



**Project Title: StayBangla - An Innovative B&B and
Property Rental System with AI Chatbot.**

Final Year Project/Thesis/Internship

Course Code: SE431

Supervised By

Mr. Md. Khaled Sohel

Assistant Professor

Department of Software Engineering

Daffodil International University

Submitted By

Md Kamrul Islam

ID: 211-35-711

Department of Software Engineering

Daffodil International University

This project report has been submitted in fulfilment of the requirements for
the degree of **Bachelor of Science in Software Engineering**

APPROVAL

APPROVAL


This project titled on “StayBangla - An Innovative B&B and Property Rental System with AI Chatbot.”, submitted by **Md Kamrul Islam (ID: 211-35-711)** to the Department of Software Engineering, Daffodil International University has been accepted as satisfactory for the partial fulfillment of the requirements for the degree of Bachelor of Science in Software Engineering and approval as to its style and contents.

BOARD OF EXAMINERS



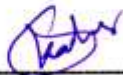
Dr. A. H. M. Saifullah Sadi
Professor
Department of Software Engineering
Faculty of Science and Information Technology Daffodil
International University

Chairman



Dr. Rubaiyat Islam
Associate Professor
Department of Software Engineering
Faculty of Science and Information Technology
Daffodil International University

Internal Examiner 1



Dr. Md. Abdul Kader
Associate Professor
Department of Software Engineering
Faculty of Science and Information Technology
Daffodil International University

Internal Examiner 2



Nuruzzaman Faruqi
Assistant Professor
Department of Software Engineering
Faculty of Science and Information Technology
Daffodil International University

Internal Examiner 3



Md. Mostafiz Khan
Managing Director
Tecognize Solutions Limited

External Examiner

DECLARATION

I hereby declare that I completed this project under the supervision of **Mr. Md. Khaled Sohel**, Assistant Professor, Department of Software Engineering, Daffodil International University. I further declare that this project is my original work accomplished in completion of the requirements for the B.Sc. in Software Engineering, and that neither the entire work nor any part of it has been submitted for the award of any other degree at this or any other institution.



(Supervisor's Signature)

Mr. Md. Khaled Sohel

Assistant Professor

Department of Software Engineering

Daffodil International University



(Student's Signature)

Md Kamrul Islam

ID: 211-35-711

Department of Software Engineering

Daffodil International University

ACKNOWLEDGEMENTS

The completion of this project was possible under the guidance, support and encouragement of various committed individuals and organizations. I could not have been successful without their assistance.

I am indeed happy that Daffodil International University has provided me with the opportunity to work on my project and resources I required. I am particularly indebted to my supervisor, **Mr. Md. Khaled Sohel** whose idea, feedback, and motivation were so instrumental in coming up with StayBangla.

I would like to thank my teachers, friends, and classmates and their valuable contributions and unlimited support. My family has been of great help to me, especially when it comes to supporting me in this difficult and rewarding process.

I would like to thank real estate professionals and tourists whose opinion influenced the design and use of the StayBangla platform. They have been invaluable in terms of their support.

DEDICATION

I therefore declare that I have done this project under the oversight of **Mr. Md. Khaled Sohel, Assistant Professor**, Department of Software Engineering, Daffodil International University. Also declare that neither entire record nor any portion of this record has been submitted somewhere else for my degree.

ABSTRACT

StayBangla is a straightforward Bed-and-Breakfast, short-term and property management site. It assists in managing the listings and booking and communication with guests in real time. The platform is equipped with a chatbot that is smart and assists guests in locating properties. StayBangla simplifies the process of interaction between guests and hosts by integrating chat support with booking services.

The site maintains the calendar and provides online booking, as well as enables quick bookings. Before booking, guests can watch photos and videos. An in-built chat assistant answers general questions and refers more complicated problems to employees. Hosts make, receive and maintain all booking, messaging and maintenance through a single dashboard.

StayBangla minimizes repetitive tasks and prevents frequent and typical booking errors. The chat assistant also ranks basic questions to be answered first hence leaving the staff to attend to more important concerns of the guests leading to a smoother operation and quick response.

The platform also incorporates safety. Each guest screen will have a distinct Emergency button which will directly call 999 to get quick assistance provided by national emergency services of Bangladesh.

The essence of StayBangla is to simplify and assure the short stay holiday in Bangladesh. The platform decreases the workload of the owners of property, offers fast and helpful assistance to the guests when they need it, and strives to make the stay experience as safe, comfortable, and guided as possible since the booking to the checkout.

TABLE OF CONTENTS

| | |
|---|-----|
| APPROVAL | ii |
| ACKNOWLEDGEMENTS | iv |
| DEDICATION | v |
| ABSTRACT | vi |
| TABLE OF CONTENTS | vii |
| LIST OF TABLES | x |
| LIST OF FIGURES | xii |
| CHAPTER 1 INTRODUCTION | 1 |
| Background | 1 |
| Project Planning and Initiation | 2 |
| Feasibility Study | 2 |
| 1.3 Target User Profile and Tentative Elicitation Process | 4 |
| 1.3.1 Target User | 4 |
| 1.3.2 User profile | 4 |
| 1.3.3 Elicitation Process | 7 |
| 1.4 Project Block Diagram | 8 |
| 1.5 System Requirements | 8 |
| 1.5.1 Hardware Requirements | 8 |
| 1.5.2 Software Requirements | 8 |
| 1.5.3 Constraints and Dependencies | 9 |
| 1.6 Project Scheduling | 9 |
| 1.7 Summary | 10 |
| CHAPTER 2 DESIGN AND IMPLEMENTATION | 11 |
| 2.1 Introduction | 11 |

| | |
|---|----|
| 2.2 Functional Requirements | 11 |
| 2.3 Non-Functional Requirements | 14 |
| 2.4 Object-oriented System design using UML | 16 |
| 2.4.1 Use Case Diagram | 16 |
| 2.4.2 Case Description | 17 |
| Table 4-1: Registration & Login | 18 |
| Table 4-2: Property Listing Management | 20 |
| Table 4-3: Search and Filter Properties | 21 |
| 2.4.3 Activity Diagram | 34 |
| 2.4.4 Sequence Diagram | 54 |
| 2.4.5 Class Diagram | 74 |
| 2.4.6 ER Diagram | 75 |
| 2.5 Coding: Appendix B | 76 |
| 2.6 Summary | 76 |
| Chapter 3 Software Testing | 77 |
| 3.1 Introduction | 77 |
| 3.2 Testing Features | 77 |
| 3.2.1 Feature to Be Tested | 77 |
| 3.3 Testing Strategies | 78 |
| 3.4 System Testing (Test Cases with Report) | 79 |
| 3.5 Summary | 84 |
| Chapter 4 Deployment and Maintenance | 85 |
| 4.1 Introduction | 85 |
| 4.2 SRLC (Software Release Life Cycle) | 85 |
| Chapter 5 User Manual | 88 |
| 5.1 Introduction | 88 |

| | |
|-----------------------------|----|
| 5.2 Project Functionalities | 88 |
| 5.2.1 Guest Functionalities | 88 |
| 5.2.2 Host Functionalities | 88 |
| 5.2.3 Admin Functionalities | 89 |
| 5.2.4 Stay Bangla System | 89 |
| 5.3 Summary | 95 |
| Chapter 6 Project Summary | 96 |
| 6.1 Introduction | 96 |
| 6.2 Project Limitation | 96 |
| 6.3 Scope | 96 |
| 6.4 Future Work | 97 |
| 6.5 Conclusion | 97 |

LIST OF TABLES

| | | |
|------------|---|----|
| Table 1-1 | User Profile for Guest | 4 |
| Table 1-2 | User Profile for Host | 5 |
| Table 1-3 | User Profile for Admin | 6 |
| Table 2-1 | Login and Registration | 10 |
| Table 2-2 | Property Listing Management | 10 |
| Table 2-3 | Search and Filter Properties | 11 |
| Table 2-4 | Booking and Reservation | 11 |
| Table 2-5 | Payment Processing | 11 |
| Table 2-6 | Review and Rating System | 11 |
| Table 2-7 | Messaging System | 11 |
| Table 2-8 | User Profile Management | 12 |
| Table 2-9 | Booking Cancellation and Refunds | 12 |
| Table 2-10 | Notifications and Alerts | 12 |
| Table 2-11 | Property Availability Management | 12 |
| Table 2-12 | User Support and Dispute Resolution | 12 |
| Table 2-13 | Chatbot | 13 |
| Table 3-1 | Non-Functional Requirement | 14 |
| Table 4-1 | Registration & Login (Use Case) | 16 |
| Table 4-2 | Property Listing Management (Use Case) | 18 |
| Table 4-3 | Search and Filter Properties (Use Case) | 20 |
| Table 4-4 | Booking and Reservation (Use Case) | 21 |
| Table 4-5 | Payment Processing (Use Case) | 22 |
| Table 4-6 | Review and Rating System (Use Case) | 22 |
| Table 4-7 | Messaging System (Use Case) | 23 |
| Table 4-8 | User Profile Management (Use Case) | 23 |
| Table 4-9 | Booking Cancellation and Refunds (Use Case) | 24 |
| Table 4-10 | Notifications and Alerts (Use Case) | 26 |

| | | |
|------------|--|----|
| Table 4-11 | Property Availability Management (Use Case) | 28 |
| Table 4-12 | User Support and Dispute Resolution (Use Case) | 30 |
| Table 4-13 | Chatbot (Use Case) | 31 |
| Table 5-1 | Testing Strategies | 77 |
| Table 6-1 | User Registration (Test Case) | 78 |
| Table 6-2 | User Login (Test Case) | 80 |
| Table 6-3 | Property Listing Creation by Host (Test Case) | 81 |
| Table 6-4 | Booking and Reservation (Test Case) | 82 |
| Table 6-5 | Booking Cancellation (Guest) (Test Case) | 82 |
| Table 6-6 | Review and Rating System (Guest) (Test Case) | 83 |

LIST OF FIGURES

| Figure No. | Title | Page |
|-------------------|---|-------------|
| Figure 1-1 | System Block Diagram | 7 |
| Figure 2-1 | Project Scheduling | 9 |
| Figure 3-1 | Use Case Diagram | 15 |
| Figure 4-1 | Login and Sign Up (Activity Diagram) | 33 |
| Figure 4-2 | Property Listing Management (Host) (Activity Diagram) | 34 |
| Figure 4-3 | Property Listing Management (Admin) (Activity Diagram) | 35 |
| Figure 4-4 | Search and Filter Properties (Guest) (Activity Diagram) | 36 |
| Figure 4-5 | Booking and Reservation (Guest) (Activity Diagram) | 37 |
| Figure 4-6 | Booking and Reservation (Host) (Activity Diagram) | 38 |
| Figure 4-7 | Payment Processing (Admin) (Activity Diagram) | 39 |
| Figure 4-8 | Review and Rating System (Guest) (Activity Diagram) | 40 |
| Figure 4-9 | Review and Rating System (Host) (Activity Diagram) | 41 |
| Figure 4-10 | Review and Rating System (Admin) (Activity Diagram) | 42 |
| Figure 4-11 | Messaging System (Guest) (Activity Diagram) | 43 |
| Figure 4-12 | Messaging System (Host) (Activity Diagram) | 43 |
| Figure 4-13 | User Profile Management (Guest & Host) (Activity Diagram) | 44 |
| Figure 4-14 | User Profile Management (Admin) (Activity Diagram) | 45 |
| Figure 4-15 | Booking Cancellation and Refund (Guest) (Activity Diagram) | 46 |
| Figure 4-16 | Booking Cancellation and Refund (Host) (Activity Diagram) | 47 |
| Figure 4-17 | Booking Cancellation and Refund (Admin) (Activity Diagram) | 47 |
| Figure 4-18 | Notification and Alert (Guest, Host and Admin) (Activity Diagram) | 48 |
| Figure 4-19 | Property Availability Management (Host) (Activity Diagram) | 49 |
| Figure 4-20 | User Support and Dispute Resolution (Guest) (Activity Diagram) | 50 |
| Figure 4-21 | User Support and Dispute Resolution (Host) (Activity Diagram) | 50 |
| Figure 4-22 | User Support and Dispute Resolution (Admin) (Activity Diagram) | 51 |
| Figure 4-23 | Chatbot (Guest) (Activity Diagram) | 52 |

| | | |
|-------------|---|----|
| Figure 4-1 | Login and Sign Up (Sequence Diagram) | 53 |
| Figure 4-2 | Property Listing Management (Host) (Sequence Diagram) | 54 |
| Figure 4-3 | Property Listing Management (Admin) (Sequence Diagram) | 55 |
| Figure 4-4 | Search and Filter Properties (Guest) (Sequence Diagram) | 56 |
| Figure 4-5 | Booking and Reservation (Guest) (Sequence Diagram) | 57 |
| Figure 4-6 | Booking and Reservation (Host) (Sequence Diagram) | 58 |
| Figure 4-7 | Review and Rating System (Guest) (Sequence Diagram) | 59 |
| Figure 4-8 | Review and Rating System (Host) (Sequence Diagram) | 59 |
| Figure 4-9 | Review and Rating System (Admin) (Sequence Diagram) | 60 |
| Figure 4-10 | Messaging System (Guest) (Sequence Diagram) | 61 |
| Figure 4-11 | Messaging System (Host) (Sequence Diagram) | 61 |
| Figure 4-12 | User Profile Management (Guest & Host) (Sequence Diagram) | 62 |
| Figure 4-13 | User Profile Management (Admin) (Sequence Diagram) | 63 |
| Figure 4-14 | Booking Cancellation and Refund (Guest) (Sequence Diagram) | 64 |
| Figure 4-15 | Booking Cancellation and Refund (Host) (Sequence Diagram) | 65 |
| Figure 4-16 | Booking Cancellation and Refund (Admin) (Sequence Diagram) | 66 |
| Figure 4-17 | Notification and Alert (Guest, Host and Admin) (Sequence Diagram) | 67 |
| Figure 4-18 | Property Availability Management (Host) (Sequence Diagram) | 68 |
| Figure 4-19 | User Support and Dispute Resolution (Guest) (Sequence Diagram) | 69 |
| Figure 4-20 | User Support and Dispute Resolution (Host) (Sequence Diagram) | 70 |
| Figure 4-21 | User Support and Dispute Resolution (Admin) (Sequence Diagram) | 71 |
| Figure 4-22 | Chatbot (Guest) (Sequence Diagram) | 72 |
| Figure 6-1 | Class Diagram | 73 |
| Figure 7-1 | ER Diagram | 74 |
| Figure 8-1 | Home Page | 90 |
| Figure 8-2 | Listing Page | 91 |
| Figure 8-3 | Registration | 91 |
| Figure 8-4 | Login | 92 |
| Figure 8-5 | Admin Dashboard | 92 |

| | | |
|-------------|------------------------|----|
| Figure 8-6 | Host Dashboard | 93 |
| Figure 8-7 | Host Listing Page | 93 |
| Figure 8-8 | Booking Checkout Page | 94 |
| Figure 8-9 | Guest Payment History | 94 |
| Figure 8-10 | Guest NID Verification | 95 |

CHAPTER 1 INTRODUCTION

Background

StayBangla is also increasing the short-term rentals and B&B hosting as there is an increase in people traveling in Bangladesh to work, study, and local tourism. Yet, the majority of the property owners continue to deal with the bookings and communication with the guests manually, which, in most cases, leads to delayed responses, inaccurate flow of information and scheduling mistakes. The customers also expect the more accurate information, promptness of the confirmation, and simpler means of connecting with the help in case of necessity.

StayBangla was designed to eliminate this loophole by providing guests and hosts one waiting place where they can make booking, support, and routine inquiries. The platform will help the property owners take less unnecessary work, enhance the consistency of the responses with the guests and simplify the overall stay experience and make it more structured.

The target of this project will be B&Bs and short-stay rental spaces in Bangladesh. It includes listing management, real-time booking flow, and communication tools, support request handling, and an emergency call feature that allows the users to be directly connected to 999 in case of emergency assistance.

Project Planning and Initiation

Feasibility Study

An elementary feasibility check was done to establish that StayBangla will work. Can be utilized successfully in practical environments of B&Bs in Bangladesh. This step helps confirm market requirement, technical potential, and can be sustained in the long term before full development begins.

Phase 1 Preliminary Analysis & Project Scope Definition

The property rental business has evolved tremendously in past few years, simply because of digital platforms which have enhanced the need of easy online booking. To meet this purpose, StayBangla is being built in order to offer a convenient and trustworthy property rental both the experience of property owners and guests. This venture will provide an opportunity to interlink property. Owners and renters through providing an easy platform to book, manage properties and communicate with guests.

The scope of the StayBangla project includes:

- Designing a responsive and user-friendly web application.
- Developing individual dashboards for property owners, guests, and administrators.
- Enabling communication between guests and property owners through messaging.
- Incorporating booking and payment processing features.
- Implementing an AI chatbot for simple questions and answers for guests.

Phase 2 Market Research

Market research has revealed that the particular percentage of travelers that would be interested in using short-term bookings services online in Bangladesh are not yet provided. Still, according to different studies and market analysis, there has been a great growth in interest and usage. By 2030, the travel and tourism market of Bangladesh is likely to increase considerably, and the short-term rental market is also a considerable portion of it.

It is expected that 63 percent of the travel and tourism revenue in Bangladesh will be generated by 2030 from online sales. This signifies increasing curiosity in on-line services. An analysis researched that online analysis is done by 62% of those planning their trips, mostly on social media. Currently, the market share of online travel agents (OTAs) is approximately 4-6% of the travel demand in Bangladesh. Nevertheless, this may increase to more than 45 percent in the year 2025. The travel accommodation market will continue to expand to 1.68 in 2025 up to its current user number of 1.5 million by 2030. This tendency gives a definite opportunity to StayBangla to deliver online short-term rentals in Bangladesh. StayBangla has a chance to provide domestic and convenient solutions on behalf of the property owners and the travelers.

Phase 3 Technical Feasibility Analysis

The technical stack incorporates:

Backend:

- Framework: Laravel 11.23.0
- PHP: 8.1+
- Database: MySQL
- Architecture: MVC with service layer

Frontend

- Build: Vite 4.0
- CSS: Bootstrap 5.2.3, SASS
- JavaScript: Vanilla JS, Axios
- UI: Blade templates

APIs

- SSLCommerz: Payments
- Google Maps: Location/Maps
- Firebase: Push notifications
- Groq: AI chatbot
- Pusher: Real-time updates

The chosen technologies are all mature, supported, and suited to production.

Phase 4 Financial Feasibility Analysis

It was also created with little financial resources since the open-source tools were used. No paid-software was developed. It deployed and was tested on Localhost using XAMPP. In a production establishment, there will be minimal recurring costs of hosting, email facilities and even APIs in case they are required.

1.3 Target User Profile and Tentative Elicitation Process

1.3.1 Target User

- **Guest:** An individual willing to rent.
- **Host:** A registered individual who rents out properties.
- **Administrator:** Staff responsible for managing the system.

1.3.2 User profile

Table 1: User Profile for Guest

| User Class | Note on Characteristics |
|--------------|--|
| Type of user | Individuals looking to rent a property for a short-term stay (vacations, business trips, etc.) |
| Age range | Typically, 18-65 years old (with a broad distribution across various age groups) |

| | |
|---------------------|---|
| Frequency of use | Occasional (Guests may use the platform for trips a few times a year, but not daily) |
| Mandatory | Internet connection (to browse and make bookings) |
| Computer experience | Varies from beginner to advanced (some guests may have limited tech experience, while others are more tech-savvy) |
| Education | Varies (can range from high school graduates to college degrees, generally educated) |
| Goal | Find and book accommodations that fit their needs (budget, location, amenities) |
| Language skills | Medium to high proficiency in English (primary language should be well understood) |
| Number of users | High (the largest user base) |
| Training | No training required; platform should be intuitive and easy to use |
| Others system use | Typically uses other travel apps like Airbnb, Booking.com, Google Maps, etc. |
| Way of working | Self-directed (browses independently, reads reviews, makes bookings at their convenience) |

Table 1-1: User Profile for Guest

Table 2: User Profile for Host

| User Class | Note on Characteristics |
|---------------------|---|
| Type of user | Property owners or managers offering their property for rent (e.g., homes, apartments, rooms) |
| Age range | Typically, 25-55 years old (but can range from younger entrepreneurs to older individuals seeking passive income) |
| Frequency of use | Frequent (daily or weekly, as they need to manage bookings, availability, and guest communication) |
| Mandatory | Internet connection, basic property management tools |
| Computer experience | Intermediate (hosts may not be highly technical, but they should have some experience with online platforms) |

| | |
|-------------------|--|
| Education | Varies (from high school graduates to college degrees or higher) |
| Goal | Rent out their property efficiently, manage guest interactions, ensure a smooth experience for both themselves and guests |
| Language skills | Medium to high proficiency in English (primary language should be well understood) |
| Number of users | Medium (fewer hosts than guests, but crucial for platform success) |
| Training | Minimal training required (but a simple onboarding process should be included for new hosts, especially for property management) |
| Others system use | Similar platforms like Airbnb, Vrbo, and Booking.com. They may also use calendars, payment systems, or accounting software |
| Way of working | More structured (hosts have specific tasks, such as updating property listings, managing bookings, and interacting with guests) |

Table 1-2: User Profile for Host

Table 3: User Profile for Admin

| User Class | Note on Characteristics |
|---------------------|---|
| Type of user | Platform administrators managing backend operations, user support, and platform maintenance |
| Age range | Typically, 20-50 years old (more experienced individuals handling platform operations) |
| Frequency of use | Very frequent (daily use for monitoring, handling issues, and managing platform activities) |
| Mandatory | Internet connection, access to platform's admin tools |
| Computer experience | High (admins are very familiar with technology, web tools, and management software) |

| | |
|-------------------|--|
| Education | College or higher education, likely in business, IT, or management |
| goal | Ensure platform is running smoothly, support hosts and guests, manage platform security, handle user accounts and content moderation |
| Language skills | High proficiency in English (should be able to communicate clearly and effectively with users) |
| Number of users | Low (only a handful of admins) |
| Training | Admins may require specific training on platform functionalities, user management, and customer support procedures |
| Others system use | Admins likely use other business tools like CRM systems, email platforms, analytics software, and payment processing tools |
| Way of working | Structured (admins follow established workflows for managing the platform and user interactions) |

Table 1-3: User Profile for Admin

1.3.3 Elicitation Process

- **Discussions and Conversations:** Interviews with Guests and hosts and platform executives to obtain information regarding their needs and challenges.
- **User Stories and Background Mapping:** Created extensive user stories and Journey maps.
- **Competitor Review:** Tested the features and the offerings of competition with such rivals as Airbnb and Booking.com and learn their strong points along with weaknesses.
- **User Review Analysis:** Reviewed other property rental feedback and reviews spreadsheets to recognize common problems and useful attributes used by users.

1.4 Project Block Diagram

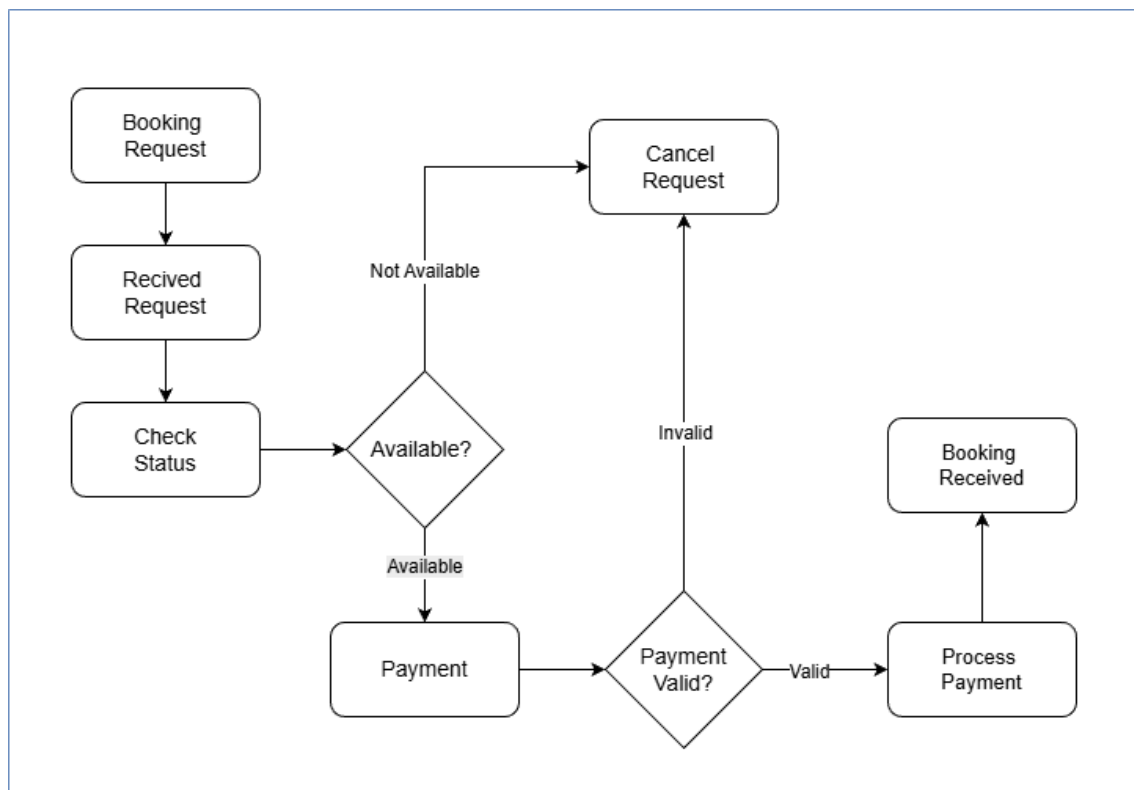


Figure 1-1: System Block Diagram

1.5 System Requirements

1.5.1 Hardware Requirements

- Computer with up-to-date browser.
- Smartphone with a modern browser.
- Internet Connectivity.

1.5.2 Software Requirements

- PHP: 8.1.0 or higher
- Database: MySQL 5.7
- Web Server: Apache 2.4
- API Services: Google Map, SSLCommerz, Groq, Firebase & Pusher

1.5.3 Constraints and Dependencies

Constraints:

- Scalability: Handle growing traffic.
- Security: Comply with PCI-DSS and GDPR.
- Availability: Ensure high uptime.
- Legal Compliance: Follow local rental laws.
- Cross-Device: Support mobile and desktop.
- Budget: Balance cost with infrastructure.
- Time-to-Market: Focus on core features.

Dependencies:

- PHP 8.1+ with required extensions
- MySQL
- Install Redis
- Web server (Apache)
- SSL certificate

1.6 Project Scheduling

The project was organized into weekly sprints. Each sprint focused on developing a specific part or module of the platform.

Tools Used:

- **Gantt Chart:** Operated to envision tasks and track task timelines.

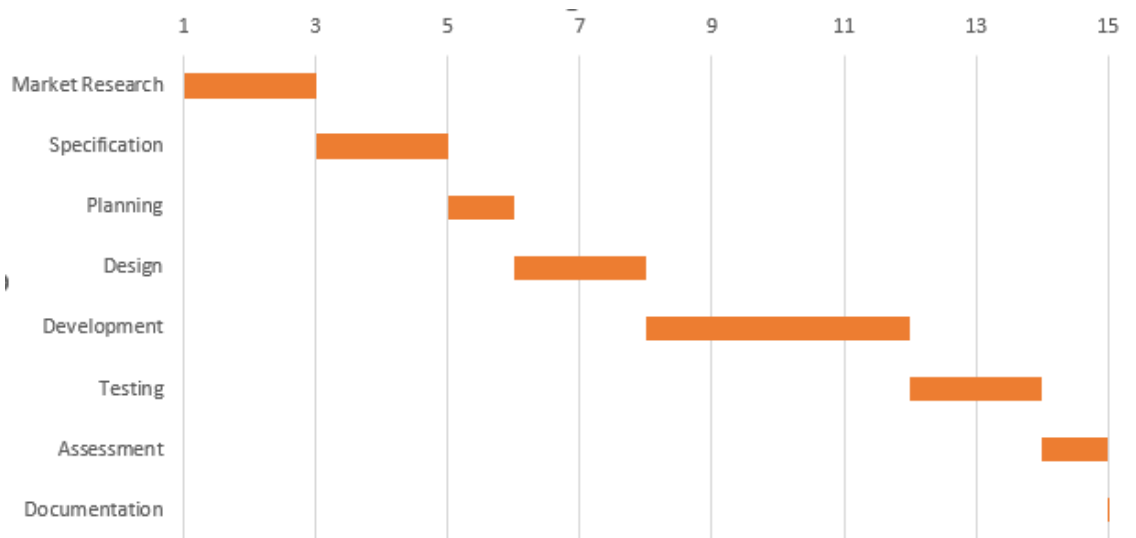


Figure 2-1: Project Scheduling

Risk Management:

- **Technical Risks:** Managed everything through modular development and regular backups.
- **Schedule Risks:** Added buffer weeks and held weekly demo sessions to check our progress.
- **Integration Risks:** Perform staged testing for critical integrations, such as payment systems and video communication APIs.

1.7 Summary

In this chapter, an overview of the StayBangla project is given starting with an overview of the project. Exposition of its history, the contextualization of the project, the problem that is identified, its purpose, and scope. It then outlines the process of the planning and initiation of a project, as well as a comprehensive feasibility study. The target user profile and the chapter also define the target user profile as well describes the elicitation process, which is applied to the project requirements. It introduces the project block diagram and gives a detailed information about the system requirements, addressing hardware, as well as software requirements. Also, the limitation and dependencies which might be influential of the project are described. The chapter would end with a conclusion description of project timeline, presentation of the project timeline and major milestones.

CHAPTER 2 DESIGN AND IMPLEMENTATION

2.1 Introduction

This chapter identifies the main features of the necessities and design of the StayBangla. It starts by giving the functional requirements that constitute the system requirements. features, then the non-functional requirements which are concerned with performance, reliability and portability.

The system design based on the object-oriented principles, and UML is discussed in the chapter. It consists of the necessary diagrams like the Use Case Diagram, Activity Diagram and the Sequence Diagram. Entity-Relationship Diagram, Diagram, Class Diagram. The following diagrams depict the. Components and interactions of system.

The last thing is that the chapter includes an appendix (B), the details of coding, and ends with. an overview of the key arguments that were covered in the chapter.

2.2 Functional Requirements

Here is a concise list of Functional Requirements (FR) for the StayBangla project, outlining the platform's key features.

| | |
|--------------------|--|
| FR01 | Login and Registration |
| Description | Guests and hosts can sign up using their email addresses. The system also helps users recover their passwords. |
| Stakeholder | Guest, Host, Admin |

Table 2-1: Login and Registration

| | |
|--------------------|--|
| FR02 | Property Listing Management |
| Description | Hosts can create, edit, and delete property listings with details like location, price, amenities, availability, and photos. Admins review and approve these listings before they go live. |
| Stakeholder | Host, Admin |

Table 2-2: Property Listing Management

| | |
|-------------|-------------------------------------|
| FR03 | Search and Filter Properties |
|-------------|-------------------------------------|

| | |
|--------------------|---|
| Description | By filtering and searching properties, guests can be able to find appropriate places to stay. They are able to make decisions depending on location, price range, guests' number and amenities. |
| Stakeholder | Guest |

Table 2-3: Search and Filter Properties

| | |
|--------------------|--|
| FR04 | Booking and Reservation |
| Description | Guests can book properties by selecting dates and making secure payments. Hosts can manually approve or automatically accept bookings. |
| Stakeholder | Guest, Host |

Table 2-4: Booking and Reservation

| | |
|--------------------|---|
| FR05 | Payment Processing |
| Description | Guests can pay for bookings through secure payment gateways like SSLCommerz. After booking confirmation, hosts can withdraw payments, with fees deducted first. |
| Stakeholder | Guest, Host, Admin |

Table 2-5: Payment Processing

| | |
|--------------------|--|
| FR06 | Review and Rating System |
| Description | Once they have been staying at a property, the guests will be able to leave a rating and review concerning their visit. These reviews can be viewed by the hosts and reacted to. |
| Stakeholder | Guest, Host |

Table 2-6: Review and Rating System

| | |
|--------------------|--|
| FR07 | Messaging System |
| Description | Guests and hosts can communicate directly through the platform's messaging system to discuss bookings, property questions, and other concerns. |
| Stakeholder | Guest, Host |

Table 2-7: Messaging System

| | |
|--------------------|--|
| FR08 | User Profile Management |
| Description | Guests and hosts maintain profiles displaying their booking history, reviews, and personal details. Admins oversee these profiles to ensure platform rules are followed. |
| Stakeholder | Guest, Host, Admin |

Table 2-8: User Profile Management

| | |
|--------------------|---|
| FR09 | Booking Cancellation and Refunds |
| Description | Guests can cancel bookings in accordance with the host's cancellation policy. Hosts can also cancel bookings, notify guests, and offer alternatives or refunds. |
| Stakeholder | Guest, Host, Admin |

Table 2-9: Booking Cancellation and Refunds

| | |
|--------------------|---|
| FR10 | Notifications and Alerts |
| Description | The platform sends notifications to guests and hosts for events like booking confirmations, cancellations, messages, and reminders. |
| Stakeholder | Guest, Host, Admin |

Table 2-10: Notifications and Alerts

| | |
|--------------------|---|
| FR11 | Property Availability Management |
| Description | Hosts can manage their property's availability using a built-in calendar. Guests can view available dates when making bookings. |
| Stakeholder | Guest, Host |

Table 2-11: Property Availability Management

| | |
|--------------------|--|
| FR12 | User Support and Dispute Resolution |
| Description | The system provides customer support for resolving disputes between guests and hosts, with admins overseeing the resolution process. |
| Stakeholder | Guest, Host, Admin |

Table 2-12: User Support and Dispute Resolution

| | |
|-------------|----------------|
| FR13 | Chatbot |
|-------------|----------------|

| | |
|--------------------|---|
| Description | The system provides 24-hours of chatbot service that provides customized property consultancy depending on the preferences. It provides the FAQ solving common questions, which makes the responses faster and the user experience more positive. |
| Stakeholder | Guest, Host, Admin |

Table 2-13: Chatbot

2.3 Non-Functional Requirements

Non-functional requirements determine the characteristics of a system that should be possessed to fulfill the performance, reliability, security, and user experience requirements. As opposed to the active requirements, which dwell on the particular behavior and actions of the system, the non-functional requirements outline how the system carries out its activities.

| Non-Functional Requirement | Description |
|----------------------------|--|
| Performance | With the platform, there should be a good number of users at a given time without delays. Decisions such as property searching, confirmation, reservations and messages must be done in a hurry. |
| Reliability | The system should be always accessible. It must have backup systems and must be in a position to recover when failures occur promptly to smooth operations even in times of trouble. |
| Security | User data, particularly payment details and personal information, must be kept safe with encryption. StayBanla must follow data security laws and offer secure login methods, along with regular security updates. |
| Scalability | As the number of users and property listings grows, the system should be able to handle the increased load without affecting performance. |
| Usability | The platform must be easy to use. Both guests and hosts should be able to navigate it without difficulty, even if they are using it for the first time. |

| | |
|-------------|---|
| Portability | The platform should be accessible across devices such as desktops, tablets, and smartphones, and work with various browsers and operating systems to ensure a consistent user experience. |
|-------------|---|

Table 3-1: Non-Functional Requirement

2.4 Object-oriented System design using UML

2.4.1 Use Case Diagram

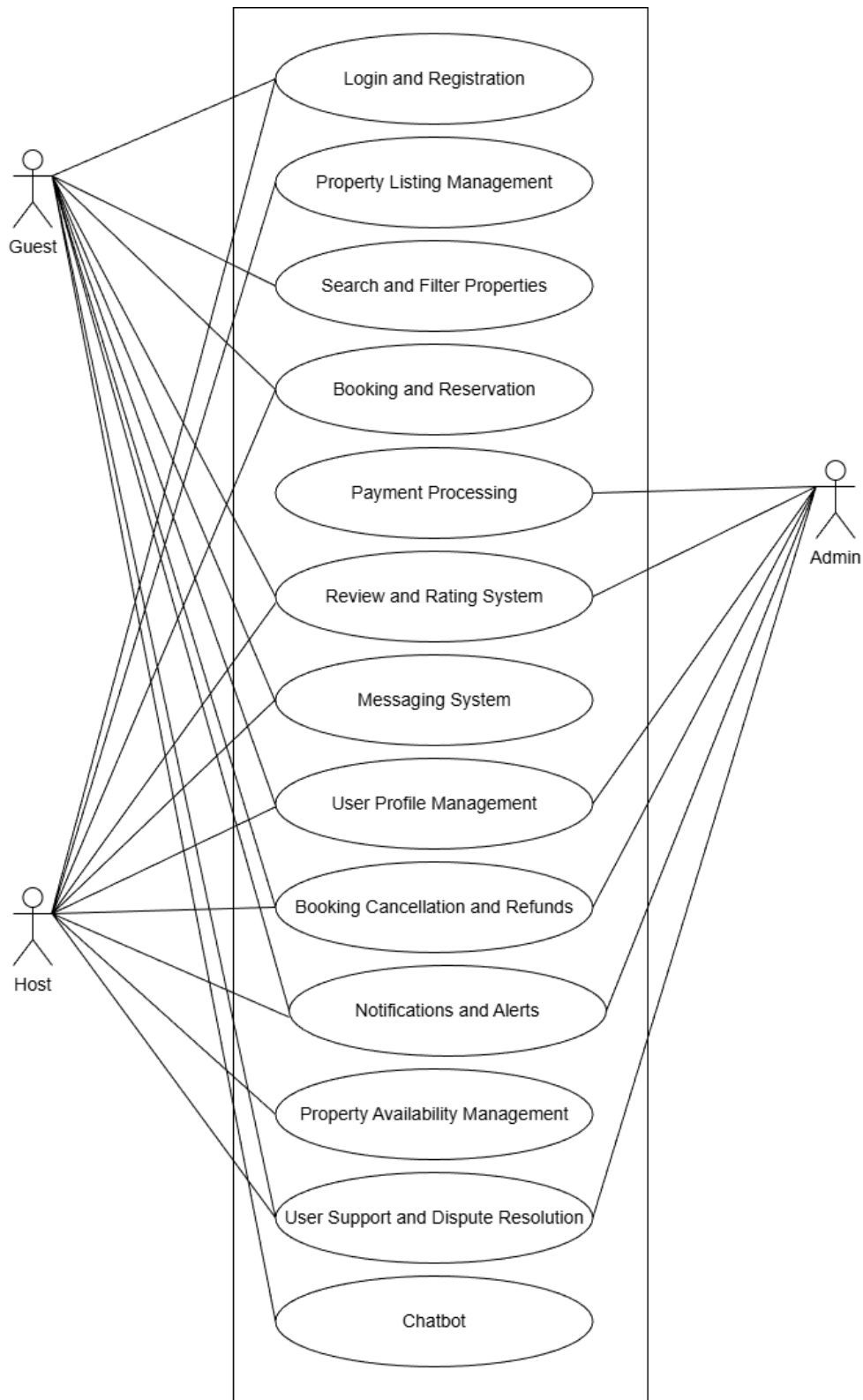


Figure 3-1: Use Case Diagram

: Use case Diagram

2.4.2 Case Description

Case Description-01: Login and Registration

| | | | | | | | | | | | | | | | | | | |
|-----------------------------|---|--|-----|--|----|--|-----|------------------------------|----|---|-----|---|----|--|-----|------------------------|--|---|
| Use Case | Registration & Login | | | | | | | | | | | | | | | | | |
| Goal | Permit users to create accounts and log in to StayBangla. | | | | | | | | | | | | | | | | | |
| Precondition | User has a valid email address and internet access. | | | | | | | | | | | | | | | | | |
| Success End Condition | The user can see "Registration/Login successful". | | | | | | | | | | | | | | | | | |
| Failed End Condition | The user can see "Registration/Login failed". | | | | | | | | | | | | | | | | | |
| Primary Actors: | Guest, Host | | | | | | | | | | | | | | | | | |
| Secondary Actors: | Admin | | | | | | | | | | | | | | | | | |
| Trigger | User attempts to log in or register in the system. | | | | | | | | | | | | | | | | | |
| Description / Main Scenario | <table border="1"> <tr> <td>1.</td> <td>System displays option 'Sign Up' or 'Login'.</td> </tr> <tr> <td>2.</td> <td>User selects Sign Up, fills in details, and chooses guest or host.</td> </tr> <tr> <td>3.</td> <td>Click 'Sign Up' to register.</td> </tr> <tr> <td>4.</td> <td>The system validates the data and creates an account.</td> </tr> <tr> <td>5.</td> <td>Display notification "Registration successful."</td> </tr> <tr> <td>6.</td> <td>A proof email is sent to the user's email address.</td> </tr> </table> | | 1. | System displays option 'Sign Up' or 'Login'. | 2. | User selects Sign Up, fills in details, and chooses guest or host. | 3. | Click 'Sign Up' to register. | 4. | The system validates the data and creates an account. | 5. | Display notification "Registration successful." | 6. | A proof email is sent to the user's email address. | | | | |
| 1. | System displays option 'Sign Up' or 'Login'. | | | | | | | | | | | | | | | | | |
| 2. | User selects Sign Up, fills in details, and chooses guest or host. | | | | | | | | | | | | | | | | | |
| 3. | Click 'Sign Up' to register. | | | | | | | | | | | | | | | | | |
| 4. | The system validates the data and creates an account. | | | | | | | | | | | | | | | | | |
| 5. | Display notification "Registration successful." | | | | | | | | | | | | | | | | | |
| 6. | A proof email is sent to the user's email address. | | | | | | | | | | | | | | | | | |
| Alternative Flows | <table border="1"> <tr> <td>1.1</td> <td>Internet connection issue</td> </tr> <tr> <td></td> <td>1.1.a. Display error: "No network connection."</td> </tr> <tr> <td>1.2</td> <td>Invalid login credentials</td> </tr> <tr> <td></td> <td>1.2.a. Display error: "Incorrect email or password."</td> </tr> <tr> <td>4.1</td> <td>Invalid Input</td> </tr> <tr> <td></td> <td>4.1.a. Display error: "Please enter a valid information"</td> </tr> <tr> <td>4.2</td> <td>Missing Sign-Up fields</td> </tr> <tr> <td></td> <td>4.2.a. Display error: "Please fill in all required fields."</td> </tr> </table> | | 1.1 | Internet connection issue | | 1.1.a. Display error: "No network connection." | 1.2 | Invalid login credentials | | 1.2.a. Display error: "Incorrect email or password." | 4.1 | Invalid Input | | 4.1.a. Display error: "Please enter a valid information" | 4.2 | Missing Sign-Up fields | | 4.2.a. Display error: "Please fill in all required fields." |
| 1.1 | Internet connection issue | | | | | | | | | | | | | | | | | |
| | 1.1.a. Display error: "No network connection." | | | | | | | | | | | | | | | | | |
| 1.2 | Invalid login credentials | | | | | | | | | | | | | | | | | |
| | 1.2.a. Display error: "Incorrect email or password." | | | | | | | | | | | | | | | | | |
| 4.1 | Invalid Input | | | | | | | | | | | | | | | | | |
| | 4.1.a. Display error: "Please enter a valid information" | | | | | | | | | | | | | | | | | |
| 4.2 | Missing Sign-Up fields | | | | | | | | | | | | | | | | | |
| | 4.2.a. Display error: "Please fill in all required fields." | | | | | | | | | | | | | | | | | |

| | | |
|----------------------|---|--|
| | 5.1 | Server error |
| | | 5.1.a. Display error: " "Unable to process your request. Try again later." |
| Quality Requirements | Password must be securely hashed. The system must respond within 3 seconds under normal conditions. | |

Table 4-1: Registration & Login

Case Description-02: Property Listing Management

| | | | | | | | | | | | | |
|-------------------------------------|--|--|----|--|----|---|----|---|----|---|----|---|
| Use Case | Property Listing Management | | | | | | | | | | | |
| Goal | Allow hosts to manage their property listings, including creating, updating, and deleting listings. | | | | | | | | | | | |
| Precondition | Host must be logged into their account. | | | | | | | | | | | |
| Success End Condition | The host successfully creates, edits, or deletes a property listing. The listing awaits admin approval before it is published on the platform. | | | | | | | | | | | |
| Failed End Condition | The host cannot create, update, or delete a property listing due to missing information, system failure, or admin rejection. | | | | | | | | | | | |
| Primary Actors: | Host | | | | | | | | | | | |
| Secondary Actors: | Admin | | | | | | | | | | | |
| Trigger | Host attempts to create, edit, or delete a property listing on the platform. | | | | | | | | | | | |
| Description / Main Success Scenario | <table border="1"> <tr> <td>1.</td> <td>Host navigates to the properties page and selects the option to add or update a listing.</td> </tr> <tr> <td>2.</td> <td>Host enters the property details, such as price, location, amenities, and availability.</td> </tr> <tr> <td>3.</td> <td>Host uploads photos of the property and provides a description.</td> </tr> <tr> <td>4.</td> <td>Host clicks "Submit" to submit the property listing for review.</td> </tr> <tr> <td>5.</td> <td>Admin reviews the listing for approval.</td> </tr> </table> | | 1. | Host navigates to the properties page and selects the option to add or update a listing. | 2. | Host enters the property details, such as price, location, amenities, and availability. | 3. | Host uploads photos of the property and provides a description. | 4. | Host clicks "Submit" to submit the property listing for review. | 5. | Admin reviews the listing for approval. |
| 1. | Host navigates to the properties page and selects the option to add or update a listing. | | | | | | | | | | | |
| 2. | Host enters the property details, such as price, location, amenities, and availability. | | | | | | | | | | | |
| 3. | Host uploads photos of the property and provides a description. | | | | | | | | | | | |
| 4. | Host clicks "Submit" to submit the property listing for review. | | | | | | | | | | | |
| 5. | Admin reviews the listing for approval. | | | | | | | | | | | |

| | | | | | | | | | | | | | | | | | |
|----------------------|---|-----|---|--|--|-----|-----------------|--|---|-----|---------------|--|--|-----|--------------------|--|---|
| | <table border="1"> <tr> <td data-bbox="611 194 687 293">6.</td> <td data-bbox="687 194 1417 293">Once approved by the admin, the listing is made visible on the platform for guests to view.</td> </tr> </table> | 6. | Once approved by the admin, the listing is made visible on the platform for guests to view. | | | | | | | | | | | | | | |
| 6. | Once approved by the admin, the listing is made visible on the platform for guests to view. | | | | | | | | | | | | | | | | |
| Alternative Flows | <table border="1"> <tr> <td data-bbox="611 750 687 813">1.1</td> <td data-bbox="687 750 1417 813">Missing or Invalid Information</td> </tr> <tr> <td data-bbox="611 813 687 1014"></td> <td data-bbox="687 813 1417 1014"> 1.1.a. Host misses a required field or enters invalid information (e.g., incomplete address, incorrect price). 1.1.b. Display error: "Please fill in all required information." </td> </tr> <tr> <td data-bbox="611 1014 687 1077">1.2</td> <td data-bbox="687 1014 1417 1077">Admin Rejection</td> </tr> <tr> <td data-bbox="611 1077 687 1279"></td> <td data-bbox="687 1077 1417 1279"> 1.2.a. Admin rejects the listing due to policy violations or incomplete details. 1.2.b. Display message: "Your listing has been rejected. Please update the information and resubmit." </td> </tr> <tr> <td data-bbox="611 1279 687 1341">2.1</td> <td data-bbox="687 1279 1417 1341">Server Error:</td> </tr> <tr> <td data-bbox="611 1341 687 1543"></td> <td data-bbox="687 1341 1417 1543"> 2.1.a. A technical error occurs while submitting the listing. 2.1.b. Display error: "Unable to process your request. Please try again later." </td> </tr> <tr> <td data-bbox="611 1543 687 1606">3.1</td> <td data-bbox="687 1543 1417 1606">Duplicate Listing:</td> </tr> <tr> <td data-bbox="611 1606 687 1807"></td> <td data-bbox="687 1606 1417 1807"> 3.1.a. Host tries to submit a listing that already exists. 3.1.b. Display error: "A listing with this address already exists. Please modify the details or try a different address." </td> </tr> </table> | 1.1 | Missing or Invalid Information | | 1.1.a. Host misses a required field or enters invalid information (e.g., incomplete address, incorrect price). 1.1.b. Display error: "Please fill in all required information." | 1.2 | Admin Rejection | | 1.2.a. Admin rejects the listing due to policy violations or incomplete details. 1.2.b. Display message: "Your listing has been rejected. Please update the information and resubmit." | 2.1 | Server Error: | | 2.1.a. A technical error occurs while submitting the listing. 2.1.b. Display error: "Unable to process your request. Please try again later." | 3.1 | Duplicate Listing: | | 3.1.a. Host tries to submit a listing that already exists. 3.1.b. Display error: "A listing with this address already exists. Please modify the details or try a different address." |
| 1.1 | Missing or Invalid Information | | | | | | | | | | | | | | | | |
| | 1.1.a. Host misses a required field or enters invalid information (e.g., incomplete address, incorrect price). 1.1.b. Display error: "Please fill in all required information." | | | | | | | | | | | | | | | | |
| 1.2 | Admin Rejection | | | | | | | | | | | | | | | | |
| | 1.2.a. Admin rejects the listing due to policy violations or incomplete details. 1.2.b. Display message: "Your listing has been rejected. Please update the information and resubmit." | | | | | | | | | | | | | | | | |
| 2.1 | Server Error: | | | | | | | | | | | | | | | | |
| | 2.1.a. A technical error occurs while submitting the listing. 2.1.b. Display error: "Unable to process your request. Please try again later." | | | | | | | | | | | | | | | | |
| 3.1 | Duplicate Listing: | | | | | | | | | | | | | | | | |
| | 3.1.a. Host tries to submit a listing that already exists. 3.1.b. Display error: "A listing with this address already exists. Please modify the details or try a different address." | | | | | | | | | | | | | | | | |
| Quality Requirements | Only the listing owner and the admin can modify or remove the property. | | | | | | | | | | | | | | | | |

Table 4-2: Property Listing Management

Case Description-03: Search and Filter Properties

| | | | | | | | | | | | | | | |
|-------------------------------------|--|--|----|------------------------------|----|---|----|---------------------------------|----|---------------------------------------|----|---|----|---|
| Use Case | Search and Filter Properties | | | | | | | | | | | | | |
| Goal | Let guests look for properties and narrow the results using filters. | | | | | | | | | | | | | |
| Precondition | The guest must be able to access the platform. | | | | | | | | | | | | | |
| Success End Condition | A list of matching properties appears. | | | | | | | | | | | | | |
| Failed End Condition | No results show up, or the system cannot process the search. | | | | | | | | | | | | | |
| Primary Actors: | Guest | | | | | | | | | | | | | |
| Secondary Actors: | System | | | | | | | | | | | | | |
| Trigger | The guest type location in the search bar or choose filter. | | | | | | | | | | | | | |
| Description / Main Success Scenario | <table border="1"> <tr> <td>1.</td> <td>Guest opens the search page.</td> </tr> <tr> <td>2.</td> <td>Guest enters a location or picks one or more filters.</td> </tr> <tr> <td>3.</td> <td>Guest clicks the search button.</td> </tr> <tr> <td>4.</td> <td>The system checks the search request.</td> </tr> <tr> <td>5.</td> <td>Properties that match the request appear.</td> </tr> <tr> <td>6.</td> <td>The guest selects a property to see more details.</td> </tr> </table> | | 1. | Guest opens the search page. | 2. | Guest enters a location or picks one or more filters. | 3. | Guest clicks the search button. | 4. | The system checks the search request. | 5. | Properties that match the request appear. | 6. | The guest selects a property to see more details. |
| 1. | Guest opens the search page. | | | | | | | | | | | | | |
| 2. | Guest enters a location or picks one or more filters. | | | | | | | | | | | | | |
| 3. | Guest clicks the search button. | | | | | | | | | | | | | |
| 4. | The system checks the search request. | | | | | | | | | | | | | |
| 5. | Properties that match the request appear. | | | | | | | | | | | | | |
| 6. | The guest selects a property to see more details. | | | | | | | | | | | | | |

| | | |
|----------------------|--|---|
| Alternative Flows | 2.1 | No Input |
| | | 2.1.a. The guest clicks search without entering anything. 2.1.b. The system asks the guest to add a location or choose a filter. |
| | 2.2 | Wrong Filter Values |
| | | 2.2.a. The guest enters values that do not make sense. 2.2.b. The system asks the guest to correct them. |
| | 4.1 | No Matching Results |
| | | 4.1.a. No property fits the selected filters. 4.1.b. The system shows a message saying no results were found. |
| | 5.1 | System Problem |
| | | 5.1.a. The system cannot load the results. 5.1.b. A message appears telling the guest to try again later. |
| Quality Requirements | Only properties that match the chosen filters should appear. | |

Table 4-3: Search and Filter Properties

Case Description-04: Booking and Reservation

| | |
|-----------------------|--|
| Use Case | Booking and Reservation |
| Goal | Let guests book a property for the days they want. |
| Precondition | The guest must be logged in. |
| Success End Condition | The booking is saved and confirmed. |
| Failed End Condition | The booking cannot be completed because of wrong dates, payment issues, or unavailability. |
| Primary Actors: | Guest |
| Secondary Actors: | Host |
| Trigger | The guest presses the “Book” button on a property page. |

| | | |
|---|--|---|
| Description / Main Success Scenario | 1. | The guest chooses check-in and check-out dates. |
| | 2. | The guest checks the booking details on the screen. |
| | 3. | The guest moves to the payment page. |
| | 4. | The guest pays for the booking. |
| | 5. | The system saves the booking. |
| | 6. | Both the guest and the host receive a confirmation message. |
| | Alternative Flows | 1.1 |
| | | 1.1.a. The dates the guest picked are not free. 1.1.b. The system asks the guest to pick new dates. |
| 1.2 | | Wrong Date Order |
| | | 2.1.a. The guest selects an end date that comes before the start date. 2.1.b. The system asks the guest to correct the dates. |
| 4.1 | | Payment Not Completed |
| | | 4.1.a. The payment fails. 4.1.b. The system cancels the attempt and tells the guest to try again. |
| 6.1 | | Host Approval Needed |
| | | 6.1.a. Some listings need the host to approve bookings. 5.1.b. The booking stays “pending” until the host accepts or rejects it. |
| Quality Requirements | Only free dates should appear as bookable. | |

Table 4-4: Booking and Reservation

Case Description-05: Payment Processing

| | |
|----------|--------------------|
| Use Case | Payment Processing |
|----------|--------------------|

| | | | | | | | | | | | | | |
|---|--|-----|-------------------------------------|----|--|-----|--|----|--|-----|--------------------------|----|---|
| Goal | Let guests pay for their bookings safely. | | | | | | | | | | | | |
| Precondition | The guest must have a booking request ready to pay for. | | | | | | | | | | | | |
| Success End Condition | The payment goes through and the booking is confirmed. | | | | | | | | | | | | |
| Failed End Condition | The payment fails or cannot be completed. | | | | | | | | | | | | |
| Primary Actors: | Guest | | | | | | | | | | | | |
| Secondary Actors: | Host, Admin | | | | | | | | | | | | |
| Trigger | The guest clicks the “Pay” button during the booking process. | | | | | | | | | | | | |
| Description / Main Success Scenario | <table border="1"> <tr> <td>1.</td> <td>The guest goes to the payment page.</td> </tr> <tr> <td>2.</td> <td>The guest selects a payment method.</td> </tr> <tr> <td>3.</td> <td>The guest enters the required payment details.</td> </tr> <tr> <td>4.</td> <td>The system sends the payment to the payment gateway.</td> </tr> <tr> <td>5.</td> <td>The payment is accepted.</td> </tr> <tr> <td>6.</td> <td>The booking is confirmed and both the guest and host are notified.</td> </tr> </table> | 1. | The guest goes to the payment page. | 2. | The guest selects a payment method. | 3. | The guest enters the required payment details. | 4. | The system sends the payment to the payment gateway. | 5. | The payment is accepted. | 6. | The booking is confirmed and both the guest and host are notified. |
| 1. | The guest goes to the payment page. | | | | | | | | | | | | |
| 2. | The guest selects a payment method. | | | | | | | | | | | | |
| 3. | The guest enters the required payment details. | | | | | | | | | | | | |
| 4. | The system sends the payment to the payment gateway. | | | | | | | | | | | | |
| 5. | The payment is accepted. | | | | | | | | | | | | |
| 6. | The booking is confirmed and both the guest and host are notified. | | | | | | | | | | | | |
| Alternative Flows | <table border="1"> <tr> <td>3.1</td> <td>Incorrect Payment Details</td> </tr> <tr> <td></td> <td>3.1.a. The guest enters wrong card or account details. 3.1.b. The system asks the guest to fix the information.</td> </tr> <tr> <td>4.1</td> <td>Payment Declined</td> </tr> <tr> <td></td> <td>4.1.a. The bank or payment provider rejects the payment. 4.1.b. The system notifies the guest and asks them to try again.</td> </tr> <tr> <td>5.1</td> <td>Gateway Unavailable</td> </tr> <tr> <td></td> <td>5.1.a. The payment service (e.g., SSLCommerz) is temporarily down. 5.1.b. The system asks the guest to try a different payment method or return later.</td> </tr> </table> | 3.1 | Incorrect Payment Details | | 3.1.a. The guest enters wrong card or account details. 3.1.b. The system asks the guest to fix the information. | 4.1 | Payment Declined | | 4.1.a. The bank or payment provider rejects the payment. 4.1.b. The system notifies the guest and asks them to try again. | 5.1 | Gateway Unavailable | | 5.1.a. The payment service (e.g., SSLCommerz) is temporarily down. 5.1.b. The system asks the guest to try a different payment method or return later. |
| 3.1 | Incorrect Payment Details | | | | | | | | | | | | |
| | 3.1.a. The guest enters wrong card or account details. 3.1.b. The system asks the guest to fix the information. | | | | | | | | | | | | |
| 4.1 | Payment Declined | | | | | | | | | | | | |
| | 4.1.a. The bank or payment provider rejects the payment. 4.1.b. The system notifies the guest and asks them to try again. | | | | | | | | | | | | |
| 5.1 | Gateway Unavailable | | | | | | | | | | | | |
| | 5.1.a. The payment service (e.g., SSLCommerz) is temporarily down. 5.1.b. The system asks the guest to try a different payment method or return later. | | | | | | | | | | | | |

| | |
|----------------------|--|
| | |
| Quality Requirements | All payment details must be encrypted. |

Table 4-5: Payment Processing

Case Description-06: Review and Rating System

| | | | | | | | | | | | | | | |
|---|--|--|----|---|----|---|----|---|----|--|----|---|----|---|
| Use Case | Review and Rating System | | | | | | | | | | | | | |
| Goal | Permit guests to rate and review the properties they stayed in. | | | | | | | | | | | | | |
| Precondition | The guest has completed their stay at the property. | | | | | | | | | | | | | |
| Success End Condition | The guest's review and rating are submitted and displayed. | | | | | | | | | | | | | |
| Failed End Condition | The review was not submitted due to errors or missing information. | | | | | | | | | | | | | |
| Primary Actors: | Guest | | | | | | | | | | | | | |
| Secondary Actors: | Host | | | | | | | | | | | | | |
| Trigger | The guest is prompted to leave a review after checkout. | | | | | | | | | | | | | |
| Description / Main Success Scenario | <table border="1"> <tr> <td>1.</td> <td>After checking out, the guest is asked to leave a review.</td> </tr> <tr> <td>2.</td> <td>The guest rates the property on a scale 1-5 stars and writes a comment.</td> </tr> <tr> <td>3.</td> <td>The guest clicks "Submit" to post the review.</td> </tr> <tr> <td>4.</td> <td>The review is saved and displayed on the property's page for other users to see.</td> </tr> <tr> <td>5.</td> <td>The host receives a notice of the new review.</td> </tr> <tr> <td>6.</td> <td>The review becomes visible on the platform.</td> </tr> </table> | | 1. | After checking out, the guest is asked to leave a review. | 2. | The guest rates the property on a scale 1-5 stars and writes a comment. | 3. | The guest clicks "Submit" to post the review. | 4. | The review is saved and displayed on the property's page for other users to see. | 5. | The host receives a notice of the new review. | 6. | The review becomes visible on the platform. |
| 1. | After checking out, the guest is asked to leave a review. | | | | | | | | | | | | | |
| 2. | The guest rates the property on a scale 1-5 stars and writes a comment. | | | | | | | | | | | | | |
| 3. | The guest clicks "Submit" to post the review. | | | | | | | | | | | | | |
| 4. | The review is saved and displayed on the property's page for other users to see. | | | | | | | | | | | | | |
| 5. | The host receives a notice of the new review. | | | | | | | | | | | | | |
| 6. | The review becomes visible on the platform. | | | | | | | | | | | | | |

| | | |
|----------------------|---|--|
| Alternative Flows | 3.1 | Incomplete Review: |
| | | 3.1.a. If the guest doesn't fill in all required fields, the system prompts them to complete the review. |
| | 3.2 | Editing Reviews |
| | | 3.2.a. The guest can edit their review within a specific time frame |
| | 4.1 | Admin Removal |
| | | 4.1.a. If a review violates platform guidelines, it can be flagged and removed by the admin. |
| Quality Requirements | The host should be notified within 1 minutes of a new review. | |

Table 4-6: Review and Rating System

Case Description-07: Messaging System

| | | |
|-------------------------------------|--|---|
| Use Case | Messaging System | |
| Goal | Permit guests to rate and review the properties they stayed in. | |
| Precondition | The guest has completed their stay at the property. | |
| Success End Condition | The guest's review and rating are submitted and displayed. | |
| Failed End Condition | The review was not submitted due to errors or missing information. | |
| Primary Actors: | Guest | |
| Secondary Actors: | Host | |
| Trigger | The guest is prompted to leave a review after checkout. | |
| Description / Main Success Scenario | 1. | After checking out, the guest is asked to leave a review. |
| | 2. | The guest rates the property on a scale 1-5 stars and writes a comment. |
| | 3. | The guest clicks "Submit" to post the review. |

| | | |
|----------------------|-----|--|
| | 4. | The review is saved and displayed on the property's page for other users to see. |
| | 5. | The host receives a notice of the new review. |
| | 6. | The review becomes visible on the platform. |
| Alternative Flows | 3.1 | Incomplete Review: |
| | | 3.1.a. If the guest doesn't fill in all required fields, the system prompts them to complete the review. |
| | 3.2 | Editing Reviews |
| | | 3.2.a. The guest can edit their review within a specific time frame |
| | 4.1 | Admin Removal |
| | | 4.1.a. If a review violates platform guidelines, it can be flagged and removed by the admin. |
| Quality Requirements | | The host should be notified within 1 minutes of a new review. |

Table 4-7: Messaging System

Case Description-08: User Profile Management

| | |
|-----------------------|--|
| Use Case | User Profile Management |
| Goal | Guests and hosts should have access to see and edit their personal details on the platform |
| Precondition | Note Both the guest and host must be logged in. |
| Success End Condition | The new profile details is saved and displayed properly on the user profile page. |
| Failed End Condition | The new data are not written to file and the user is alerted of that discrepancy. |
| Primary Actors: | Guest, Host |
| Secondary Actors: | System |
| Trigger | A first user selects to obtain or edit a profile. |

| | | |
|-------------------------------------|---|--|
| Description / Main Success Scenario | 1. | Here, the user accesses profile section from the dashboard. |
| | 2. | The details of the current user are presented such as name, email and contact details etc. |
| | 3. | The user modifies one or more of the fields. |
| | 4. | On the profile there is a button labeled “Save.” |
| | 5. | It’s checked against the information fill in. |
| | 6. | The changes are saved to the system and the updated is acknowledged. |
| | Alternative Flows | 3.1 |
| | 3.1.a. | It identifies that there are mandatory fields that have not been completed. |
| | 3.1.b. | It shows: “Please complete all of the required fields.” |
| | 4.1 | Invalid Data Format |
| | 4.1.a. | System can detect wrong formats |
| | 4.1.b. | It says ‘Invalid input. Please check the entered data.’ |
| | 4.2 | Connection Issue |
| | 4.2.a. | Because of a connection error, the system is not able to finish the update. |
| | 4.2.b. | It keeps saying: “Can’ t update you right now. Please try again.” |
| Quality Requirements | User data are to be safeguarded from unauthorized access. | |

Table 4-8: User Profile Management

Case Description-09: Booking Cancellation and Refunds

| | |
|----------|---|
| Use Case | Booking Cancellation and Refunds |
| Goal | Let guests and hosts cancel bookings and let the host manage refunds according to cancellation rules. |

| | | | | | | | | | | | | | | |
|-------------------------------------|---|--|-----|--|----|---|-----|-------------------------------------|----|---|-----|---------------------------------|----|--|
| Precondition | The guest or host must be registered and having a valid property or booking in the system. | | | | | | | | | | | | | |
| Success End Condition | The booking is cancelled and the refund is issued and settled. | | | | | | | | | | | | | |
| Failed End Condition | This booking is not 'cancelled' for policy constraints or technical problems. | | | | | | | | | | | | | |
| Primary Actors: | Guest, Host | | | | | | | | | | | | | |
| Secondary Actors: | Admin | | | | | | | | | | | | | |
| Trigger | User taps on booking, then proceed on cancel. | | | | | | | | | | | | | |
| Description / Main Success Scenario | <table border="1"> <tr> <td>1.</td> <td>User accesses a page which contains the booking history.</td> </tr> <tr> <td>2.</td> <td>System is displaying all bookings - both current and historical.</td> </tr> <tr> <td>3.</td> <td>User chosen booking to be canceled.</td> </tr> <tr> <td>4.</td> <td>System brings up the cancellation policy for that booking.</td> </tr> <tr> <td>5.</td> <td>User confirms the cancellation.</td> </tr> <tr> <td>6.</td> <td>System finishes the cancel process and modified the status of booking.</td> </tr> </table> | | 1. | User accesses a page which contains the booking history. | 2. | System is displaying all bookings - both current and historical. | 3. | User chosen booking to be canceled. | 4. | System brings up the cancellation policy for that booking. | 5. | User confirms the cancellation. | 6. | System finishes the cancel process and modified the status of booking. |
| 1. | User accesses a page which contains the booking history. | | | | | | | | | | | | | |
| 2. | System is displaying all bookings - both current and historical. | | | | | | | | | | | | | |
| 3. | User chosen booking to be canceled. | | | | | | | | | | | | | |
| 4. | System brings up the cancellation policy for that booking. | | | | | | | | | | | | | |
| 5. | User confirms the cancellation. | | | | | | | | | | | | | |
| 6. | System finishes the cancel process and modified the status of booking. | | | | | | | | | | | | | |
| Alternative Flows | <table border="1"> <tr> <td>2.1</td> <td>Host-Cancelled Booking</td> </tr> <tr> <td></td> <td>2.1.a. Host cancels because there is something wrong with the property 2.1.b. The system still informs the guest and refunds or replaces as necessary.</td> </tr> <tr> <td>4.1</td> <td>Booking Outside Refund Window</td> </tr> <tr> <td></td> <td>4.1.a. The system verifies the policy and determines refund is disallowed. 4.1.b. Message displayed: "We are unable to refund this booking as per the policy."</td> </tr> <tr> <td>6.1</td> <td>Payment Reversal Issue</td> </tr> <tr> <td></td> <td>6.1.a. System tries to refund it but payment gateway gives an error.</td> </tr> </table> | | 2.1 | Host-Cancelled Booking | | 2.1.a. Host cancels because there is something wrong with the property 2.1.b. The system still informs the guest and refunds or replaces as necessary. | 4.1 | Booking Outside Refund Window | | 4.1.a. The system verifies the policy and determines refund is disallowed. 4.1.b. Message displayed: "We are unable to refund this booking as per the policy." | 6.1 | Payment Reversal Issue | | 6.1.a. System tries to refund it but payment gateway gives an error. |
| 2.1 | Host-Cancelled Booking | | | | | | | | | | | | | |
| | 2.1.a. Host cancels because there is something wrong with the property 2.1.b. The system still informs the guest and refunds or replaces as necessary. | | | | | | | | | | | | | |
| 4.1 | Booking Outside Refund Window | | | | | | | | | | | | | |
| | 4.1.a. The system verifies the policy and determines refund is disallowed. 4.1.b. Message displayed: "We are unable to refund this booking as per the policy." | | | | | | | | | | | | | |
| 6.1 | Payment Reversal Issue | | | | | | | | | | | | | |
| | 6.1.a. System tries to refund it but payment gateway gives an error. | | | | | | | | | | | | | |

| | |
|----------------------|--|
| | 6.1.b. Message shown: “Refund failed. Please try again later.” |
| Quality Requirements | The host should be notified within 1 minutes of a new review. |

Table 4-9: Booking Cancellation and Refunds

Case Description-10: Notifications and Alerts

| | | | | | | | | | | | |
|-----------------------------|--|----|---|----|---|----|---|----|-----------------------|----|--|
| Use Case | Notifications and Alerts | | | | | | | | | | |
| Goal | Keep guests, hosts, and admins up to date on booking status changes, cancellations, messages, and other key actions. | | | | | | | | | | |
| Precondition | A system event must take place | | | | | | | | | | |
| Success End Condition | Notification is pushed and displayed to the right user. | | | | | | | | | | |
| Failed End Condition | The alert is not received or displayed to the user. | | | | | | | | | | |
| Primary Actors: | Guest, Host | | | | | | | | | | |
| Secondary Actors: | Admin, System | | | | | | | | | | |
| Trigger | A system event that implies user should be notified. | | | | | | | | | | |
| Description / Main Scenario | <table border="1"> <tr> <td>1.</td> <td>A booking, cancellation, message or update is made.</td> </tr> <tr> <td>2.</td> <td>The system determines who it should notify.</td> </tr> <tr> <td>3.</td> <td>The alert is being prepared by the application.</td> </tr> <tr> <td>4.</td> <td>The user is notified.</td> </tr> <tr> <td>5.</td> <td>The user reads and opens the notification.</td> </tr> </table> | 1. | A booking, cancellation, message or update is made. | 2. | The system determines who it should notify. | 3. | The alert is being prepared by the application. | 4. | The user is notified. | 5. | The user reads and opens the notification. |
| 1. | A booking, cancellation, message or update is made. | | | | | | | | | | |
| 2. | The system determines who it should notify. | | | | | | | | | | |
| 3. | The alert is being prepared by the application. | | | | | | | | | | |
| 4. | The user is notified. | | | | | | | | | | |
| 5. | The user reads and opens the notification. | | | | | | | | | | |
| Alternative Flows | | | | | | | | | | | |

| | | |
|----------------------|---|--|
| | 2.1 | Duplicate Notification Prevention |
| | | 2.1.a. System verifies whether an alert for the same event was already sent. If so, it won't send another one. |
| | 3.1 | Delivery Issue |
| | | 3.1.a. If inability to send. 3.1.b. System tries to retransmit or flips over to in-app alert. |
| | 5.1 | User Ignoring Notification |
| | | 5.1.a. The user ignoring notification is one of the major reasons, users do not open a notification when it comes on screen. |
| Quality Requirements | Alerts need to show and not disappear until the user opens or clears. | |

Table 4-10: Notifications and Alerts

Case Description-11: Property Availability Management

| | |
|-----------------------|--|
| Use Case | Property Availability Management |
| Goal | Let hosts set available / unavailable dates for their lodging to keep things in time and allow guests to book correctly. |
| Precondition | Host user should be logged in and he must have at least 1 listed property. |
| Success End Condition | Availability update is being saved and displayed properly on the property calendar. |
| Failed End Condition | The system doesn't save changes or displays old availability. |
| Primary Actors: | Host |
| Secondary Actors: | System |
| Trigger | It is the host that is opening dates up on a property. |

| | | | | | | | | | | | | | | | | | |
|-------------------------------------|---|-----|---|----|--|-----|---|----|--|-----|---|----|---|-----|----------------|--|---|
| Description / Main Success Scenario | <table border="1"> <tr> <td data-bbox="612 230 687 309">1.</td> <td data-bbox="687 230 1417 309">Host is signing up to become a host and navigates to /listings.</td> </tr> <tr> <td data-bbox="612 309 687 365">2.</td> <td data-bbox="687 309 1417 365">Host launches the calendar for that property.</td> </tr> <tr> <td data-bbox="612 365 687 421">3.</td> <td data-bbox="687 365 1417 421">Date is marked as available or unavailable by host.</td> </tr> <tr> <td data-bbox="612 421 687 477">4.</td> <td data-bbox="687 421 1417 477">Host saves the changes.</td> </tr> <tr> <td data-bbox="612 477 687 533">5.</td> <td data-bbox="687 477 1417 533">System updates the property's availability.</td> </tr> <tr> <td data-bbox="612 533 687 611">6.</td> <td data-bbox="687 533 1417 611">Visitors searching for reservations will see the new dates.</td> </tr> </table> | 1. | Host is signing up to become a host and navigates to /listings. | 2. | Host launches the calendar for that property. | 3. | Date is marked as available or unavailable by host. | 4. | Host saves the changes. | 5. | System updates the property's availability. | 6. | Visitors searching for reservations will see the new dates. | | | | |
| 1. | Host is signing up to become a host and navigates to /listings. | | | | | | | | | | | | | | | | |
| 2. | Host launches the calendar for that property. | | | | | | | | | | | | | | | | |
| 3. | Date is marked as available or unavailable by host. | | | | | | | | | | | | | | | | |
| 4. | Host saves the changes. | | | | | | | | | | | | | | | | |
| 5. | System updates the property's availability. | | | | | | | | | | | | | | | | |
| 6. | Visitors searching for reservations will see the new dates. | | | | | | | | | | | | | | | | |
| Alternative Flows | <table border="1"> <tr> <td data-bbox="612 723 687 779">3.1</td> <td data-bbox="687 723 1417 779">Overlapping Bookings</td> </tr> <tr> <td data-bbox="612 779 687 947"></td> <td data-bbox="687 779 1417 947"> 3.1.a. The host attempts to block already booked dates. 3.1.b. System halts and displays: "these dates cannot be changed because a booking already exists." </td> </tr> <tr> <td data-bbox="612 947 687 1003">3.2</td> <td data-bbox="687 947 1417 1003">Invalid Date Selection</td> </tr> <tr> <td data-bbox="612 1003 687 1081"></td> <td data-bbox="687 1003 1417 1081"> 3.2.a. Host chooses checkout date before check-in. 3.2.b. System shows: "Invalid date range." </td> </tr> <tr> <td data-bbox="612 1081 687 1137">4.1</td> <td data-bbox="687 1081 1417 1137">Save Failure</td> </tr> <tr> <td data-bbox="612 1137 687 1305"></td> <td data-bbox="687 1137 1417 1305"> 4.1.a. There is a network or system problem that prevents the save. 4.1.b. System messages: "Could not save availability. Please try again." </td> </tr> <tr> <td data-bbox="612 1305 687 1361">5.1</td> <td data-bbox="687 1305 1417 1361">Admin Override</td> </tr> <tr> <td data-bbox="612 1361 687 1485"></td> <td data-bbox="687 1361 1417 1485"> 5.1.a. Admin changes availability in the event of argument or property issue. 5.1.b. The host receives a notification of the change. </td> </tr> </table> | 3.1 | Overlapping Bookings | | 3.1.a. The host attempts to block already booked dates. 3.1.b. System halts and displays: "these dates cannot be changed because a booking already exists." | 3.2 | Invalid Date Selection | | 3.2.a. Host chooses checkout date before check-in. 3.2.b. System shows: "Invalid date range." | 4.1 | Save Failure | | 4.1.a. There is a network or system problem that prevents the save. 4.1.b. System messages: "Could not save availability. Please try again." | 5.1 | Admin Override | | 5.1.a. Admin changes availability in the event of argument or property issue. 5.1.b. The host receives a notification of the change. |
| 3.1 | Overlapping Bookings | | | | | | | | | | | | | | | | |
| | 3.1.a. The host attempts to block already booked dates. 3.1.b. System halts and displays: "these dates cannot be changed because a booking already exists." | | | | | | | | | | | | | | | | |
| 3.2 | Invalid Date Selection | | | | | | | | | | | | | | | | |
| | 3.2.a. Host chooses checkout date before check-in. 3.2.b. System shows: "Invalid date range." | | | | | | | | | | | | | | | | |
| 4.1 | Save Failure | | | | | | | | | | | | | | | | |
| | 4.1.a. There is a network or system problem that prevents the save. 4.1.b. System messages: "Could not save availability. Please try again." | | | | | | | | | | | | | | | | |
| 5.1 | Admin Override | | | | | | | | | | | | | | | | |
| | 5.1.a. Admin changes availability in the event of argument or property issue. 5.1.b. The host receives a notification of the change. | | | | | | | | | | | | | | | | |
| Quality Requirements | Hosts should not be able to change the dates a confirmed booking covers with this system. | | | | | | | | | | | | | | | | |

Table 4-11: Property Availability Management

Case Description-12: User Support and Dispute Resolution

| | |
|----------|---|
| Use Case | User Support and Dispute Resolution |
| Goal | Aid guests and hosts in getting help when things go wrong, and offer a fair way to settle disputes. |

| | | | | | | | | | | | | | |
|-------------------------------------|---|-----|--|----|---|-----|--|----|---|-----|---|--|--|
| Precondition | The user is logged in, and has an issue about booking, payment, property or communication. | | | | | | | | | | | | |
| Success End Condition | The problem is both reconsidered and resolved, with the two parties apprised of the result. | | | | | | | | | | | | |
| Failed End Condition | The question is unanswered because it lacks information, opinion, or system. | | | | | | | | | | | | |
| Primary Actors: | Guest, Host | | | | | | | | | | | | |
| Secondary Actors: | Admin | | | | | | | | | | | | |
| Trigger | A problem is reported by a user through the support area. | | | | | | | | | | | | |
| Description / Main Success Scenario | <table border="1"> <tr> <td>1.</td> <td>User opens support section from his dashboard.</td> </tr> <tr> <td>2.</td> <td>The user reports the problem and places the support request.</td> </tr> <tr> <td>3.</td> <td>System logs the requested and send out to admin.</td> </tr> <tr> <td>4.</td> <td>Admin reviews the submitted information.</td> </tr> <tr> <td>5.</td> <td>System informs guest and host about the resolution.</td> </tr> </table> | 1. | User opens support section from his dashboard. | 2. | The user reports the problem and places the support request. | 3. | System logs the requested and send out to admin. | 4. | Admin reviews the submitted information. | 5. | System informs guest and host about the resolution. | | |
| 1. | User opens support section from his dashboard. | | | | | | | | | | | | |
| 2. | The user reports the problem and places the support request. | | | | | | | | | | | | |
| 3. | System logs the requested and send out to admin. | | | | | | | | | | | | |
| 4. | Admin reviews the submitted information. | | | | | | | | | | | | |
| 5. | System informs guest and host about the resolution. | | | | | | | | | | | | |
| Alternative Flows | <table border="1"> <tr> <td>4.1</td> <td>Missing or unclear details</td> </tr> <tr> <td></td> <td>4.1.a. There is missing info and it's not allowing me to continue. 4.1.b. User is invited to elaborate more.</td> </tr> <tr> <td>4.2</td> <td>Conflict between guest and host</td> </tr> <tr> <td></td> <td>4.2.a. Both parties give conflicting statements. 4.2.b. Admin looks at evidence and choose what they feel is fair.</td> </tr> <tr> <td>5.1</td> <td>Issue outside platform responsibility</td> </tr> <tr> <td></td> <td>5.1.a. Issue is not around bookings or platform use. 5.1.b. System tells user that the issue can not a be solved.</td> </tr> </table> | 4.1 | Missing or unclear details | | 4.1.a. There is missing info and it's not allowing me to continue. 4.1.b. User is invited to elaborate more. | 4.2 | Conflict between guest and host | | 4.2.a. Both parties give conflicting statements. 4.2.b. Admin looks at evidence and choose what they feel is fair. | 5.1 | Issue outside platform responsibility | | 5.1.a. Issue is not around bookings or platform use. 5.1.b. System tells user that the issue can not a be solved. |
| 4.1 | Missing or unclear details | | | | | | | | | | | | |
| | 4.1.a. There is missing info and it's not allowing me to continue. 4.1.b. User is invited to elaborate more. | | | | | | | | | | | | |
| 4.2 | Conflict between guest and host | | | | | | | | | | | | |
| | 4.2.a. Both parties give conflicting statements. 4.2.b. Admin looks at evidence and choose what they feel is fair. | | | | | | | | | | | | |
| 5.1 | Issue outside platform responsibility | | | | | | | | | | | | |
| | 5.1.a. Issue is not around bookings or platform use. 5.1.b. System tells user that the issue can not a be solved. | | | | | | | | | | | | |

| | |
|----------------------|---|
| Quality Requirements | Record of all dispute cases shall be archived safely and maintained for future reference. |
|----------------------|---|

Table 4-12: User Support and Dispute Resolution

Case Description-13: Chatbot

| | | | | | | | | | | | | | | |
|-----------------------------|--|--|-----|--------------------------------|----|---|-----|--|----|--|----|---|----|-----------------------------|
| Use Case | Chatbot | | | | | | | | | | | | | |
| Goal | Quick answers to popular: Answer common questions, and give property recommendations according to user input. | | | | | | | | | | | | | |
| Precondition | The user must be able to access the platform. | | | | | | | | | | | | | |
| Success End Condition | The chatbot responds in a friendly and useful manner. | | | | | | | | | | | | | |
| Failed End Condition | Chatbot is not able to respond to question and prompts user "Please contact support. | | | | | | | | | | | | | |
| Primary Actors: | Guest, Host | | | | | | | | | | | | | |
| Secondary Actors: | System | | | | | | | | | | | | | |
| Trigger | The user initiates a conversation with the bot. | | | | | | | | | | | | | |
| Description / Main Scenario | <table border="1"> <tr> <td>1.</td> <td>User opens the chatbot window.</td> </tr> <tr> <td>2.</td> <td>Chatbot asks how it can help.</td> </tr> <tr> <td>3.</td> <td>User inputs a question or chooses topic.</td> </tr> <tr> <td>4.</td> <td>Chatbot processes the request.</td> </tr> <tr> <td>5.</td> <td>Chatbot produces a response, or a property suggestion if asked.</td> </tr> <tr> <td>6.</td> <td>User ends the conversation.</td> </tr> </table> | | 1. | User opens the chatbot window. | 2. | Chatbot asks how it can help. | 3. | User inputs a question or chooses topic. | 4. | Chatbot processes the request. | 5. | Chatbot produces a response, or a property suggestion if asked. | 6. | User ends the conversation. |
| 1. | User opens the chatbot window. | | | | | | | | | | | | | |
| 2. | Chatbot asks how it can help. | | | | | | | | | | | | | |
| 3. | User inputs a question or chooses topic. | | | | | | | | | | | | | |
| 4. | Chatbot processes the request. | | | | | | | | | | | | | |
| 5. | Chatbot produces a response, or a property suggestion if asked. | | | | | | | | | | | | | |
| 6. | User ends the conversation. | | | | | | | | | | | | | |
| Alternative Flows | <table border="1"> <tr> <td>4.1</td> <td>Unclear Question</td> </tr> <tr> <td></td> <td>4.1.a. Chatbot cannot understand the message.</td> </tr> <tr> <td>5.1</td> <td>Information Not Available</td> </tr> <tr> <td></td> <td>5.1.a. Chatbot has insufficient command of the subject matter to respond. 5.1.b. It recommends getting in touch with support for additional assistance.</td> </tr> </table> | | 4.1 | Unclear Question | | 4.1.a. Chatbot cannot understand the message. | 5.1 | Information Not Available | | 5.1.a. Chatbot has insufficient command of the subject matter to respond. 5.1.b. It recommends getting in touch with support for additional assistance. | | | | |
| 4.1 | Unclear Question | | | | | | | | | | | | | |
| | 4.1.a. Chatbot cannot understand the message. | | | | | | | | | | | | | |
| 5.1 | Information Not Available | | | | | | | | | | | | | |
| | 5.1.a. Chatbot has insufficient command of the subject matter to respond. 5.1.b. It recommends getting in touch with support for additional assistance. | | | | | | | | | | | | | |

| | | |
|----------------------|--|--|
| | 5.2 | Technical Issue |
| | | 5.2.a. The chatbot doesn't load or gets stuck. 5.2.b. System displayed: "Unavailable chatbot at this time." |
| Quality Requirements | Replies in Chat must be readable and comprehensible. | |

Table 4-13: Chatbot

2.4.3 Activity Diagram

Login and Sign Up

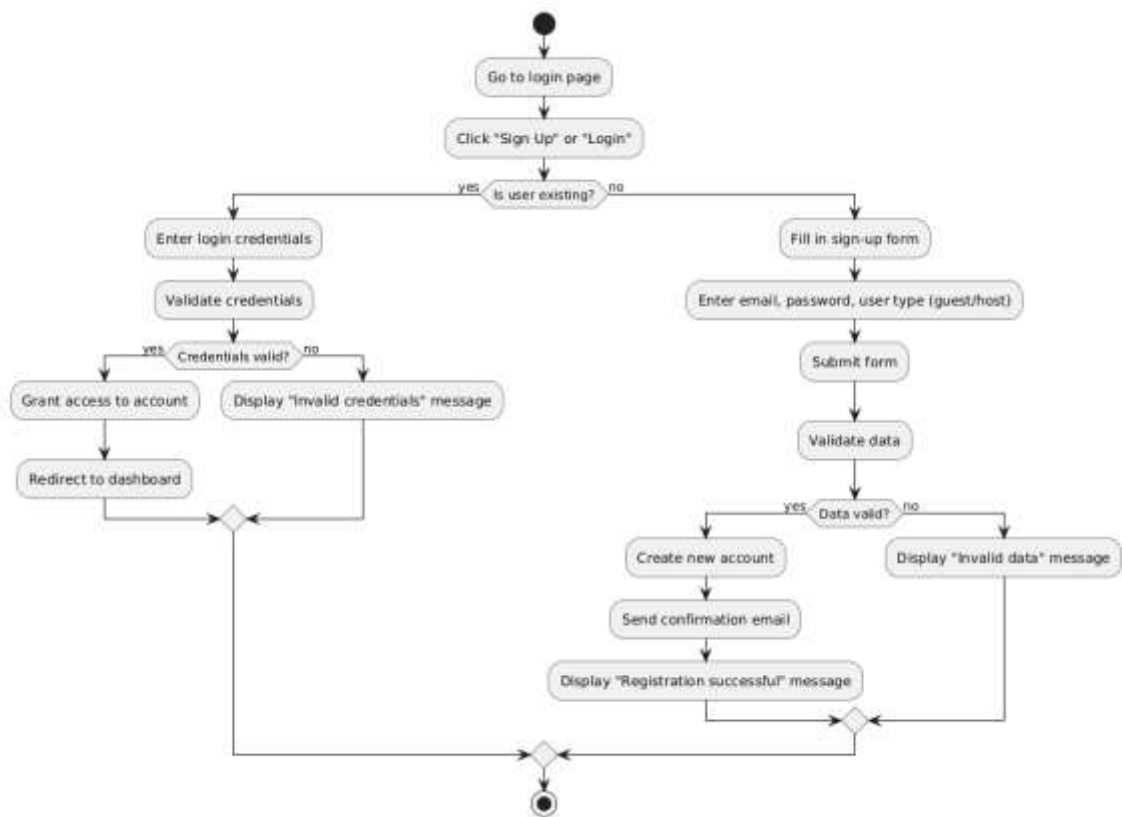


Figure 4-1: Login and Sign Up

Property Listing Management (Host)

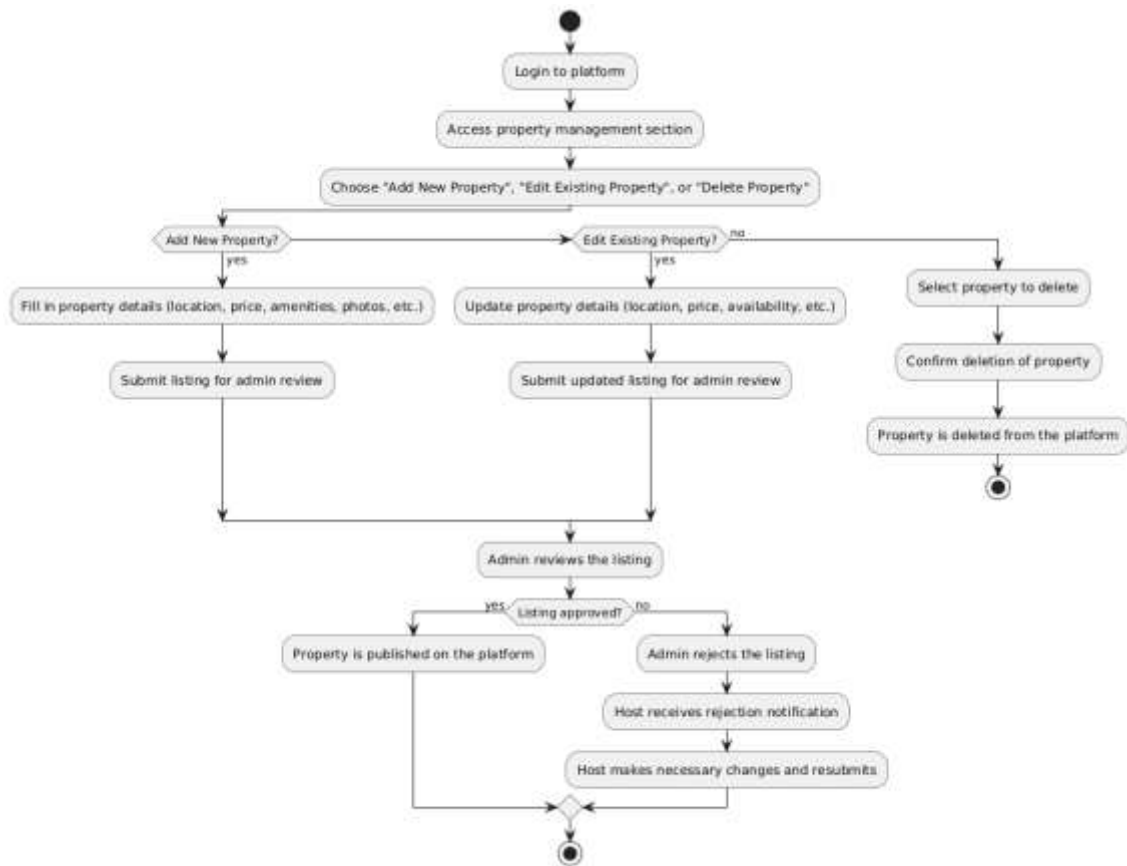


Figure 4-2: Property Listing Management (Host)

Property Listing Management (Admin)

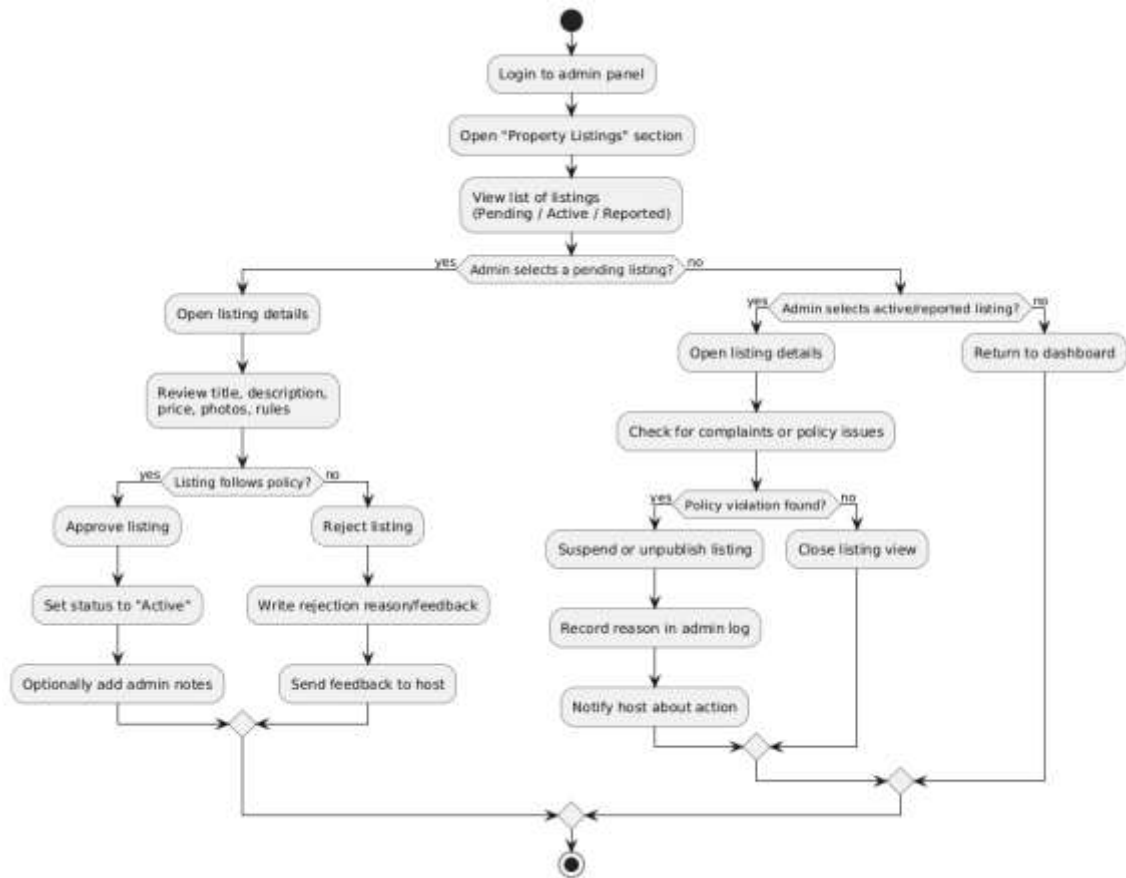


Figure 4-3: Property Listing Management (Admin)

Search and Filter Properties (Guest)

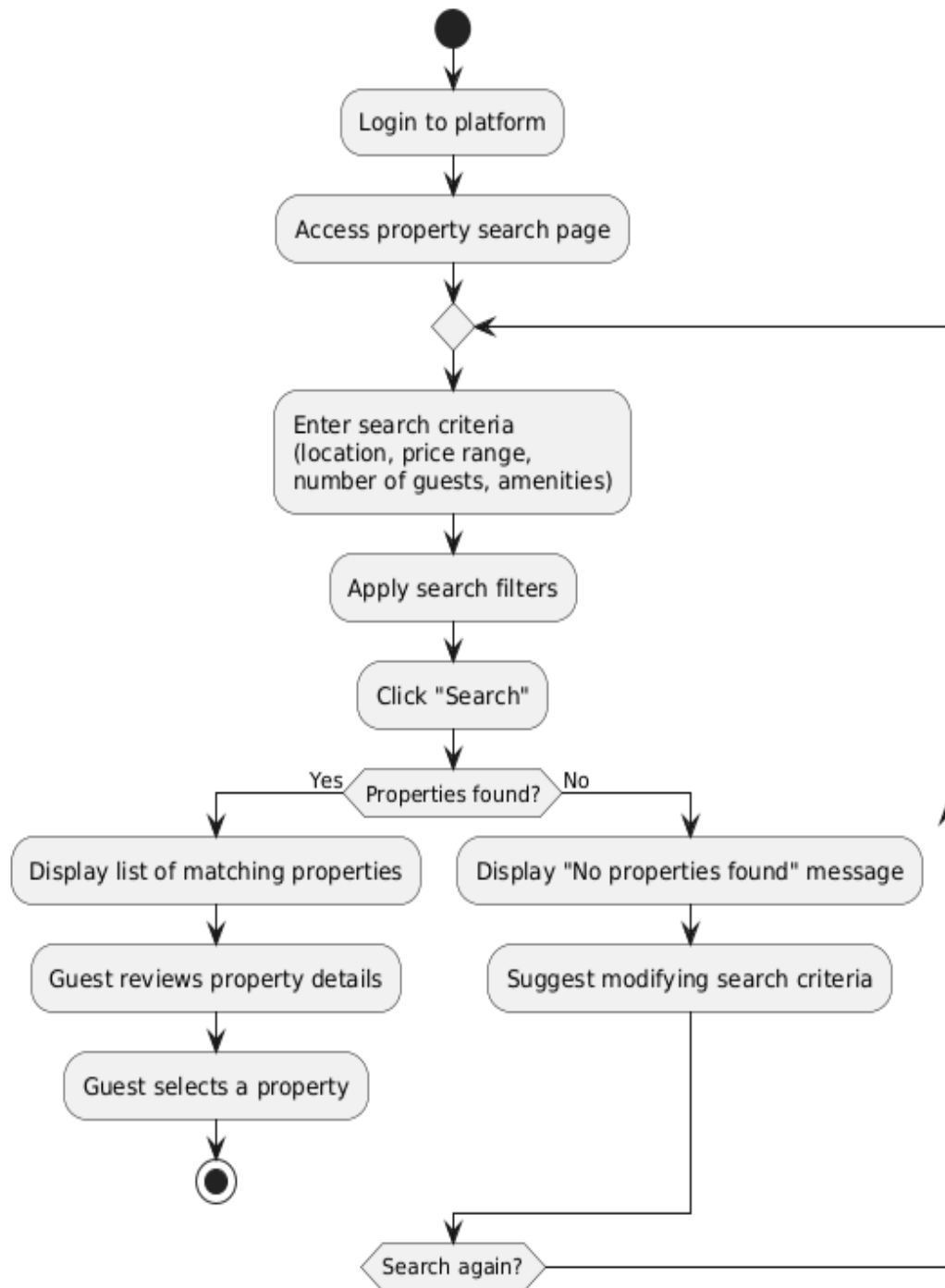


Figure 4-4: Search and Filter Properties (Guest)

Booking and Reservation (Guest)

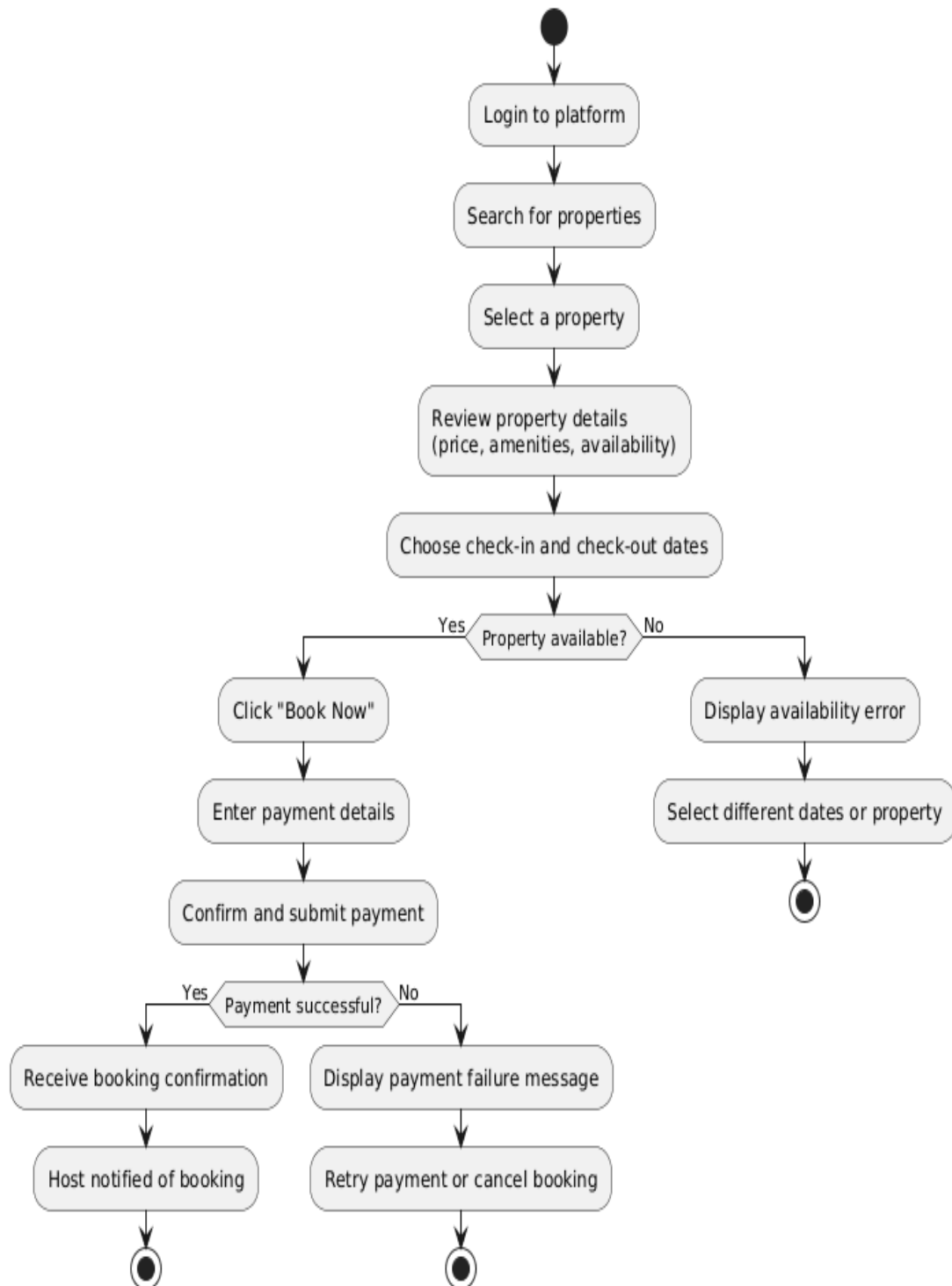


Figure 4-5: Booking and Reservation (Guest)

Booking and Reservation (Host)

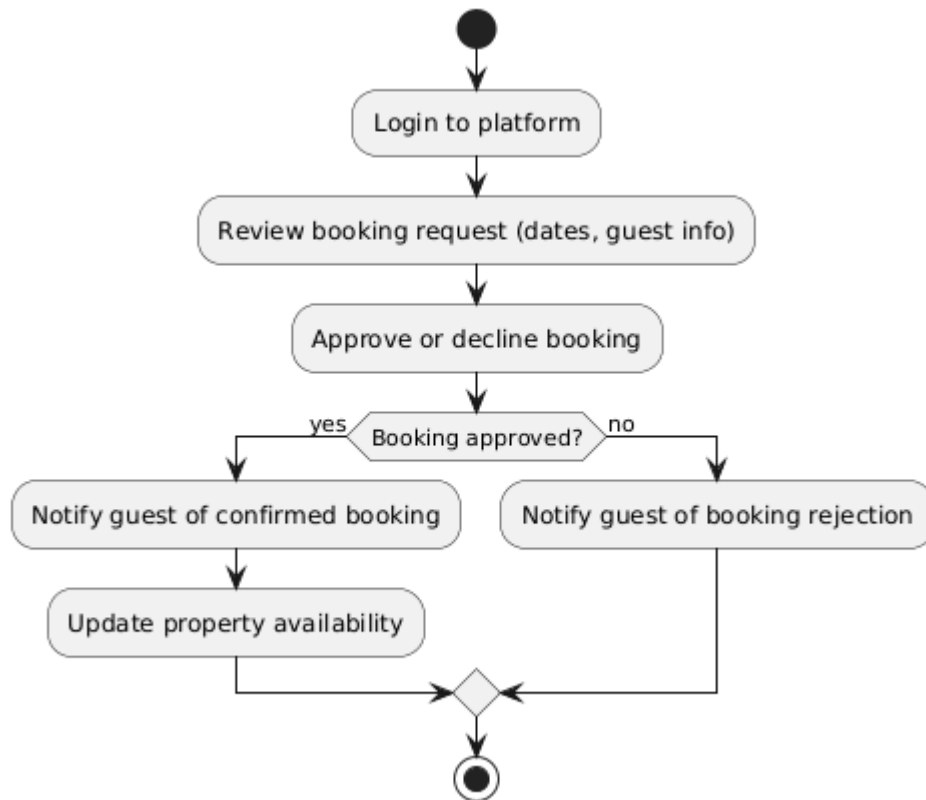


Figure 4-6: Booking and Reservation (Host)

Payment Processing (Admin)



Figure 4-7: Payment Processing (Admin)

Review and Rating System (Guest)



Figure 4-8: Review and Rating System (Guest)

Review and Rating System (Host)

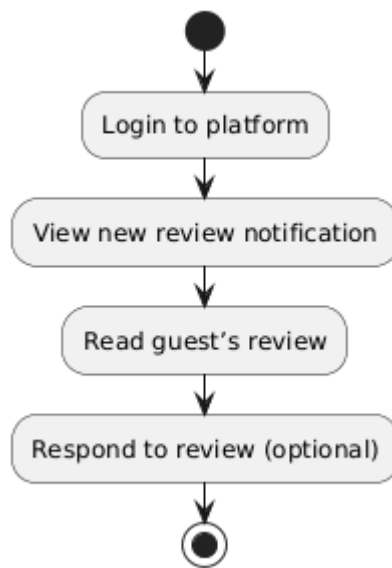


Figure 4-9: Review and Rating System (Host)

Review and Rating System (Admin)

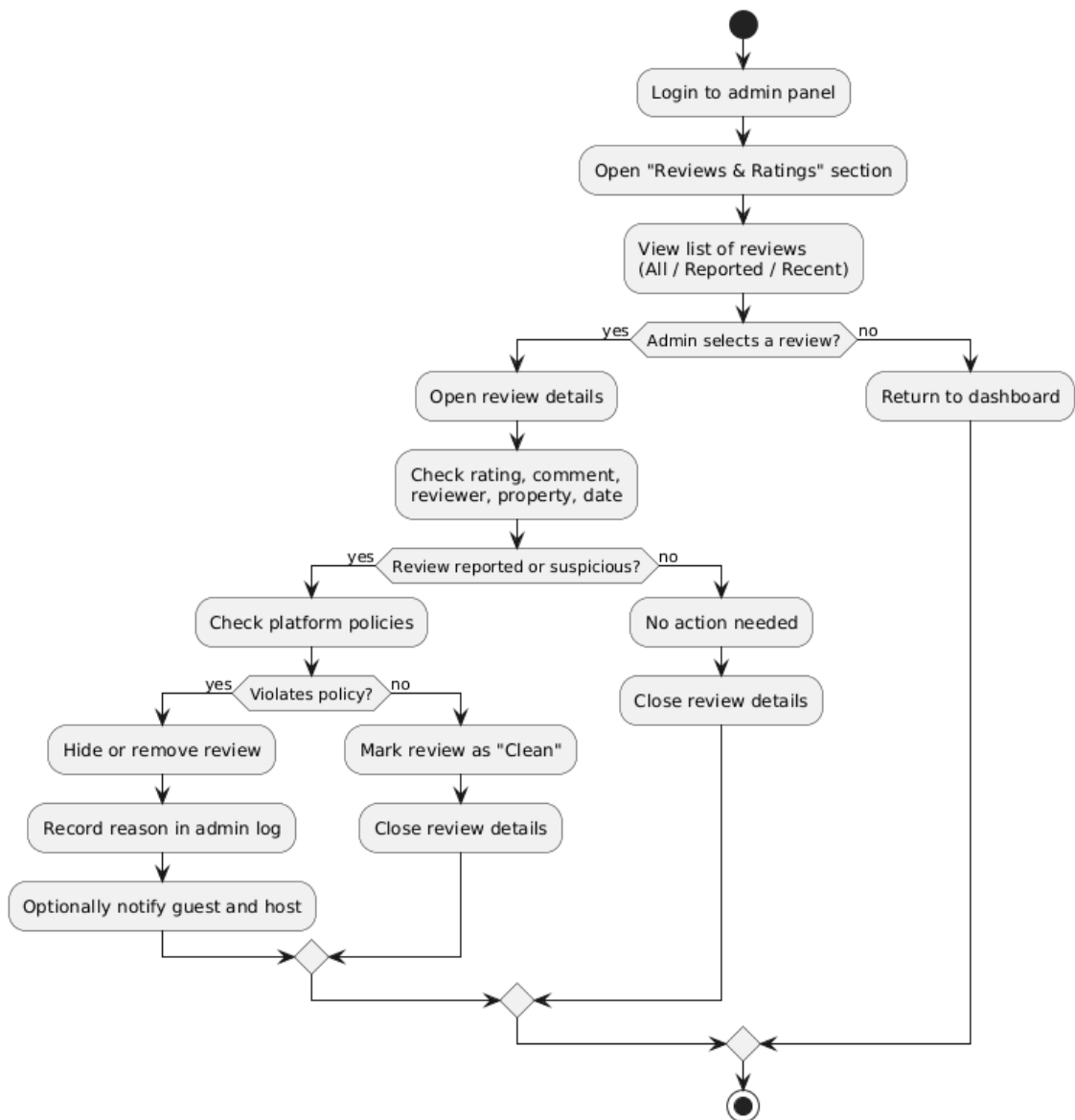


Figure 4-10: Review and Rating System (Admin)

Messaging System (Guest)

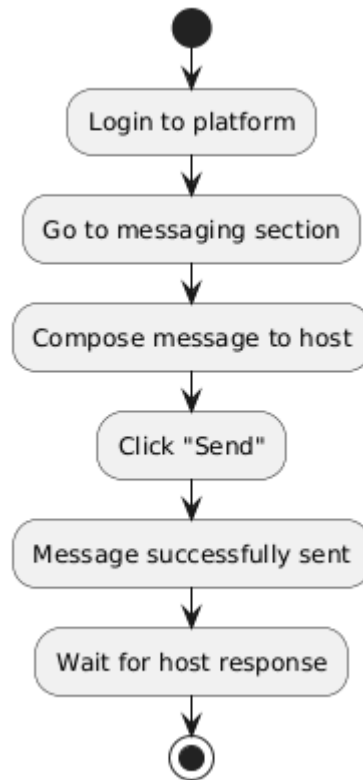


Figure 4-11: Messaging System (Guest)

Messaging System (Host)

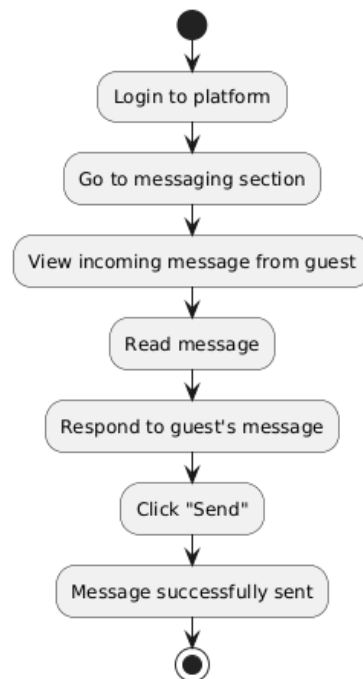


Figure 4-12: Messaging System (Host)

User Profile Management (Guest & Host)

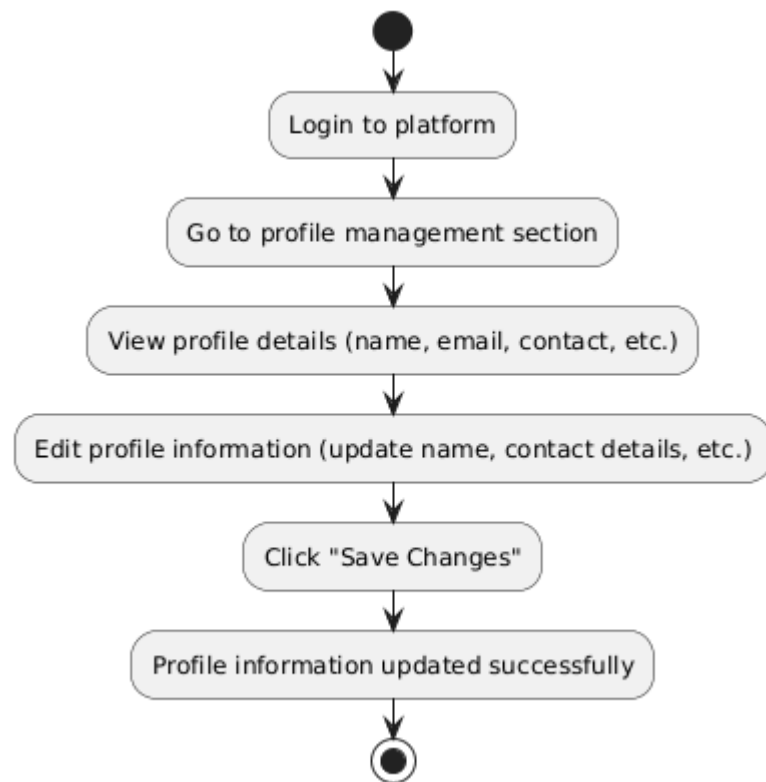


Figure 4-13: User Profile Management (Guest & Host)

User Profile Management (Admin)

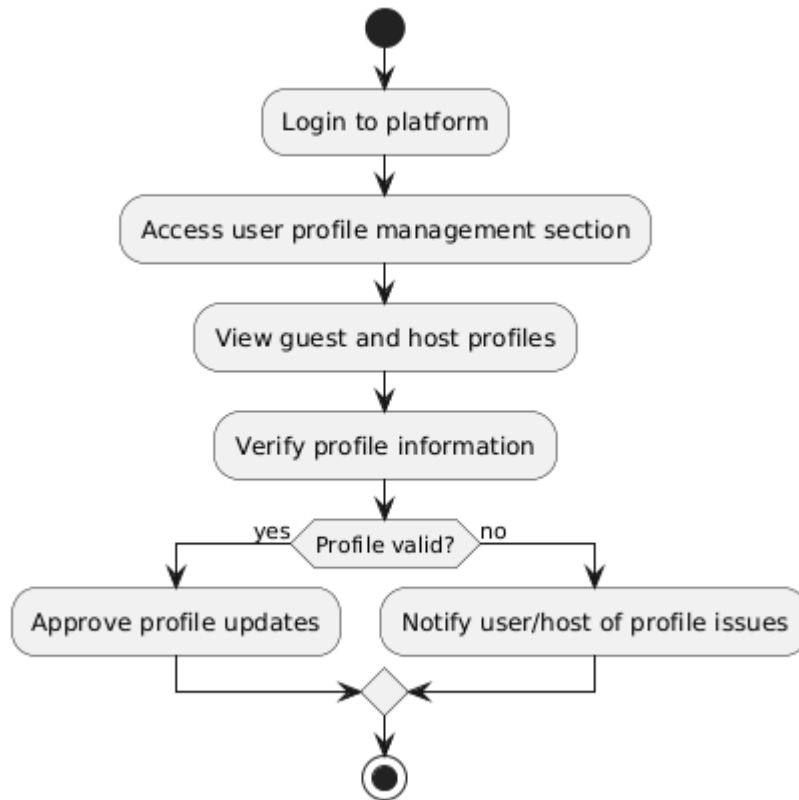


Figure 4-14: User Profile Management (Admin)

Booking Cancellation and Refund (Guest)

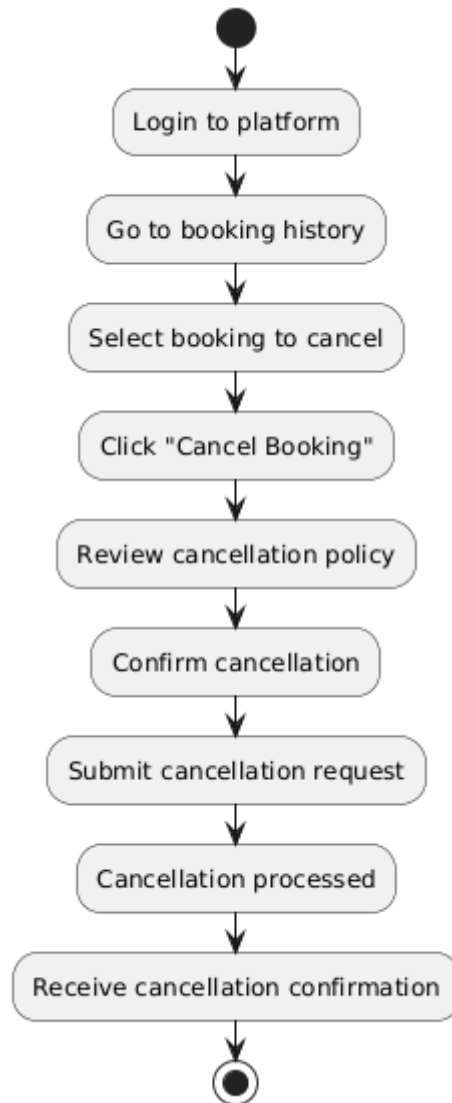


Figure 4-15: Booking Cancellation and Refund (Guest)

Booking Cancellation and Refund (Host)

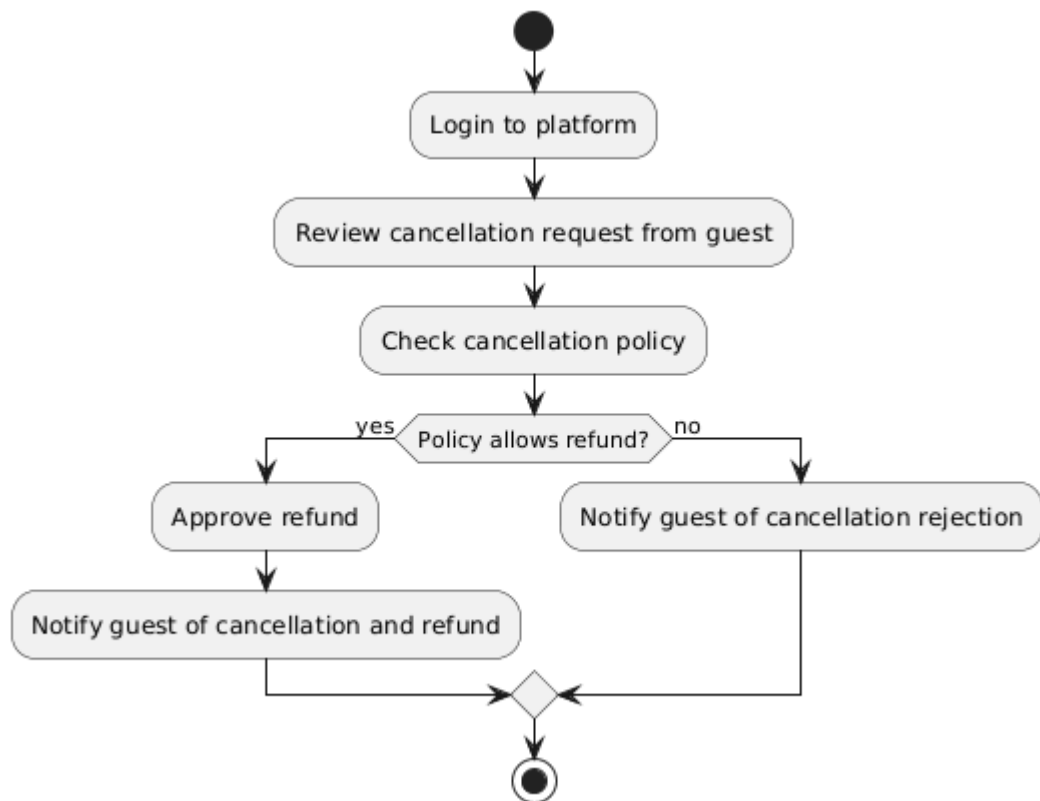


Figure 4-16: Booking Cancellation and Refund (Host)

Booking Cancellation and Refund (Admin)

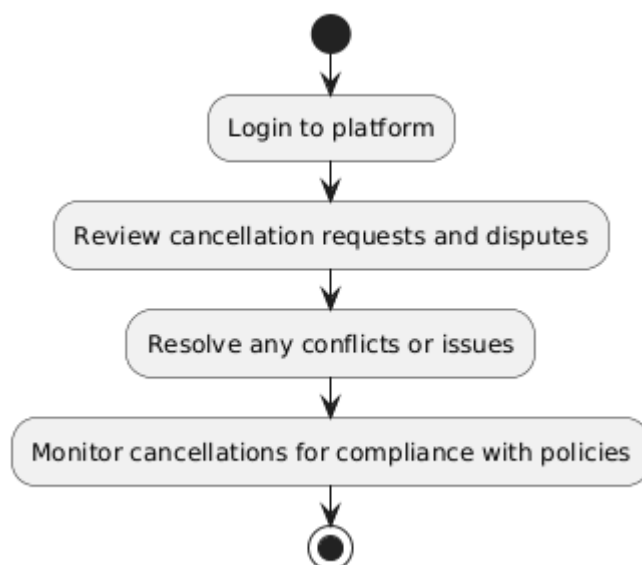


Figure 4-17: Booking Cancellation and Refund (Admin)

Notification and Alert (Guest, Host and Admin)

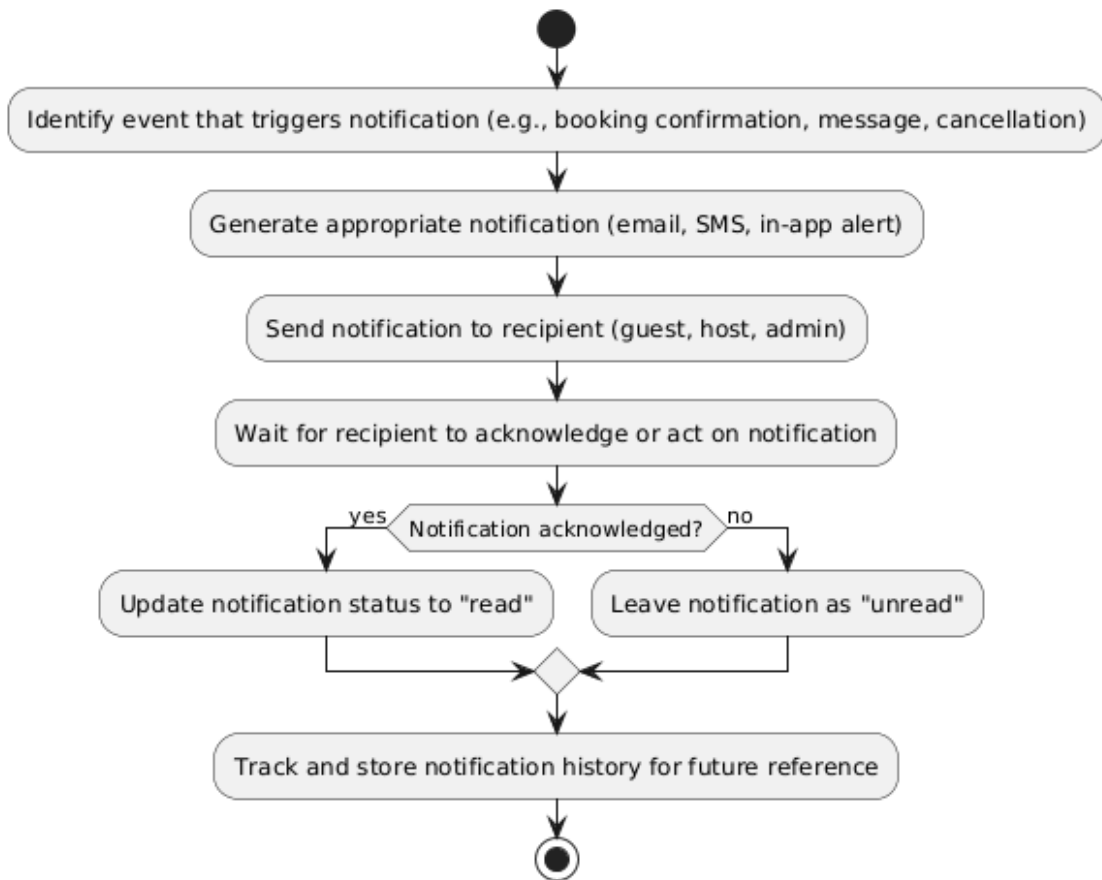


Figure 4-18: Notification and Alert (Guest, Host and Admin)

Property Availability Management (Host)

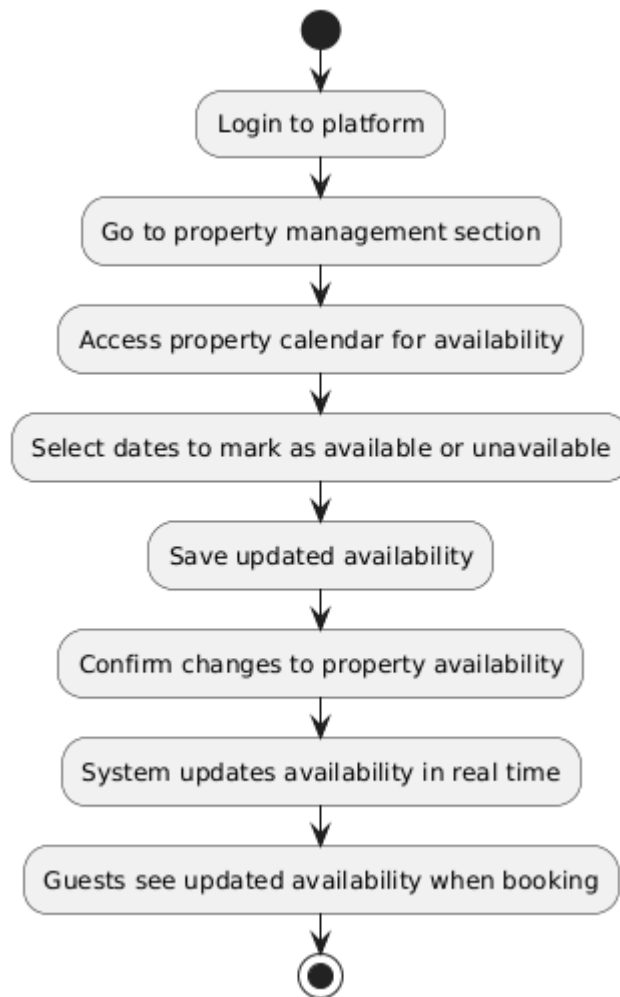


Figure 4-19: Property Availability Management (Host)

User Support and Dispute Resolution (Guest)

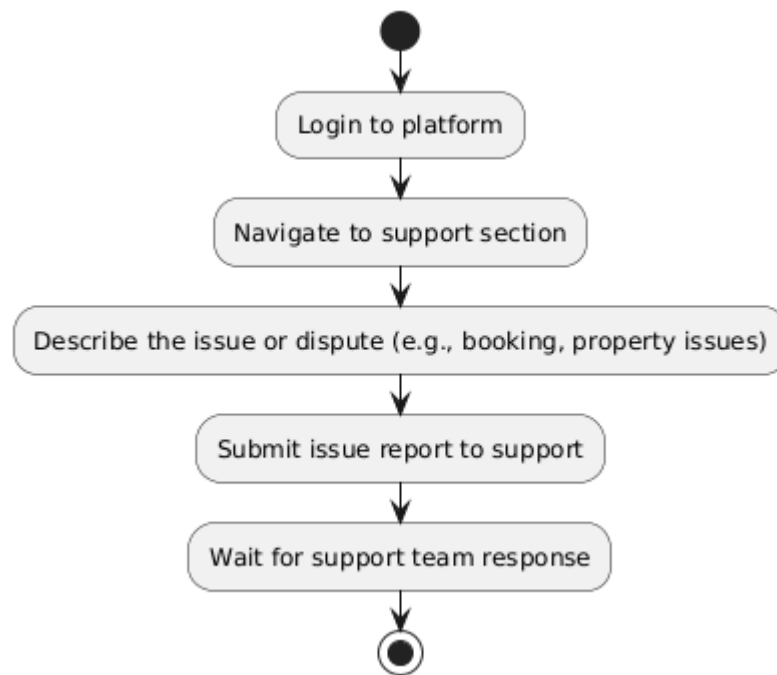


Figure 4-20: User Support and Dispute Resolution (Guest)

User Support and Dispute Resolution (Host)

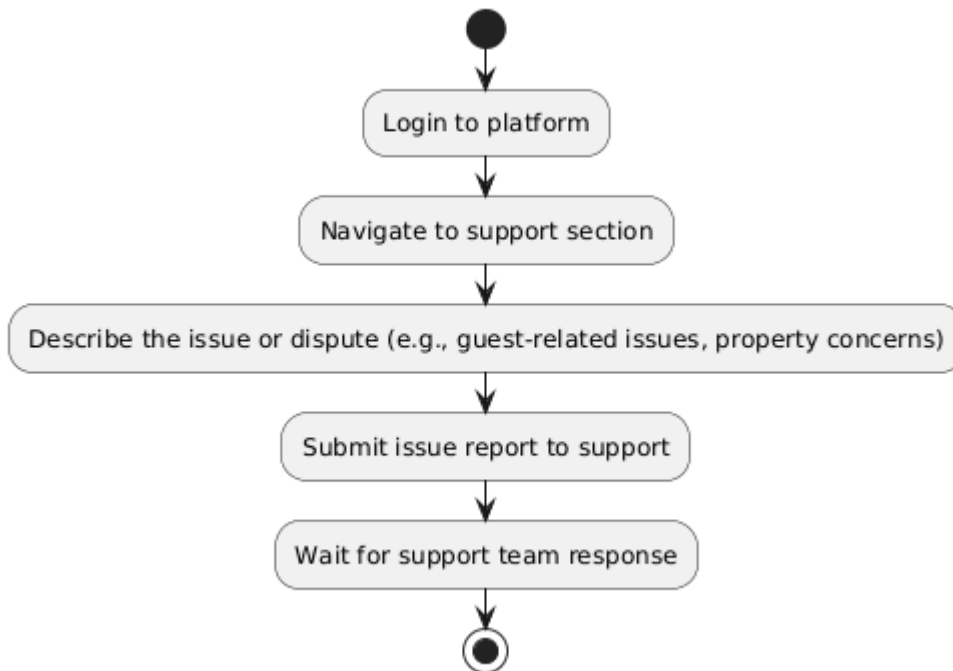


Figure 4-21: User Support and Dispute Resolution (Host)

User Support and Dispute Resolution (Admin)

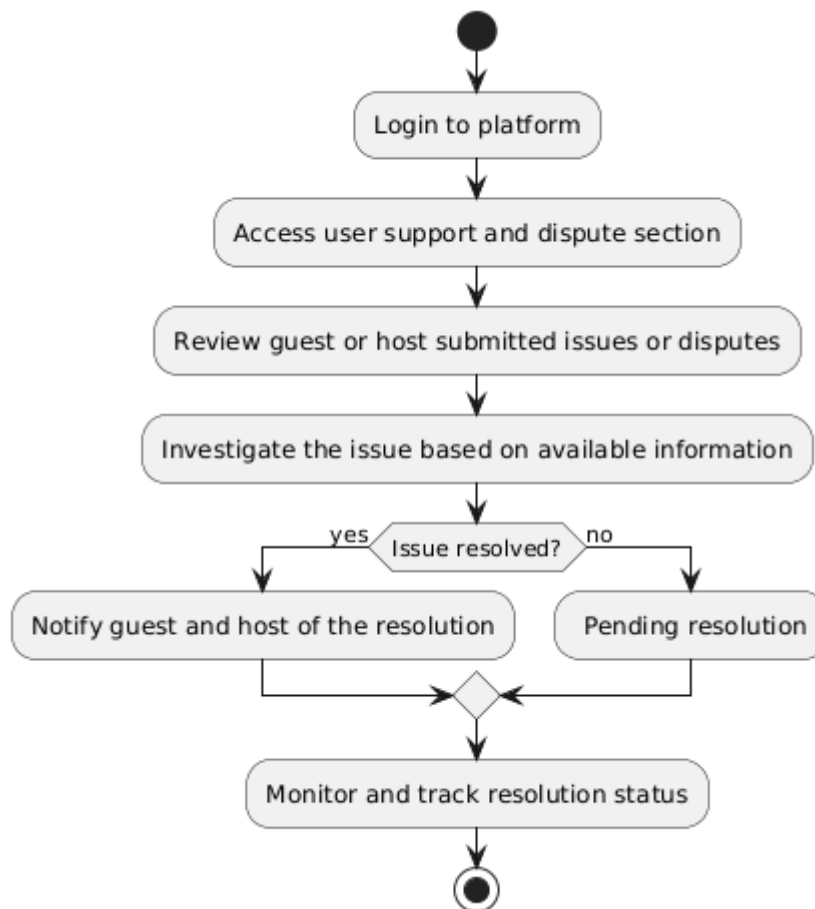


Figure 4-22: User Support and Dispute Resolution (Admin)

Chatbot (Guest)

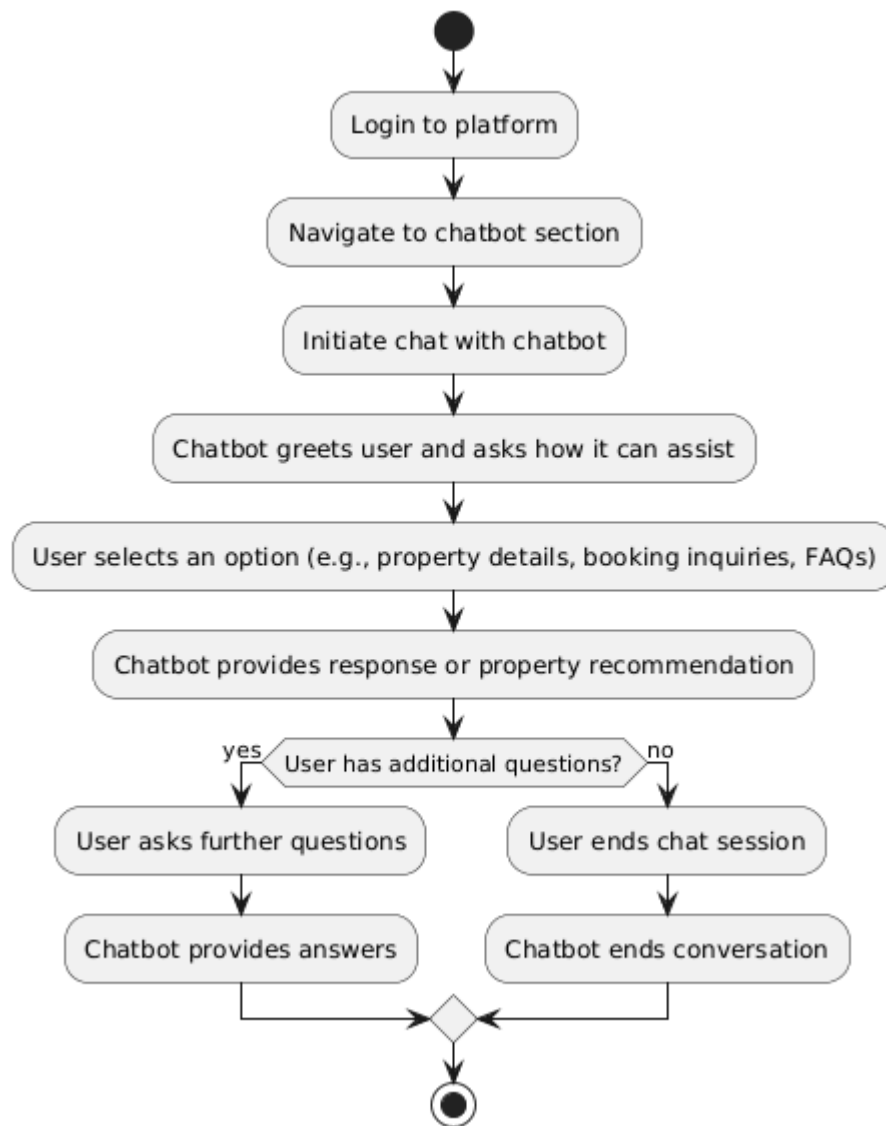


Figure 4-23: Chatbot (Guest)

2.4.4 Sequence Diagram

Login and Sign Up

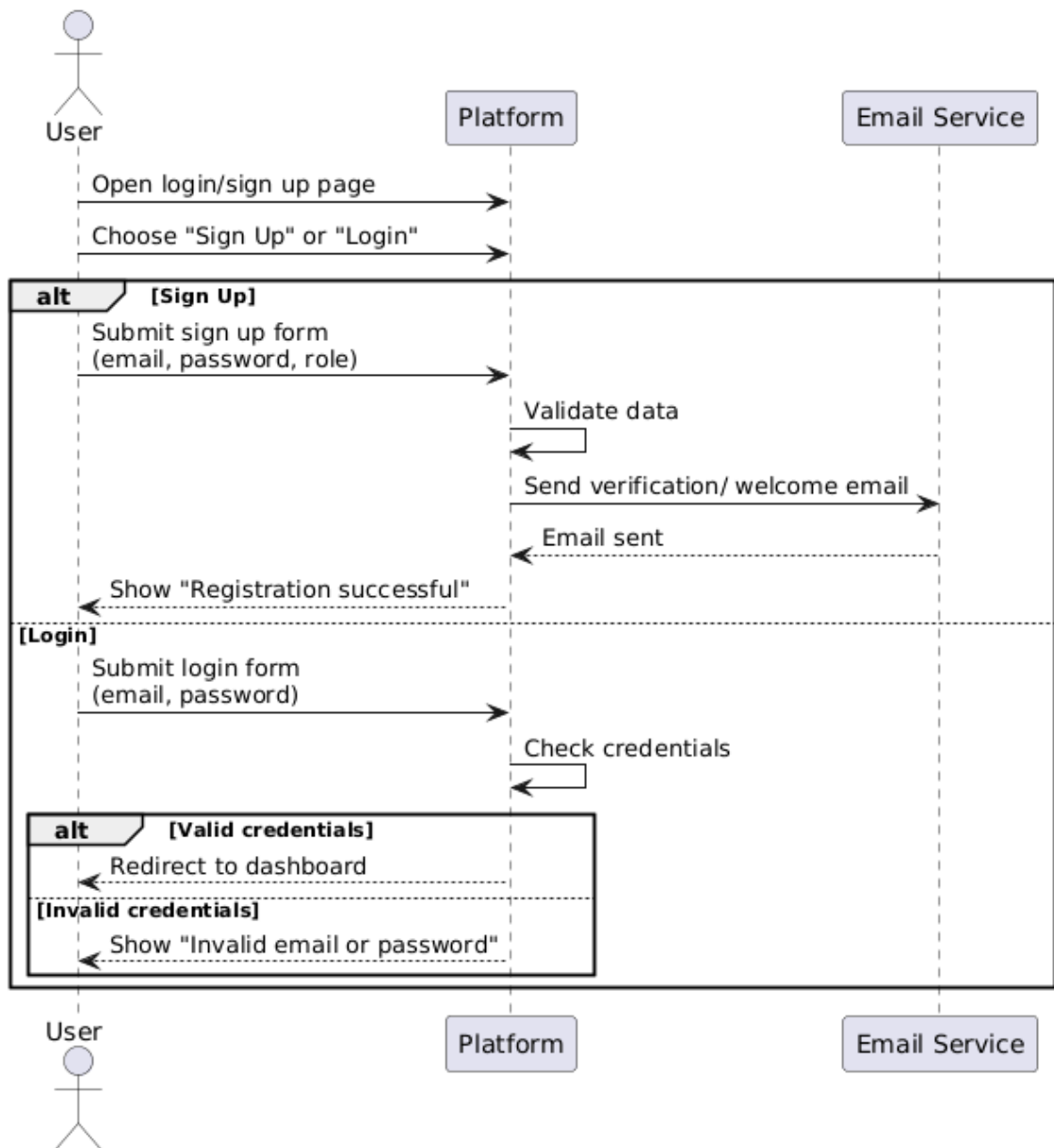


Figure 5-1: Login and Sign Up

Property Listing Management (Host)

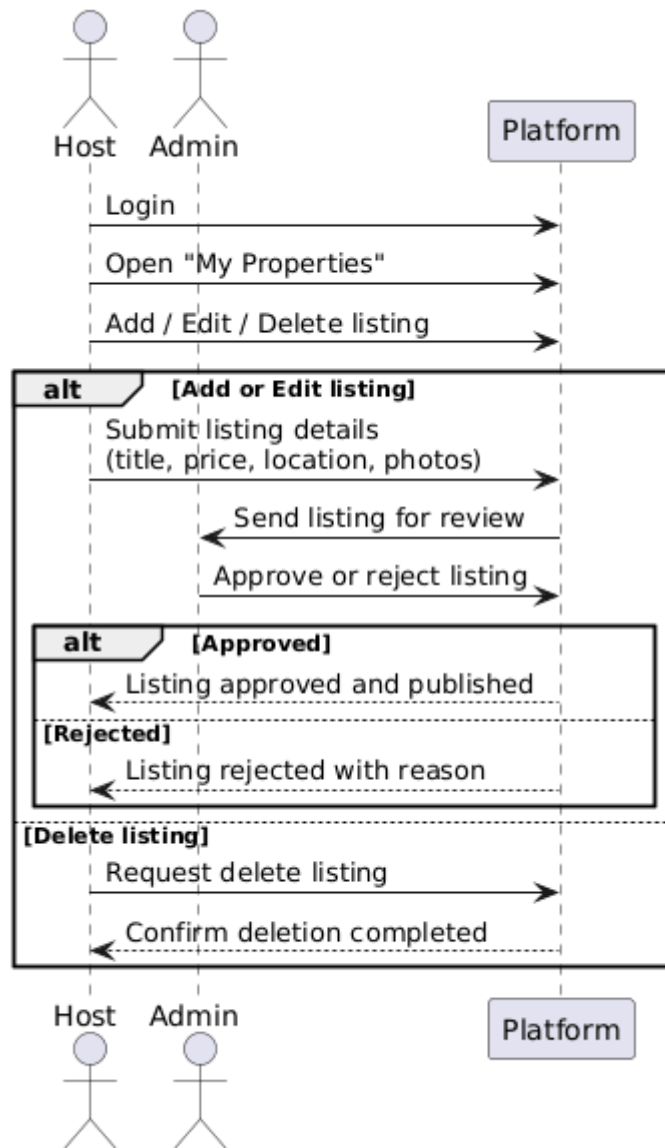


Figure 5-2: Property Listing Management (Host)

Property Listing Management (Admin)

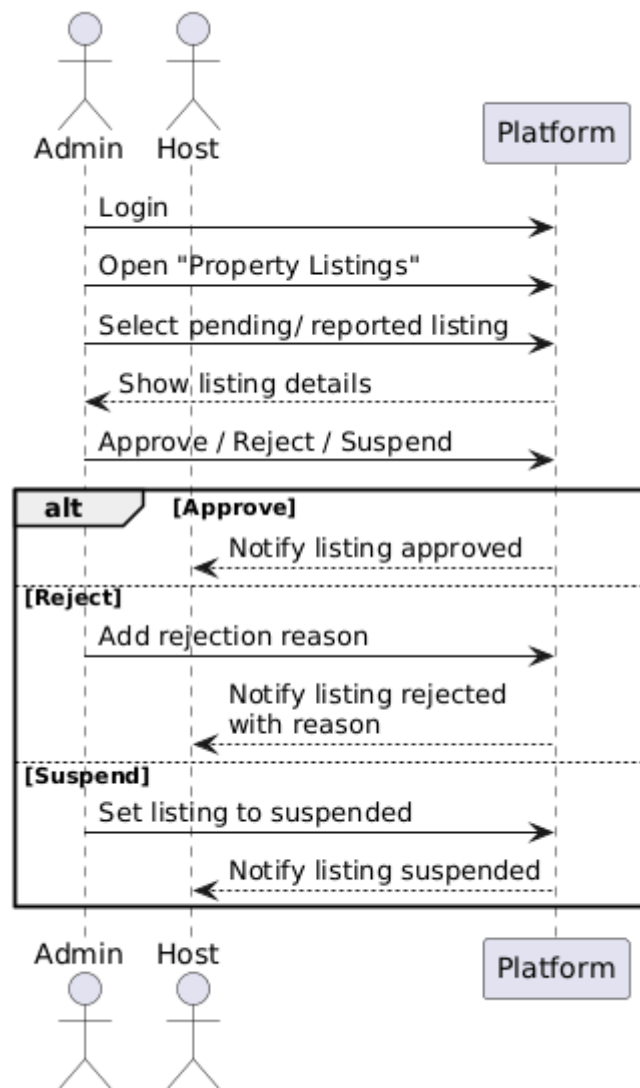


Figure 5-3: Property Listing Management (Admin)

Search and Filter Properties (Guest)

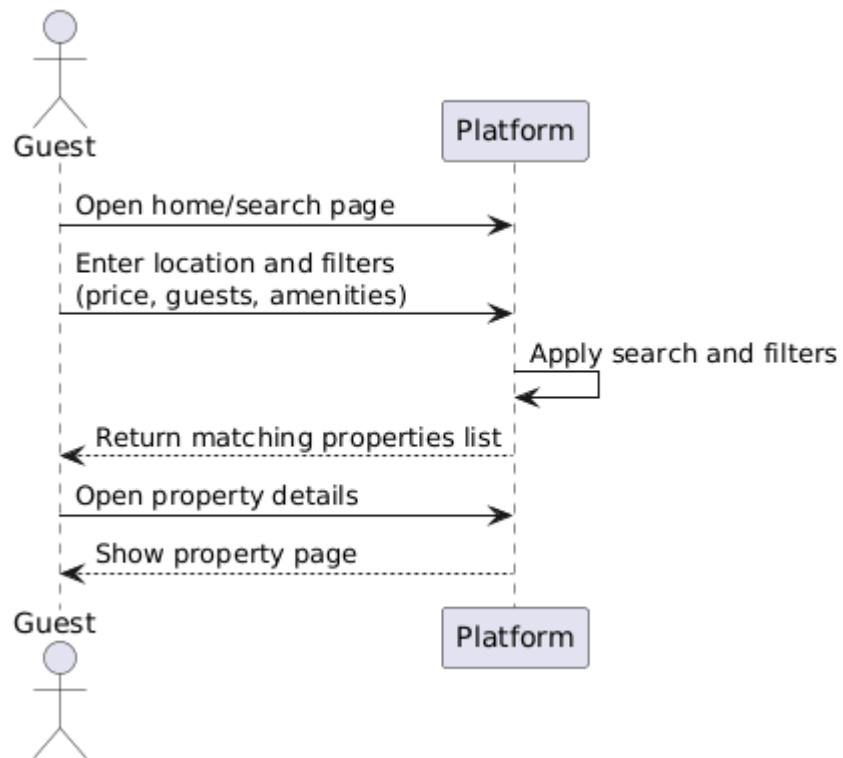


Figure 5-4: Search and Filter Properties (Guest)

Booking and Reservation (Guest)

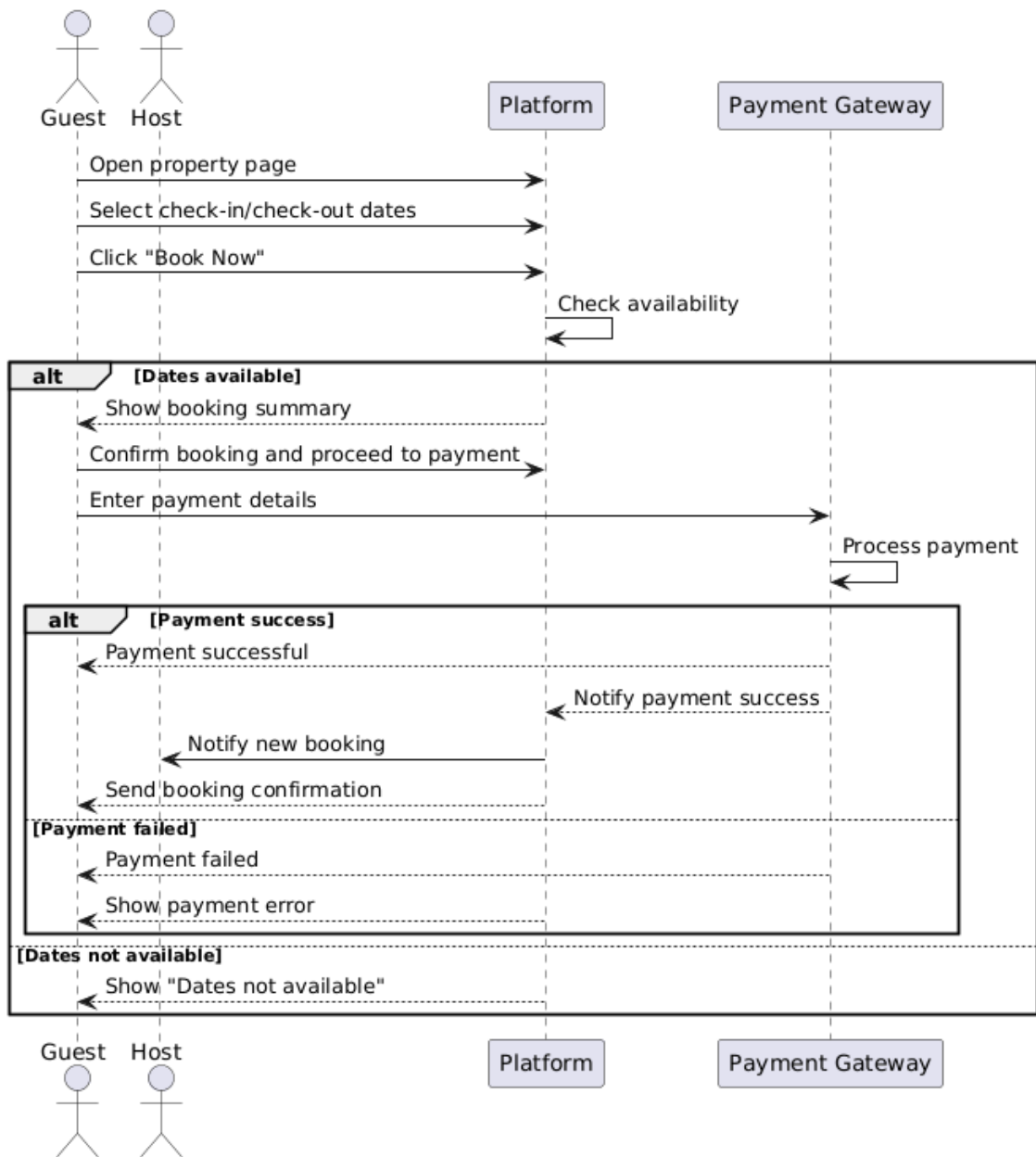


Figure 5-5: Booking and Reservation (Guest)

Booking and Reservation (Host)

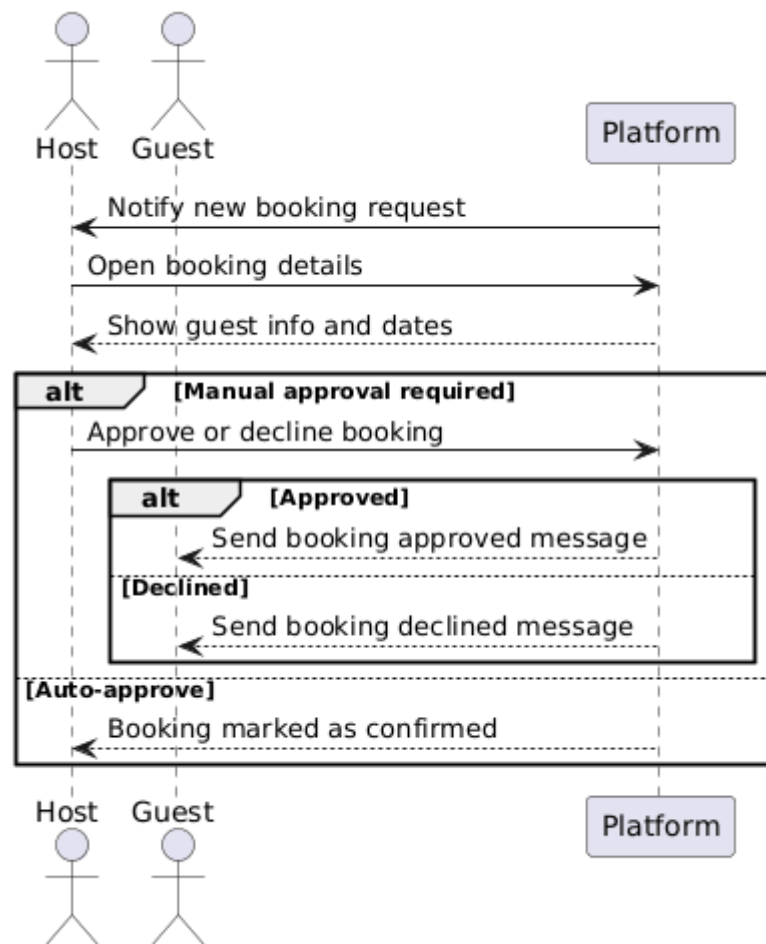


Figure 5-6: Booking and Reservation (Host)

Review and Rating System (Guest)

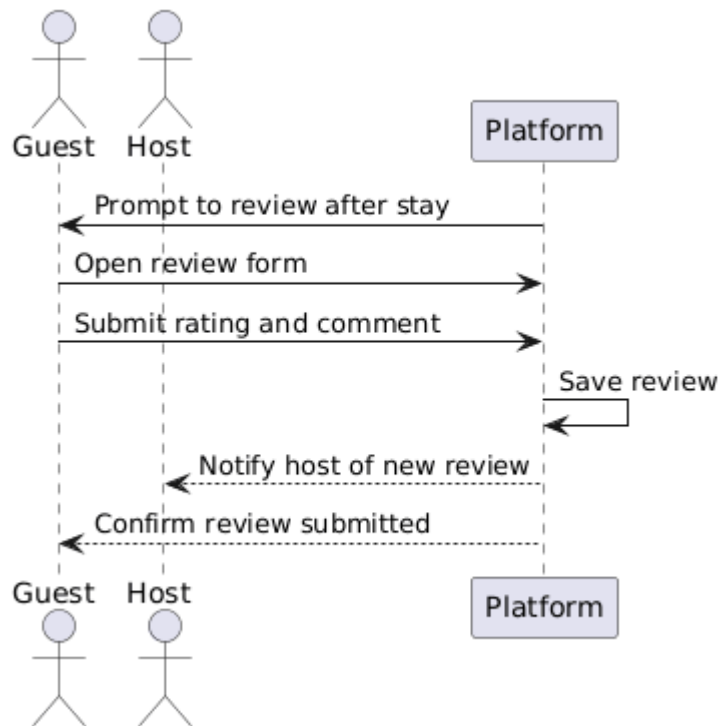


Figure 5-7: Review and Rating System (Guest)

Review and Rating System (Host)

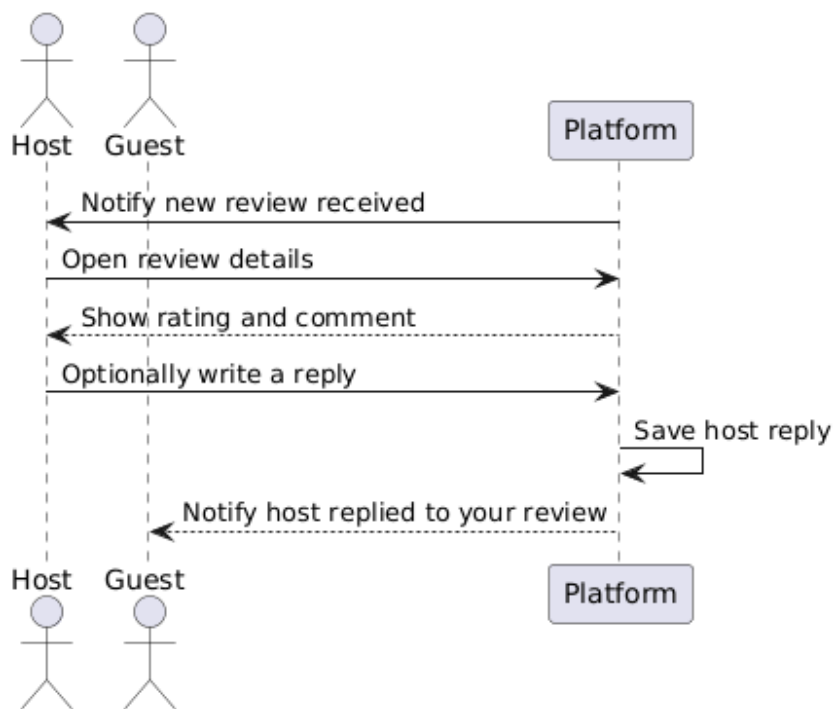


Figure 5-8: Review and Rating System (Host)

Review and Rating System (Admin)

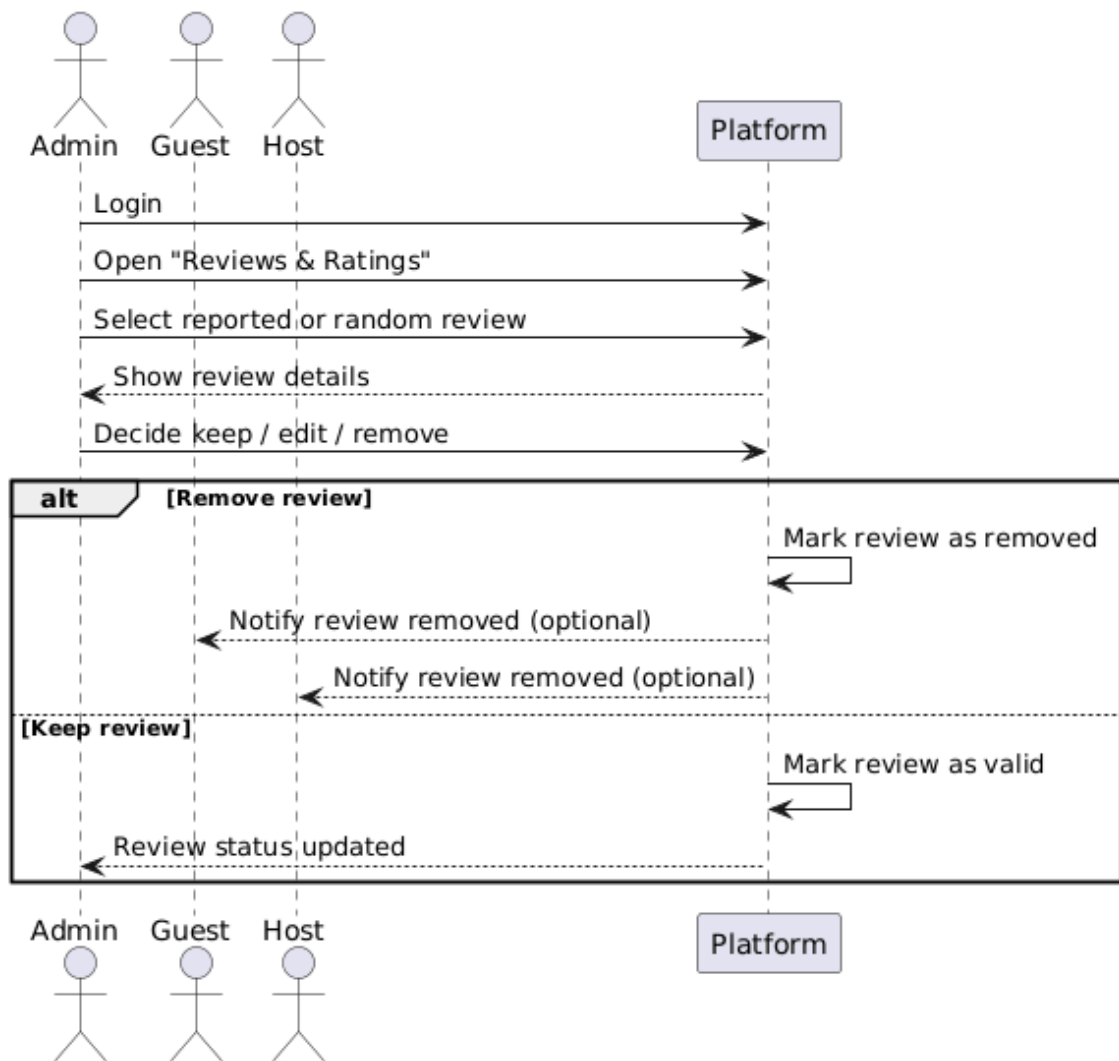


Figure 5-9: Review and Rating System (Admin)

Messaging System (Guest)

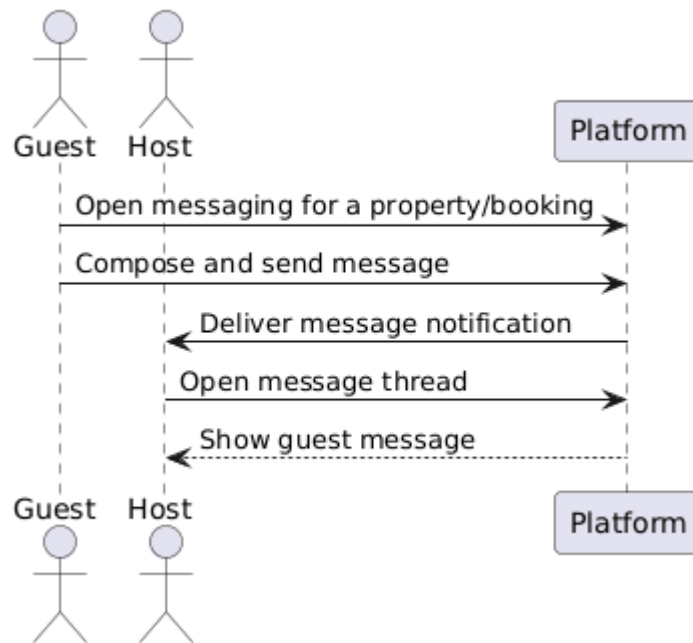


Figure 5-10: Messaging System (Guest)

Messaging System (Host)

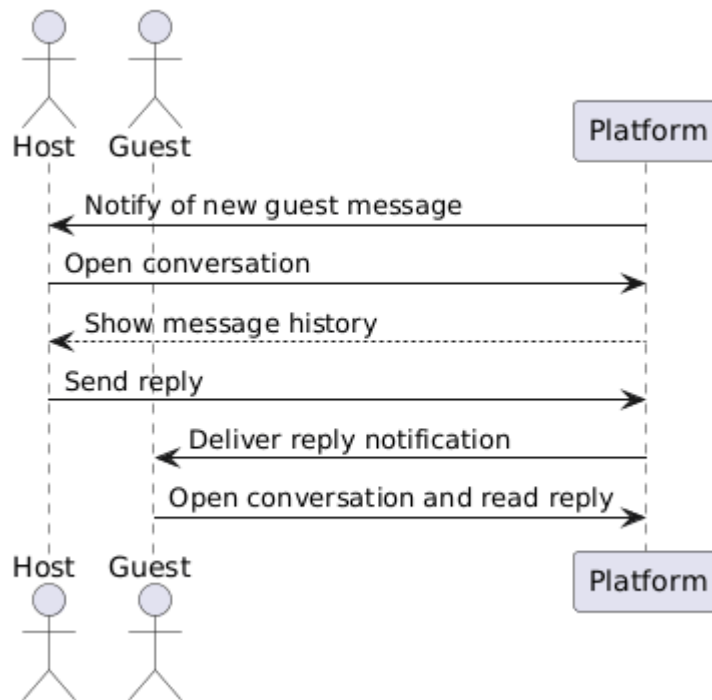


Figure 5-11: Messaging System (Host)

User Profile Management (Guest & Host)

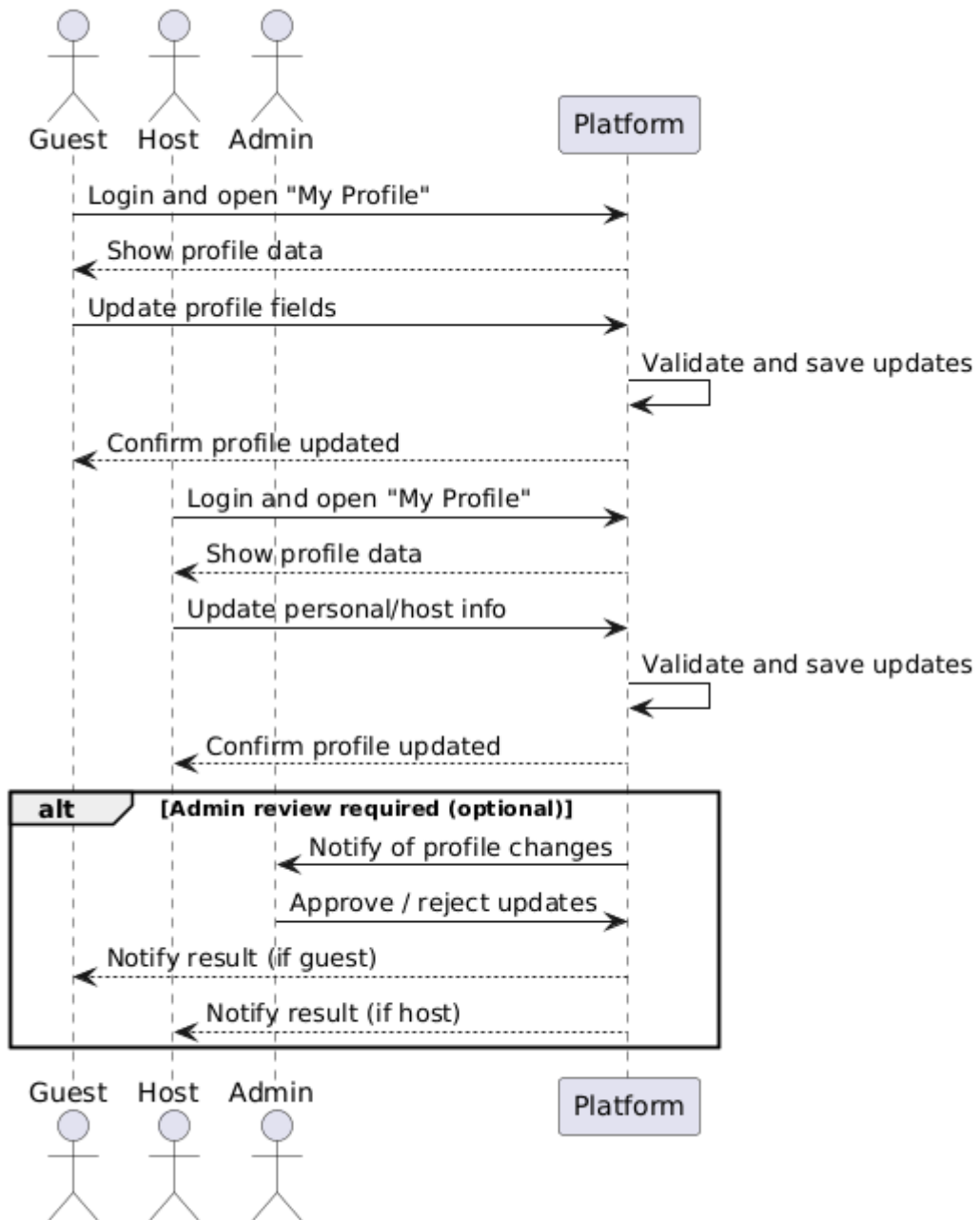


Figure 5-12: User Profile Management (Guest & Host)

User Profile Management (Admin)

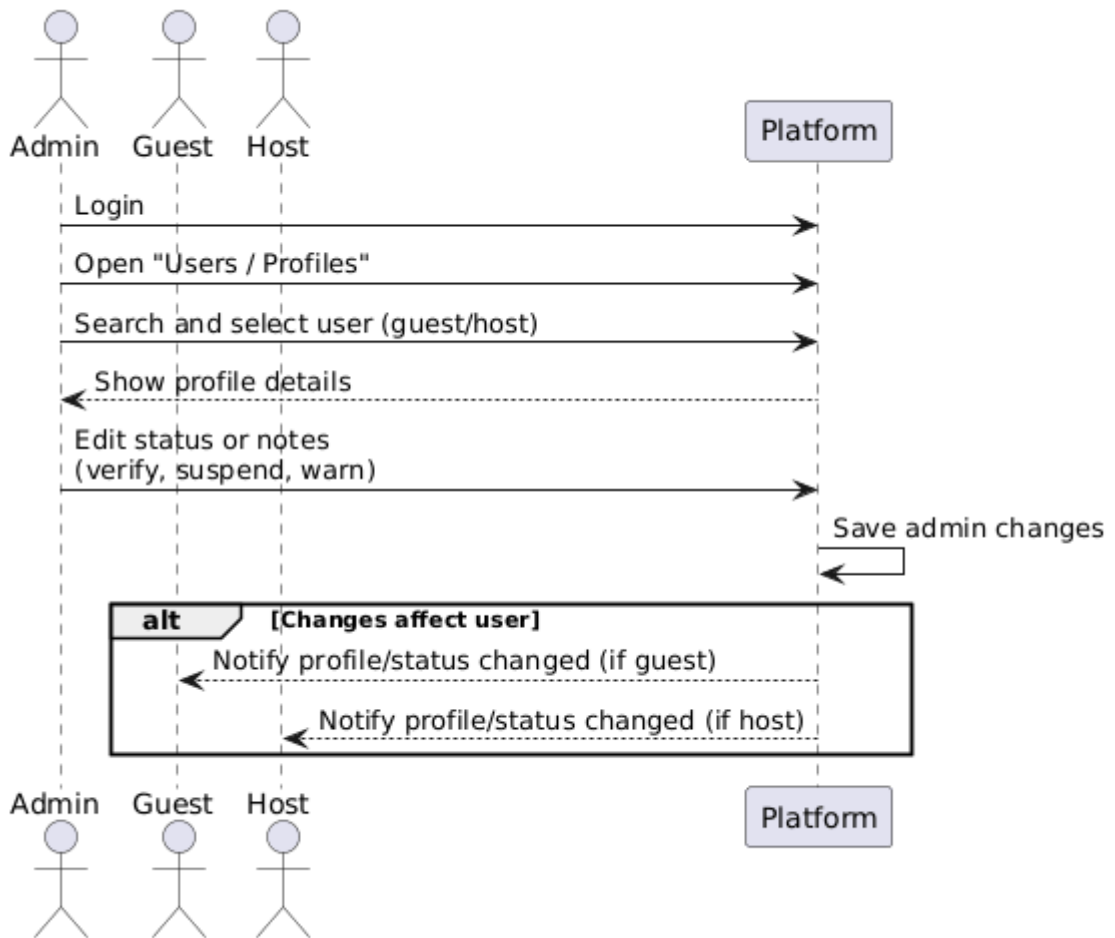


Figure 5-13: User Profile Management (Admin)

Booking Cancellation and Refund (Guest)

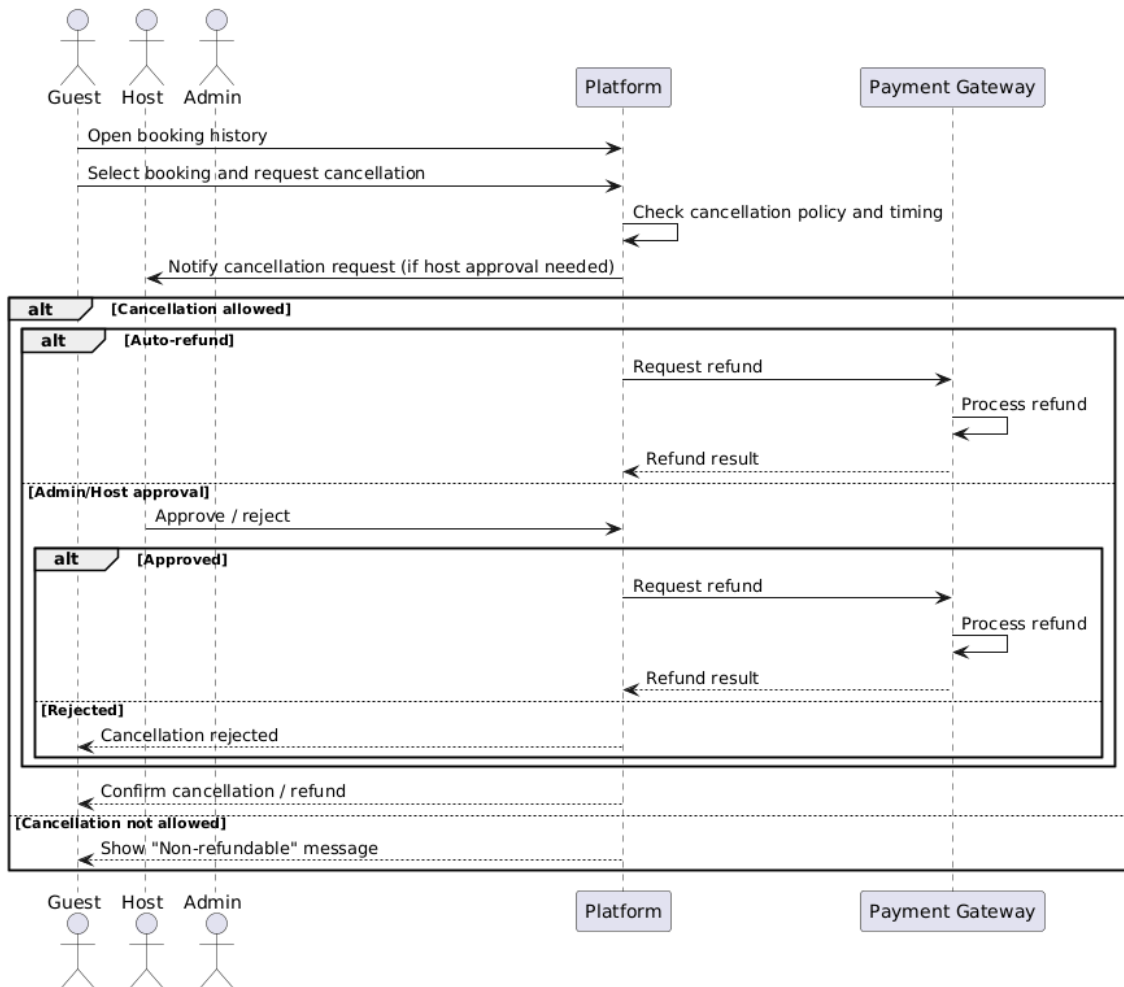


Figure 5-14: Booking Cancellation and Refund (Guest)

Booking Cancellation and Refund (Host)

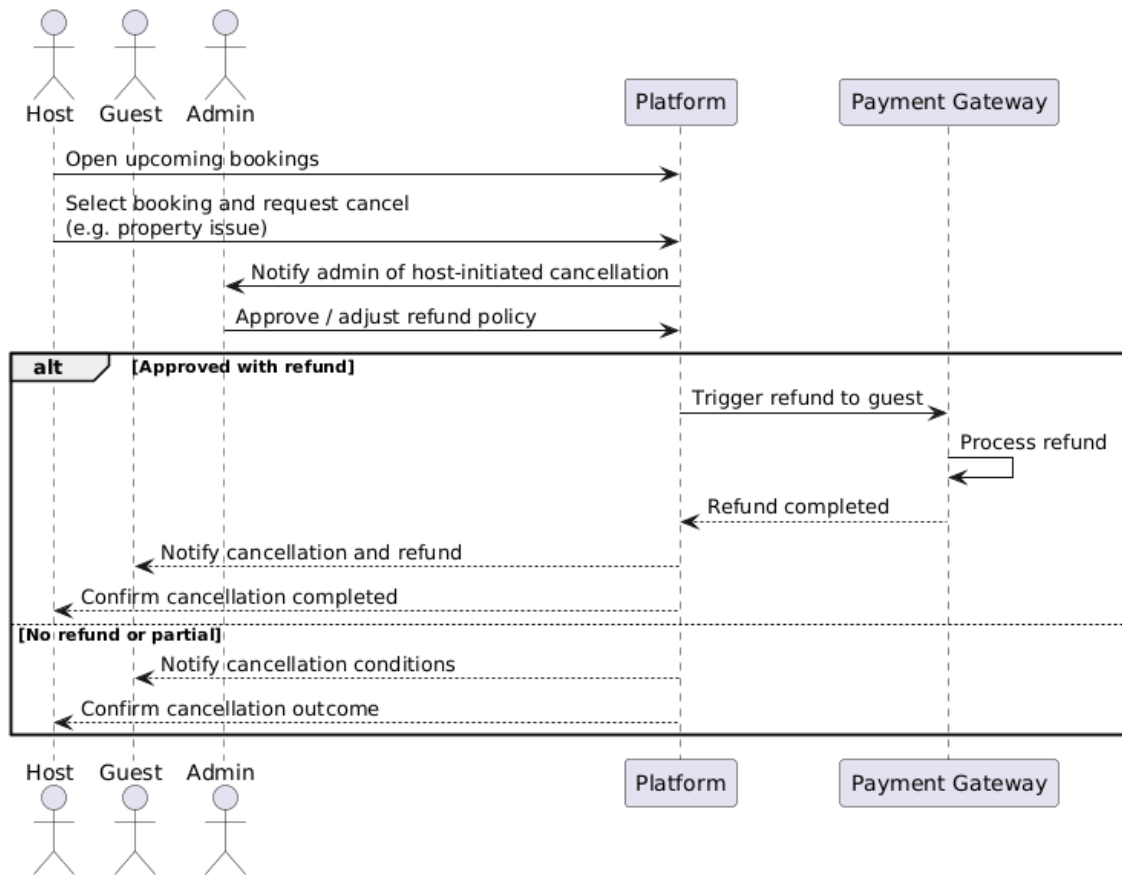


Figure 5-15: Booking Cancellation and Refund (Host)

Booking Cancellation and Refund (Admin)

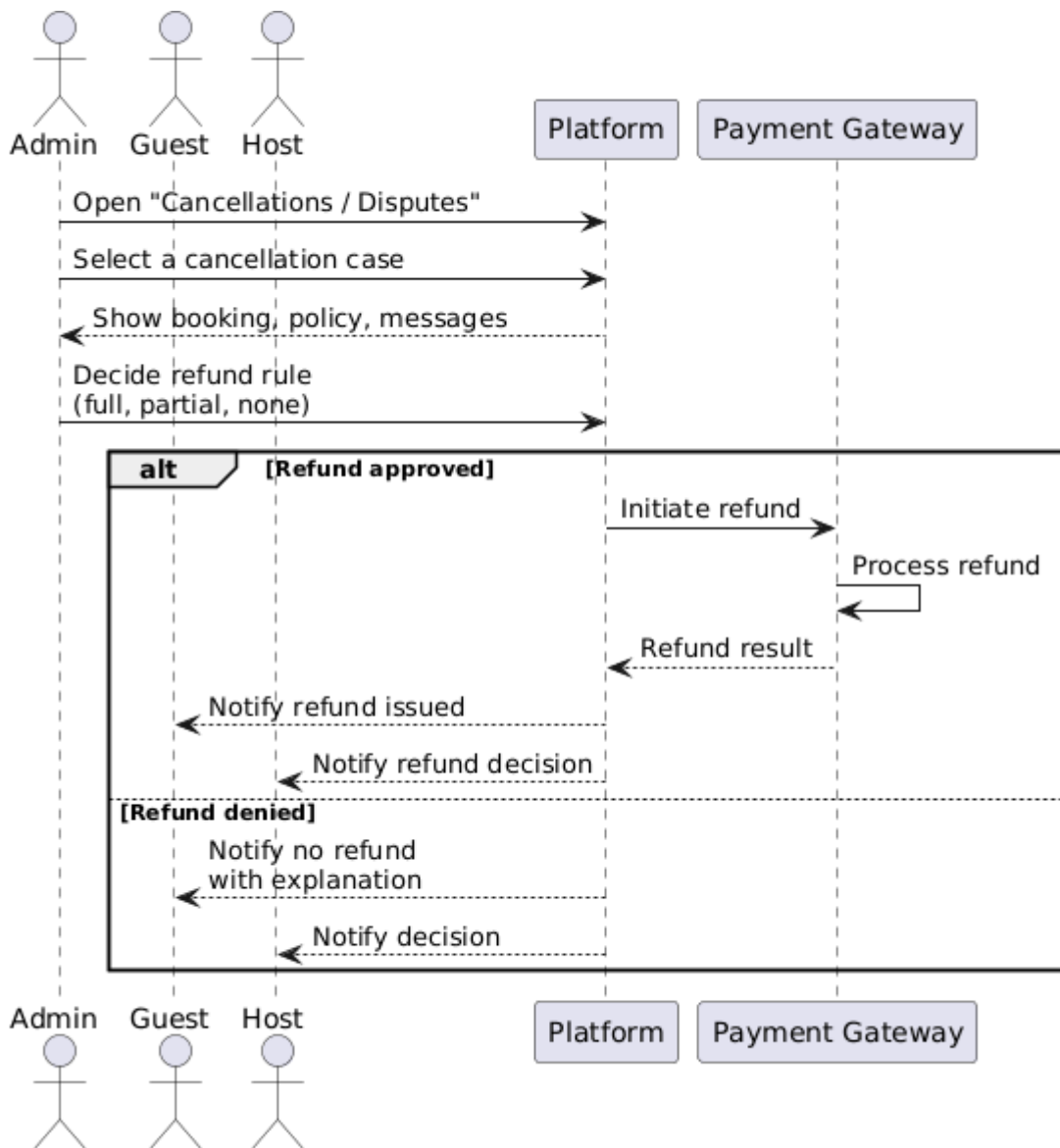


Figure 5-16: Booking Cancellation and Refund (Admin)

Notification and Alert (Guest, Host and Admin)

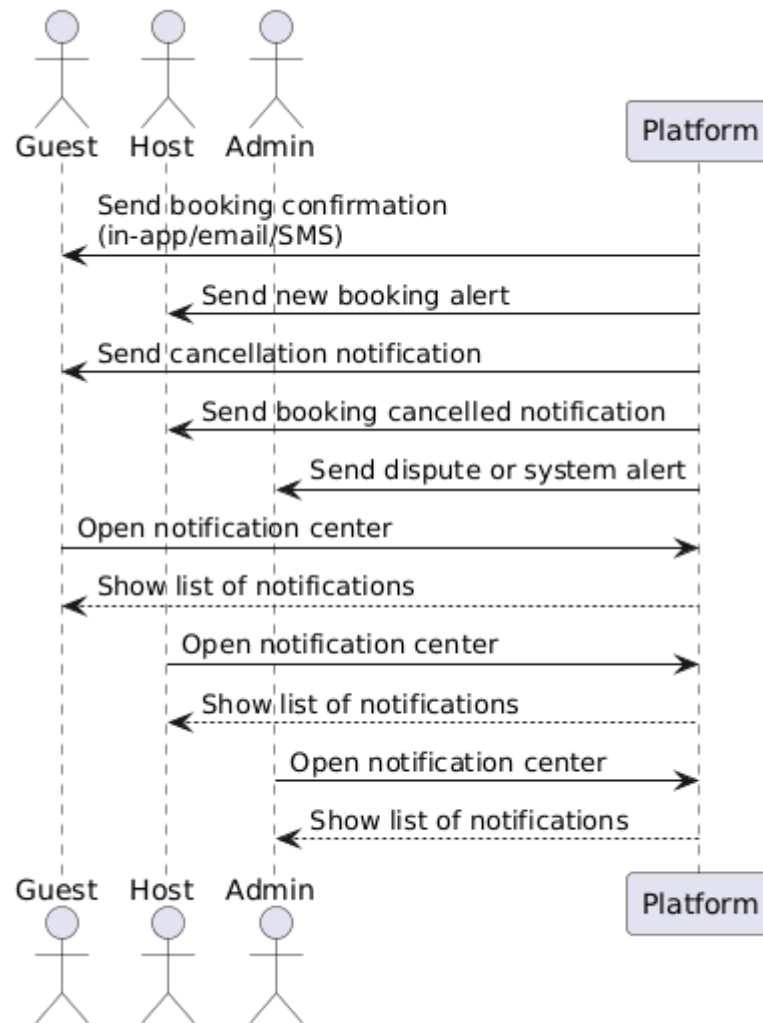


Figure 5-17: Notification and Alert (Guest, Host and Admin)

Property Availability Management (Host)

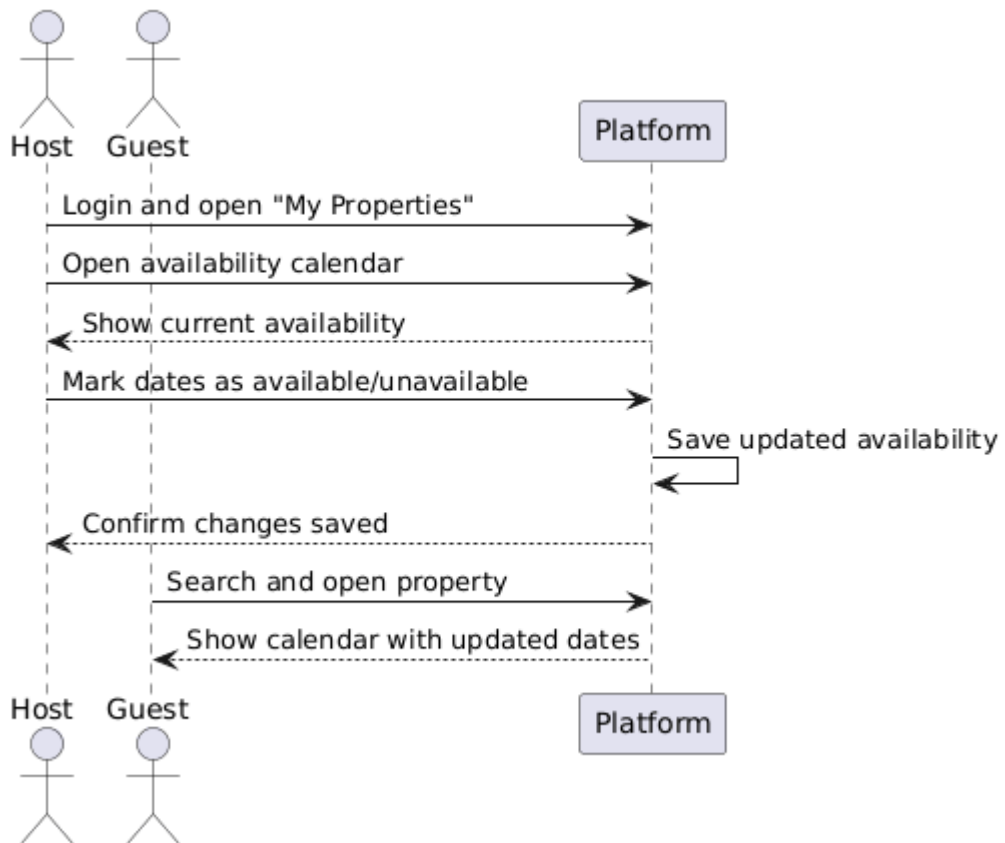


Figure 5-18: Property Availability Management (Host)

User Support and Dispute Resolution (Guest)

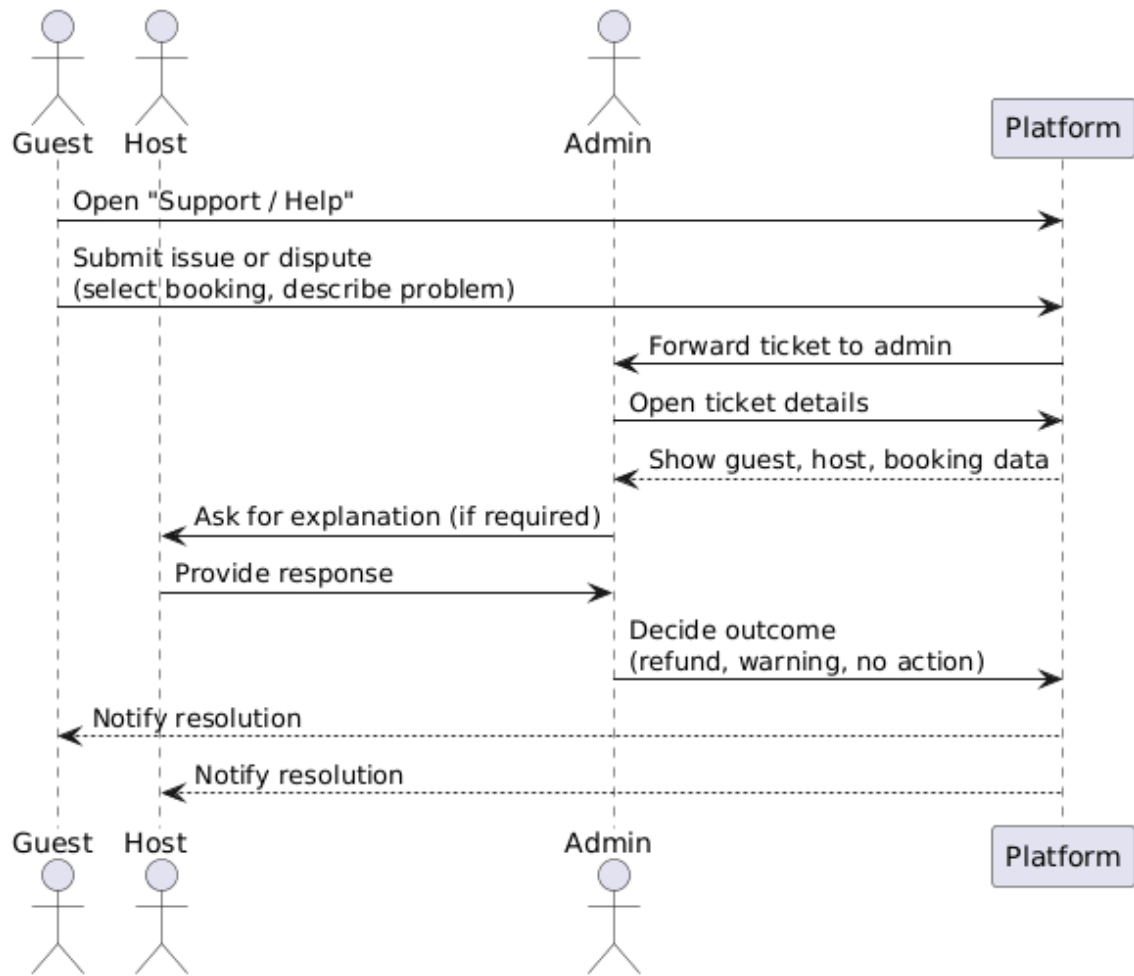


Figure 5-19: User Support and Dispute Resolution (Guest)

User Support and Dispute Resolution (Host)

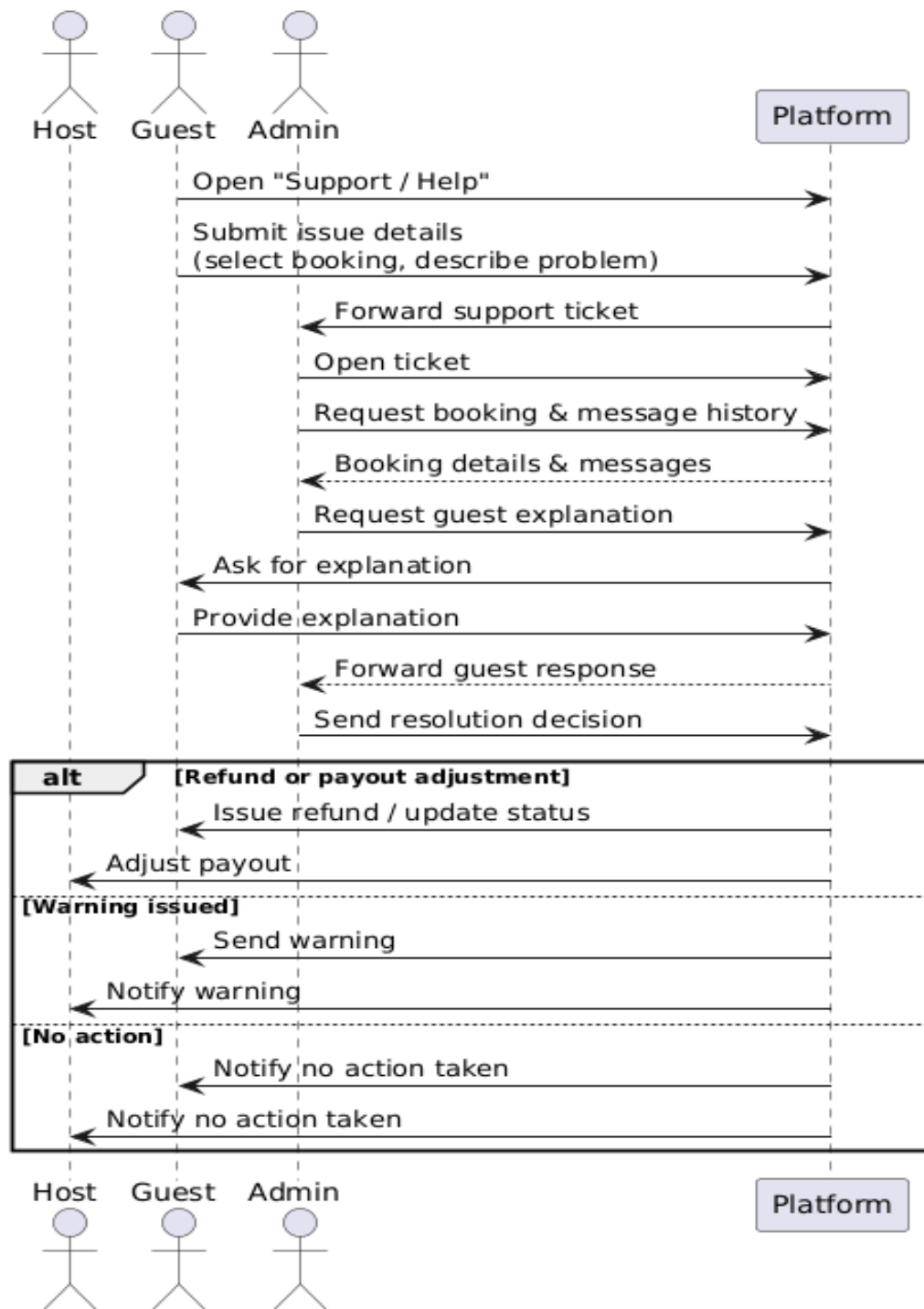


Figure 5-20: User Support and Dispute Resolution (Host)

User Support and Dispute Resolution (Admin)

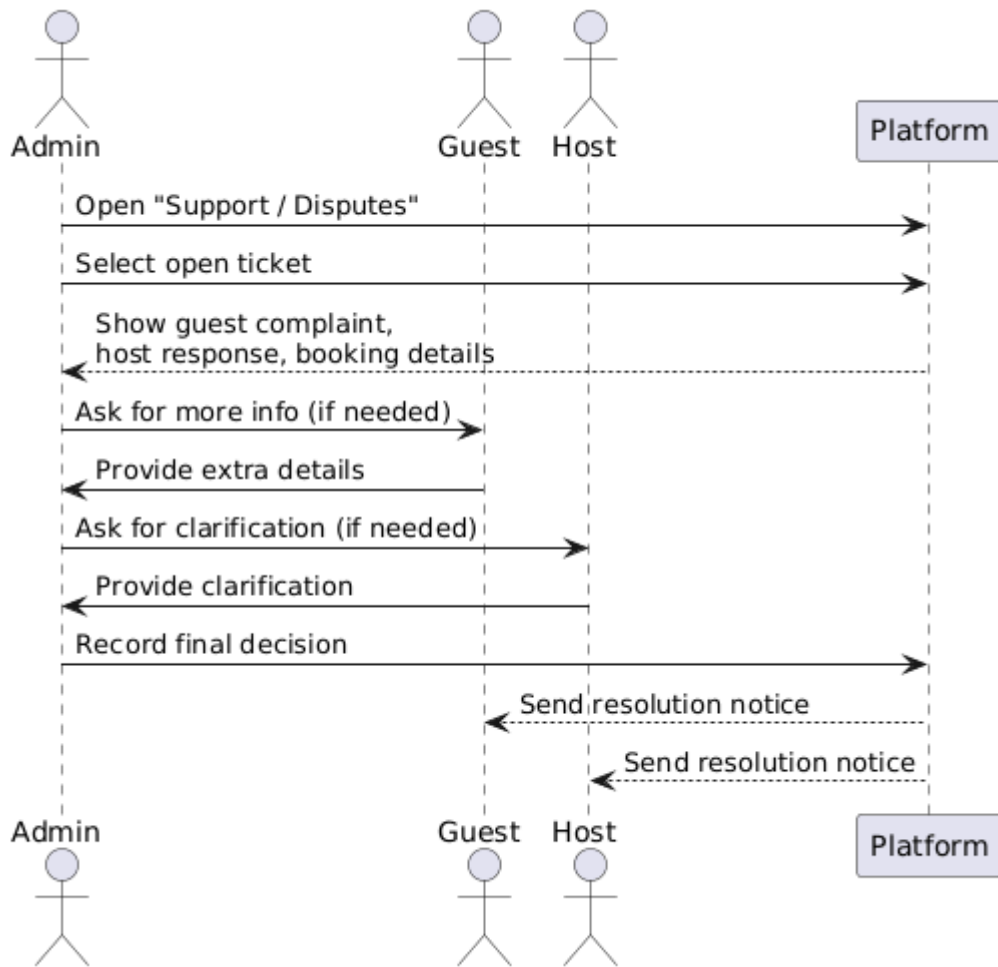


Figure 5-21: User Support and Dispute Resolution (Admin)

Chatbot (Guest)

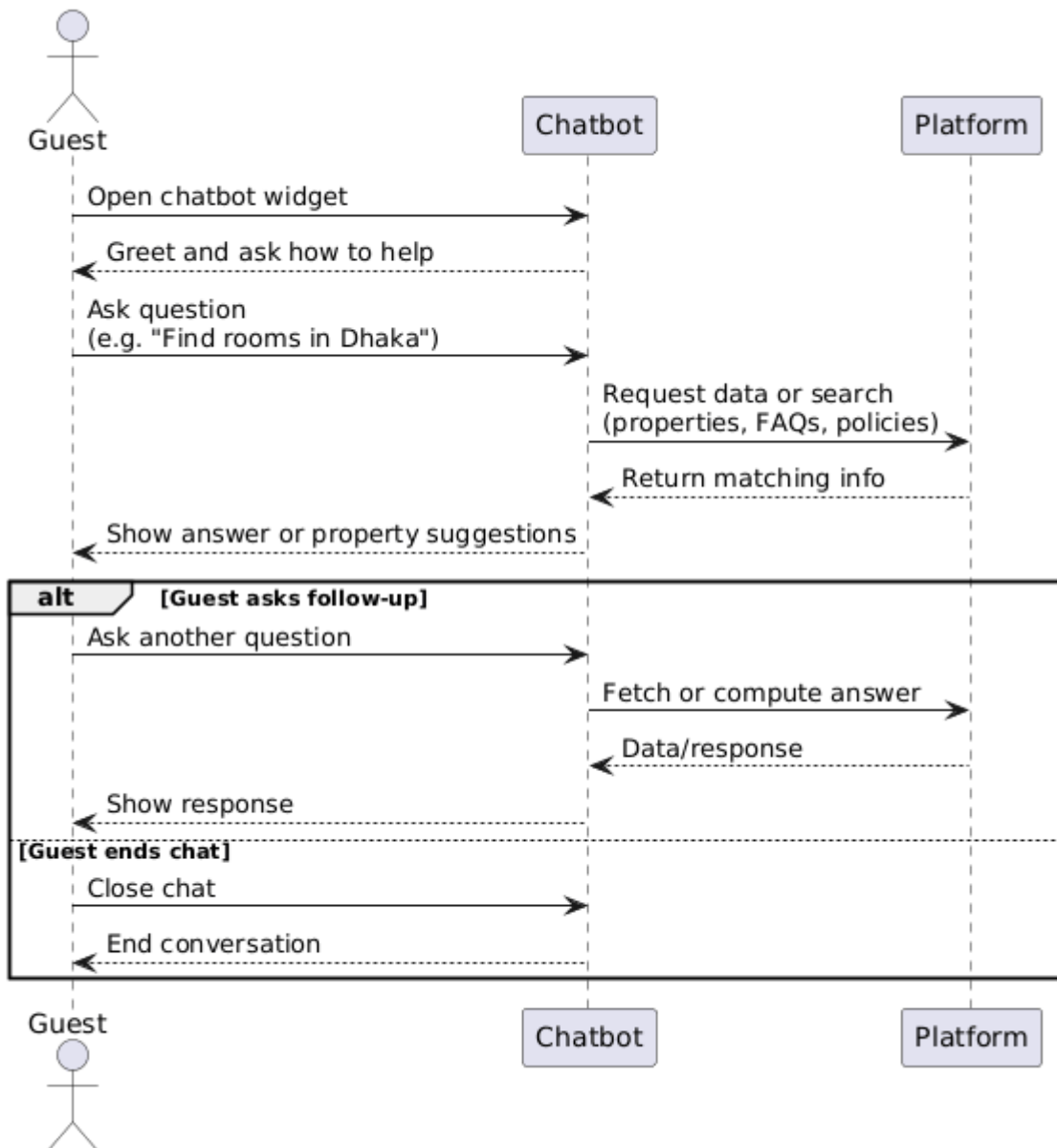


Figure 5-22: Chatbot (Guest)

2.4.6 ER Diagram

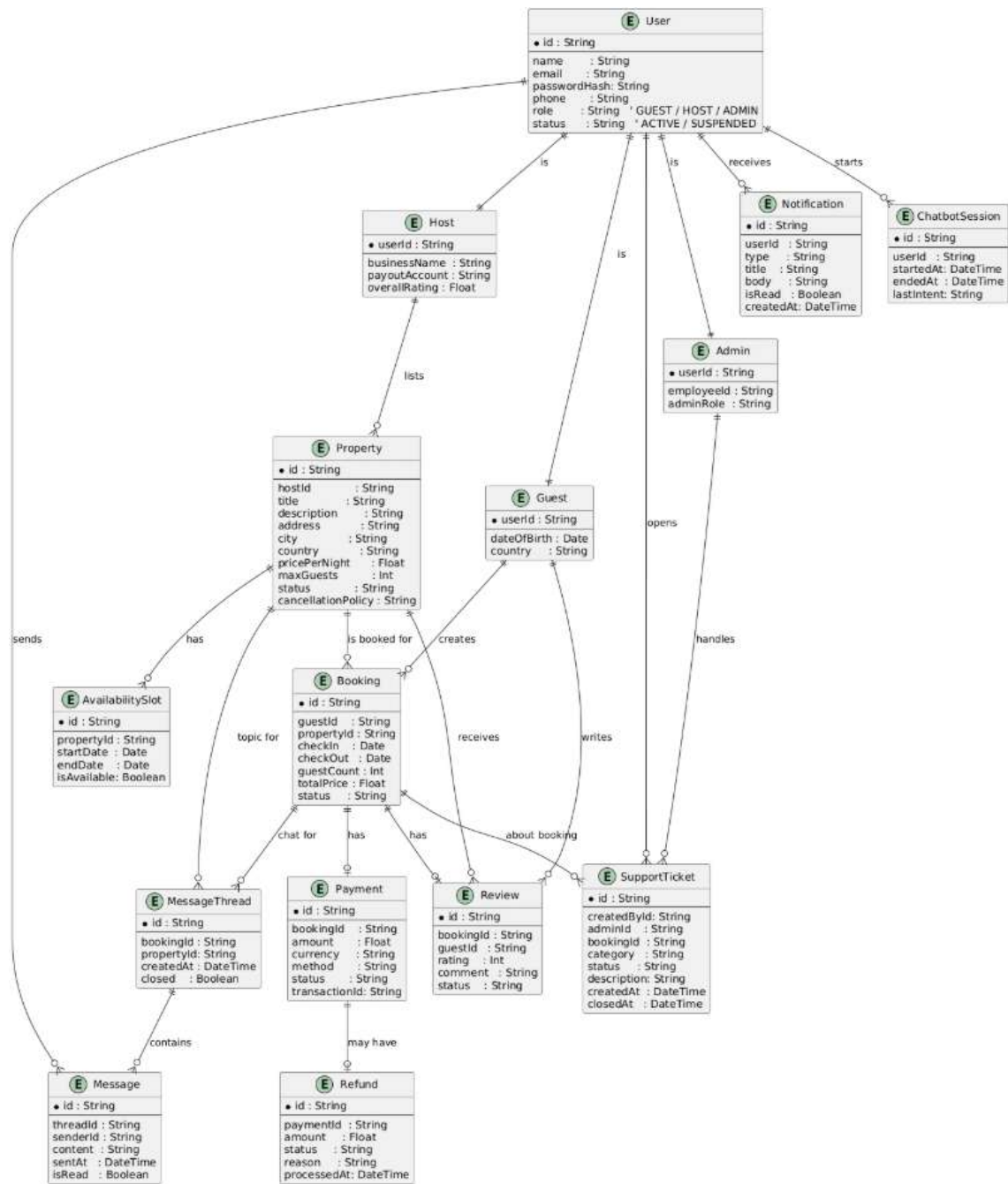


Figure 7-1: ER Diagram

2.5 Coding: Appendix B

- Database Schema
- Model
- Controller
- Service Layer
- Trait
- Route
- Helper
- Configuration
- Middleware
- API

2.6 Summary

In chapter 2, the general design and implementation plan of the StayBangla B&B and Property Rental System is stated. It describes the functional and non-functional requirements, which are then complemented by detailed UML diagrams, use case, activity, sequence, and class as well as ER models, which describe system behavior and data relationship. The chapter also describes the organization of the main modules, user accounts, property management, booking flow, payments, messaging, reviews, notifications, support tickets and chatbot interactions.

The implementation part identifies the use of Laravel (PHP) in the back-end development, MySQL in the database management and a web-based interface among the guests, hosts, and admins. The design models and implementation details taken together give a good outline of how the system functions and how every component functions in the overarching architecture.

Chapter 3 Software Testing

3.1 Introduction

StayBangla is a web-based application that allows the guests to book the properties, the hosts to list their offerings, and the administrators to receive payments and resolve conflicts. Due to the interdependence of these features, imperfection in one sphere may break the whole process of booking and cause financial problems.

Software testing is an exercise that is required to ascertain the functionality and reliability of any software. This will consist of navigating through user routes like sign-up, property listing, booking, cancellation, messaging, and support. The test levels applied are a number of testing levels: unit test to verify the functions, integration test to verify the communication between modules, and system/user acceptance test to verify the system overall performance. These also contribute to the credibility of StayBangla prior to implementation.

3.2 Testing Features

3.2.1 Feature to Be Tested

- a. **User Account & Authentication:** Register, Login, Logout, Forgot Password, and Role-based access control (Guest and Host).
- b. **Property Listing Management:** Host can create, edit, and remove listings, upload photos, and manage availability without errors.
- c. **Search & Filter:** Verifies that guests can browse for a property by location, price, dates, and amenities, and that the results load correctly.
- d. **Booking & Reservation Flow:** Validates the date was selected, confirms renter bookings, prices out costs, and avoids double bookings.
- e. **Payment Processing:** Opens service to test payment submission, transaction status, failure handling and refund eligibility.
- f. **Review & Rating:** Ensures guests can leave reviews, hosts receive replies, and admins can moderate inappropriate content.
- g. **Messaging System:** Ensures guests and hosts can send messages via a reliable, real-time thread.
- h. **Notifications:** Filters alerts for bookings, payments, cancellations, messages, and system events.
- i. **Support & Dispute Handling:** Verifies ticket generation, admin events, and log entries in the support flow.
- j. **Chatbot:** It tests quick responses, the accuracy of FAQs, and basic recommendation prompts.

3.3 Testing Strategies

| Test Type | Test Approach | Pass Criteria | Fail Criteria |
|-----------------------|---|--|---|
| Functional Testing | Test all features described in the project, including account management and property listing, booking logic, payment handling, and reviews. | All functions can be used successfully without crashes or errors. | Any of the expectations is not met. |
| Usability Testing | Take Real end users for a walk-through of the application. Capture feedback on user experience (UX) elements, such as navigation, instructions, and task progression. | Good feedback, easy navigation, and no problems with user interaction. | The feedback is negative, or you cannot perform simple operations such as booking or reviewing on this property page. |
| Security Testing | Conduct penetration testing and SQL injection testing, and verify the use of HTTPS and password hashing. | No vulnerabilities found, and secure (SSL) communication is in place for sensitive data. | Any reporting of security threats, weaknesses (e.g., credentials exposed, data breach). |
| Database Testing | Test CRUD (Create, Read, Update, and Delete) operations + Data consistency. | The data is inserted, updated, and deleted without errors. | Due to database query errors or unexpected datasets. |
| Performance Testing | Load test the application with the help of the load testing tools so that multiple users can jump into the platform in parallel. So, I want to measure how long it takes for any response and resource usage. | I can browse the site perfectly fine during high traffic, no lag or timeouts. | Indications that the website is very slow or not responding at all when fully loaded with simultaneous users. |
| Compatibility Testing | Now it is time to open the website on different devices like Desktop, Mobile, and Browsers (Chrome, Firefox, etc.) | The website is fully functional on every device and in every browser. | A significant portion of users or markets can't use the website on some devices/browsers. |

Table 5-1: Testing Strategies

3.4 System Testing (Test Cases with Report)

Test Case ID: 01

| Test Case: 3.4.1 | | Test Case Name: User Registration | | | | | |
|---|---------------------|--|----------------------------|--------------|-------------------------|--------|--|
| System: StayBangla | | Subsystem: User Authentication | | | | | |
| Designed by: Md. Kamrul Islam | | | | | | | |
| Executed by: Md Kamrul Islam | | | | | | | |
| Description | | The user registers for the StayBangla system by providing valid information. | | | | | |
| Pre-condition | | The user accesses the registration page. | | | | | |
| Step | First and Last Name | Email | Password & Retype Password | Guest / Host | Response | Status | Comment |
| 1 | Md Kamrul Islam | kamrul.kafi1@gmail.com | 123456 | Guest | Registration successful | Pass | Guest registration is successful with valid information. |
| 2 | | kamrul.kafi1@gmail.com | 123456 | Guest | Name field empty | Fail | Guest must input a name. |
| 3 | Md Kamrul Islam | kamrul98081@gmail.com | 123456 | Host | Registration successful | Pass | Host registration is successful with valid information. |
| 4 | Md Kamrul Islam | | 123456 | Host | Email field empty | Fail | Host must input a email. |
| Post-condition: The user is successfully registered, and the registration process is considered successful with valid information. | | | | | | | |

Table 6-1: User Registration

Test Case ID: 02

| | | | | | |
|---|------------------------|--|-------------------------|--------|---|
| Test Case: 3.4.2 | | Test Case Name: User Login | | | |
| System: StayBangla | | Subsystem: User Authentication | | | |
| Designed by: Md. Kamrrul Islam | | | | | |
| Executed by: Md Kamrul Islam | | | | | |
| Description | | The user login for the StayBangla system by providing valid information. | | | |
| Pre-condition | | The user accesses the login page. | | | |
| Step | Email | Password | Response | Status | Comment |
| 1 | kamrul.kafil@gmail.com | 123456 | Login successful | Pass | Guest login is successful with valid information. |
| 2 | | 123456 | Email field empty | Fail | Guest must input an Email. |
| 3 | kamrul98081@gmail.com | 123456 | Registration successful | Pass | Host login is successful with valid information |
| 4 | kamrul98081@gmail.com | | Password field empty | Fail | Host must input a Password. |
| Post-condition: The user is successfully login, and redirect to the dashboard. | | | | | |

Table 6-2: User Login

Test Case ID: 03

| Test Case: 3.4.3 | | Test Case Name: Property Listing Creation by Host | | | | | | | |
|--------------------------------------|--|---|---------------|-------------|------------|--------------|--|--------|--|
| System: StayBangla | | Subsystem: Property Management (Host) | | | | | | | |
| Designed by: Md. Kamrul Islam | | | | | | | | | |
| Executed by: Md Kamrul Islam | | | | | | | | | |
| Description | | Host creates a new property listing with valid details. | | | | | | | |
| Pre-condition | | The host is logged in and navigates to the property management page. | | | | | | | |
| Step | Title | Description | Price | Location | Max Guests | Availability | Response | Status | Comment |
| 1 | Modern City Apartm ent in the Heart of Dhaka | 2 Bedr oom s,2 Bath s | 3000tk/ night | Dhaka | 4 | Availa ble | Property listing submitte d for review | Pas s | Property was submitted successfull y. |
| 2 | Beachfr ont Cottage with Stunnin g Sea Views | 3 Bedr oom s,2 Bath s | 4000tk/ night | Cox's Bazar | 6 | Availa ble | Property listing submitte d for review | Pas s | Property was submitted successfull y. |
| 3 | Eco Friendl y Lodge in the Sundarb ans | 1 Bedr oom, 1 Bath | 5000tk/ night | Khulna | 2 | Unava ilable | Property submissi on failed, availabil ity issue | Fail | Host ensure availability is set correctly. |
| Post-condition: | | The property is listed successfully and submitted for admin approval. | | | | | | | |

Table 6-3: Property Listing Creation by Host

Test Case ID: 04

| | | | | | | | | | |
|---------------------------|--|--|--|--|--|--|--|--|--|
| Test Case: 3.4.4 | | Test Case Name: Booking and Reservation | | | | | | | |
| System: StayBangla | | Subsystem: Booking System (Guest) | | | | | | | |

| Designed by: Md. Kamrrul Islam | | | | | | | | |
|--|---|--|----------------|--------|------------------|---------------------------------------|--------|--|
| Executed by: Md Kamrul Islam | | | | | | | | |
| Description | | A guest successfully books a property and receives confirmation. | | | | | | |
| Pre-condition | | Guest is logged in and selects a property. | | | | | | |
| Step | Property | Check-in Date | Check-out Date | Guests | Payment Details | Response | Status | Comment |
| 1 | Modern City Apartment in the Heart of Dhaka | 01/11/2025 | 03/11/2025 | 3 | bKash (Valid) | Booking confirmed | Pass | Booking successfully made and confirmed. |
| 2 | Beachfront Cottage with Stunning Sea Views | 05/11/2025 | 10/11/2025 | 4 | Card (Valid) | Booking confirmed | Pass | Booking successfully made and confirmed. |
| 3 | Eco Friendly Lodge in the Sundarbans | 10/11/2025 | 12/11/2025 | 2 | Rocket (Invalid) | Payment failed, booking not confirmed | Fail | Invalid payment details rejected. |
| Post-condition: Booking is confirmed, and guest receives booking details via email. | | | | | | | | |

Table 6-4: Booking and Reservation

Test Case ID: 05

| | |
|---------------------------------------|--|
| Test Case: 3.4.5 | Test Case Name: Booking Cancellation |
| System: StayBangla | Subsystem: Booking Cancellation (Guest) |
| Designed by: Md. Kamrrul Islam | |
| Executed by: Md Kamrul Islam | |

| | | | | | | | |
|--|---|--|-----------------------------|-----------------------|--------------------------------------|---------------|---|
| Description | | The guest cancels a booking successfully according to the cancellation policy. | | | | | |
| Pre-condition | | The guest has an active booking. | | | | | |
| Step | Booking | Cancellation Policy | Cancellation Request | Refund Process | Response | Status | Comment |
| 1 | Modern City Apartment in the Heart of Dhaka | Full refund if cancelled 72 hrs in advance | Cancel booking | Process refund | Booking cancelled , refund processed | Pass | Booking successfully cancelled with refund. |
| 2 | Beachfront Cottage with Stunning Sea Views | No refund for late cancellations | Cancel booking | No refund | Booking cancelled , no refund | Pass | No refund processed, cancellation successful. |
| 3 | Eco Friendly Lodge in the Sundarbans | 50% refund if cancelled within 72 hrs | Cancel booking | Process refund | Booking cancelled , refund processed | Pass | Booking successfully cancelled with refund. |
| Post-condition: Guest's booking is cancelled, and the applicable refund is processed. | | | | | | | |

Table 6-5: Booking Cancellation (Guest)

Test Case ID: 06

| | |
|---------------------------------------|--|
| Test Case: 3.4.6 | Test Case Name: Review and Rating |
| System: StayBangla | Subsystem: Review and Rating System (Guest) |
| Designed by: Md. Kamrrul Islam | |
| Executed by: Md Kamrul Islam | |
| Description | Guest leaves a review for a property after their stay. |
| Pre-condition | The guest has completed their stay at the property. |

| Step | Property | Rating | Review | Response | Status | Comment |
|--|---|---------|--|---------------------------------------|--------|---|
| 1 | Modern City Apartment in the Heart of Dhaka | 5 stars | "Amazing experience" | Review submitted successfully | Pass | Review submitted successfully and visible to Host. |
| 2 | Beachfront Cottage with Stunning Sea Views | 4 stars | "Good, but needs improvement in cleanliness" | Review submitted successfully | Pass | Review submitted successfully and visible to Host. |
| 3 | Eco Friendly Lodge in the Sundarbans | 0 stars | "Didn't like the property" | Review not submitted, rating required | Fail | Guest must provide a rating before submitting a review. |
| Post-condition: Review is visible to Host and included in the property's ratings. | | | | | | |

Table 6-6: Review and Rating System (Guest)

3.5 Summary

Chapter 3 explains how the StayBangala system was tested so as to ensure that core features are functioning properly and reliably. It discusses the test strategy, the primary types of tests (unit, integration, system and user acceptance) and some of the test cases which includes, login, property listings, reservation, payments, reviews, messages, notifications, support and chat bot. Simple pass/fail criteria are also described in the chapter and how testing facilitated the detection and correction of defects prior to deployment.

Chapter 4 Deployment and Maintenance

4.1 Introduction

StayBangla Property Rental site creation and Upkeep. The StayBangla Property Rental platform will need proper implementation and maintenance, such that it will run without issue and be deployed safely and securely with a high level of scalability. Deployment is the process of taking the application to production and making sure that everything is functional. Maintenance means keeping the track of the system performance, bug fixing, security patches, and addition of more capacity to serve more users. This implicates that the platform must be backed up regularly, user-feedbacked, and optimized on a regular basis in order to ensure there is a smooth user experience.

4.2 SRLC (Software Release Life Cycle)

Planning & Requirement Gathering

Objective:

- Understand the purpose of the platform, the needs of the users, and the functionality they will need (i.e. listings of the properties, booking system, payment gateway, user control, etc.)
- Specify the system's technical, functional, and non-functional requirements.

Activities:

- Stakeholder meetings (Guests, Hosts, Admins)
- Feature list creation and prioritization
- Project scope and resource allocation at the outset

Design & Prototyping

Objective:

- Design the system layout, databases, and user interfaces as needed.
- Build prototypes of the significant features to get user validation sooner.

Activities:

- Design UML Diagrams (use case, activity, sequence, and class diagrams)
- Front-end user interface (UI) wireframes
- Database design (ER and schema design)

Development

Objective:

- Prepare and start writing code for the project according to the design.
- Build out the front-end and back-end of the platform, ensuring it meets all functional and non-functional requirements.

Activities:

- Back-end: Write server-side code with Laravel (authentication, booking, payment, etc.)
- Front-end: Develop responsive UIs using HTML, CSS, and a JS framework.
- Database Setup: Write a MySQL schema and implement CRUD operations.

Testing (QA)

Objective:

- Make sure the deployed system is bug-free, secure, and stable.
- Do thorough testing of functionality, performance, and security.

Activities:

- Test discrete components and functions.
- Testing components' functional compatibility together (e.g., testing the data format between the booking system and the payment gateway).
- Stress the system for load testing
- Verify that the platform is safe from threats.
- Validate that Real Users (Guests, Hosts, Admin) can use the system without a hassle.

Deployment

Objective:

- Put the platform in that live production environment and let end users use it.

Activities:

- Set up production servers.
- Set up databases, file storage, and network services.
- Rolling out the app code to production.
- Configure domain names and SSL certificates for secure access.

Release

Objective:

- Bring the platform to life and make it available to users so that individual and organisations can start using it. Assure a smooth shift from prototype to production.

Activities:

- Publish the site to the public and users.
- Deal with any unknown problems or bugs straight after the launch.
- Support users with basic queries (e.g., forgotten passwords, booking issues).

Maintenance & Updates

Objective:

- Maintain the platform's smooth operation by resolving bugs and implementing security fixes, and by deploying new features/updates.

Activities:

- Bug Fixing: Browse the updates for user-reported bugs and errors for StayBangla!
- Security Patch Updates: Ensure that security patches are updated regularly.
- New Feature: Release new features based on user feedback, e.g., more payment options and better search filters.
- Performance Monitoring: Monitor server performance regularly and scale as required.

End of Life (EOL)

Objective:

- Terminate the platform's lifetime at a specific time or when a major release is required.

Activities:

- User data can be migrated or archived at any time when see fit.
- Let users know when the platform is expected to reach EOL.
- Migrate users to a new version of the platform or service when applicable.

Chapter 5 User Manual

5.1 Introduction

StayBangla is a web-based platform that connects Guests looking for short-term stays with Hosts offering properties. The system includes features such as property listings, search and filtering, booking and payments, messaging, reviews, notifications, cancellation management, a support ticket system, and an AI chatbot for user assistance.

This user manual explains how to use the system from the perspective of all three user types: Guest, Host, and Admin.

5.2 Project Functionalities

5.2.1 Guest Functionalities

1. Create an account
2. Log in and access the dashboard
3. Search for properties
4. Apply filters (price, location, guests, amenities)
5. View property details
6. Check the availability calendar
7. Book a property
8. Make online payments
9. View and manage bookings
10. Cancel bookings
11. Request refunds (based on policy)
12. Leave reviews and ratings
13. Message hosts
14. Receive notifications
15. Use a chatbot for help and recommendations
16. Update personal profile

5.2.2 Host Functionalities

1. Create a host account
2. Log in to the host dashboard
3. Add new property listings
4. Upload property photos
5. Edit or delete property listings
6. Manage availability calendar
7. Set pricing and amenities
8. View booking requests
9. Accept or decline bookings
10. Message guests
11. View and respond to guest reviews
12. Manage host profile
13. Receive notifications

5.2.3 Admin Functionalities

1. Log in to the admin dashboard
2. Review and approve property listings
3. Reject or request edits on listings
4. Manage all user accounts (Guest/Host)
5. Suspend, activate, or update profiles
6. Monitor all bookings
7. Oversee payments and refunds
8. Handle disputes and support tickets
9. Manage system-wide notifications
10. Track platform activity
11. Ensure policy compliance

5.2.4 Stay Bangla System

1. Protected Login and Registration
2. Messaging System Integrated
3. Automatic notifications push
4. AI chatbot asset to the guest
5. Local Payment gateway integration
6. Booking and availability engine
7. Review and rating module
8. Help ticket system
9. Data repository and control

Home Page

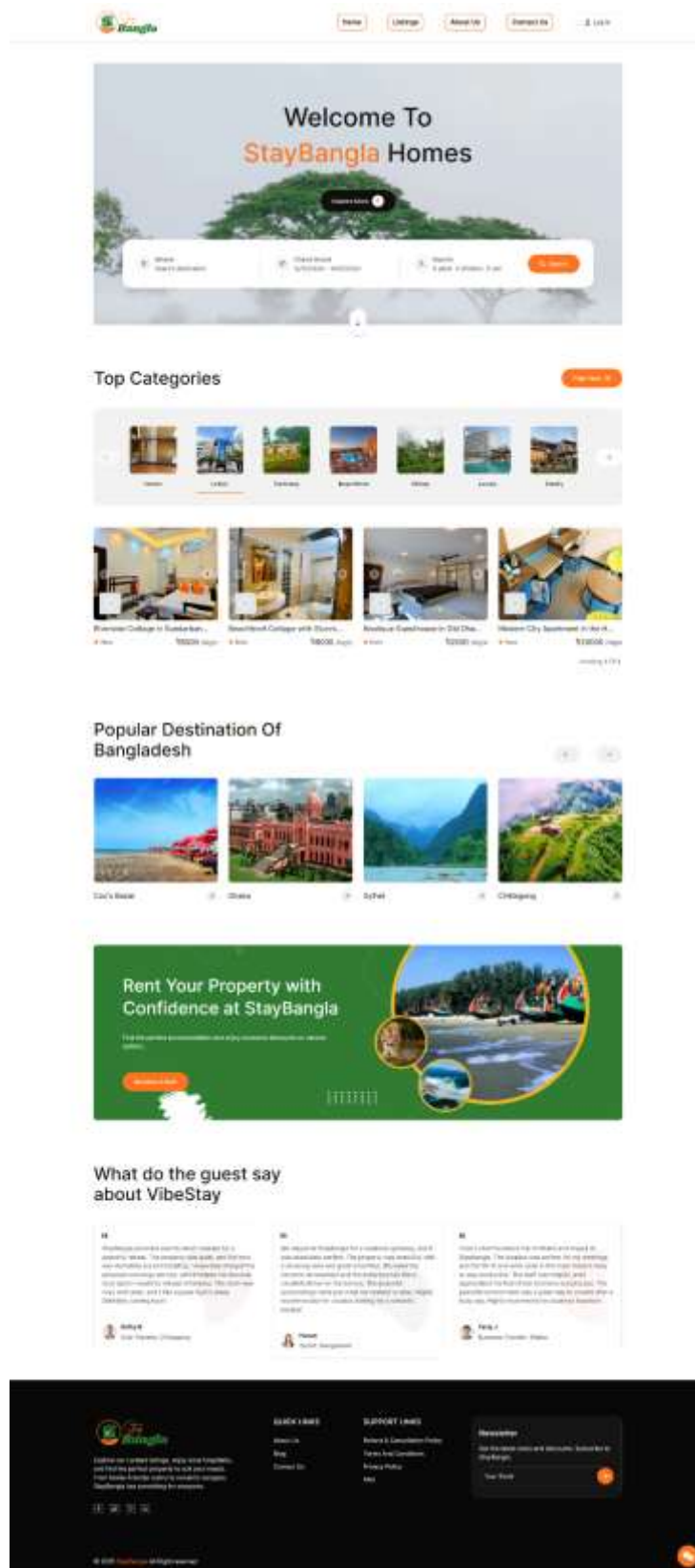


Figure 8-1: Home Page

Listing Page

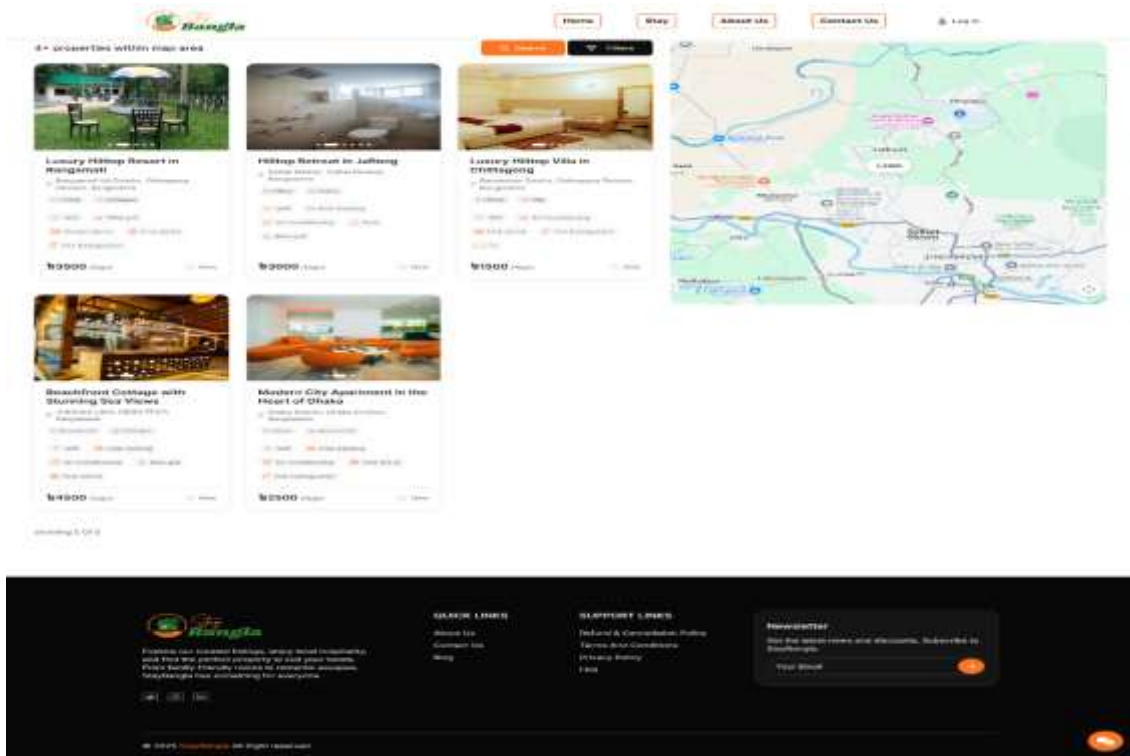


Figure 8-2: Listing Page

Registration

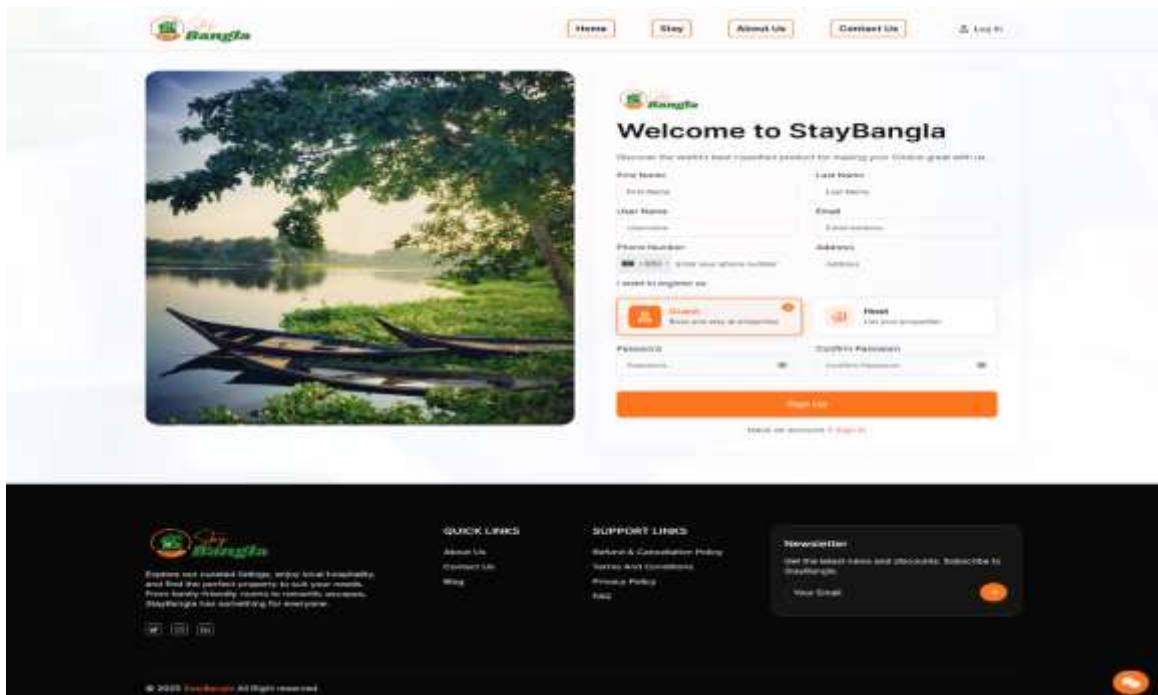


Figure 8-3: Registration

Login

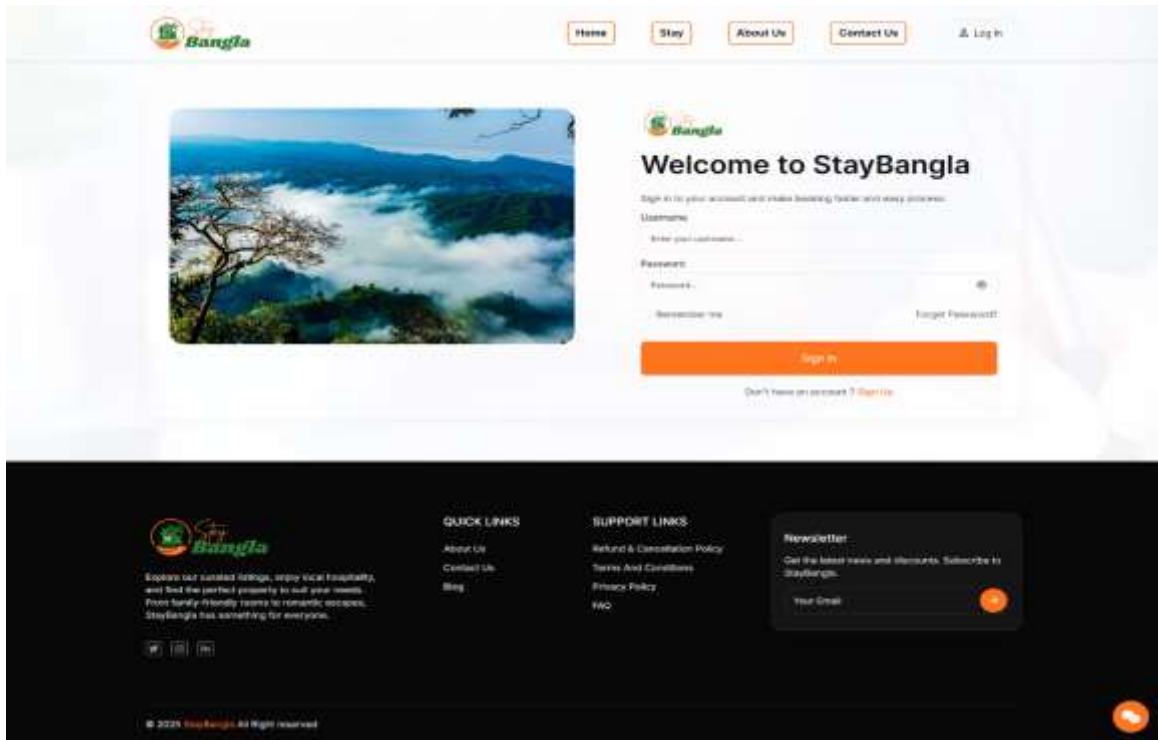


Figure 8-4: Login

Admin Dashboard

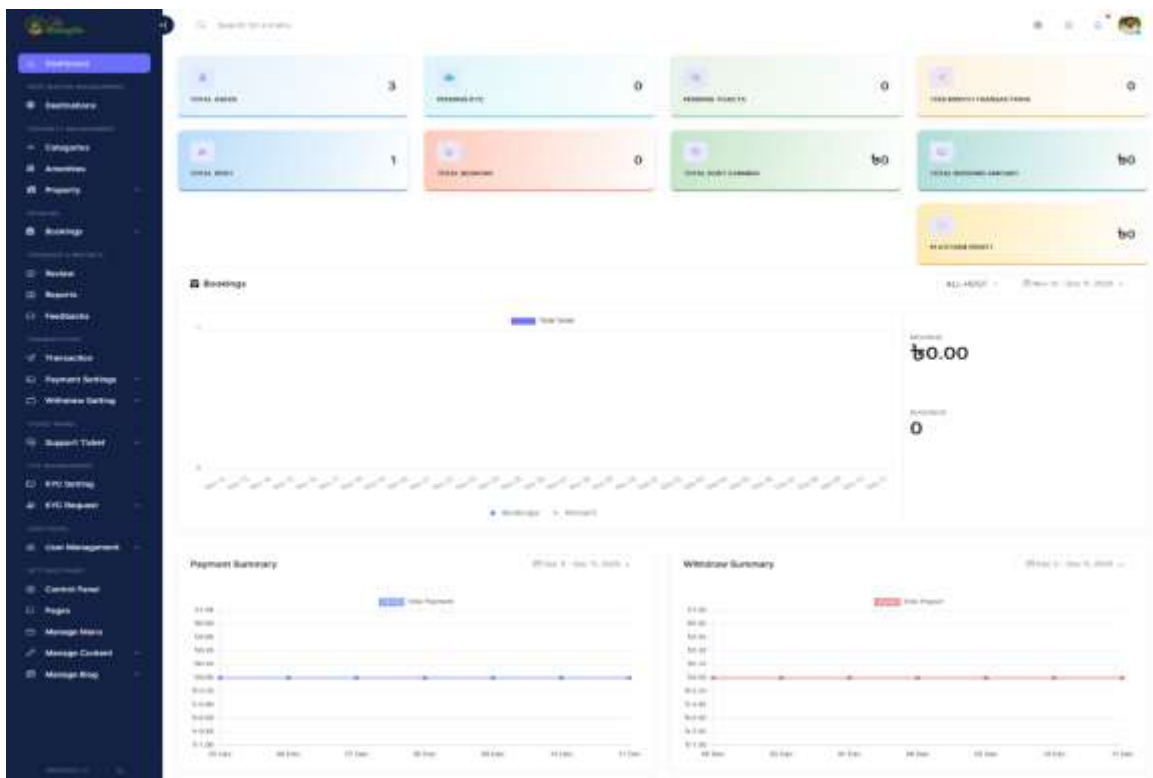


Figure 8-5: Admin Dashboard

Host Dashboard

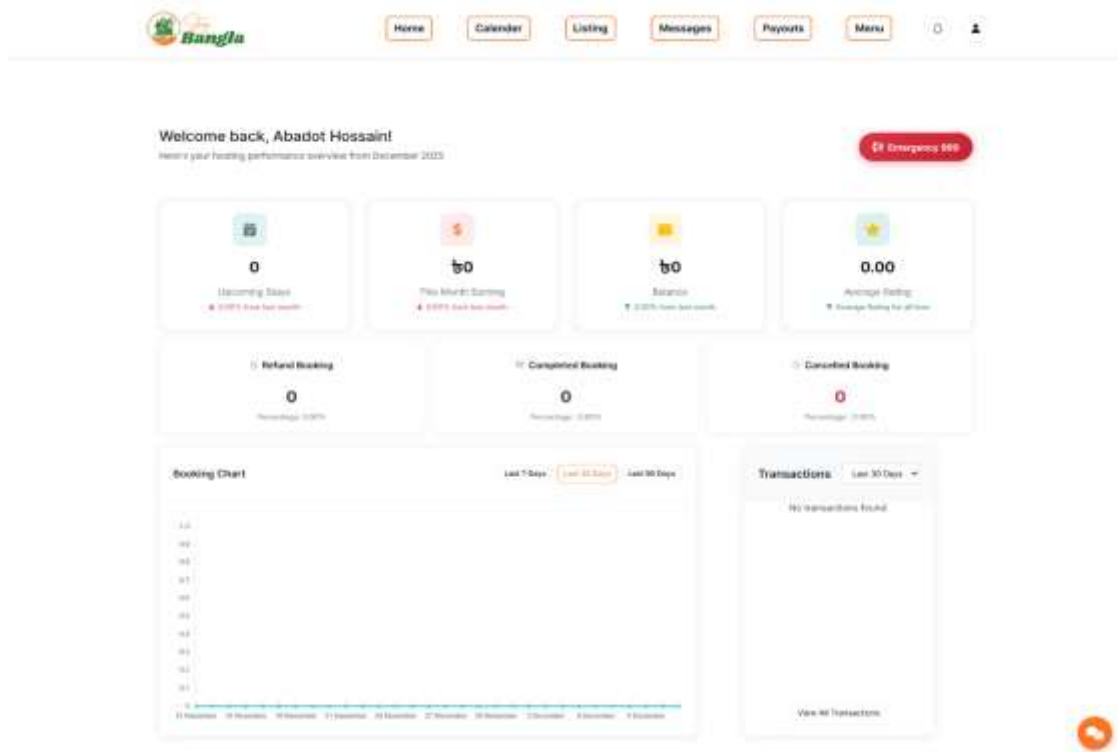


Figure 8-6: Host Dashboard

Host Listing Page

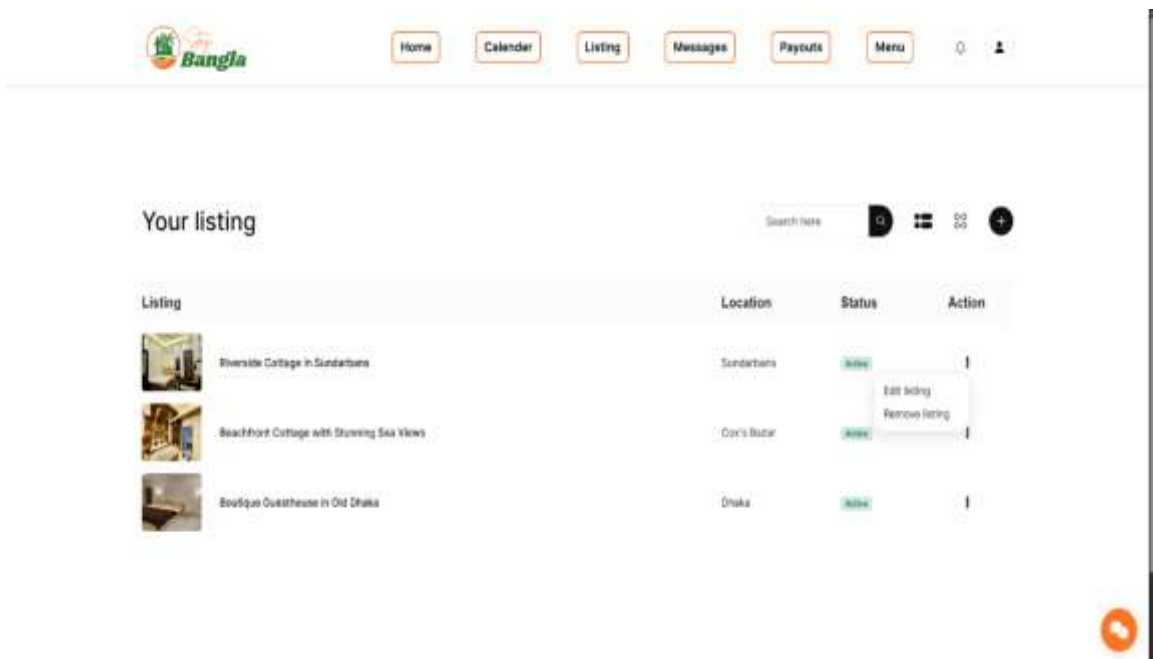


Figure 8-7: Host Listing Page

Booking Checkout Page

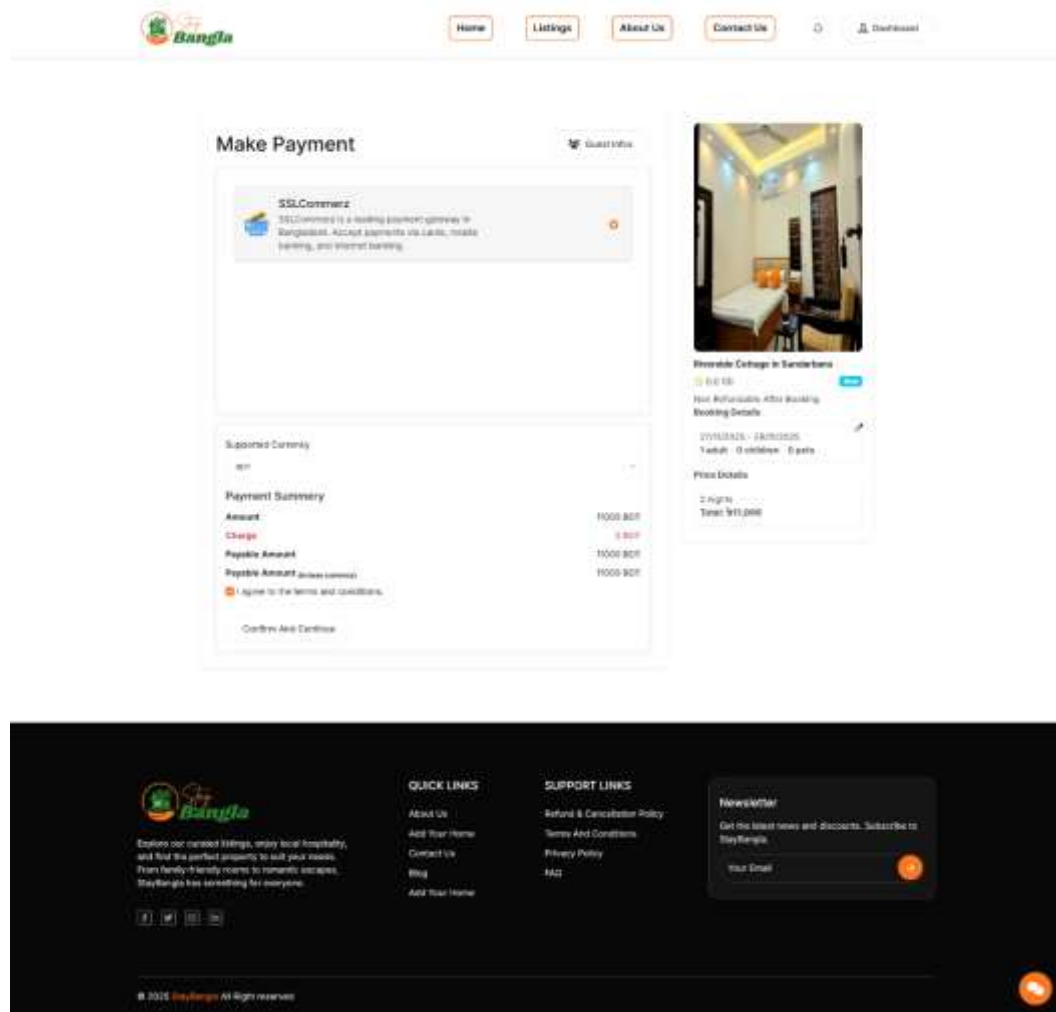


Figure 8-8: Booking Checkout Page

Guest Payment History

Account > Payment History

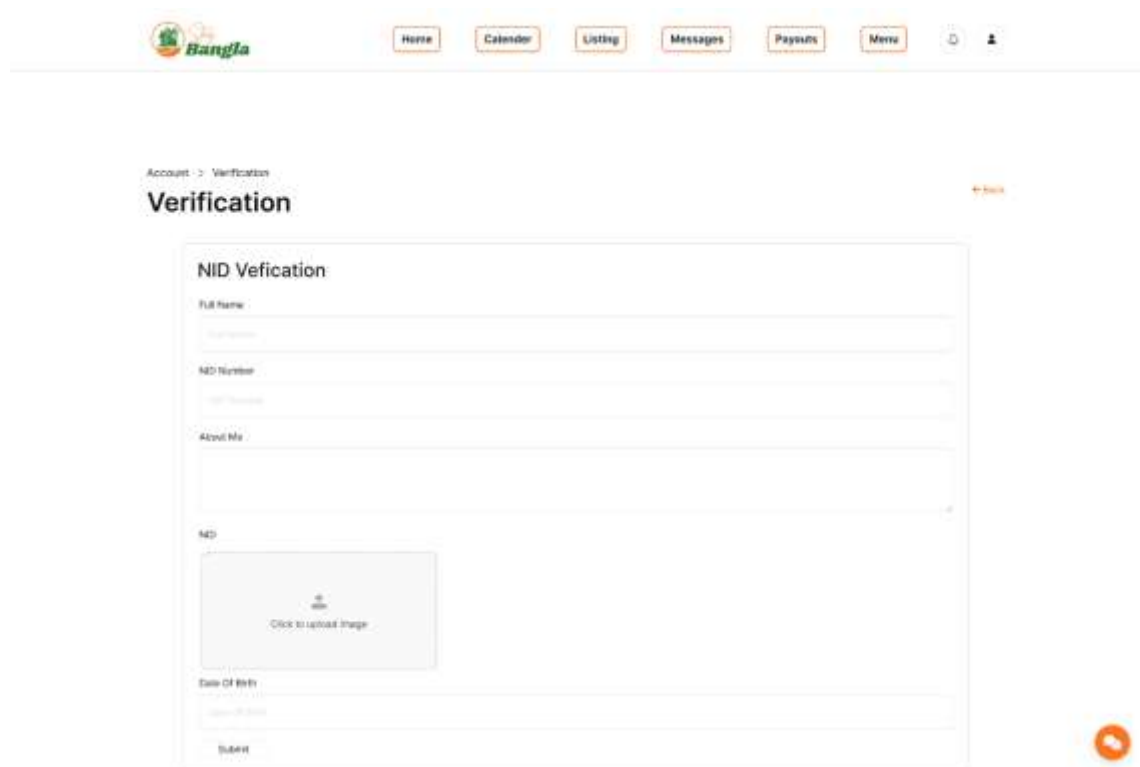
Payment History

Filter

| Method | Transaction | Amount | Status | Action |
|------------|----------------|--------|---------|--------|
| SSLCommerz | D4202026133143 | 1000 | Success | |

Figure 8-9: Guest Payment History

Guest NID/Passport Verification



The screenshot shows the 'Verification' page of the StayBangla website. At the top, there is a navigation bar with the StayBangla logo and buttons for Home, Calendar, Listing, Messages, Payouts, and Menu. Below the navigation bar, the page title is 'Verification'. The main content area contains a form titled 'NID Vefication' (note the typo) with the following fields: 'Full Name', 'NID Number', 'About Me', 'NID' (with a placeholder image and the text 'Click to upload image'), and 'Date Of Birth'. A 'SUBMIT' button is located at the bottom of the form. There is also a 'Back' link in the top right corner of the form area.

Figure 8-10: Guest Passport Verification

5.3 Summary

The fifth chapter describes the interaction of users with the StayBangla system and identifies the principal features that are offered to the Guests, Hosts, and Admins. It offers the norms of instructions and actions, which involve the creation of accounts, logging in, searching of properties, bookings, payments, and NID verification. In the case of hosts, the chapter discusses the property listing page and dashboard. The admin dashboard is also presented. In general, it can be seen as a useful manual to the working process and the knowledge of the main functionality of the platform.

Chapter 6 Project Summary

6.1 Introduction

StayBangla Property Rental is a full service of property rental and B&B solution which targets international space rental market. Visitors are able to find and make a reservation of properties and to review. The site hosts post their property (or properties) to be found. Book guests, take and maintain bookings and store all property information. Admins oversee facilitation of the proper running of a platform and adherence to rules and regulations. Moreover, an AI-driven chatbot is used to respond to user queries and recommend properties, which would even more beautify user experience and support.

6.2 Project Limitation

Geographical reach: The system is designed to operate in specific regions and may struggle to scale globally without additional localization features.

Internet Dependence: Both host and guest need fast internet, so it may not be applicable in many places where the internet is either too slow.

Payment Gateway: Limited integration with a few payment gateways (such as SSLcommerz) prevents users from checking out with other regional and international payment options.

Platform Compatibility: The platform is compatible with up-to-date browsers and devices; however, not all browsers or devices will provide full functionality.

6.3 Scope

Listings: Hosts can modify their property listing(s). Listings are reviewed and approved by admins.

Search and Booking: Guests can search for properties using filters (price, location, amenities) and book them.

User accounts: Hosts and guests can register, log in to their profiles, and keep track of their bookings.

Chatbot: In-built chatbot AI to help users book, request a property, and answer some general questions 24/7.

Payment and Review: Guests can safely pay for their booking and review properties; hosts can respond to reviews.

Admin Panel: Admins manage user and listing profiles and ensure the service rules are followed.

6.4 Future Work

Mobile App: Creating a mobile application that will make the process more user-friendly and easier.

AI: The app elevates the AI chatbot to the next level by incorporating machine learning for personalized recommendations and responses.

More Payment Gateways: Introducing more international and regional payment gateways to give the user a plethora of choices.

Machine Learning Home Management: Smart home connections which allow hosts to automate property availability, pricing and bookings.

6.5 Conclusion

The StayBangla Property Rental B&B is a complete one solution rental management and booking system. It provides a free-flowing experience both to the guests and the hosts and adopts technology to enhance interaction as well as simplify operation. The task is not comprehensive in scope, in territory, or in payment, but it does provide a platform to be further developed. My system will further grow in territory and enhance my users by adding mobile applications and AI optimizations to it, expanding its market penetration and allowing the system to better satisfy its users in the property rental industry

