



Web-Based Hotel Reservation System

Submitted By

Md. Shakwat Mozumder Alif

ID: 221-35-1057

Supervised By

Mr. Md. Rajib Mia

Lecturer (Senior scale)

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**

@ All right Reserved by Daffodil International University

©Daffodil International University



Department of Software Engineering
Faculty of Science and Information Technology
Supervisor Approval Form

Fall 2025	B.Sc. In SWE	Campus: DSC
-----------	--------------	-------------

Student Name	Student ID
Md. Shakwat Mozumder Alif	221-35-1057

Project/Thesis Information	
Project/Thesis Title	Web-Based Hotel Reservation System
Type of work	Project

Supervisor information	
Supervisor Name	Mr. Md. Rajib Mia
Supervisor Initial	RM
Completed Credit till now	136
How many credits in this semester	9
Supervisor Consent	<input checked="" type="checkbox"/> Yes <input type="checkbox"/> No

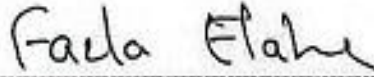
Supervisor Signature

Md. Rajib Mia
Lecturer (Senior Scale)
Department of Software Engineering
Faculty of Science and Information Technology
Daffodil International University

APPROVAL

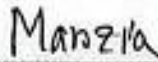
This project titled on "Web-Based Hotel Reservation", submitted by Md. Shakwat Mozumder Alif (ID: 221-35-1057) to the Department of Software Engineering, Daffodil International University has been accepted as satisfactory for the partial fulfillment of the requirements for the degree of Bachelor of Science in Software Engineering and approval as to its style and contents.

BOARD OF EXAMINERS



Dr. Fazla Ealhe
Assistant Professor & Associate Head
Department of Software Engineering
Faculty of Science and Information Technology
Daffodil International University

Chairman



Dr. Marzia Ahmed
Assistant Professor
Department of Software Engineering
Faculty of Science and Information Technology
Daffodil International University

Internal Examiner 1



Dr. Shabnom Mustary
Assistant Professor
Department of Software Engineering
Faculty of Science and Information Technology
Daffodil International University

Internal Examiner 2



Md. Rajib Mia
Lecturer (Senior Scale)
Department of Software Engineering
Faculty of Science and Information Technology
Daffodil International University

Internal Examiner 3



Mohammad Abul Kashem, PhD
Professor
Department of Computer Science and Engineering
DUET, Bangladesh

External Examiner

SUPERVISOR'S DECLARATION

I hereby declare that I have checked this project and in my opinion, this project is adequate in terms of scope and quality for the award of the degree of Bachelor of Science.



Md. Rajib Mia
Lecturer (Senior Scale)
Department of Software Engineering
Faculty of Science and Information Technology
Daffodil International University

(Supervisor's Signature)

Full Name : **Mr. Md. Rajib Mia**
Position : **Lecturer (Senior Scale)**
Date : 25 November 2025

STUDENT'S DECLARATION

I hereby declare that the work in this project is based on my original work except for quotations and citations which have been duly acknowledged. I also declare that it has not been previously or concurrently submitted for any other degree at Daffodil International University or any other institution.



(Student's Signature)

Full Name : **Md. Shakwat Mozumder Alif**

ID Number : **221-35-1057**

Date : 25 November 2025

ACKNOWLEDGEMENTS

First of all, I'm grateful to The Almighty Allah for making us eligible to complete this project.

Then I would like to thank my supervisor "Mr. Md. Rajib Mia", Department of Software Engineering. I'm extremely grateful and indebted to his expert, sincere and valuable guidance and encouragement extended to us.

I take this opportunity to record our sincere thanks to all the faculty members of the Department of Software Engineering for their help and encouragement.

Last but not least, I would like to thank my parents, for their unconditional support, love and without this we would not have come this far.

DEDICATION

I therefore declare that I have done this project under the oversight of “**Mr. Md. Rajib Mia**”, “**Lecturer (Senior Scale)**”, 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

Luxe Hotel is a Web-Based Hotel Reservation System for the solution of hotel room bookings. For travelers or for business trips Luxe Hotel can be a very good choice and helpful to book a room through quick and easy clicks.

This system is very easy for customers to find rooms, they can read details about the hotel and book a room. Customers has the free will of booking by choosing how many guests they will have.

A user friendly interface also guarantees that the users and administrators can experience seamless booking process, and check room availability and many more.

TABLE OF CONTENT

DECLARATION	i
TITLE PAGE	ii
ACKNOWLEDGEMENTS	vi
DEDICATION	vii
ABSTRACT	viii
LIST OF TABLES	
LIST OF FIGURES	
CHAPTER 1 INTRODUCTION	1
1.1 Background	1
1.1.1 Context and Relevance	1
1.1.2 Problem Identification	1
1.1.3 Purpose and Justification	1
1.1.4 Scope	2
1.2 Project Planning and Initiation	2
Feasibility Study (Step-by-Step)	2
1.3 Target User Profile and Tentative Elicitation Process	3
1.3.1 Target User	3
1.3.2 User profile	3
1.3.3 Elicitation Process	5
1.4 Project Block Diagram	6
1.5 System Requirements	6
1.5.1 Hardware Requirements	6

1.5.2 Software Requirements	7
1.5.3 Constraints and Dependencies	7
1.6 Project Scheduling	8
1.7 Summary	8
CHAPTER 2 DESIGN AND IMPLEMENTATION	10
2.1 Introduction	10
2.2 Functional Requirements	10
2.3 Non-Functional Requirements	12
2.3.1 Performance	12
2.3.2 Reliability	12
2.3.3 Portability	12
2.4 Object-oriented System design using UML	14
2.4.1 Use Case Diagram	14
2.4.2 Case Description	15
2.4.3 Activity Diagram	25
2.4.4 Sequence Diagram	31
2.4.5 Class Diagram	34
2.4.6 ER Diagram	35
2.5 Coding: Appendix A	36
2.6 Summary	37
CHAPTER 3 SOFTWARE TESTING	38
3.1 Introduction	38
3.2 Testing Features	39
3.2.1 Feature to Be Tested	39
3.3 Testing Strategies	39

3.3.1 Test Approach	39
3.3.2 Pass/Fail Criteria	40
3.4 System Testing (Test Cases with Report)	40
3.5 Summary	41
CHAPTER 4 DEPLOYMENT AND MAINTENANCE	42
4.1 Introduction	42
4.2 Try to follow the SRLC (software release life cycle)	42
CHAPTER 5 USER MANUAL	44
5.1 Introduction	44
5.2 Project Functionalities	44
5.3 Summary	50
CHAPTER 6 PROJECT SUMMARY	51
6.1 Introduction	51
6.2 Project Limitation	51
6.3 Scope	51
6.4 Future Work	51
6.5 Conclusion	51
REFERENCES	53

LIST OF TABLES

Table 1: User Profile for Guest

Table 2: User Profile for Manager

Table 3: User Profile for Admin

LIST OF FIGURES

Figure 1: System Block Diagram

Figure 2: Use case Diagram

Figure 3: Activity Diagram

Figure 4: Sequence Diagram

Figure 5: Class Diagram

Figure 6: ER Diagram

Project Functionalities Figures

Figure: Sign In

Figure: Admin login

Figure: Sign Up

Figure: Add Reservation

Figure: Homepage

Figure: AI Chat bot

Figure: Admin Dashboard

Figure: Add New Room

CHAPTER 1 INTRODUCTION

1.1 Background

Online reservation are taking the place of the old methods in this hotel booking process, such as phone calls and in-person hotel reservations. Luxe Hotel was created for practical and effective booking systems.

Luxe Hotel is a web-based application which has its unique design, makes it easy for users to search, view, and reserve rooms. It also features an admin dashboard for reservation administration that is very user-friendly and effective.

1.1.1 Context and Relevance

The use of digital system in the hotel management industry is now a necessity in the modern world in terms of efficiency and customer satisfaction. The old manual systems of booking hotels may result in miscommunication, overbooking, and delay of services. In the era of fast web technologies development, the majority of hotels switch to online reservation, customer information and real-time-availability management systems. Finding a room, reserving a room and making a purchase are all available online through a hotel reservation system that enables a customer to go online and have all the convenient options and availability of the online hotel reservation system.

1.1.2 Problem Identification

Most of the small and medium hotels are still using manual booking records or old system without automation and synchronization. The outcome of this is frequent occurrences like the double booking, inaccurate availability of rooms, and lack of proper communication between the staff and the customers. In addition to that, customers have to be inconvenienced when attempting to make reservation by phone or using third party sites. Lack of unified and easy to use online system forms inefficiency and discontent to both the hotel administrators and guests.

1.1.3 Purpose and Justification

This project aims to plan and implement a Web-Based Hotel Reservation system that will automate the whole system through which hotel rooms are booked including room choice and confirmation of payment. The system will set out to make the operations of the hotel management more efficient and make the system user-friendly and interactive. The customers will be able to view room details easily, make bookings and confirmation immediately. The administrators are able to handle room inventory, check availability, and real-time reservation.

1.1.4 Scope

The proposed project is the development of a dynamic and friendly web based hotel reservation platform. It encompasses basic features like the registration of users, the authentication of a user during his/her login, search and booking of rooms, booking management and an administrative dashboard as the system to manage the hotel operations.

1.2 Project Planning and Initiation

This feasibility study establishes that the Luxe Hotel web-based hotel reservation system is viable both in technical, financial and operational terms. It offers a solid basis upon which the implementation is capable of being effective, and the intention of the project which is to improve customer experience with the help of digitalization.

Phase 1 Preliminary Analysis & Project Scope Definition:

The first stage of the project was dedicated to the establishment of the necessity of a digital hotel reservation system and the scope of the project. It was noted during the analysis that the conventional booking systems are time-intensive and are subject to human error.

Phase 2 Market Feasibility Analysis:

The hospitality sector is quickly shifting to digital management system as there is a growing population of online clients who like making a reservation via websites or mobile phone applications. According to research and study performed in the market, online booking platforms enhance customer satisfaction and operational efficiency.

Phase 3 Technical Feasibility Analysis:

The technical approach of the system is possible with the use of modern web development technologies. It is built using Typescript, Next JS, React JS, Tailwind CSS, Node JS and PostgreSQL to manage the database. The local server will be based on the MS Visual Studio code. The project does not need any special hardware, it can be installed in any machine with moderate specifications of the system.

Phase 4 Financial Feasibility Analysis:

As the system is created as an academic undertaking, no significant financial expenses are part of it except personal computer equipment and internet access. Visual studio code,

Postgres, and all the software required are free and open-source. The economic benefit would be the saving of the operation cost incurred by the hotel by the automation of the manual booking processes.

1.3 Target User Profile and Tentative Elicitation Process

1.3.1 Target User

- User
- Manager
- Admin

1.3.2 User profile

Table 1: User Profile for Guest

User Class	Note on Characteristics
Type of user	Guest
Age range	18 to 60 years
Frequency of use	Occasional or frequent (depending on travel plans)
Mandatory	No
Computer experience	Basic to Intermediate
Education	General education level or above
goal	To Search, View, and Book hotel rooms easily
Language skills	Basic English Comprehension
Number of users	Unlimited (depends on hotel size and reach)
Training	None required
Others system use	Web browsers (Chrome, Edge, Firefox)
Way of working	Independent browsing and booking

Table 2: User Profile for Manager

User Class	Note on Characteristics
Type of User	Manager
Age Range	25 – 55 years
Frequency of Use	Daily or as required for updates
Mandatory	Yes
Computer Experience	Intermediate to advanced
Education	Graduate level or relevant experience in hotel management
Goal	To manage room availability, reservations, and customer data efficiently
Language Skills	English proficiency required
Number of Users	1–3 (depending on staff size)
Training	Basic system orientation/training session
Other System Use	Database and admin dashboard access
Way of Working	Office-based or remote web access

Table 3: User Profile for Admin

User Class	Note on Characteristics
Type of User	System Administrator
Age Range	25 – 50 years
Frequency of Use	Regular – used daily for monitoring and maintenance
Mandatory	Yes – required for system operation and security oversight
Computer Experience	Advanced – knowledge of database management and web systems
Education	Bachelor’s degree in Computer Science or Software Engineering preferred
Goal	To maintain system performance, security, and user account integrity
Language Skills	Professional English proficiency
Number of Users	Typically 1 (main admin) + 1 backup admin
Training	Technical training on server setup, database backup, and software updates
Other System Use	Web server (MS Visual Studio Code), database tools (PostgreSQL)
Way of Working	Remote or on-site access through secured network

1.3.3 Elicitation Process

A systematic process of elicitation was undertaken in order to obtain the right user requirements to the Luxe Hotel reservation system. This was aimed at having a clear understanding of the customer and administrative requirements and converting them into system requirements.

Methods Used:

1. **Interviews:** The interviews with prospective users (travelers and hotel staff) will be informal as they will be required to establish the necessary features of the system including booking, cancellations, and room availability management.
2. **Questionnaires:** Simple online questionnaires were given out to get feedback on desired web designs, color template and navigability.

3. Observation: Monitored the hotel booking procedures manually to determine possible areas of inefficiencies that can be automated.

4. Document Review: Examined the current hotel management systems and compared the functionality of the systems with the purpose of defining peculiarities of Luxe Hotel.

1.4 Project Block Diagram

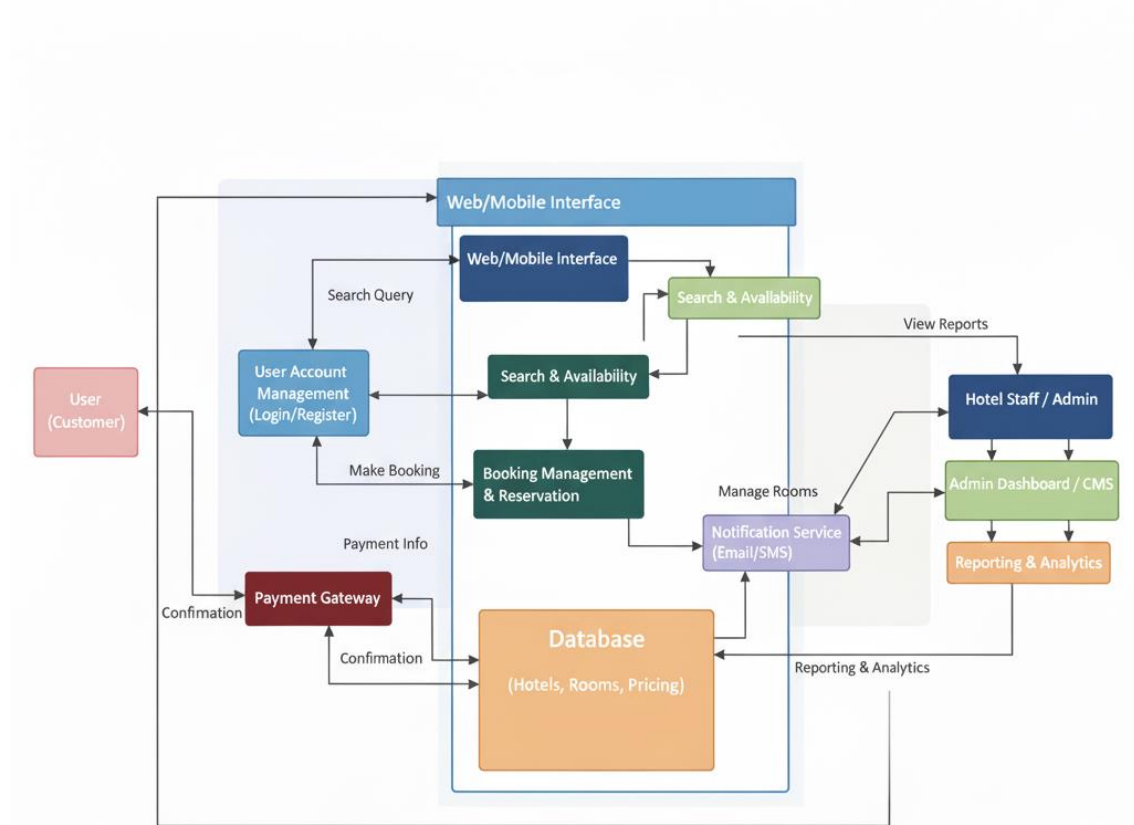


Figure 1: System Block Diagram

1.5 System Requirements

1.5.1 Hardware Requirements

Processor : 1.6 GHZ or any faster processors

Ram : 4 GB or more

Space Disk : 4 GB or more

1.5.2 Software Requirements

Operating System: Windows 10 or more

Front-End : Next JS, Tailwind CSS, Typescript

Framework : React JS

Backend : Node JS

Code Editor : MS Visual Studio Code

Database : PostgreSQL

Web Browser : Google Chrome

Web Server : Localhost

1.5.3 Constraints and Dependencies

Type	Description
Hardware Constraints	The overall performance depends on speed of the processor, memory, and storage capacity of the computer running in the live server. Low-end hardware might result slower response times while handling multiple requests.
Software Constraints	The system requires a local server environment such as MS Visual Studio code . Compatibility issues might arise if older versions are used.
Browser Dependency	The system is optimized for modern browsers like Google Chrome and Microsoft Edge . Display or functionality issues may occur in outdated browsers.
Network Constraints	Since the system runs locally, it cannot be accessed externally unless deployed to an online web server. Any network configuration issue may prevent users from accessing the application.
Security Limitations	Basic security measures like session validation and form validation are implemented. But full-scale protection (SSL) encryption, advanced authentication) is beyond the current project's academic scope.
Data Dependency	The system relies heavily on the PostgreSQL database. Any database connection failure, corruption, or improper configuration will disrupt the booking and data retrieval processes.

Type	Description
Scalability Limitation	The current version is designed for small to medium-sized hotels. For large-scale implementation, additional optimization and load balancing would be required.
Maintenance Dependency	System updates, backups, and bug fixes depend on administrator access and technical knowledge in database management.

1.6 Project Scheduling

Phase No.	Project Phase	Major Activities	Estimated Duration (Week)	Time Period (Week)
1	Project Planning and Analysis	Requirement gathering, feasibility study	2	W1 –W2
2	System Design	UML diagrams, database schema design, and user interface layout	2	W3 –W4
3	Implementation / Coding	Front-end and back-end development (Next JS, Tailwind CSS, TypeScript, React JS, Node JS)	3	W5 –W7
4	Testing and Debugging	Unit testing, system testing, and fixing of errors	2	W8 –W9
5	Deployment and Documentation	Local deployment through MS Visual Studio Code project documentation, and final report writing	1	W10
6	Presentation and Review	Preparing presentation slides, demonstration, and supervisor review	1	W11

1.7 Summary

This chapter presented an overview of the Luxe Hotel web-based reservation system project, including its background, purpose and the development philosophy. The project was presented as an online project designed to automate hotel room bookings and control by means of an easy to use web platform. The Background section talked about the context, relevance and intent of the project and how web-based automation can result in efficiency and customer satisfaction in the hospitality industry. The Project Planning and Initiation section discussed the stages of the feasibility study: technical, market, financial,

and preliminary analysis of the project to verify the feasibility of the project. Then it was followed by Target User Profile and Elicitation Process that targeted the main stakeholders such as customers, administrators and system managers and described how user requirements were obtained by using interviews, surveys and observation. The section on System Requirements outlined the hardware, software, and system requirements to be used in ensuring smooth operations. Lastly, the Project Scheduling section showed the development schedule and risk mitigation plan, which will take into account all the stages of the project, starting with planning and ending with the deployment in a well-organized and effective way.

CHAPTER 2 DESIGN AND IMPLEMENTATION

2.1 Introduction

In this chapter the detailed design and implementation of the Luxe Hotel web-based Reservation system has been given. It dwells upon the issue of the translation of the system requirements, both non-functional and functional, into a working software solution. The chapter delineates the architectural design, UML diagrams and coding structure that come together to be the foundation of the system.

2.2 Functional Requirements

The Luxe Hotel web-based reservation system is set to standardize its hotel booking system and make sure that it manages customer and room data efficiently. Functional requirements establish what the system needs to do and what it does not need to do in order to meet the needs of the users.

FR01	View Homepage
Description	The system displays the homepage where users can view general information about the hotel, available room types, facilities, and navigation options to other pages.
Stakeholder	Manager, User (Guest), Admin

FR02	Login
Description	Admin must log in using valid credentials to access their dashboard. Incorrect credentials will display an error message.
Stakeholder	Admin

FR03	Search Room
Description	User can search available rooms based on date, room type, and number of guests. The system displays filtered results dynamically.
Stakeholder	Manager, User (Guest), Admin

FR04	Submit Reservation
Description	User can select a desired room, provide necessary information, and submit a booking request. The system saves the reservation in the database and keeps the room reserved.
Stakeholder	Admin, Manager, User (Guest)

FR05	View Confirmation
Description	After submitting a booking, the reservation confirmation page showing booking ID, room details, check-in/out dates, and payment status can be viewed.
Stakeholder	Admin, Manager

FR06	Manage Rooms(Add/Edit/Delete)
Description	Admin can add new rooms, edit room details (type, price, description, availability), or delete outdated room listings.
Stakeholder	Admin

FR07	Manage Accounts
Description	The admin can view, edit, or deactivate user accounts as needed to maintain system security and accuracy.
Stakeholder	Admin

FR08	Confirm/Reject Reservations
Description	The admin reviews incoming booking requests and can confirm or reject them. Confirmation updates the room status, while rejection frees it for new reservations.
Stakeholder	Admin

FR09	Check-In/Check-Out Guests
Description	The admin can mark a guest as “Checked-In” when they arrive and “Checked-Out” upon departure. The system updates the room status accordingly.

Stakeholder	Admin
--------------------	-------

FR10	View Transaction
Description	The admin can view all completed transactions, including reservation details, total charges, and customer information for reporting purposes.
Stakeholder	Admin

2.3 Non-Functional Requirements

Non-functional requirements establish the quality, performance, and operating condition in which the Luxe Hotel web-based reservation system has to operate. These requirements are necessary to make sure that the system is not only working in a correct way but also offers the user experience being smooth, secure, and reliable.

2.3.1 Performance

The system should be responsive in a fast and efficient manner to the user. The loading time of the pages is a factor that should be less than 3 seconds, given normal working conditions. Even when the number of users and records grows, database queries and room searches should be able to give results quickly. System responsiveness is enhanced by caching and optimized SQL queries when the system is at its peak usage.

2.3.2 Reliability

This system should be working round the clock without any sudden crashes or loss of data. Before committing to the database, all the booking and registration transactions are authenticated as a way of guaranteeing data integrity. PostgreSQL is used to carry out backup and restore to ensure that information is not lost in case of failure.

2.3.3 Portability

The system is also platform-free, and it could be executed on any operating system provided it supports a local web server environment like the MS Visual Studio Code. It is also availed to use any contemporary web-based browser, which is essential because users and administrators can utilize it regardless of the device they are using.

2.3.4 Usability

The interface is user-friendly, easy and aesthetically similar. The platform is user-friendly and therefore, a user does not need to be trained to use it. Booking forms, buttons, and the navigation menu are designed in a logical manner to provide a smooth user experience. The layout is responsive in order to adjust to other screen dimensions.

2.3.5 Security

Authors are established to safeguard user accounts and booking information. The database contains the passwords of the users safely. The prevention of unauthorized access and SQL injection attacks is provided with input validation and server-side checks. Session management helps to secure the users by keeping them authenticated during their use.

2.3.6 Maintainability

The code base is modular in nature making the Codebase easy to debug and update systems. Separation of programming scripts and database connections is based on functionality and this makes them easier to maintain. Additional features in the future like the inclusion of online payments or reports automation can be added with minimum structural modifications.

2.3.7 Scalability

The system architecture supports expansion as the hotel industry grows. New room categories, additional features, or more users can be accommodated without redesigning the entire application. Database normalization ensures efficient data handling even with a larger dataset.

2.3.8 Availability

The system should be available for use 24/7 when deployed online. During the local development phase (via <http://localhost:8080/>), it remains available whenever the local Server and PostgreSQL servers are active. Future deployment on a hosted web server will allow global accessibility.

2.4 Object-oriented System design using UML

2.4.1 Use Case Diagram

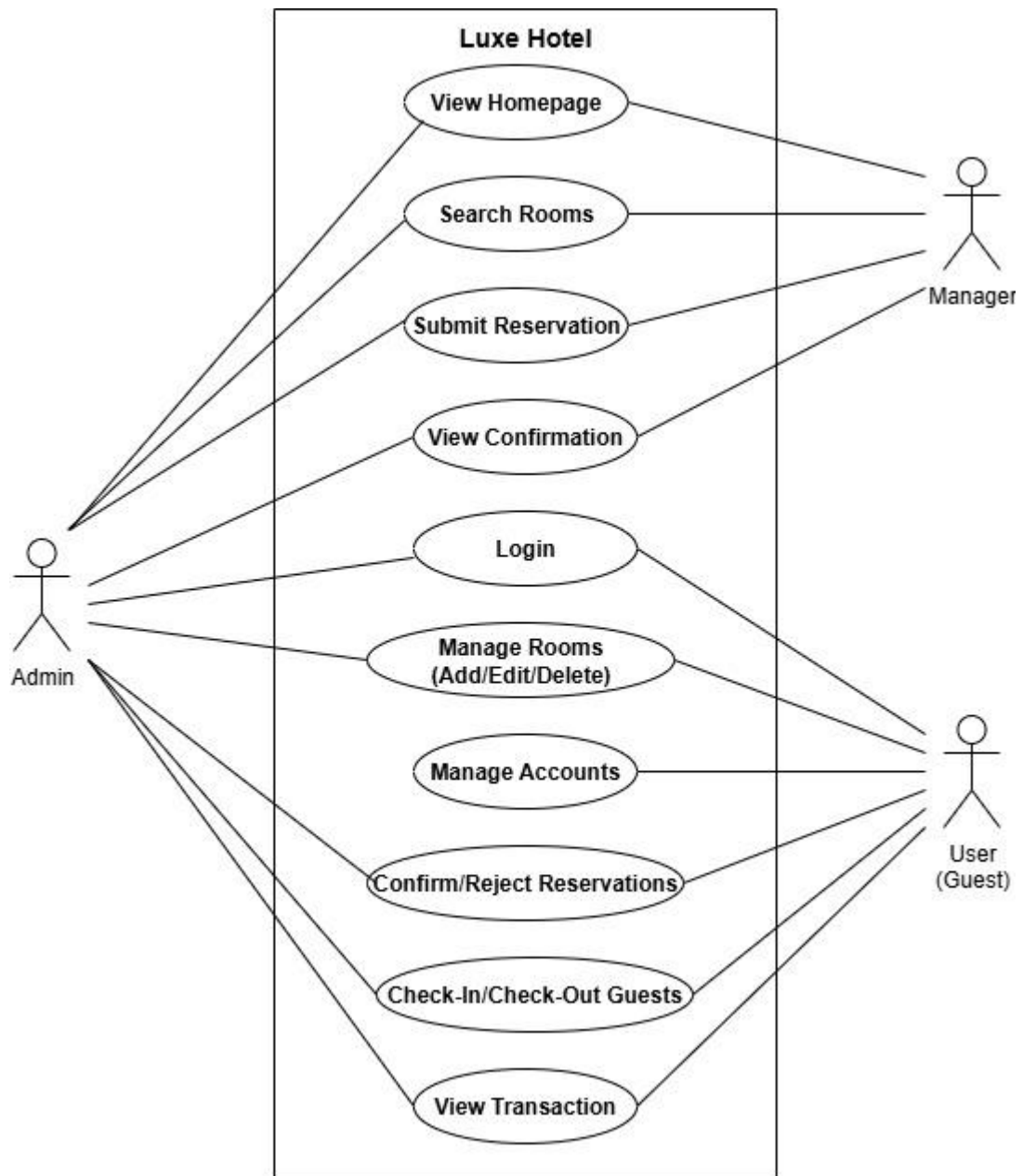


Figure 2: Use case Diagram

2.4.2 Case Description

Case Description-01: Login

Use Case	Login	
Goal	Admin logs into the system to perform management tasks.	
Preconditions	Admin has valid login credentials.	
Success End Condition	Admin is authenticated and redirected to the admin dashboard.	
Failed End Condition	Access denied due to invalid credentials.	
Primary Actors:	Admin	
Secondary Actors:	Authentication System	
Trigger	Admin submits login credentials.	
Description / Main Success Scenario	Step	Action
	1	Admin navigates to the login page.
	2	Admin enters username and password.
	3	System verifies credentials.
	4	Admin is granted access and redirected.
Alternative Flows	Step	Branching Action
	1	Invalid credentials
	2	System shows login error message.
Quality Requirements	Step	Requirement
	1	Login response must return in less than 2 seconds.

Case Description-02: Manage Rooms

Use Case	Manage Rooms	
Goal	Admin adds, edits, or deletes room information.	
Preconditions	Admin is logged in.	
Success End Condition	Room information is updated in the system.	
Failed End Condition	Changes not saved or database error occurs.	
Primary Actors:	Admin	
Secondary Actors:	Hotel Database	
Trigger	Admin selects "Manage Rooms" option.	
Description / Main Success Scenario	Step	Action
	1	Admin selects Manage Rooms from dashboard.
	2	System displays room management panel.
	3	Admin edits or creates room entries.
	4	System updates database.
Alternative Flows	Step	Branching Action
	1	Input error
	2	System prompts admin to correct input.
Quality Requirements	Step	Requirement
	1	Changes must reflect in the database within 2 seconds.

Case Description-03: Manage Accounts

Use Case	Manage Accounts	
Goal	Admin manages guest user accounts.	
Preconditions	Admin is logged in.	
Success End Condition	Account changes saved successfully.	
Failed End Condition	No changes made due to input or system error.	
Primary Actors:	Admin	
Secondary Actors:	User Account Database	
Trigger	Admin selects "Manage Accounts".	
Description / Main Success Scenario	Step	Action
	1	Admin navigates to the account management panel.
	2	Admin adds, edits, or deletes user accounts.
	3	System updates the account database.
Alternative Flows	Step	Branching Action
	1	Invalid account input
	2	System prompts admin to correct input.
Quality Requirements	Step	Requirement
	1	All account changes should be completed within 2 seconds.

Case Description-04: Confirm/Reject Reservations

Use Case	Confirm/Reject Reservations	
Goal	Admin confirms or rejects room reservation requests.	
Preconditions	Admin is logged in and reservations exist.	
Success End Condition	Reservation is either confirmed or rejected.	
Failed End Condition	No action taken on the reservation.	
Primary Actors:	Admin	
Secondary Actors:	Reservation System Database	
Trigger	Admin opens reservation list and selects a request.	
Description / Main Success Scenario	Step	Action
	1	Admin views pending reservations.
	2	Admin selects a reservation.
	3	Admin confirms or rejects the reservation.
	4	System updates the reservation status.
Alternative Flows	Step	Branching Action
	1	Admin does not act
	2	System retains reservation in pending state.
Quality Requirements	Step	Requirement
	1	Status update should occur in under 2 seconds.

Case Description-05: Check-In/Check-Out Guests

Use Case	Check-In/Check-Out Guests	
Goal	Admin records guest check-in and check-out.	
Preconditions	Reservation is confirmed.	
Success End Condition	Guest is checked in or out and record is updated.	
Failed End Condition	Check-in or check-out fails.	
Primary Actors: Secondary Actors:	Admin Reservation and Guest Record Database	
Trigger	Admin selects guest reservation on arrival or departure.	
Description / Main Success Scenario	Step	Action
	1	Admin accesses the reservation records.
	2	Admin selects check-in or check-out action.
	3	System updates the guest status accordingly.
Alternative Flows	Step	Branching Action
	1	Guest not found
	2	System shows error and prompts to re-check reservation ID.
Quality Requirements	Step	Requirement
	1	Status update must complete within 2 seconds.

Case Description-06: View Transaction

Use Case	View Transaction	
Goal	Admin views completed financial transactions.	
Preconditions	Transactions have occurred; admin is logged in.	
Success End Condition	Transaction records are displayed.	
Failed End Condition	System fails to display records.	
Primary Actors: Secondary Actors:	Admin Payment System Database	
Trigger	Admin navigates to the transaction report section.	
Description / Main Success Scenario	Step	Action
	1	Admin clicks on "View Transactions".
	2	System retrieves transaction data.
	3	System displays transaction history.
Alternative Flows	Step	Branching Action
	1	Database error
	2	System displays an error and retry option.
Quality Requirements	Step	Requirement
	1	Results should appear within 3 seconds.

Case Description-07: Search Rooms

Use Case	Search Rooms	
Goal	To allow users to search for available rooms.	
Preconditions	Homepage is successfully loaded.	
Success End Condition	List of available rooms is displayed.	
Failed End Condition	No results are shown or search fails.	
Primary Actors: Secondary Actors:	Admin, Manager, User (Guest) Database (for room availability)	
Trigger	User submits a room search query.	
Description / Main Success Scenario	Step	Action
	1	User enters search criteria (e.g., date, room type).
	2	System checks room availability.
	3	System displays list of matching rooms.
Alternative Flows	Step	Branching Action
	1	No rooms match criteria
	2	System notifies user of no availability.
Quality Requirements	Step	Requirement
	1	Results should be returned within 5 seconds.

Case Description-08: View Confirmation

Use Case	View Confirmation	
Goal	User wants to view their reservation confirmation.	
Preconditions	User has submitted a reservation.	
Success End Condition	Confirmation details are displayed.	
Failed End Condition	Confirmation cannot be retrieved.	
Primary Actors:	Admin, Manager	
Secondary Actors:	Reservation System Database	
Trigger	User requests to view confirmation after submission.	
Description / Main Success Scenario	Step	Action
	1	User clicks to view confirmation.
	2	System retrieves reservation details.
	3	System displays confirmation.
Alternative Flows	Step	Branching Action
	1	Reservation not found
	2	System displays an error and asks user to re-check.
Quality Requirements	Step	Requirement
	1	Confirmation must load within 3 seconds.

Case Description-09: Submit Reservation

Use Case	Submit Reservation	
Goal	The guest wants to submit a reservation for a selected hotel room.	
Preconditions	Guest has searched and selected an available room.	
Success End Condition	Reservation is successfully submitted and stored in the system.	
Failed End Condition	Reservation is not submitted due to errors or lack of availability	
Primary Actors:	Admin, Manager, User (Guest)	
Secondary Actors:	Hotel Reservation System (backend)	
Trigger	Guest clicks the "Submit Reservation" button after selecting a room.	
Description / Main Success Scenario	Step	Action
	1	Guest selects a room after browsing.
	2	System displays reservation form.
	3	Guest fills in details (dates, number of guests, etc.) and submits.
	4	System validates input and room availability.
	5	System stores reservation and displays confirmation.
Alternative Flows	Step	Branching Action
	1	Input is incomplete or invalid
	2	System prompts the guest to correct the input.
	3	Room is no longer available
	4	System notifies the user and suggests alternative rooms.
Quality Requirements	Step	Requirement
	1	Confirmation should be displayed within 5 seconds of successful submission.

Case Description-10: View Homepage

Use Case	View Homepage	
Goal	The user wants to view the hotel homepage for general information.	
Preconditions	None	
Success End Condition	Homepage is displayed to the user. .	
Failed End Condition	Homepage is displayed to the user.	
Primary Actors: Secondary Actors:	Admin, Manager, User (Guest) None	
Trigger	User navigates to the hotel reservation system URL.	
Description / Main Success Scenario	Step	Action
	1	User accesses the system through a browser.
	2	System loads and displays the homepage. .
Alternative Flows	Step	Branching Action
	1	Page fails to load
	2	Display error message and prompt to reload.
Quality Requirements	Step	Requirement
	1	Homepage should load within 3 seconds under normal network conditions.

2.4.3 Activity Diagram

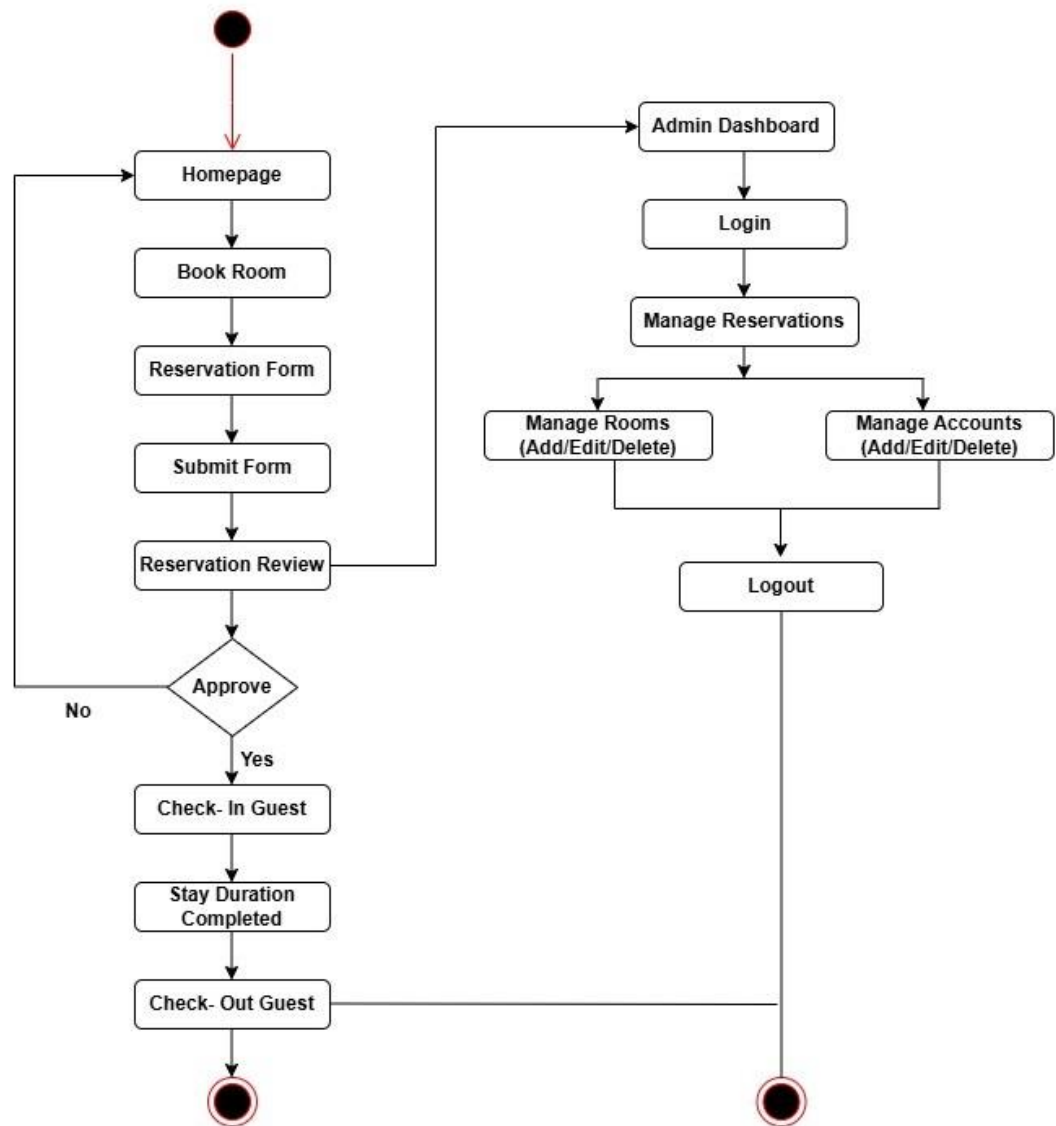
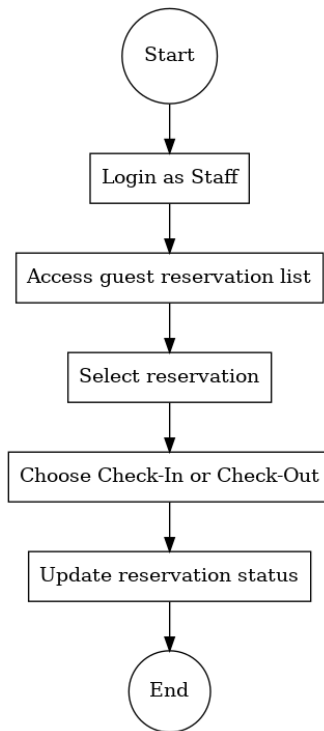
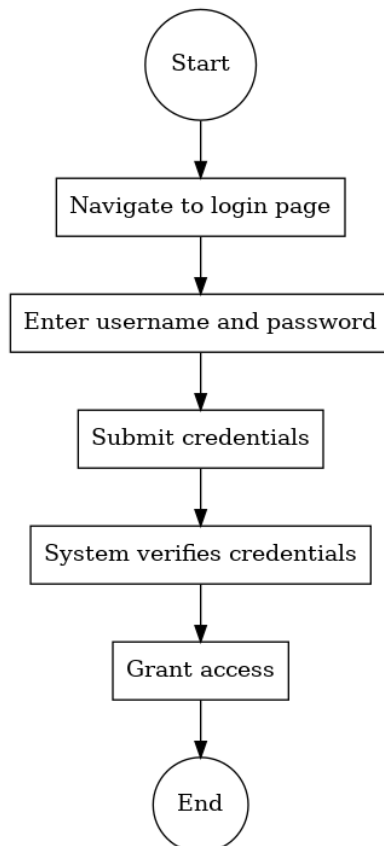


Figure 3: Activity Diagram

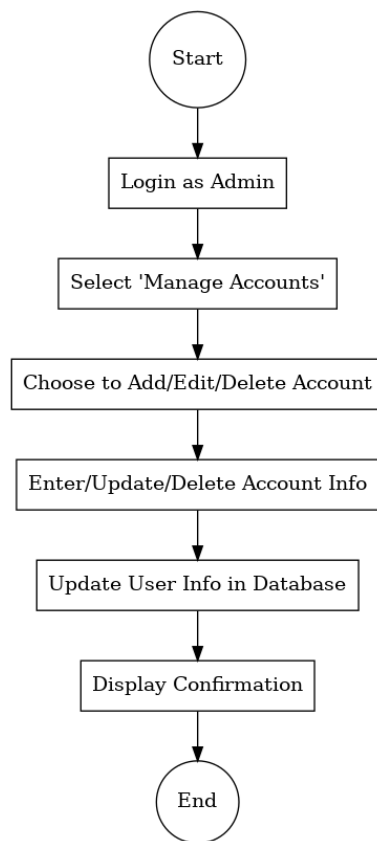
Activity Diagram 1: Check-In/Check-out Guests



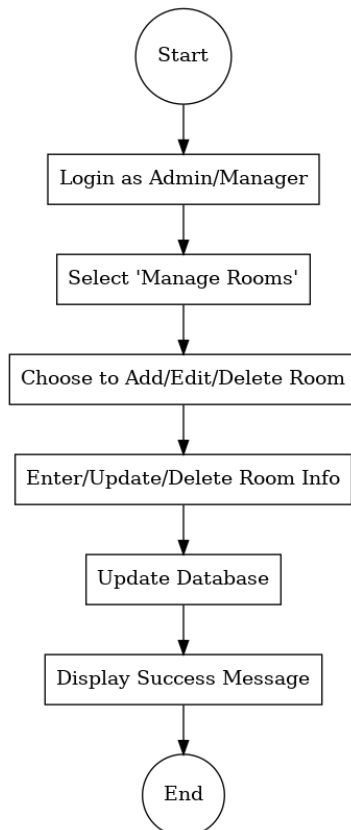
Activity Diagram 2: Login



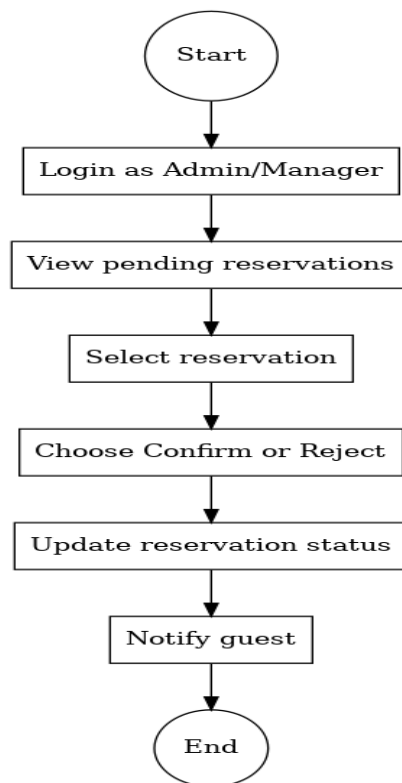
Activity Diagram 3: Manage Accounts



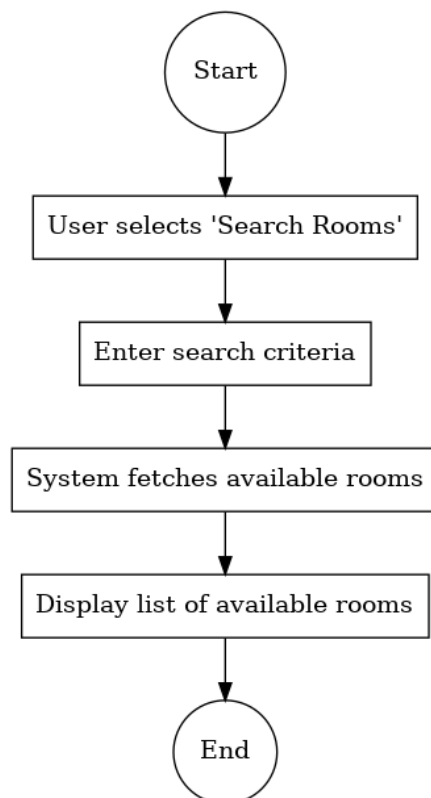
Activity Diagram 4: Manage Rooms



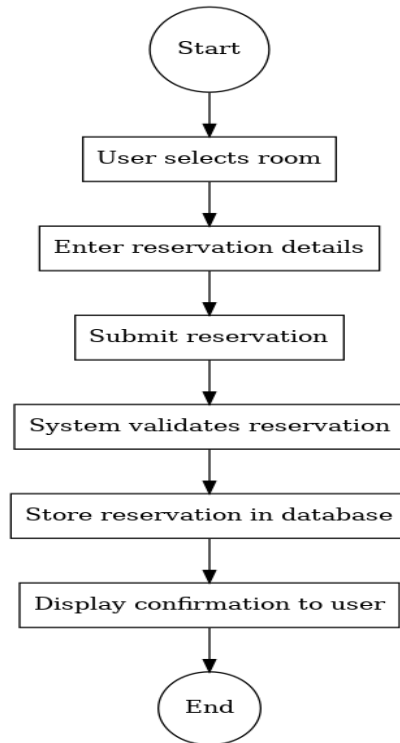
Activity Diagram 5: Confirm/Reject Reservations



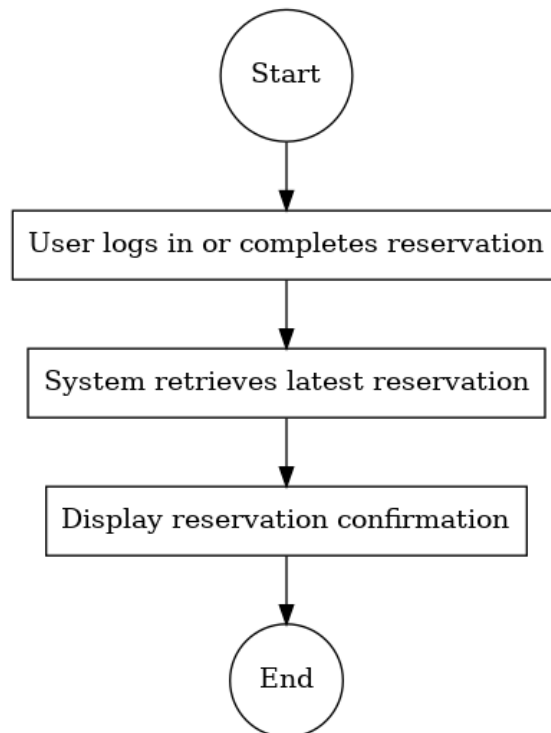
Activity Diagram 6: Search Rooms



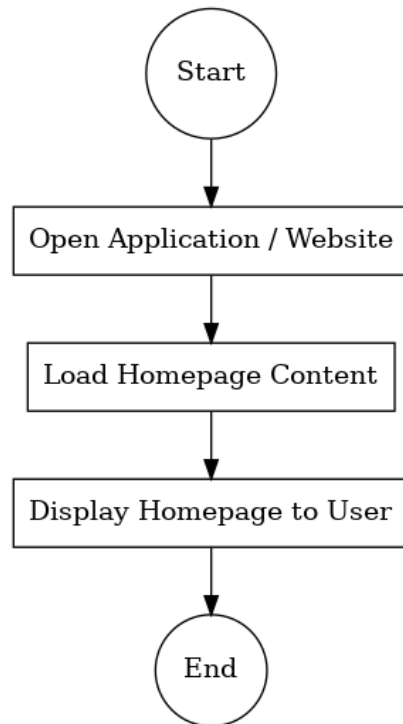
Activity Diagram 7: Submit Reservation



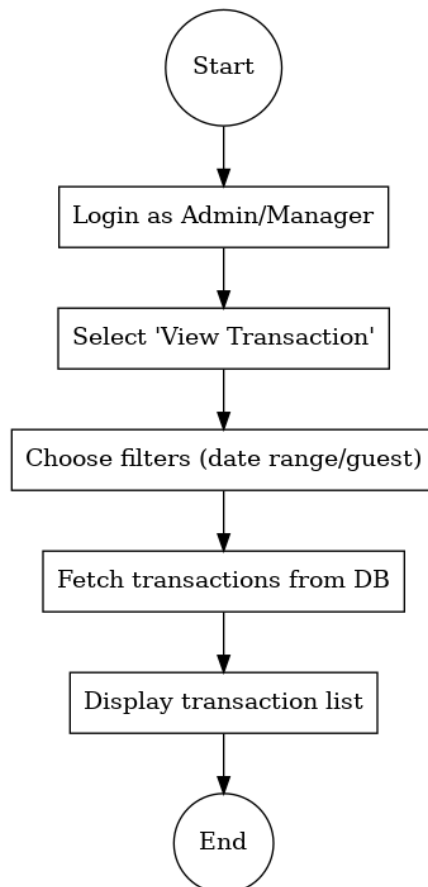
Activity Diagram 8: View Confirmation



Activity Diagram 9: View Homepage



Activity Diagram 10: View Transaction



2.4.4 Sequence Diagram

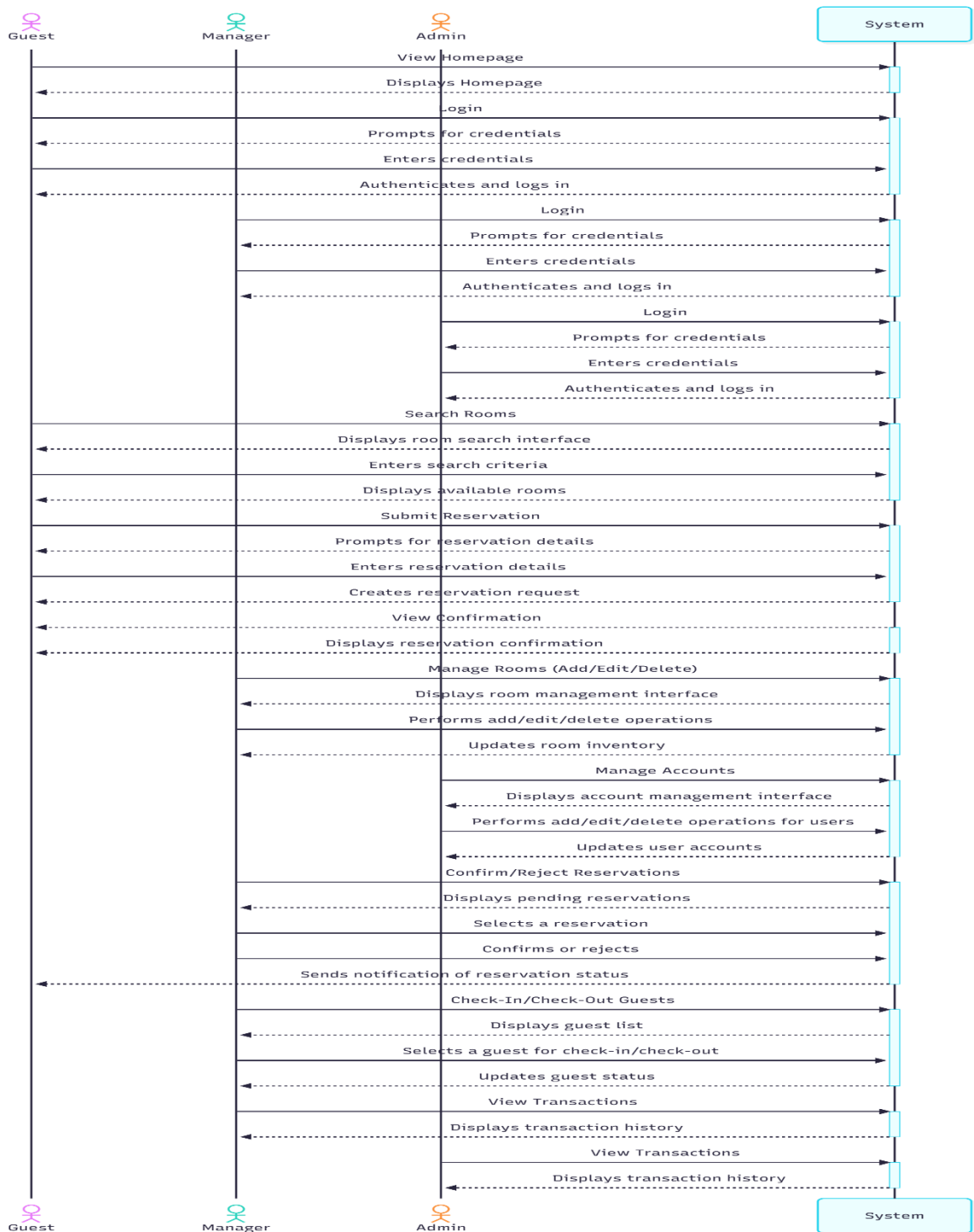
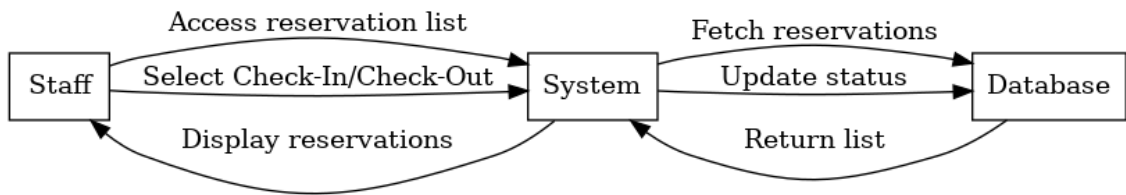
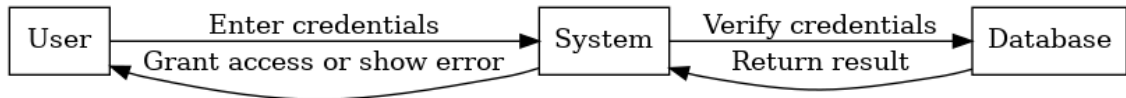


Figure 4: Sequence Diagram

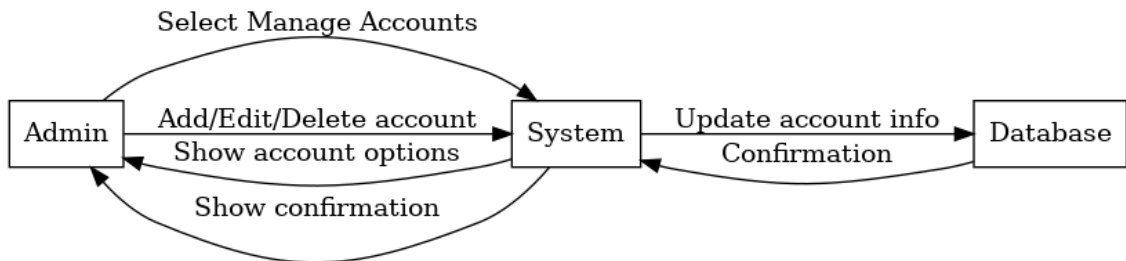
Sequence Diagram 1: Check-In/check-Out Guests



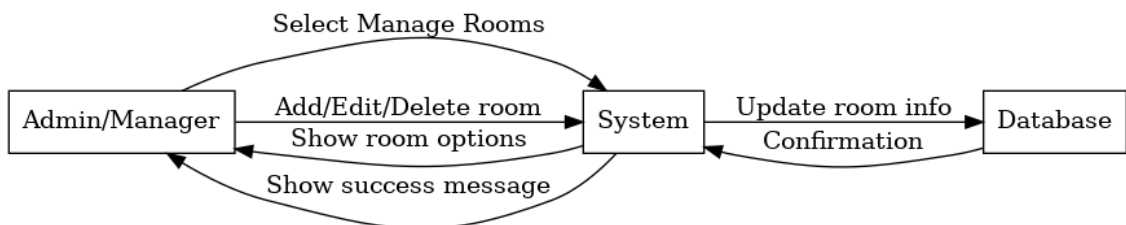
Sequence Diagram 2: Login



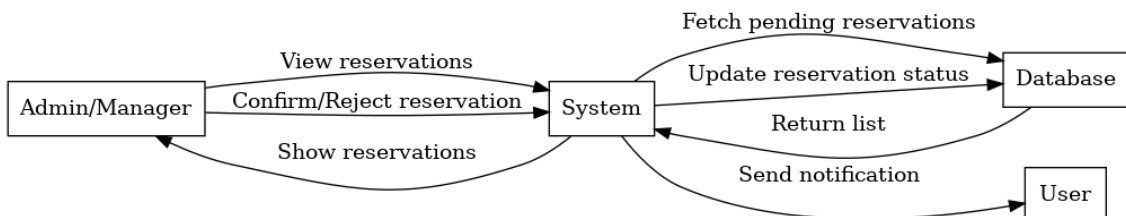
Sequence Diagram 3: Manage Accounts



Sequence Diagram 4: Manage Rooms



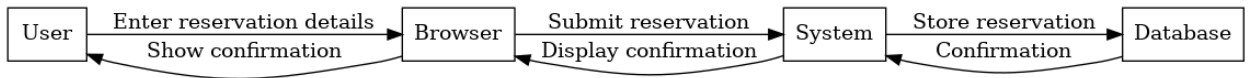
Sequence Diagram 5: Confirm/Reject Reservations



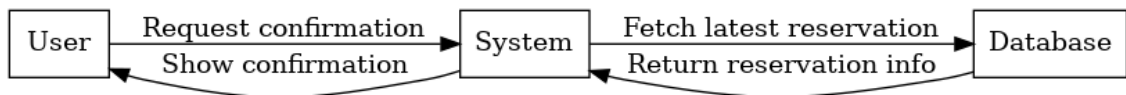
Sequence Diagram 6: Search Rooms



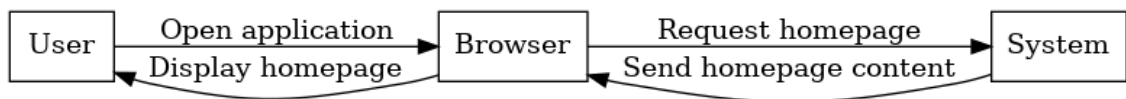
Sequence Diagram 7: Submit Reservation



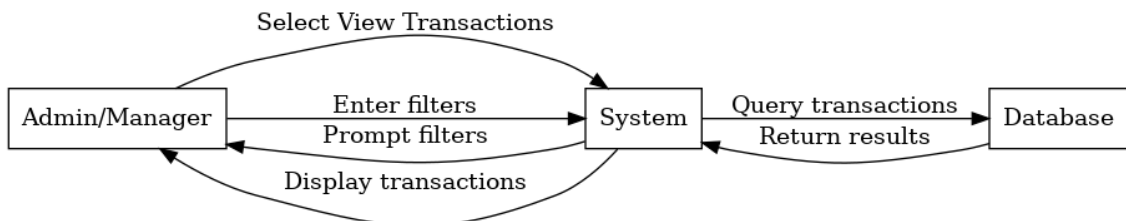
Sequence Diagram 8: View Confirmation



Sequence Diagram 9: View Homepage



Sequence Diagram 10: View Transaction



2.4.5 Class Diagram

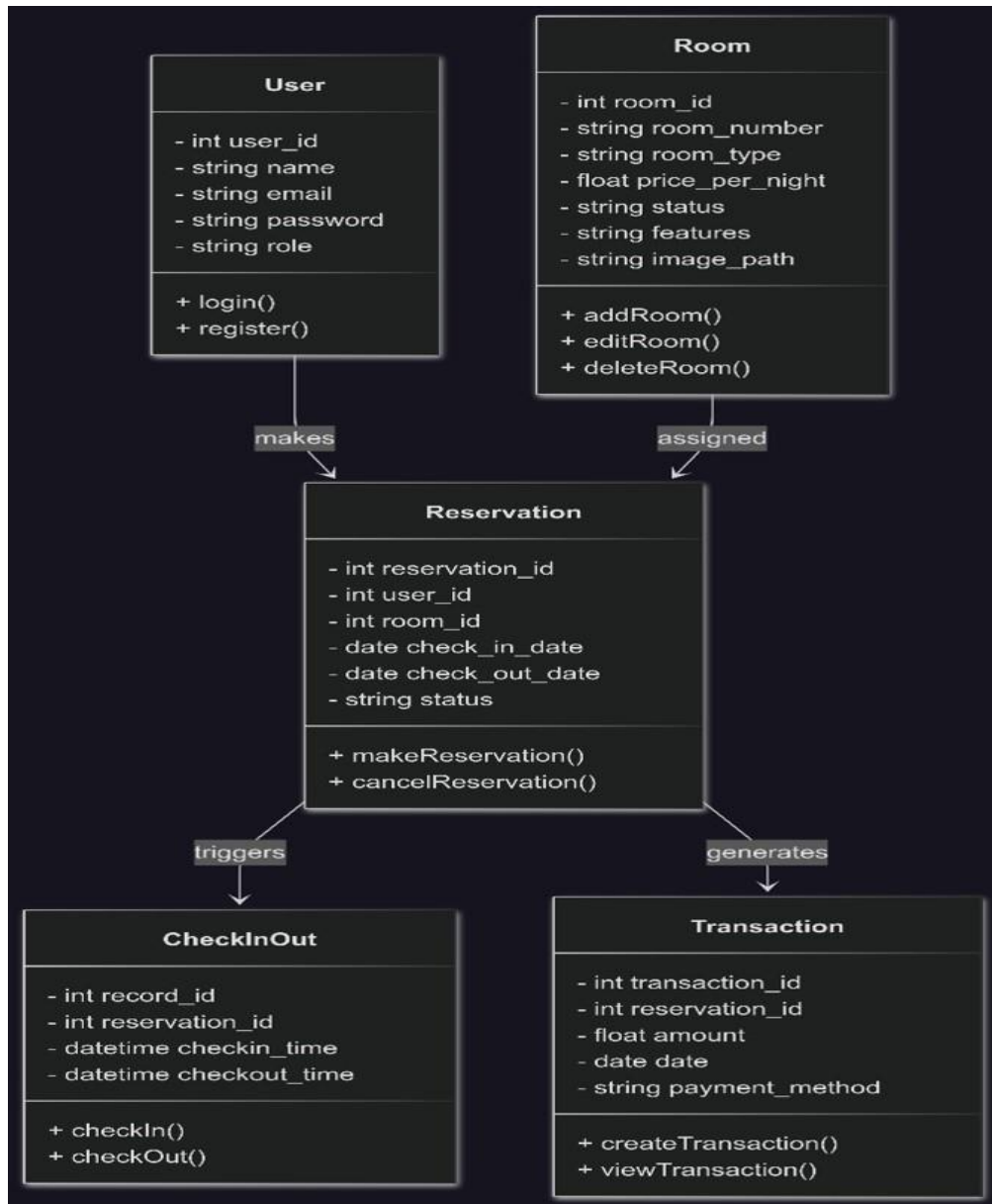


Figure 5: Class Diagram

2.4.6 ER Diagram

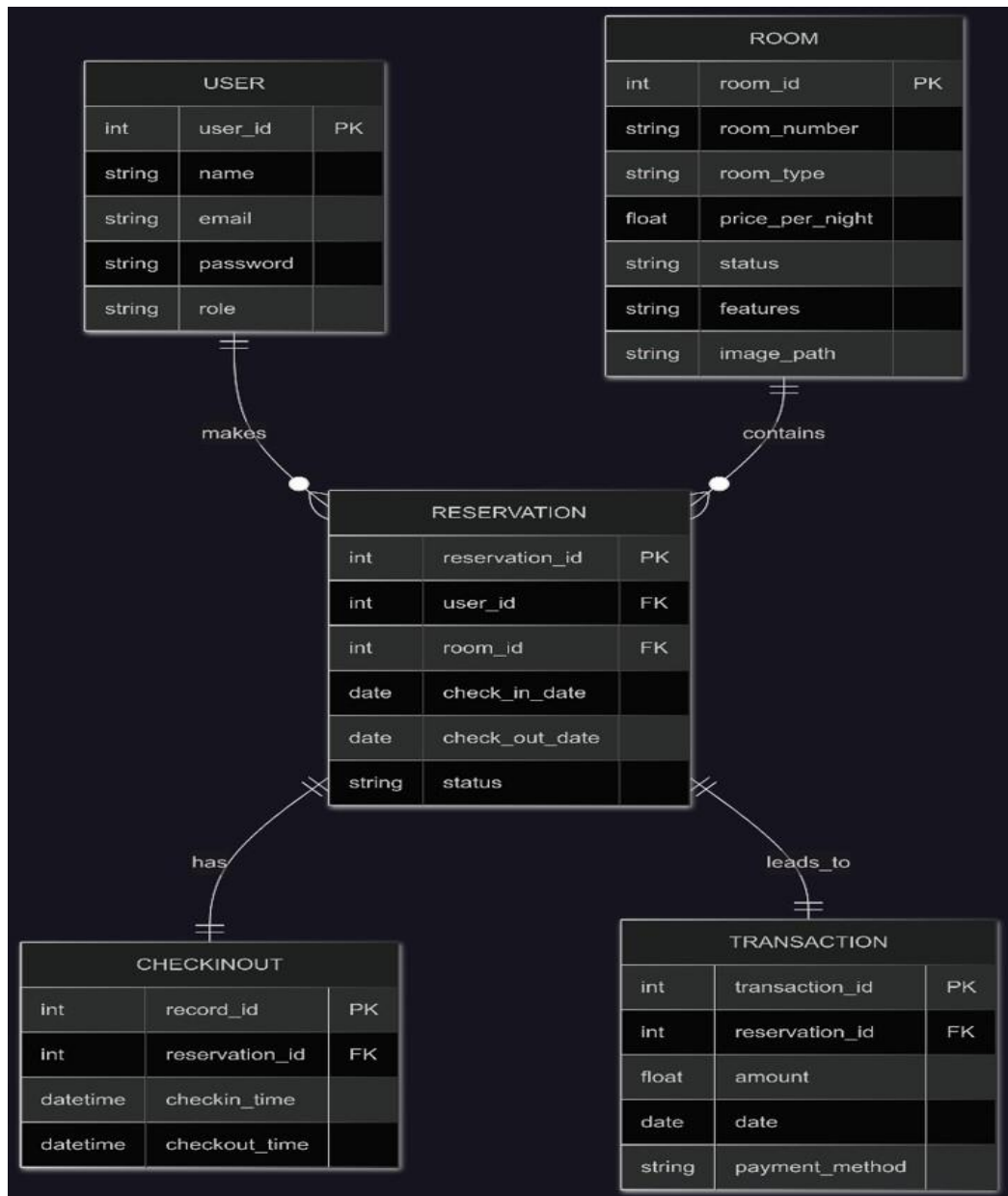


Figure 6: ER Diagram

2.5 Coding: Appendix A

```
Aconst loadRooms = async () => {
  try {
    setIsLoading(true);
    const data = await roomsApi.getAllRooms();
    const mappedRooms: Room[] = data.map((dbRoom: any) => ({
      id: dbRoom.id,
      name: dbRoom.name,
      description: dbRoom.description,
      price: Number(dbRoom.price),
      capacity: dbRoom.capacity,
      size: dbRoom.size,
      bedType: dbRoom.bed_type,
      amenities: dbRoom.amenities,
      image: dbRoom.image_url || "",
      available: dbRoom.available,
    }));
    setRooms(mappedRooms);
  } catch (error) {
    console.error('Error loading rooms:', error);
    toast({
      title: "Error",
      description: "Failed to load rooms. Please refresh the page.",
      variant: "destructive",
    });
  } finally {
    setIsLoading(false);
  }
};
```

2.6 Summary

This chapter outlined the design and development of Luxe Hotel web-based hotel reservation. It described the transformation of the functional and non-functional requirements of the system to a working solution through the concepts of object-oriented designs and UML modeling.

Different system components such as the **use case, activity, sequence, class, and ER diagrams** were developed to visualize interactions and data flow between users and the system. Next JS, PostgreSQL, React JS, Tailwind CSS, Node JS, and Type Script were the implementation tools that were used to build a dynamic, responsive and efficient web based application.

All in all, planning and execution were connected in this chapter as each design element was uniquely to the goal of the project provision of a reliable and user-friendly online hotel reservation system.

Chapter 3 Software Testing

3.1 Introduction

Luxe Hotel web-based reservation system is developed in a manner that facilitates hotel management and web bookings using a user-friendly web platform. The system enables visitors to easily reach the home page, search the page, read on the types of the rooms, the facilities and the hotel information. The login feature enables the guest and administrative users to access their respective dashboards in a secure manner. Guests are provided with the search options in terms of availability, date, room type and number of guests so that they could make the right decisions regarding making the booking. After locating the right room, the user has the opportunity to make a reservation with the help of submission of reservation information, and the reservation can be viewed with the confirmation of the booking ID, room details as well as the check-in/check-out dates. On the administrative level, the hotel managers and admins will be able to add, edit, or delete room listings when necessary. They are also able to control accounts and make sure that the user and staff profiles are current. Moreover, the system provides the admins with the power to accept or deny bookings, thereby making sure that only legitimate bookings are made. The administrators are able to carry out check-in and check-out business during guest stays that automatically transfer room status in the system. The view transaction is also an option that allows the admin to track all financial and booking transactions to make accurate reporting and business analysis. All in all, this system combines all these features into a single complete solution, which serves as a system to manage a hotel reservation system efficiently, automatized and user-friendly.

3.2 Testing Features

3.2.1 Feature to Be Tested

- a. User Registration
- b. User Login
- c. Search Rooms
- d. Submit Reservation
- e. View Confirmation
- f. Manage Rooms
- g. Manage Accounts
- h. Confirm/Reject Reservations
- i. Check-In / Check-Out
- j. View Transactions

3.3 Testing Strategies

3.3.1 Test Approach

The testing was done based on black-box testing approach whereby, the inputs and the desired output were used to test whether the system worked or failed without performing an internal code analysis.

The phases in which testing was carried out were as follows:

Unit Testing - Each of the modules (registration, login, booking) was tested separately.

Integration Testing - Modules as booking and database connection were tested to interact.

System Testing - The entire system was put through a simulated real-life scenario to test the functionality of all the functions.

User Acceptance Testing (UAT) - The system was put through an environment of testing by prospective users (students and supervisors) in order to test its usability and satisfaction.

The testing environment adopted:

- **Platform:** MS Visual Studio Code
- **Browser:** Google Chrome / Microsoft Edge
- **URL:** <http://localhost:8080/>

3.3.2 Pass/Fail Criteria

Criteria	Description
Pass	The system meets the expected outcome without any errors, and the operation completes successfully.
Fail	The system does not meet expected results, produces an error, or fails to perform the required operation.
Pending	The feature is under observation after initial modification.

3.4 System Testing

Test Case	Feature	Test Description	Expected Output	Actual Output	Result
1	Registration	Register new user with valid data	User account created successfully	As expected	Pass
2	Login	Login with valid credentials	Redirects to user dashboard	As expected	Pass
3	Search Rooms	Search available rooms by date	Displays available room list	As expected	Pass
4	Submit Reservation	Book a room with valid details	Booking saved and confirmation displayed	As expected	Pass
5	View Confirmation	Check booking status	Confirmation details shown correctly	As expected	Pass
6	Manage Rooms	Add new room in admin panel	Room added successfully	As expected	Pass
7	Manage Accounts	Update user account details	Account updated in database	As expected	Pass
8	Confirm/Reject Reservation	Approve booking request	Reservation confirmed and room status updated	As expected	Pass
9	Check-In/Out	Mark guest as checked-in	Status changed to "Checked-In"	As expected	Pass
10	View Transactions	View all completed transactions	Displays correct data summary	As expected	Pass

3.5 Summary

The Luxe Hotel web-based reservation system software testing ensured that all modules of the system are operating well and as per the expectations of the users. The system was well tested through manual and automated testing. All the key functionalities such as booking, logging, room management and viewing of transactions were done successfully without critical errors.

The validity of the testing was that the system is stable, reliable, and can be deployed on a local or online server.

Chapter 4 Deployment and Maintenance

4.1 Introduction

The chapter addresses the process of deploying and maintenance of Luxe Hotel Web-Based Reservation System. Deployment involves the transfer of the project to operational environment after development and maintenance is the assurance that the system remains to perform efficiently once it is implemented. In this project, the system was locally configured with the use of MS Visual Studio Code, which is the combination of Node JS and PostgreSQL servers, and thus it could be launched using the port of <http://localhost:8080/>. This arrangement gave a platform upon which the guest and the admin modules could be tested successfully. Upon confirmation, the system would be easily transferred to an online hosting platform to be accessed by everyone. Maintenance entails constant updates, database backups, bug fixing and upgrades on the basis of user feedback to ensure that the system is secure, efficient and user friendly.

4.2 Try to follow the SRLC (software release life cycle)

The Luxe Hotel web-based reservation system software release was done using a planned life cycle in order to deliver a stable and functional version. The SRLC phases are described below:

1. Alpha Version (Initial Development Phase)

In the first version, the key functionality included the user registration, the user log-in, room search, and reservation. Internal testing would be used to point out and correct major errors or performance problems.

2. Beta Version (Testing and Review Phase)

Once there was the initial stability, the system was offered to control testing by some of the chosen users including students and supervisors. The feedbacks received assisted in enhancing usability, interface layout and validation process.

3. Release Candidate (Pre-Deployment Phase)

An almost final version was developed, which included all the functional modules like the administration room control, reservation confirmation, and the check-in/check-out management. This version was tested on integration, reliability and performance.

4. Final Release (Deployment Phase)

The system was completely implemented locally by the use of the link <http://localhost:8080/> using the MS Visual Studio Code. The entire functionality was also successfully tested and the final version was deemed to be stable and ready to be used by both guests and administrators.

5. Maintenance Phase (Post-Deployment)

Periodic maintenance is done to maintain the stability of the system. It will involve bug fixing, upgrading Supabase/PostgreSQL when necessary, optimizing database, and data backup. Such features as online payment integration, the improvement of mobile responsiveness, and security can be implemented in the future.

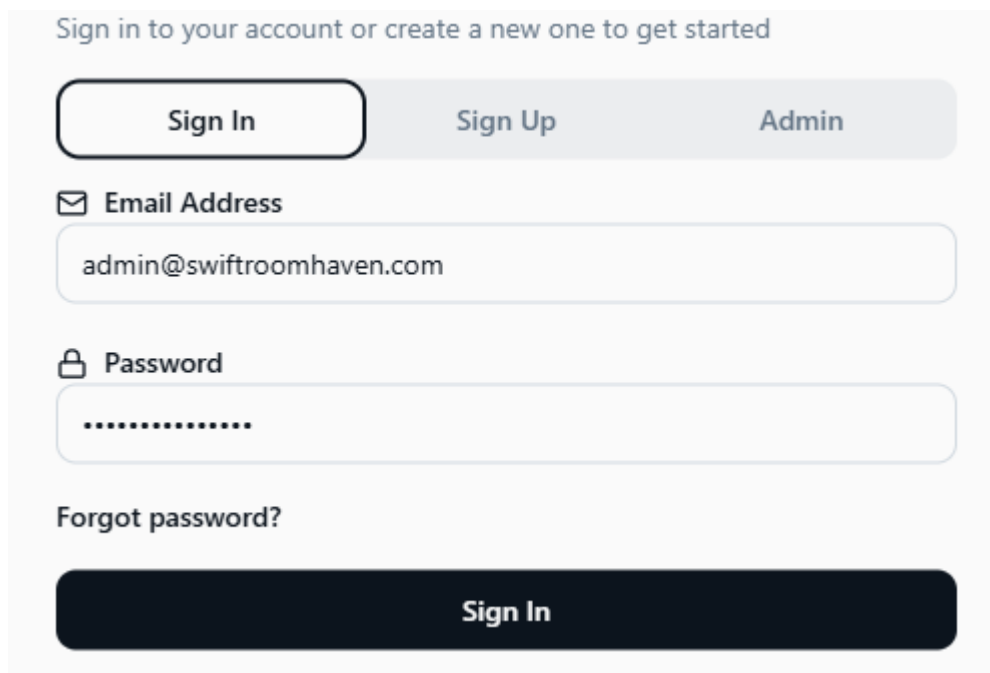
Chapter 5 User Manual

5.1 Introduction

The User Manual will contain step-by-step procedures on the use of the Luxe Hotel web-based reservation system. It serves to make both visitors and management familiar with the features of the system and learn how to navigate the interface. The user registration, login, room search, reservation submission and confirmation viewing are some of the functionalities that are explained in the manual. To administrators, it is an explanation of how to handle rooms, user accounts, reservations, check-ins/check-outs, and records of the transactions. Every part of the manual shows the intended purpose of the feature, the steps that one has to follow to achieve the task, and the anticipated system responses. This manual aims at making sure that every user can use the system effectively even without prior technical knowledge.

5.2 Project Functionalities

Sign In:



The screenshot shows a sign-in form with the following elements:

- Header: "Sign in to your account or create a new one to get started"
- Buttons: "Sign In" (active), "Sign Up", and "Admin"
- Form Fields:
 - Email Address: "admin@swiftroomhaven.com"
 - Password: "....."
- Link: "Forgot password?"
- Submit Button: "Sign In"

Figure: Sign In

Admin login:

Sign in to your account or create a new one to get started

[Sign In](#) [Sign Up](#) [Admin](#)

Admin Access Only
Requires authorized admin credentials

✉ Admin Email

🔒 Admin Password

[Sign In as Admin](#)

Figure: Admin login

Sign Up:

Sign in to your account or create a new one to get started

[Sign In](#) [Sign Up](#) [Admin](#)

[Create Account](#)

Figure: Sign Up

Add Reservation:

Reserve Deluxe Room ×

Complete the form below to book your stay

Deluxe Room
450 sq ft • Up to 2 guests

\$399
per night

Check-in Date

Check-out Date

Number of Guests

Maximum capacity: 2 guests

Guest Information

[Update your information in your profile settings](#)

Figure: Add Reservation

Homepage:

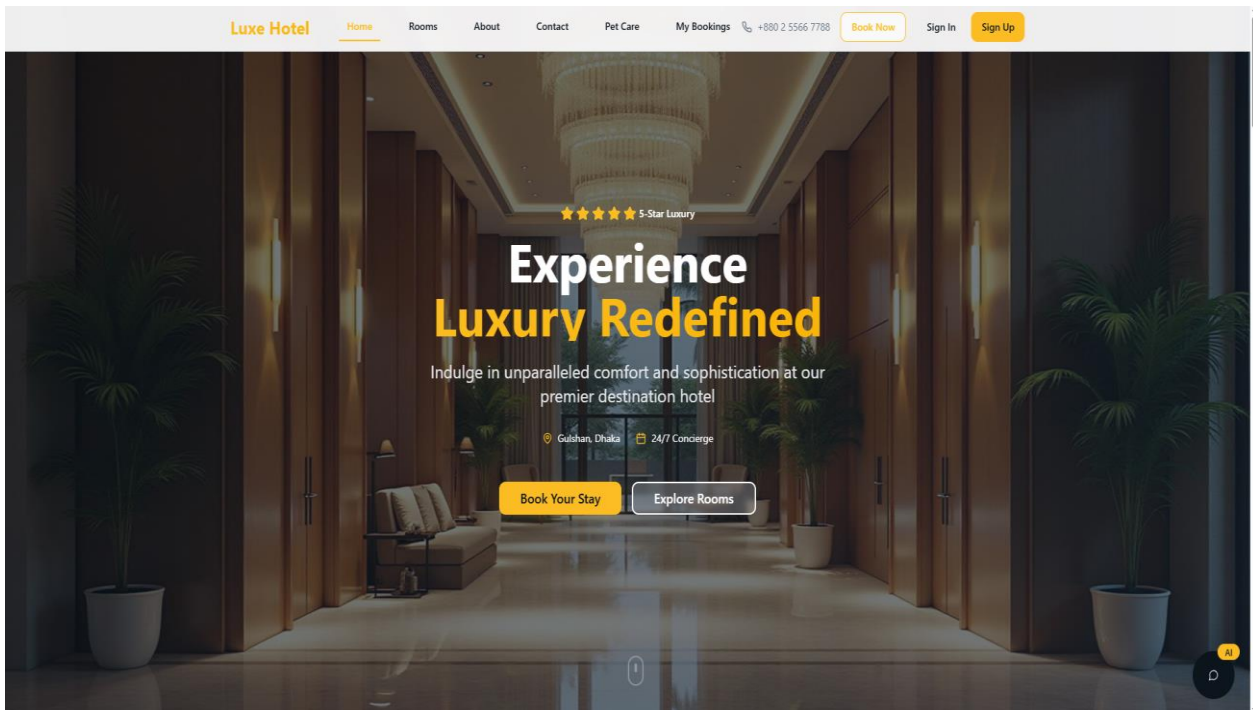


Figure: Homepage

AI Chat bot:

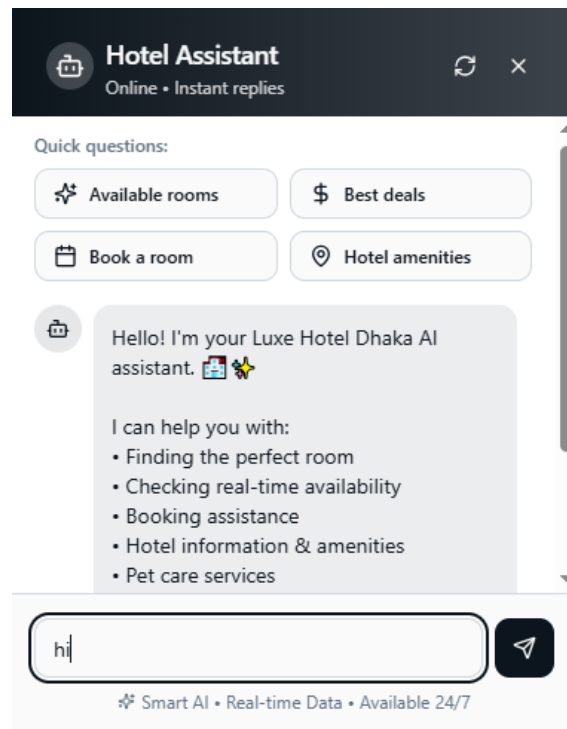


Figure: AI Chat bot

Admin Dashboard:

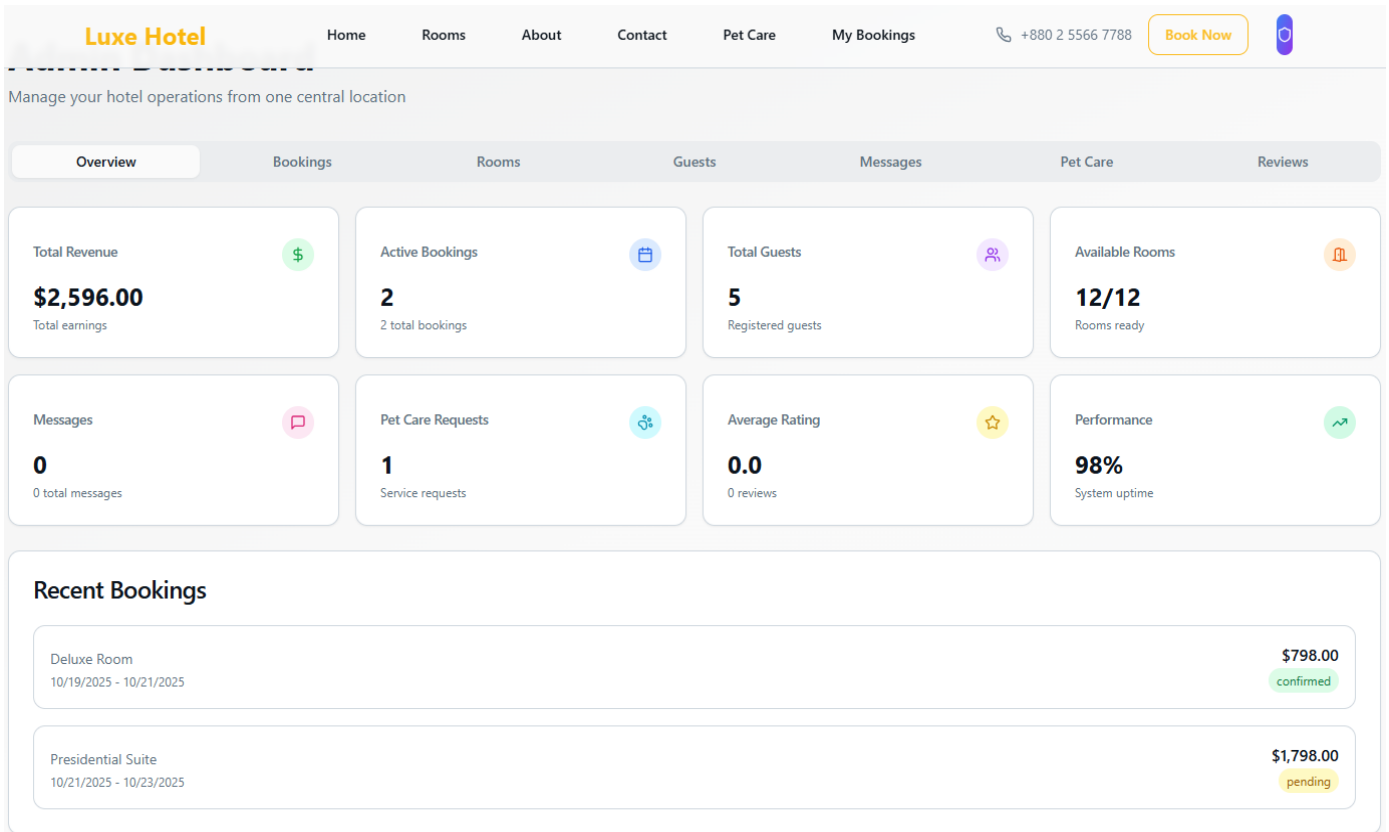


Figure: Admin Dashboard

Add New Room:

The 'Add New Room' form is used to create a new room listing. It includes the following fields and options:

- Room Name ***: Text input field.
- Price per Night ***: Text input field.
- Description ***: Text area for a detailed description.
- Capacity ***: Text input field.
- Size ***: Text input field (e.g., 300 sq ft).
- Bed Type ***: Text input field (e.g., King Bed).
- Amenities (comma-separated) ***: Text input field (e.g., Free Wi-Fi, Coffee Maker, Room Service).
- Image URL**: Text input field (e.g., /assets/room-image.jpg).
- Room is available for booking**: A toggle switch.
- Buttons**: 'Cancel' and 'Create Room' buttons.

Figure: Add New Room

5.3 Summary

This chapter had the most important functionalities of the Luxe Hotel web-based reservation system and displayed how the various modules are interrelated to give a complete hotel management solution. The Sign In and Sign Up will enable users to safely place their accounts as well as log in or out. General page Homepage gives a preview of the rooms to be stayed at, hotel information and links to other sections that can be easily accessed. The Reservation Now option allows the guest to book the rooms online instantly. On the administrative end, the Admin Login provides access to the Admin Dashboard on which hotel staff may monitor the work of the system, control the information about rooms, and process customer reservations. The Add New Room option helps administrators to update the room inventory easily. The AI Chat bot also helps users, as it responds instantly and helps users by giving guidance, which increases interaction and support. All these features collectively aim at the establishment of a smooth automated and easy customer experience as well as the administrator experience.

Chapter 6 Project Summary

6.1 Introduction

This chapter gives a summary of the Luxe Hotel web-based reservation system and highlights its main findings, drawbacks, and contributions. The project was aimed at automating hotel room booking, control of guests and administrative activities and enhancing efficiency in the hospitality industry. It comprises the necessary functions like user registration, room searching, online booking, administrative control panel, which are available after using a web-based interface at the address of localhost: <http://localhost:8080/>.

6.2 Project Limitation

Despite the fact that the system is effective in achieving its core objectives, there are some limitations associated with it. An external access is also restricted as the project is hosted on a local server (MS Visual Studio Code) and not on a live web domain. Payment illustration (solar card, mobile banking) is yet to be integrated, and automated email confirmation services as well as SMS confirmation services are not provided. Moreover, it is based on just a single entity of a hotel and does not offer sophisticated analytics modules. Such limitations are mainly caused by the scarcity of time, scope, and resources in the academic project context.

6.3 Scope

The Luxe Hotel web-based reservation system has a broad scope that involves the entire process of online hotel booking and management. It enables the guests to search rooms, book rooms and check booking confirmations and the administrators to log in to manage rooms, reservations, customer accounts and transactions. The project is based on functionality, simplicity, and user experience in the context of one-hotel. It lacks the multi-branch integration of the hotel, the online payment gateways, and the integration of the third-party API, which is deemed to be too broad of the current academic scope.

6.4 Future Work

The Luxe Hotel web-based reservation system can be greatly expanded in functionality in the future. Its planned upgrades are: Online Payment Integration (e.g., bKash, Nagad, Rocket). Booking Confirmations/Reminders Email/SMS. Android and iOS Mobile Application Version to enhance accessibility. Multi-Hotel or Chain Management System to be used on a bigger scale. Elaborated Analytics Dashboard of administration reporting and revenue monitoring. AI Chat bot Improvement of natural language processing to serve customers better.

6.5 Conclusion

The Luxe Hotel web based reservation system is effective in fulfilling its role of offering a digital system of hotel booking and management. It reduces human error through

automation of manual processes, and enhances efficiency of the services. Visitors find it very simple to explore and reserve rooms, and the administration effectively manages the business using a centralized board. The project illustrates the implementation of the web technologies (Next JS, PostgreSQL, React JS, Tailwind CSS, Node JS, Type script) and reflects the significance of the software engineering principles in the real-life context. In general, the system is a great milestone to modernizing the hotel management and can be considered as a good base of future improvements and research.

REFERENCE

1. <https://app.diagrams.net>
2. <https://www.w3schools.com>
3. <https://www.geeksforgeeks.org>
4. <https://www.lucidchart.com/pages/data-flow-diagram>
5. <https://github.com>
6. <https://www.tutorialspoint.com>

Plagiarism Report:

221-35-1057

ORIGINALITY REPORT

8%

SIMILARITY INDEX

3%

INTERNET SOURCES

1%

PUBLICATIONS

6%

STUDENT PAPERS


PRIMARY SOURCES

1	Submitted to Daffodil International University Student Paper	2%
2	dspace.daffodilvarsity.edu.bd:8080 Internet Source	1%
3	Submitted to UC, Irvine Student Paper	1%
4	www.coursehero.com Internet Source	1%
5	Submitted to Macquarie University Student Paper	1%
6	Submitted to Informatics Education Limited Student Paper	<1%
7	Submitted to Central Queensland University Student Paper	<1%
8	Submitted to University of Hertfordshire Student Paper	<1%
9	Submitted to Kuwait University Student Paper	<1%
10	Submitted to NCC Education Student Paper	<1%
11	Submitted to Brunel University Student Paper	<1%
12	Submitted to University of Greenwich Student Paper	<1%

13	learninglandscape.com Internet Source	<1 %
14	www.systoolsgroup.com Internet Source	<1 %
15	etd.aau.edu.et Internet Source	<1 %
16	V. Sharmila, S. Kannadhasan, A. Rajiv Kannan, P. Sivakumar, V. Vennila. "Challenges in Information, Communication and Computing Technology", CRC Press, 2024 Publication	<1 %
17	da Silva Linhares Vieira, Lia. "Development of a Web Application for Real-Time Inference in AI Models for Autonomous Driving.", Universidade do Porto (Portugal) Publication	<1 %

Exclude quotes Off Exclude matches Off
Exclude bibliography Off

Account Clearance



- Dashboard
- Student Profile
- Payment Ledger
- Registration/Exam Clearance
- Registered Course
- Result
- Routine
- Live Result
- Teaching Evaluation
- Scholarship
- Convocation Apply
- Certificate & Transcript
- Laptop
- Mentor Meeting
- Transport Card Apply
- Student Application
- Logout

MD. SHAKWAT MOZUMDER ALIF
221-35-1057

Dashboard

Student Portal

Total Payable

759,200.00

Total Paid

760,000.00

Total Due

-800.00

Total Other

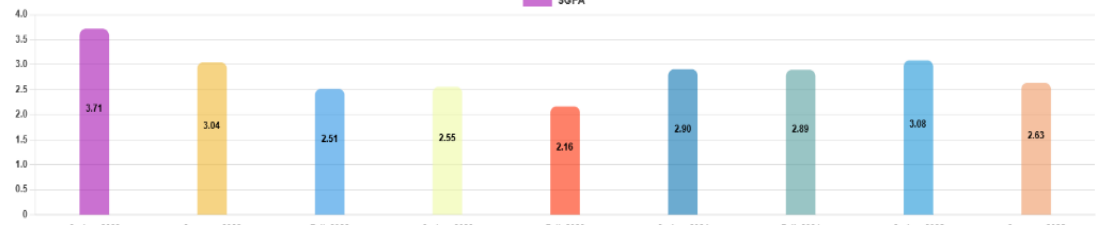
6,100.00

Today's Routine - Thursday


No routine available for today.

Semester Wise Result

Semester-wise SGPA Performance



Semester	SGPA
Spring, 2022	3.71
Summer, 2022	3.04
Fall, 2022	2.51
Spring, 2023	2.55
Fall, 2023	2.16
Spring, 2024	2.90
Fall, 2024	2.89
Spring, 2025	3.08
Summer, 2025	2.63



- Dashboard
- Student Profile
- Payment Ledger
- Registration/Exam Clearance
- Registered Course
- Result
- Routine
- Live Result
- Teaching Evaluation
- Scholarship
- Convocation Apply
- Certificate & Transcript
- Laptop
- Mentor Meeting
- Transport Card Apply
- Student Application
- Logout

MD. SHAKWAT MOZUMDER ALIF
221-35-1057

Registration/Exam Clearance

SL	Semester	Registration	Mid-Term Exam	Final Exam
1	Summer 2025 (252)	✓	✗	✓
2	Fall 2025 (253)	✓	✗	✓

2025 © - Developed by Daffodil International University