Objective: Offer users a centralized space for managing bookings and profile details.
- Process:
 - Users can navigate to their dashboard from the landing page.
 - The profile section allows updating personal details.
 - The booking history section displays past and upcoming reservations with PG names, booking dates, and statuses.

2.1.3 Owner Functions

1. Managing PG Listings

- Objective: Allow property owners to maintain and update accommodation details.
- Process:
 - Owners log in to access their dashboard.
 - They can add new PG listings, update existing details like rent, amenities, and photos, or remove outdated listings.
 - Listings are automatically updated in the user search results.

2. Handling Bookings & Customer Interactions

- Objective: Enable owners to efficiently manage reservations and communicate with tenants.
- Process:
 - Owners can view incoming booking requests in their dashboard.
 - They have the option to approve or decline bookings based on availability.
 - Confirmed bookings reflect in the user's dashboard, ensuring smooth interaction between tenants and owners.

Summary

StayHub is a user-centric accommodation platform that bridges the gap between tenants and property owners. Users can search, book, and manage PG accommodations effortlessly, while owners can oversee listings and bookings through an intuitive dashboard. The platform ensures secure transactions, streamlined user experience, and efficient property management.

2.2 NON-FUNCTIONAL REQUIREMENTS

2.2.1 Interface

Go to Appendix B for user interfaces

2.2.2 Performance

- Number of Concurrent Users: Shall able to handle at least 1000 orders per second
- Order product or service:

The system is susceptible to any temporary server failure since it uses the strong feature of Struts 2 and Hibernate. Hence the examination will be continued even if the sever gets disconnected in between the examination.

2.2.3 Other Requirements:

- Hardware Interfaces

The SPMS is expected to function on Intel PIII 900 MHz Processor equivalent or above, 128 MB RAM, 20 GB HDD.

- Software Interfaces

The SPMS shall work on MS Windows operating systems family (MS Windows 98, MS Windows NT Workstation, MS Windows 2000, MS Windows XP). It configures to work with Oracle database. This System works on Apache Tomcat server. It uses browser IE 5.0 & above. It uses IIS 5.0 server.

2.2 NON-FUNCTIONAL REQUIREMENTS

2.2.1 Interface

Go to Appendix B for user interfaces

2.2.2 Performance

- Number of Concurrent Users: Shall able to handle at least 1000 orders per second
- Order product or service:
The system is susceptible to any temporary server failure since it uses the strong feature of Struts 2 and Hibernate. Hence the examination will be continued even if the sever gets disconnected in between the examination.

2.2.3 Constraint

Home Bazaar shall able to handle at least 1000 Booking/Sec

- **Other Requirements:**

Hardware and Network Interfaces:

Back-end Server Configuration:

- Intel Pentium-IV Processor
- 8 GB RAM

Front-end Client Configuration:

- AMD RYZEN 5 Processor
- 128 MB SDRAM
- 10 GB Hard Disk Drive
- 104 Keys Keyboard
- PS2 Mouse with pad

Software Interfaces:

Software configuration for back-end Services:

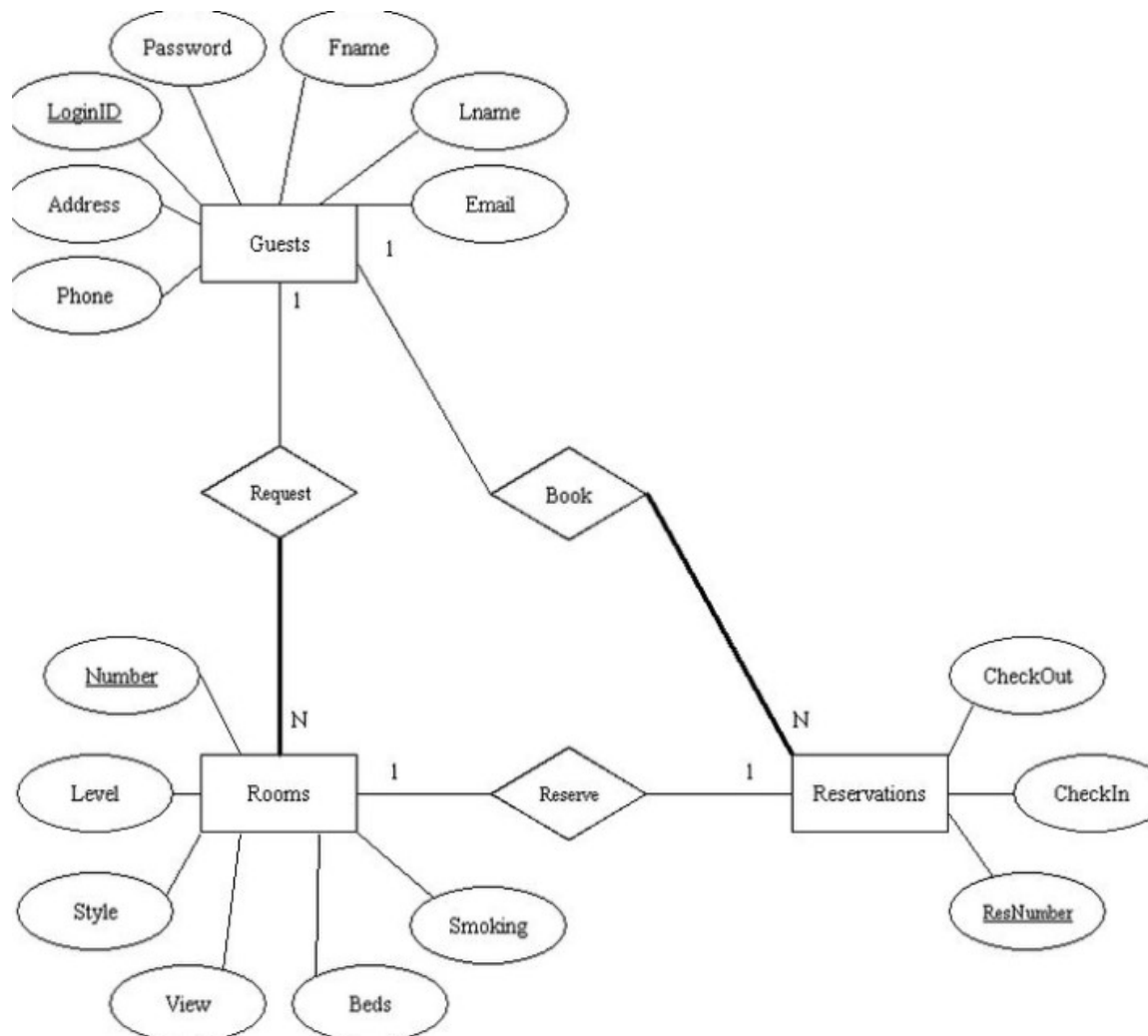
- Java EE
- Spring Boot 4 , JPA, Razor Pay, Spring Authentication
- MySQL8
- STS 3.9.18

Software configuration for front-end Services:

- ReactJS 6, Redux 2.1
- HTML, CSS, JS
- Bootstrap 4.2
- VS Code

3.DIAGRAMS

1.1 Entity Relationship Diagram:



Use Case Diagram

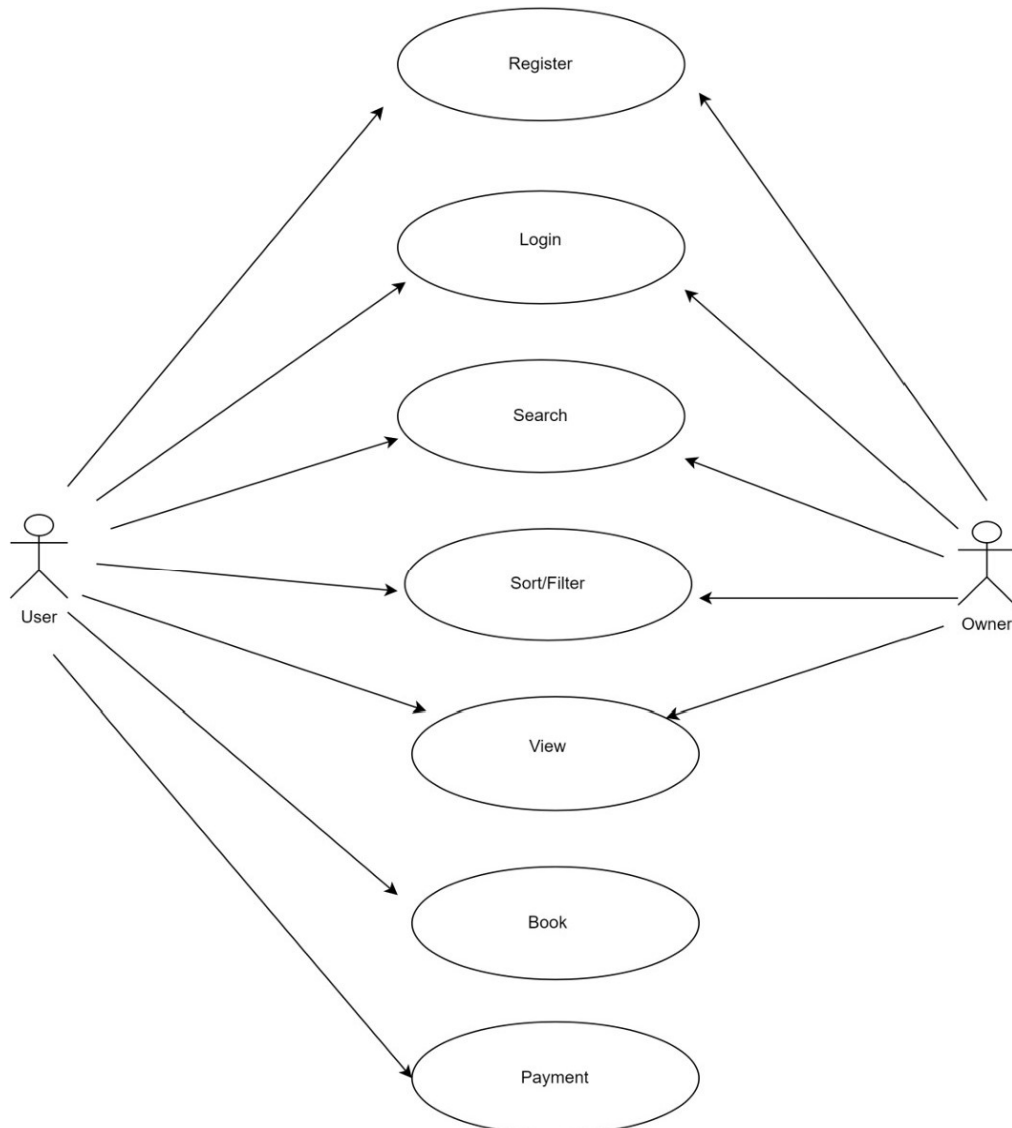
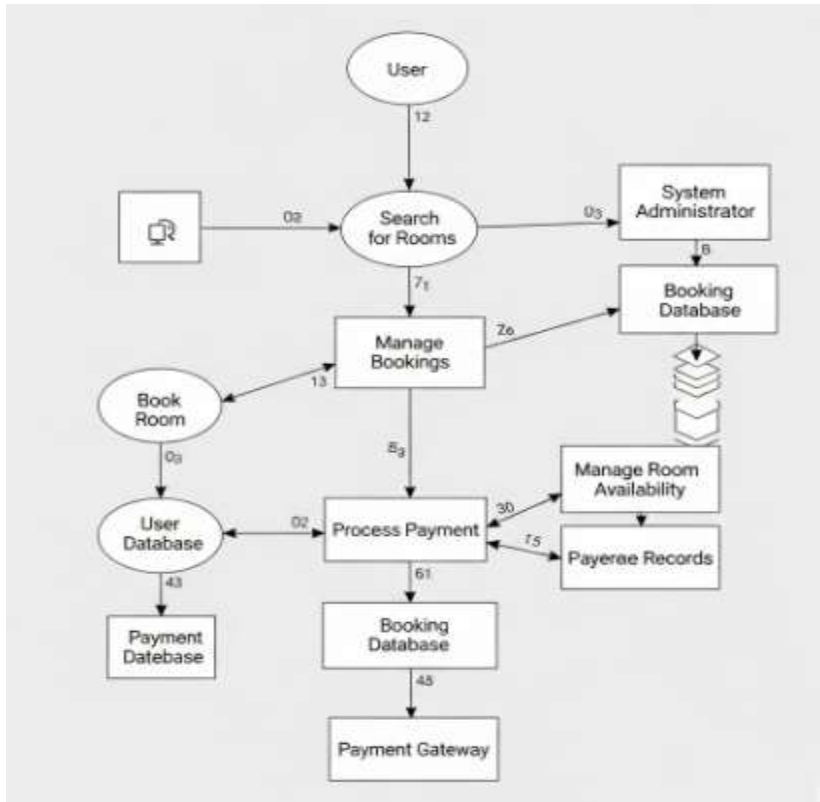


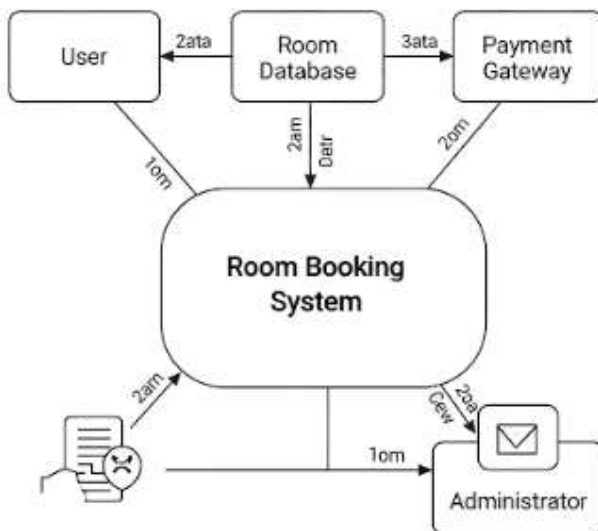
Fig. Use Case Diagram for StayHub

1. Data-Flow Lev(0-2) Diagram.

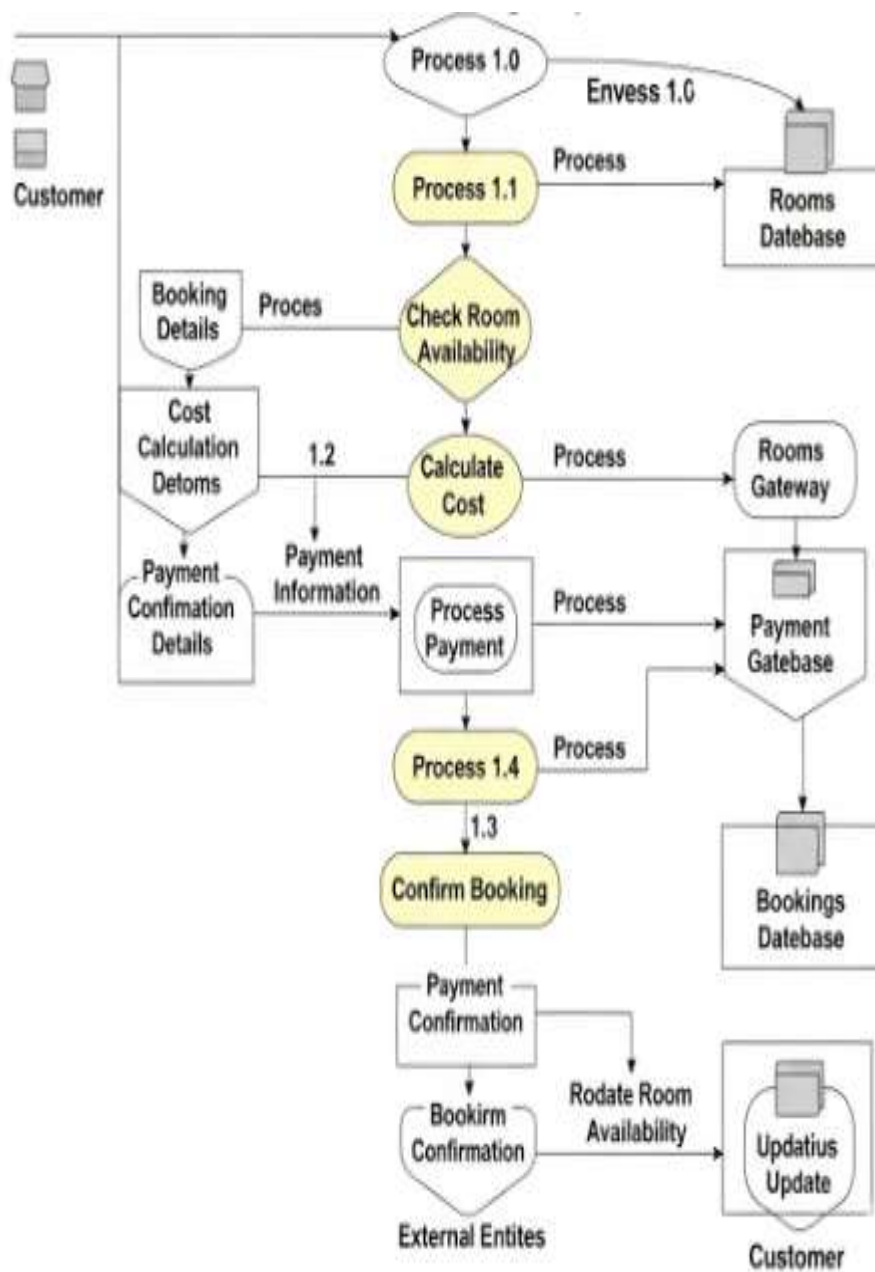
Level 0



Level-1

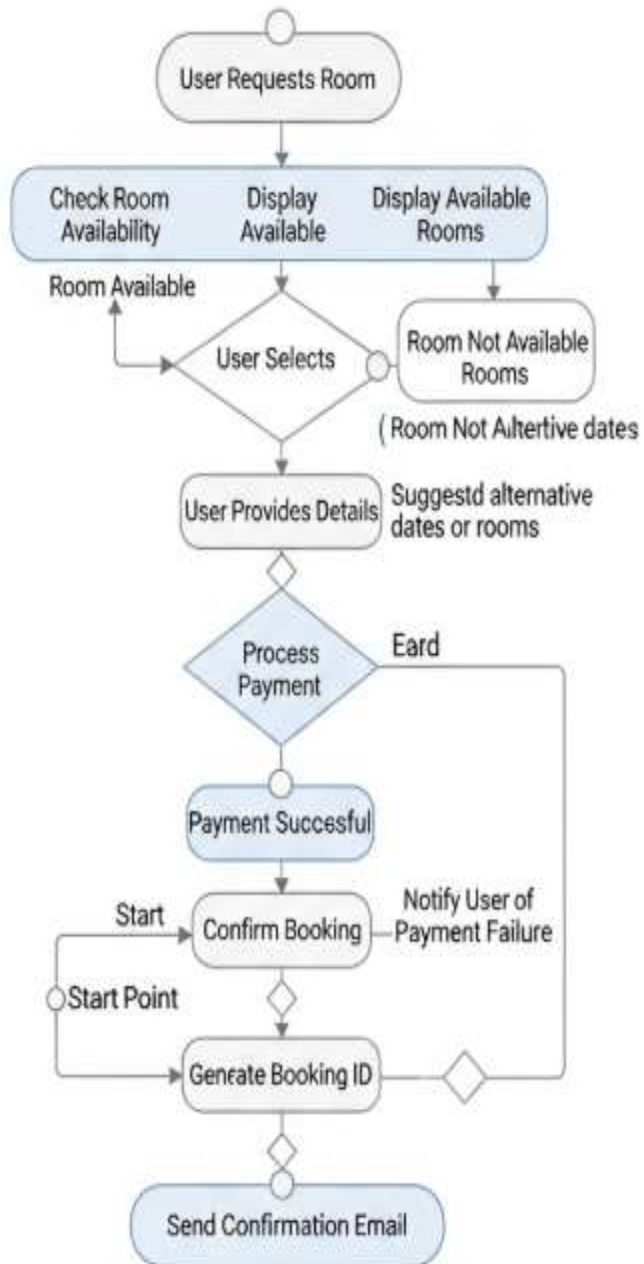


Level 2

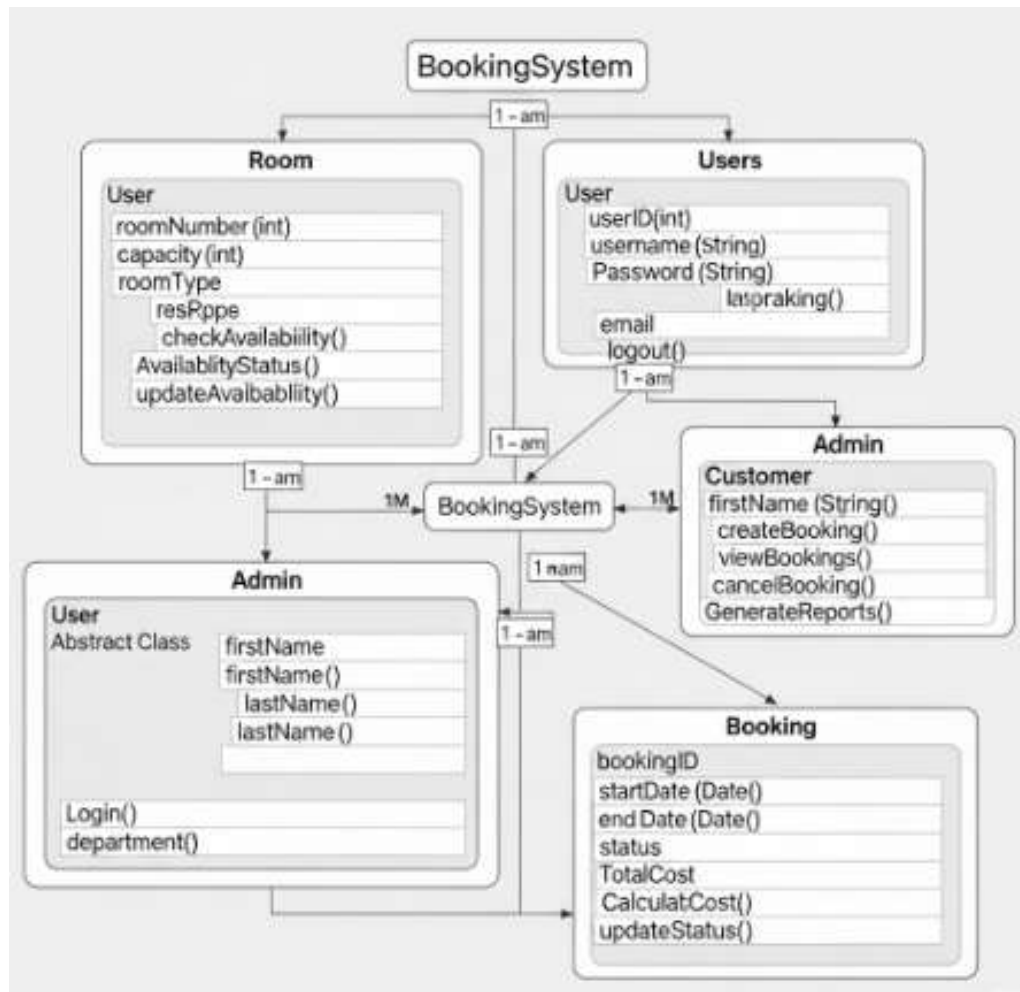


1.1 Activity Diagram :

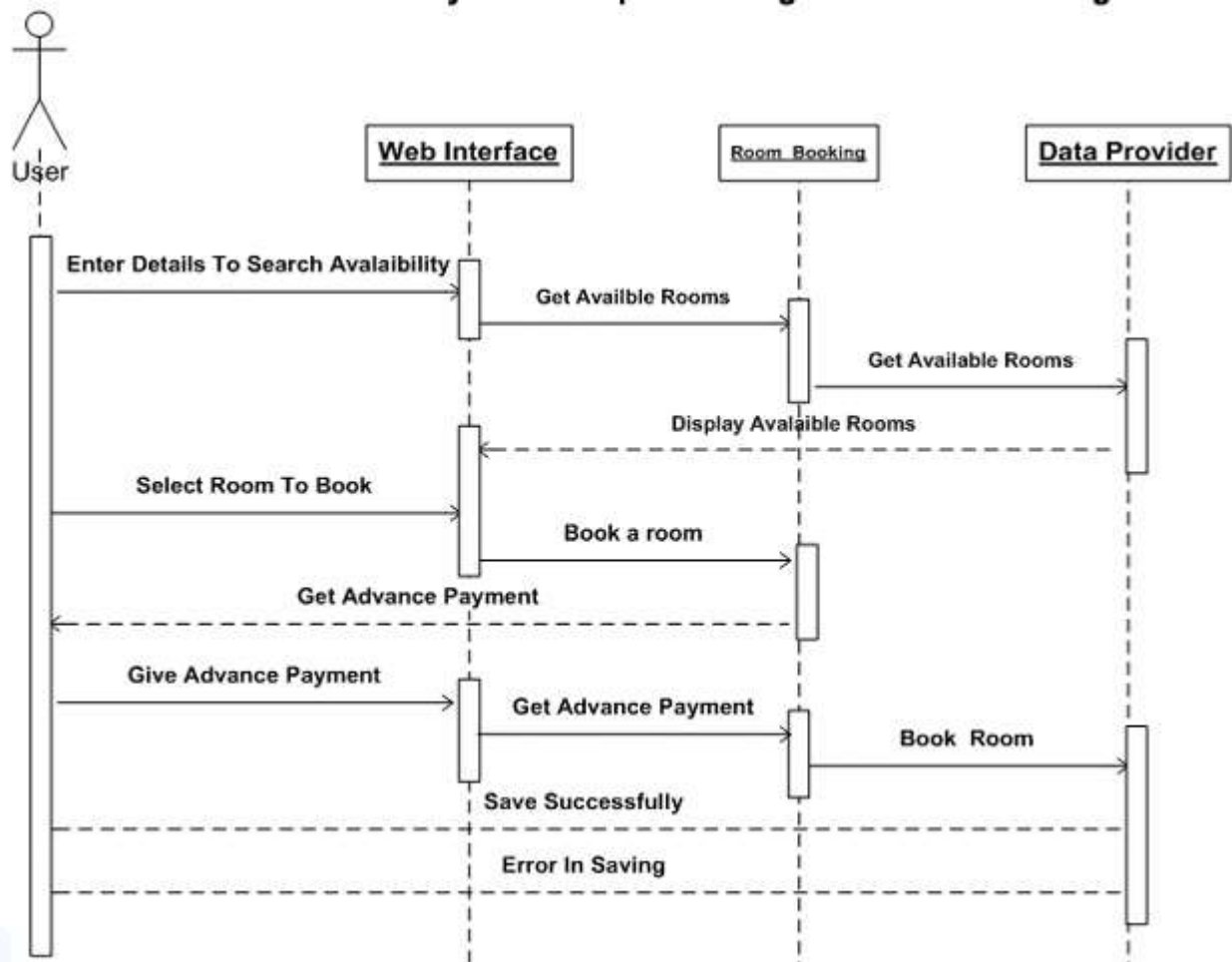
1. User Activity Diagram



1.2 Class Diagram:



1.3 Sequence Diagram



3. DESIGN

3.1 Database Design

The following table structures depict the database design.

Table1: User Table Schema

Field	Type	Null	Key	Default	Extra
id	bigint	NO	PRI	NULL	auto_incremen t
created_on	datetime(6)	YES		NULL	
updated_on	datetime(6)	YES		NULL	
email	varchar(255)	NO	UNI	NULL	
first_name	varchar(255)	NO		NULL	
last_name	varchar(255)	NO		NULL	
password	varchar(255)	NO		NULL	
role	enum('ADMIN','OWNER','USER')	NO		NULL	

Table 2: Owner

Field	Type	Null	Key	Default	Extra
id	bigint	NO	PRI	NULL	auto_increment
created_on	datetime(6)	YES		NULL	
updated_on	datetime(6)	YES		NULL	
email	varchar(255)	NO	UNI	NULL	
first_name	varchar(255)	NO		NULL	
last_name	varchar(255)	NO		NULL	
password	varchar(255)	NO		NULL	
role	enum('ADMIN', 'OWNER','USER')	NO		NULL	

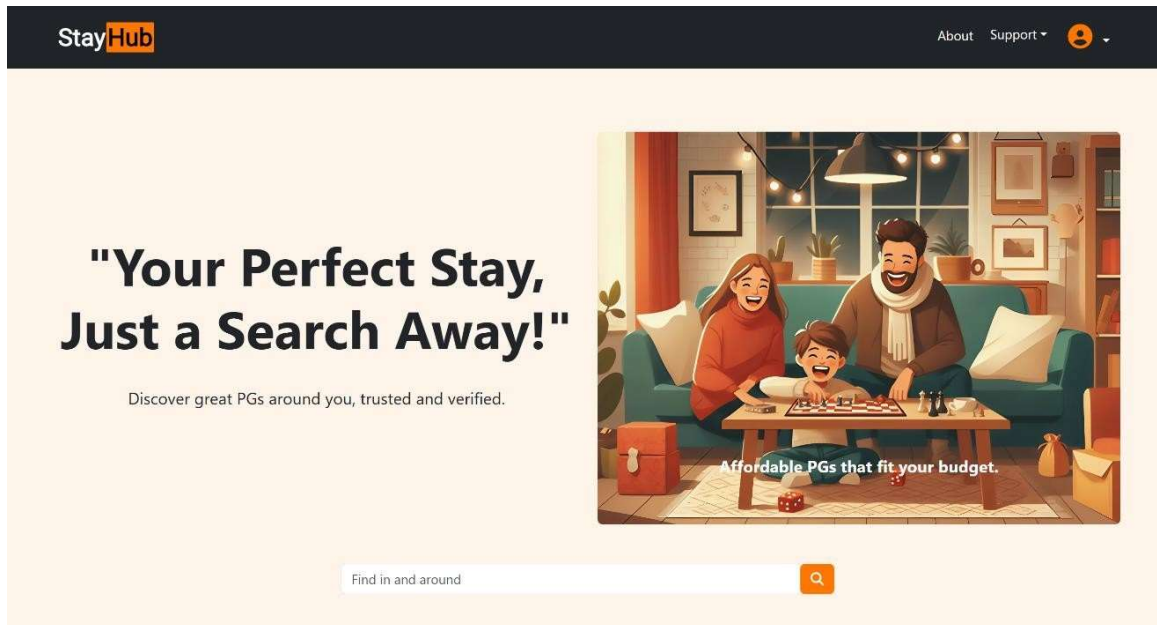
Table 3: PG's

Field	Type	Null	Key	Default	Extra
id	bigint	NO	PRI	NULL	auto_increment
created_on	datetime(6)	YES		NULL	
updated_on	datetime(6)	YES		NULL	
location	varchar(255)	NO		NULL	
name	varchar(255)	NO		NULL	
rent	double	NO		NULL	
owner_id	bigint	NO	MUL	NULL	

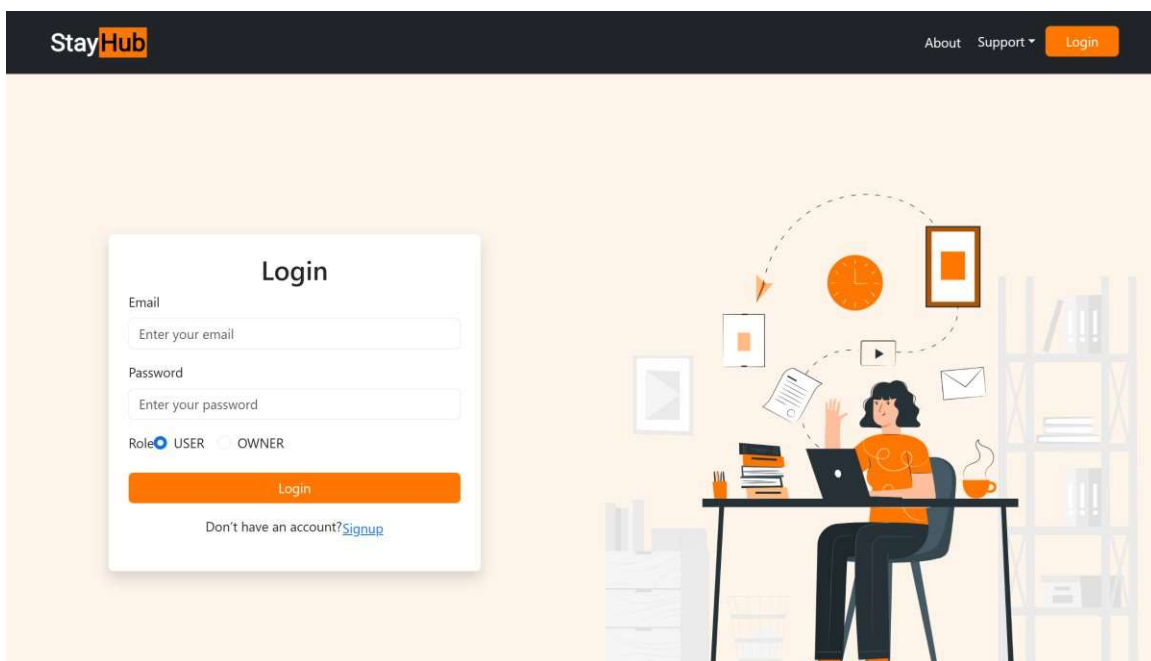
Table 4: Bookings

Field	Type	Null	Key	Default	Extra
id	bigint	NO	PRI	NULL	auto_incre m ent
created_on	datetime(6)	YES		NULL	
updated_on	datetime(6)	YES		NULL	
amount_pai d	double	NO		NULL	
check_in_ da te	date	NO		NULL	
check_out_ date	date	NO		NULL	
status	enum('APP ROVED','P ENDING',' REJECTED')	YES		NULL	
owner_id	bigint	YES	MUL	NULL	
pg_id	bigint	NO	MUL	NULL	
user_id	bigint	NO	MUL	NULL	

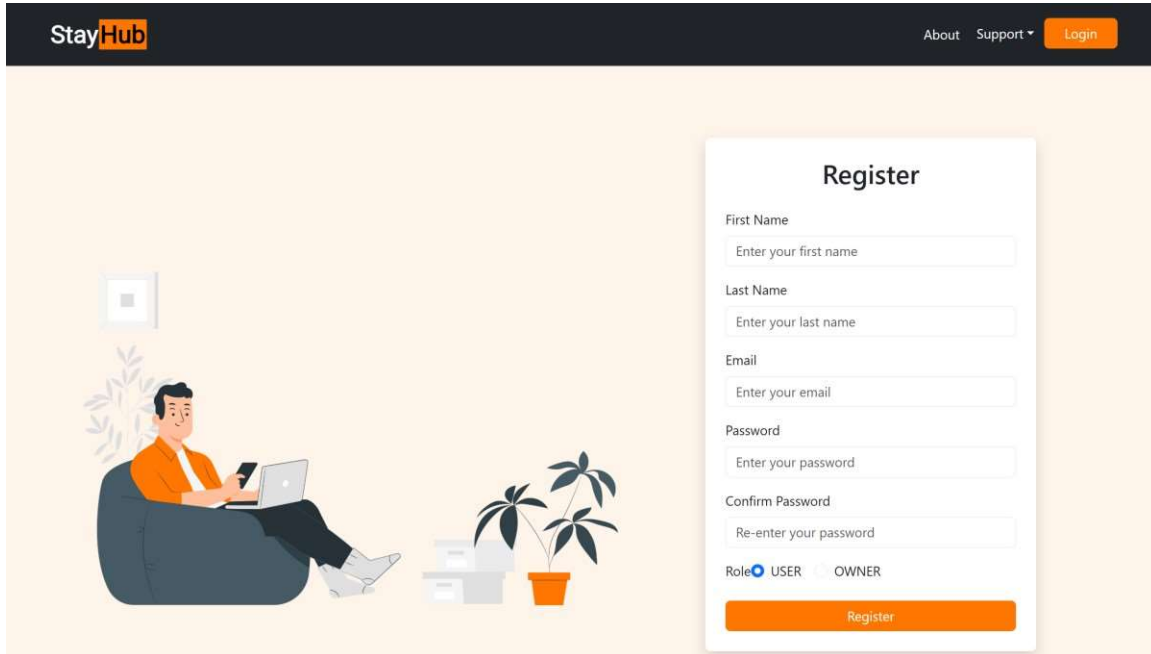
Landing Page:



Login:



Registration:



The registration form is titled "Register" and is located on the right side of the page. It includes fields for First Name, Last Name, Email, Password, and Confirm Password. There are also radio buttons for Role (USER or OWNER) and a Register button.

StayHub About Support Login

Register

First Name
Enter your first name

Last Name
Enter your last name

Email
Enter your email

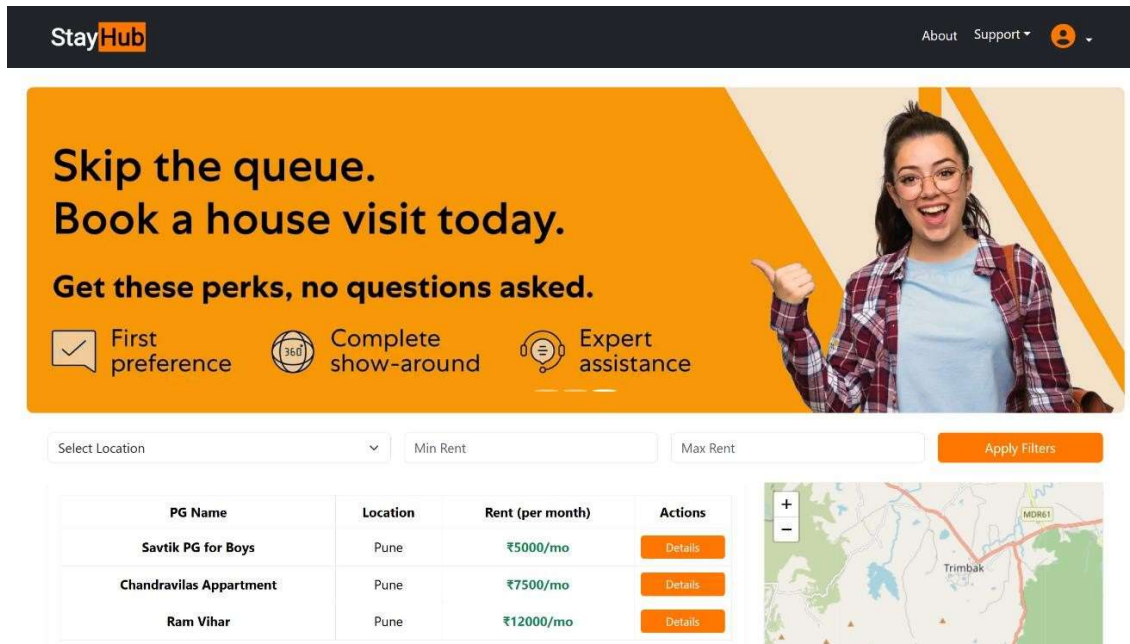
Password
Enter your password

Confirm Password
Re-enter your password

Role ☒ USER ☐ OWNER

Register

List of PG's:



The PG listing page features a banner with the text "Skip the queue. Book a house visit today. Get these perks, no questions asked." and a woman pointing. Below the banner are filters for location, min rent, and max rent. A table lists PGs with their names, locations, and monthly rents. A map of Trimbak is also shown.

StayHub About Support

Skip the queue. Book a house visit today. Get these perks, no questions asked.

- First preference
- Complete show-around
- Expert assistance

Select Location Min Rent Max Rent Apply Filters

PG Name	Location	Rent (per month)	Actions
Savtik PG for Boys	Pune	₹5000/mo	Details
Chandravilas Appartment	Pune	₹7500/mo	Details
Ram Vihar	Pune	₹12000/mo	Details

Map showing location: Trimbak

Details Page

[About](#)
[Support](#)

[← Back](#)

Reserve Now

Check-in:

Check-out:

☒ I agree to the [terms & conditions](#).

[Reserve Now](#)

Chandravilas Appartment

Location: Pune

Rent: ₹7500/month

[Amenities](#)
[Services](#)
[Details](#)

Services

AC
High-Speed WiFi

Owner profile-

[About](#)
[Support](#)

Owner Dashboard

[← Back](#)

- [Profile](#)
- [+ Add PG](#)
- [Manage PGs](#)
- [Manage Bookings](#)

Owner Profile

First Name: Ram
Last Name: Charan
Email: ram@gmail.com

Add PG's (Owner)-

PG added successfully!

Owner Dashboard
 ← Back
 Profile
 + Add PG
 Manage PGs
 Manage Bookings

+ Add PG
 PG Name
 Enter PG Name
 Location
 Enter PG Location
 Rent (per month)
 Enter Rent Amount
 Add PG

Manage PG's (Owner)-

About Support

Owner Dashboard
 ← Back
 Profile
 + Add PG
 Manage PGs
 Manage Bookings

Manage PGs

PG Name	Location	Rent (per month)	Actions
Savtik PG for Boys	Pune	₹5000	Edit Delete
Chandravilas Appartment	Pune	₹7500	Edit Delete
Ram Vihar	Pune	₹12000	Edit Delete

Manage bookings (Owner)-

About Support

Owner Dashboard
 ← Back
 Profile
 + Add PG
 Manage PGs
 Manage Bookings

Manage Bookings

Booking ID	PG Name	User Email	Status	Actions
1	Chandravilas Appartment	nishant362001@gmail.com	PENDING	Approve Reject
2	Ram Vihar	ritesh@gmail.com	PENDING	Approve Reject
3	Savtik PG for Boys	genelia@gmail.com	PENDING	Approve Reject

User profile-

StayHub

AboutSupport

Hi, nishant362001

Dashboard

Logout

← Back

User Dashboard

Home

Profile

Previous Bookings

User Profile

First Name: Nishant

Last Name: Deshmukh

Email: nishant362001@gmail.com

Previous bookings (User)-

StayHub

AboutSupport

← Back

User Dashboard

Home

Profile

Previous Bookings

Previous Bookings

#	PG Name	Booking Date	Status
1	Chandravilas Appartment	2/10/2025	PENDING

ONCLUSION

Stayhub addresses key requirements for modern accommodation and booking management systems, providing a solid foundation for further development and improvement. Its modular design and adherence to security best practices position it well for continued success and adaptation in the evolving hospitality landscape.

The Stayhub project exemplifies a sophisticated approach to property and booking management by leveraging cutting-edge technologies and methodologies. This application is designed to meet the needs of both guests and hosts, ensuring a seamless experience for managing properties, bookings, and payments while adhering to stringent security standards. Its blend of modern technologies, robust security, and user-centric design positions it as a valuable tool for businesses seeking to enhance their hospitality capabilities. The project's architecture and features provide a strong foundation for ongoing development, making it adaptable to future technological advancements and market trends.

REFERENCES:

<http://www.google.com>

<http://www.airbnb.com>

<http://www.webdevelopersjournal.com/>

<http://www.sty.com>

<http://www.w3.org>

<http://www.wikipedia.org>

<http://www.delta.com>

<http://www.priceline.com>