TRIBHUVAN UNIVERSITY

INSTITUTE OF SCIENCE AND TECHNOLOGY

SOFTWARE ENGINEERING (BIT 302)

A PROJECT REPORT ON

"Hotel Booking System"

Submitted By:

Sabu Bhuju [BIT 535/078]

Bidhisha K.C. [BIT 512/078]

Kripa Banskota [BIT 521/078]

Grishma Acharya [BIT 517/078]

Submitted To:

Ramesh Singh Saud

Padmakanya Multiple Campus

Bagbazar, Kathmandu

ACKNOWLEDGEMENT

We would like to express our deepest gratitude to our respected teacher, **Mr. Ramesh Singh Saud**, for their outstanding guidance and unwavering support in completing this project. Their vast knowledge, passion for the subject, and insightful discussions have significantly deepened our understanding, making this project an exceptional learning experience.

We also wish to extend our sincere appreciation to the college administration and faculty for providing us with the essential resources and opportunities needed to successfully undertake this project.

We are confident that the knowledge and skills we have gained through this project will be invaluable in our future endeavors, contributing to both our personal and professional growth

ABSTRACT

Many people around the world seek comfortable and convenient accommodation, whether it's for business travel or a leisurely getaway. The demand for a Hotel Booking System that efficiently manages hotel reservations and room information is high among hospitality providers. The Hotel Booking System is designed as a dynamic website specifically tailored for the hotel industry. This application enables hotels to manage various room types based on availability, pricing, and amenities. With this system, hotels can customize their offerings to meet diverse customer needs and preferences.

Additionally, the system includes a search module that allows administrators to easily locate, update, or enhance room listings. This module can also be extended to a customer-facing application, enabling guests to find and book the ideal room that fits their budget and preferences. The primary goal is to assist hotel providers in efficiently managing room bookings, ensuring a seamless experience for both business and leisure travelers. The system maintains a centralized repository to streamline booking processes and facilitate easy access to hotel information.

Contents

List of Figures

List of Tables

Abbreviations

CSS: Cascading Style Sheet

DFD: Data Flow Diagram

ER: Entity-Relationship (Diagram)

HTML: Hypertext Markup Language

HTTP: Hypertext Transfer Protocol

PHP: Hypertext Preprocessor

JS: JavaScript

SQL: Structured Query Language

UI: User Interface

Chapter 1: Introduction

1.1 Introduction

Online hotel booking system revolutionized the way people plan their trips. It offers convenience, a wide range of options, and the ability to compare prices and amenities easily. With just a few clicks or taps, travellers can find and book accommodations that suit their preferences and budget, making the entire process more efficient and enjoyable.

Online hotel booking refers to the process of reserving hotel accommodations through the internet. Instead of visiting a hotel in person or calling to make a reservation, travellers can use websites or mobile apps dedicated to hotel booking. These platforms typically provide a vast selection of hotels, ranging from budget to luxury, along with detailed information about each property, such as photos, amenities, room types, and customer reviews. Users can compare prices, read reviews, and make reservations instantly, often receiving confirmation emails or digital vouchers shortly after booking. This method offers convenience, flexibility, and transparency, empowering travellers to plan their trips with ease from anywhere with an internet connection.

The system will indeed help the hotel management and the esteemed staff members to manage and steer the hotel's functionality and transactions to realize its maximum potential in addition to its competence in the hotel business field. The objective of this project is to provide online room booking process for hotel management.

1.2 Problem Statement

The existing hotel reservation process heavily relies on a manual filing system, wherein guests can make room reservations either through travel agents or by directly contacting the hotel to inquire about availability and book their preferred rooms. However, this method is plagued with inefficiencies and inconsistencies. The manual nature of the system makes it time-consuming for both guests seeking reservations and staff managing bookings. Consequently, errors and discrepancies often occur in the final reservations, leading to potential customer dissatisfaction and operational challenges.

By automating this process through an online booking system, operations can be streamlined, accuracy can be improved, and guest satisfaction can be enhanced. The implementation of an online reservation system aims to address the issues faced by the current manual system. This system will allow guests to conveniently reserve their desired rooms through the internet, eliminating errors and inconsistencies associated with reservations.

1.3 Objectives

The basic objectives of the proposed project are:

1. To facilitate online reservations through the internet.

2. To implement automated methods for data entry.

3. To establish effective and dependable communication within the hotel.

4. To prevent data entry mistakes by utilizing input masks.

5. To allow authorized individuals to easily modify data.

6. To enforce security measures in order to prevent unauthorized access to guest records.

7. To enable quick and effortless retrieval of guest records and data for efficient reference purposes.

1.4 Scope and Limitations

1.4.1 Scope

Hotel booking system offer a comprehensive range of services to streamline and enhance staycation. Their scope typically includes:

User-Friendly Interfaces: The system will feature intuitive and easy-to-navigate interfaces, ensuring a seamless experience for users of all technical skill levels.

Instant Confirmation: Users will receive immediate confirmation of their bookings, reducing waiting times and enhancing the overall experience.

Extensive Listings: The system will provide a comprehensive list of services, ranging from budget to luxury options, to cater to diverse customer preferences.

User Reviews and Ratings: The system will feature user-generated reviews and ratings for hotel, helping potential customers make informed choices based on real experiences.

1.4.2 Limitations

While hotel booking system offer numerous benefits, they also face certain limitations.

Some of the limitations are:

Dependence on Internet Access: Users without reliable internet access may face difficulties in using the system, especially in remote areas where connectivity is poor.

Technical Glitches: Unexpected technical issues or server downtime can disrupt the booking process, leading to customer dissatisfaction.

Complexity for Non-Tech-Savvy Users: Individuals who are not comfortable with technology might find it challenging to navigate the system, leading to frustration and potential abandonment of the booking process.

1.5 Report Organization

This project report is divided into four chapters. The chapters are as follows:

Chapter 1: Introduction – This chapter contains a brief introduction about the project.

Chapter 2: Methodology and Requirements – This chapter contains specific approaches and criteria used to conduct the research or project

Chapter 3: System Design and Implementation – The part of the system along with the design process is mentioned in this chapter.

Chapter 5: Conclusion and Lesson Learned – The summary of the project along with the improvements and recommendations for future projects have been mentioned in this chapter.

Chapter 2: Methodology and Requirements Analysis

2.1 Methodology

The waterfall model is used to build the system. This model divides the entire software development process into distinct phases. Each phase must be finished before the next one can begin, with no overlap between phases. As a result, the output of one phase serves as the input for the next phase in a sequential manner.

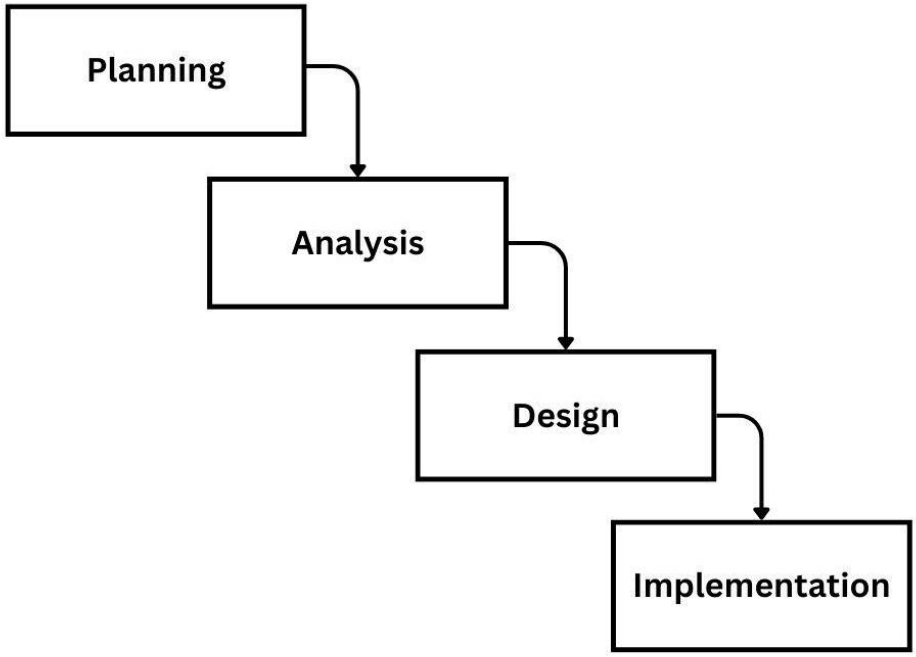


Figure 1: Waterfall Model for Software Development Process

We chose to develop the Hotel booking system using the waterfall model because of its simplicity. We were familiar with the project objectives and the planned course of action, so the waterfall model seemed to be the best method for implementation. The major steps we followed while using the waterfall model of development are as follows:

i. **Planning:** In this phase, we planned the specifics of our system by gathering insights from the surroundings and nearby people. We also collected facts to understand the necessity of this system.

ii. **Analysis:** We analyze past projects, as well as the current systems in use, and the challenges we have encountered. Additionally, we assessed the needs and

expectations of potential customers to determine the requirements for our project. Subsequently, we established the specific requirements, identified the target users, and selected the platforms for our system.

iii. **Design:** During the design phase, we created database schemas, interface designs, process models, and more. In short, we designed everything from how our system will look to how it will work. We also designed workflows and architectural designs for our system.

iv. **Implementation:** In the implementation phase, we proceeded with coding and testing. We put into action all the studies and designs we created in earlier development phases. Furthermore, we tested our system by developing various test cases for both unit and system testing.

2.2 Requirement Analysis

2.2.1 Functional Requirements for Admin

- **User Registration and Profile Management:**

  Users can create and manage their profiles, including personal information, contact details, and preferences.

- **Authentication and Authorization:**

  Secure login and access control mechanisms to ensure that users and administrators can access only their permitted functions.

- **Room Booking Management:**

  - Check Room Availability: Users can view available rooms based on their criteria such as date, room type, and amenities.
  - Book Rooms: Users can book available rooms by selecting their preferences and making a reservation.
  - View Booking History: Users can view their past and upcoming bookings in their profile.

- Change Booking: Users can modify their existing bookings, such as changing the dates or room type.
- Cancel Booking: Users can cancel their bookings if their plans change.

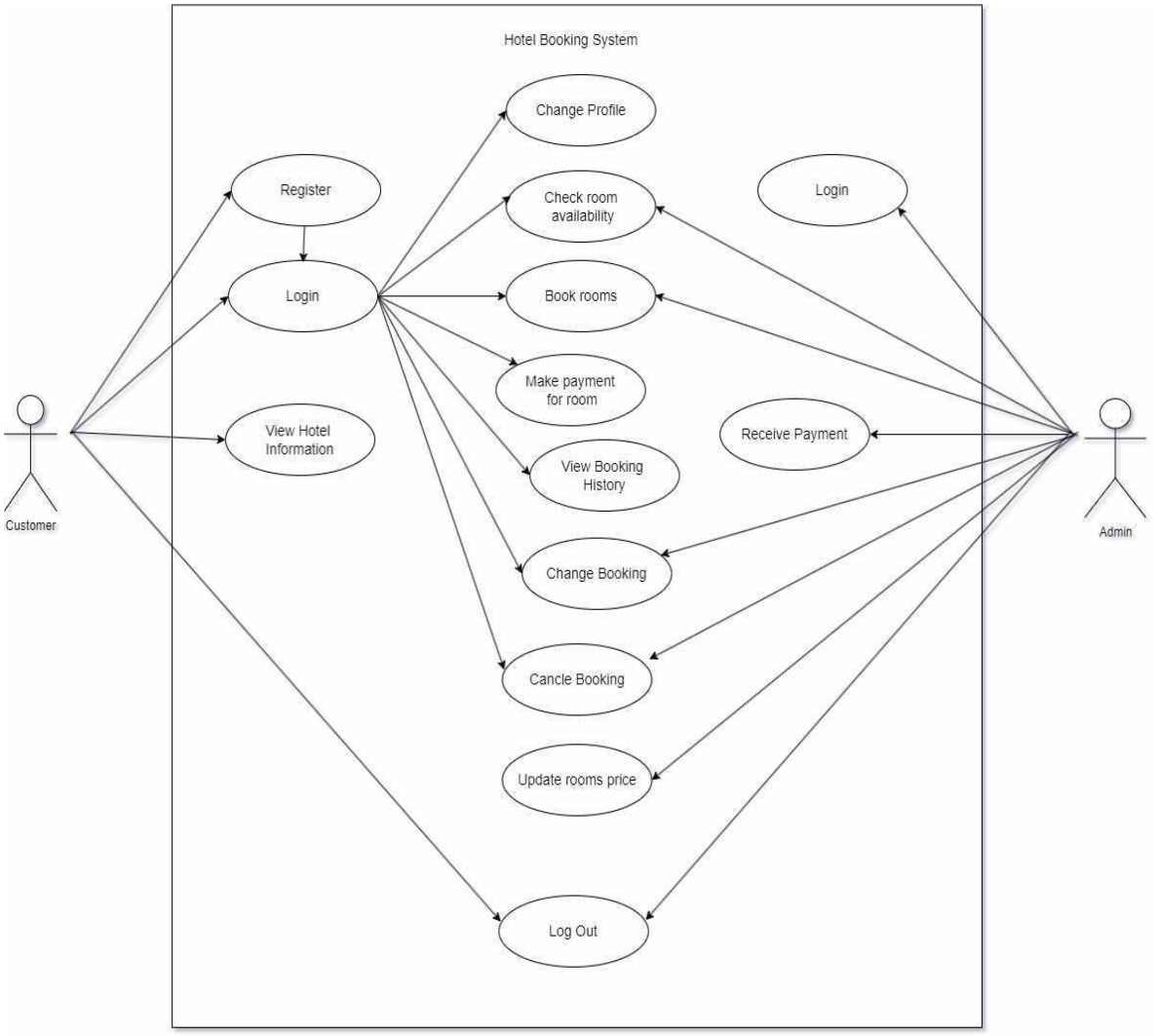
- **Payment Management:**

- Make Payment for Room: Users can make payments for their room bookings through various payment methods such as credit cards, debit cards, and online banking.
- Receive Payment: The system processes and confirms payments, updating the booking status accordingly.

- **Hotel Information Management:**

- View Hotel Information: Users can access detailed information about the hotel, including amenities, location, and policies.
- Admin Management: Administrators can manage hotel operations, including user roles, booking approvals, and system configurations.
- Update Room Prices: Administrators can update the pricing of rooms based on seasons, demand, and special promotions.
- Session Management: The system manages user sessions to ensure security and a seamless user experience, handling login states and timeouts.

Use-Case Diagram:



2.2.2 Non-functional Requirements

1. Performance: The system responds to user actions within short time and handle multiple booking transactions per second during peak hours.

2. Usability: These website has appropriate user interface and adequate information to guide the user in order to use the website.

3. Reliability: Ensure the system consistently performs its intended functions without failure.

4. Maintainability: These website is capable to secure the data and easily retrieve the data.

5. Scalability: These system can further modified in future

6. Portability: The website is portable as it is online website running across the internet.

7. Security: This website provide user and authentication so that only the legitimate user are allowed to use the website

2.3 Feasibility Study

1. Technical Feasibility:

This project results a very simple and user-friendly outcome. The technical feasibility in the proposed system deals with the technology used in the system. It deals with the hardware and software used in the system whether they are of latest technology or not. It happens that after a system is prepared a new technology arises and the user wants the system based on that technology. This system uses windows platform, HTML, CSS, PHP, MYSQL making our project Hotel Booking system technically feasible.

2. Economic Feasibility:

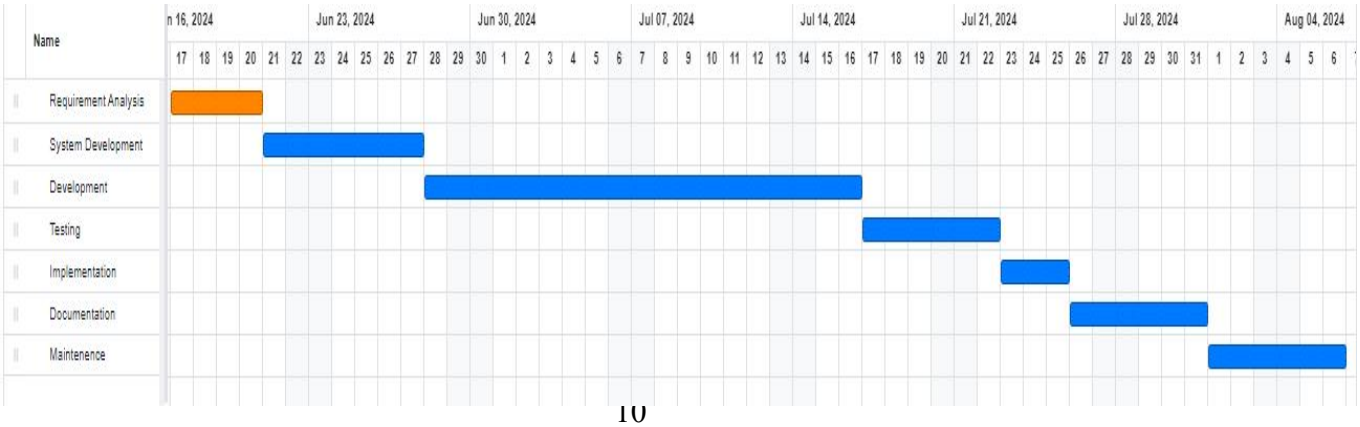
This project is economically feasible in the sense that the money which were invested in purchasing register to keep data during short periods are now to be invested in a computer which do not need to be changed every year. So, a onetime investment in computer reduces expenses of the company. And a project itself is feasible as every software used to make it are easily available in internet.

Schedule Feasibility:

The project was done for duration of four month from June-August 2024 from the stage of analysis; data collection and system development; for the purpose of collecting righteous information to enable us develop the system. The following Gantt chart provides an estimated timeline for the development of the hotel reservation system:

Gantt Chart:

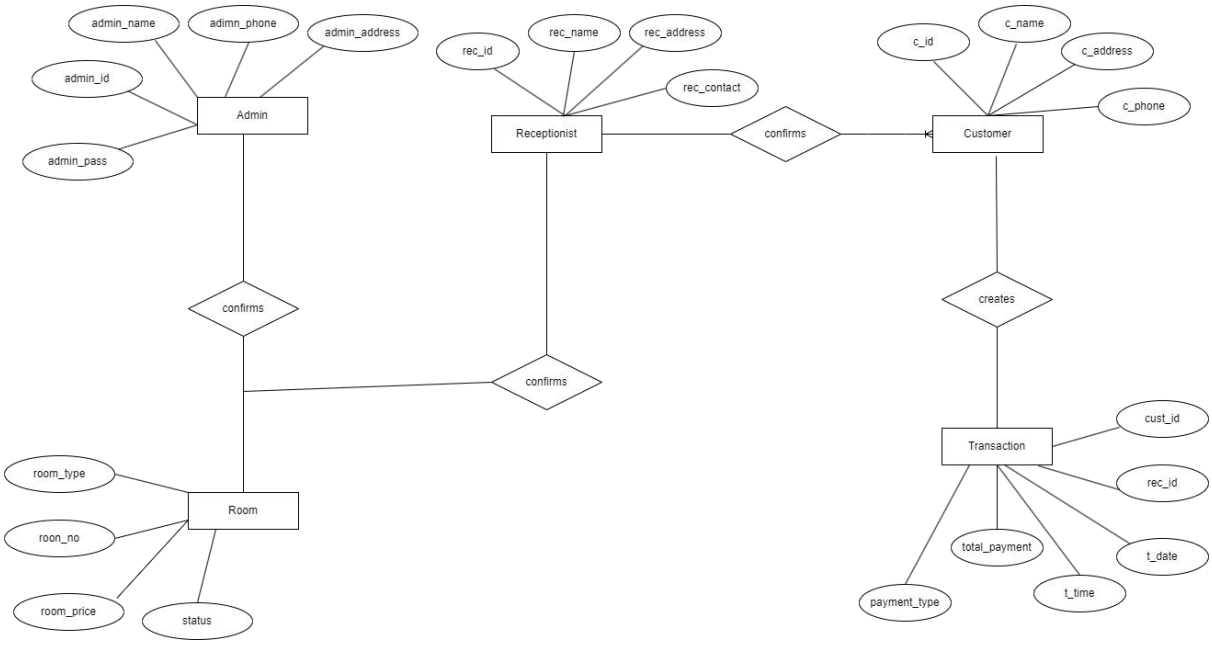
Task	Start Date	End Date
Requirement Analysis	06/20/2024	06/20/2024
System Design	06/21/2024	06/27/2024
Development	06/28/2024	07/16/2024
Testing	07/17/2024	07/22/2024
Implementation	07/23/2024	07/25/2024
Documentation	07/26/2024	07/31/2024
Maintenance	08/01/2024	08/06/2024



2.4 Data Modeling

2.4.1 ER Diagram

Data modeling is essential for illustrating how entities interact and relate within a database. One of the primary tools for this purpose is the Entity-Relationship (ER) Model, which serves as a high-level conceptual framework for designing databases. The ER Model is widely recognized and utilized in the conceptual design phase of database applications due to its ability to visually represent data relationships and structures. Many database design tools incorporate variations of the ER Model, making it a fundamental component in database design and planning.



2.5 Process Modelling

DFD (Data Flow Diagram)

A data flow diagram (DFD) is a visual tool employed to outline and examine how data moves through a system. It serves as a foundational element for system development, illustrating the flow and transformation of data from its initial input through various processing stages to its final output.

Context Level DFD

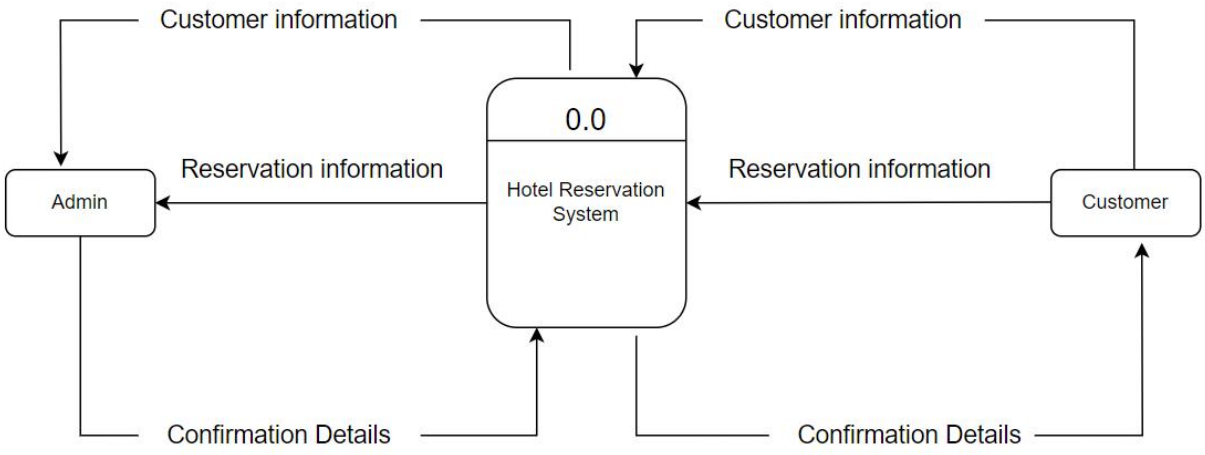


Figure 5: Context level DFD for Hotel Reservation System

The figure above presents the context-level Data Flow Diagram (DFD) for the **Hotel Reservation System**. It shows the main interactions where the admin inputs customer information and reservation details, and the system processes this data to provide reservation and confirmation details back to the admin. Similarly, customers submit their information, and the system returns the corresponding reservation and confirmation details. This diagram offers a high-level view of the system, focusing on the core processes and interactions.

Level-1 DFD

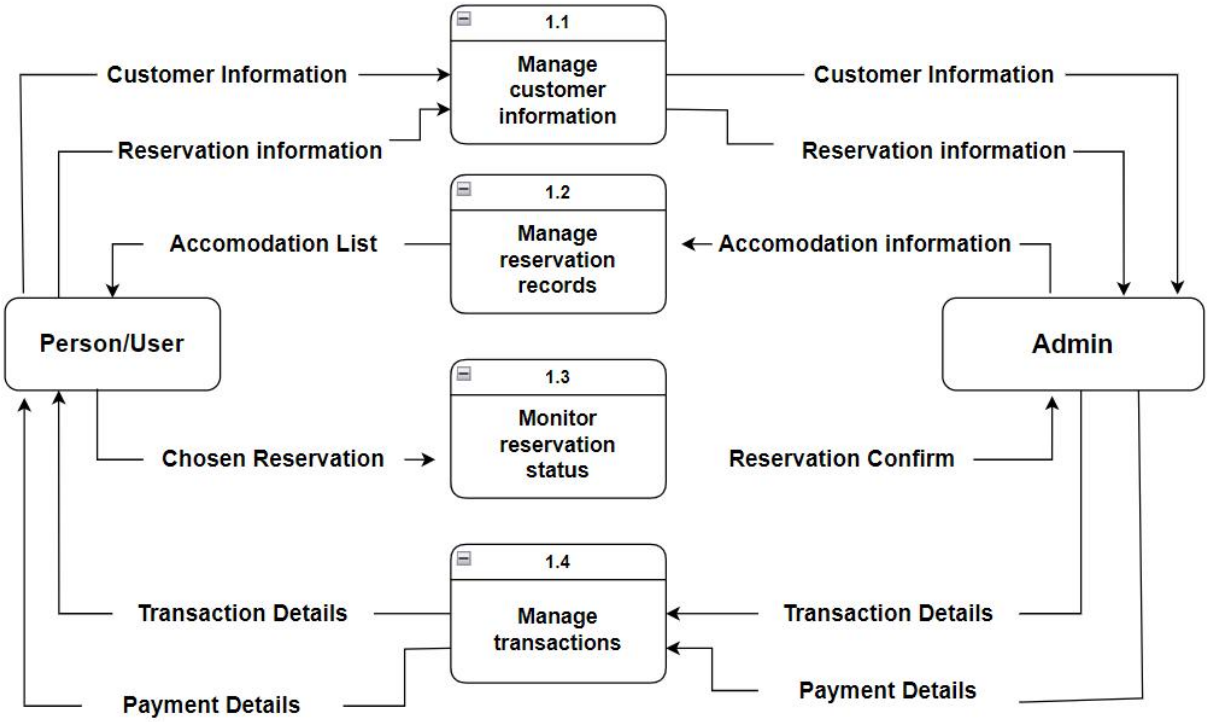


Figure 6: Level 1 DFD for Hotel Reservation System

The figure above presents the Level 1 Data Flow Diagram (DFD) for the **Hotel Reservation System**, detailing the main processes involved. It includes **Manage Customer Information**, where customer details are handled; **Manage Reservation Records**, which processes reservation data provided by both the admin and the user; **Monitor Reservation Status**, allowing users to view their booking status and providing confirmation to the admin; and **Manage Transactions**, which processes financial transactions and payment details from both the admin and the user. This DFD provides a clear overview of the system's workflow and interactions.

Chapter 3: System Design and Implementation

3.1. Architecture of system

Architectural design is considered the basis and the first phase before bringing the idea to reality. In the same way, it mixes design, understood as the creative process, and architecture, which is based on the creation and presentation of solutions at a technical level

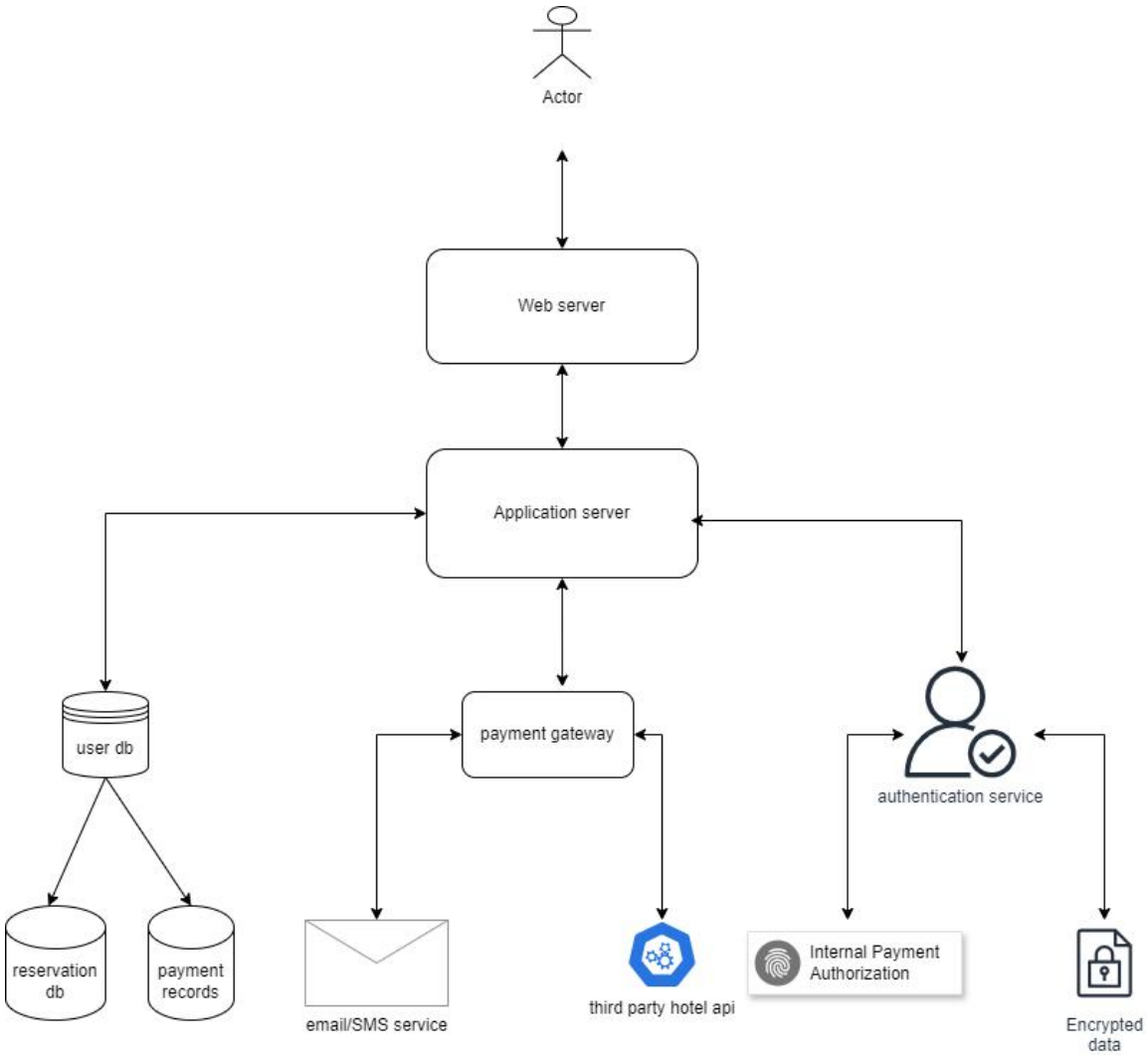


Figure 7: Architecture of Hotel Reservation System

Workflow Overview:

1. The Actor initiates a request (e.g., to book a room) through the web server.
2. The web server forwards this request to the application server.
3. The application server interacts with the user database to retrieve user data and may query the reservation database for availability.
4. If the user proceeds to payment, the application server communicates with the payment gateway, which processes the payment and interacts with external payment services.
5. Payment details are recorded in the payment records database.
6. Upon successful payment, the application server may use the email/SMS service to send a confirmation to the user.
7. The application server may also interact with third-party hotel APIs for additional services or data.
8. Throughout the process, the authentication service ensures secure access and the internal payment authorization service validates payment transactions.
9. Sensitive data, such as user credentials or payment information, is encrypted to ensure security.

3.2. UI design

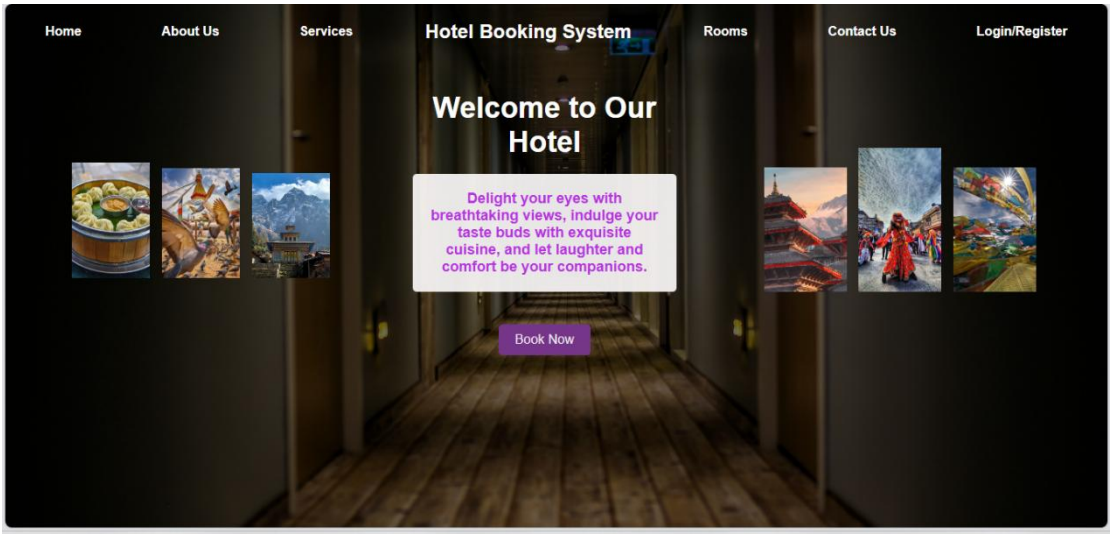


Figure 8: Home Page

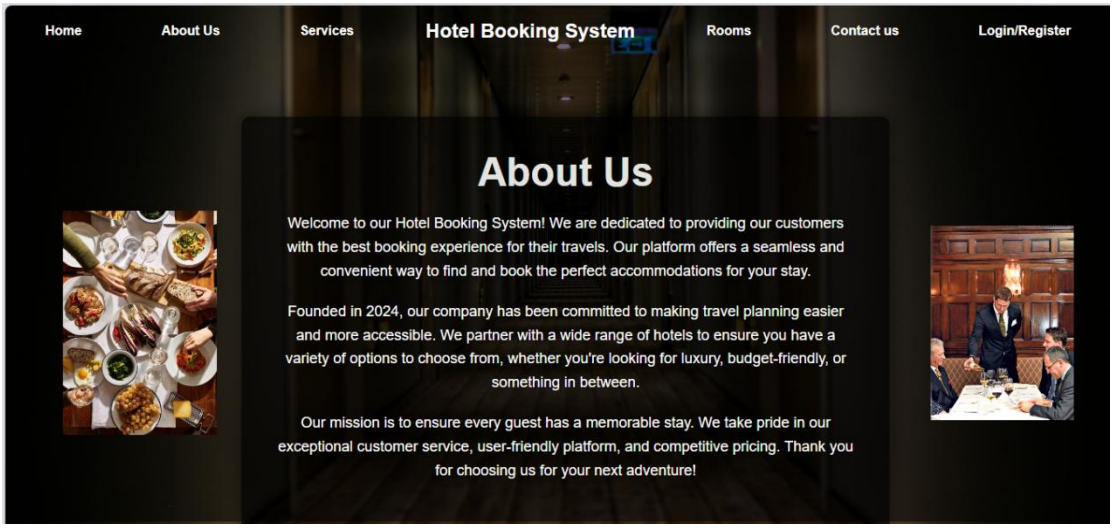


Figure 9: About Hotel Booking System

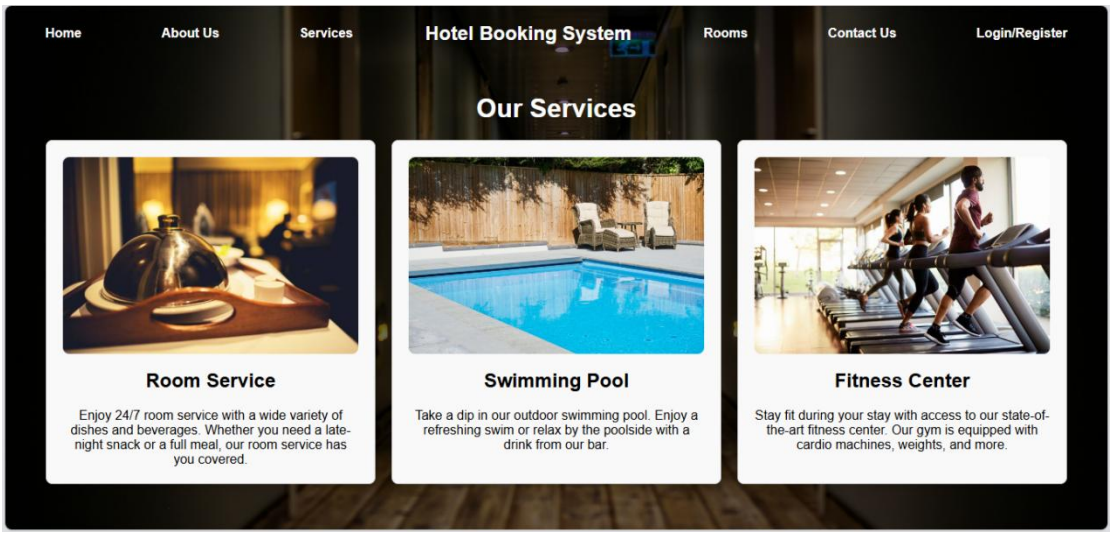


Figure 10: Services

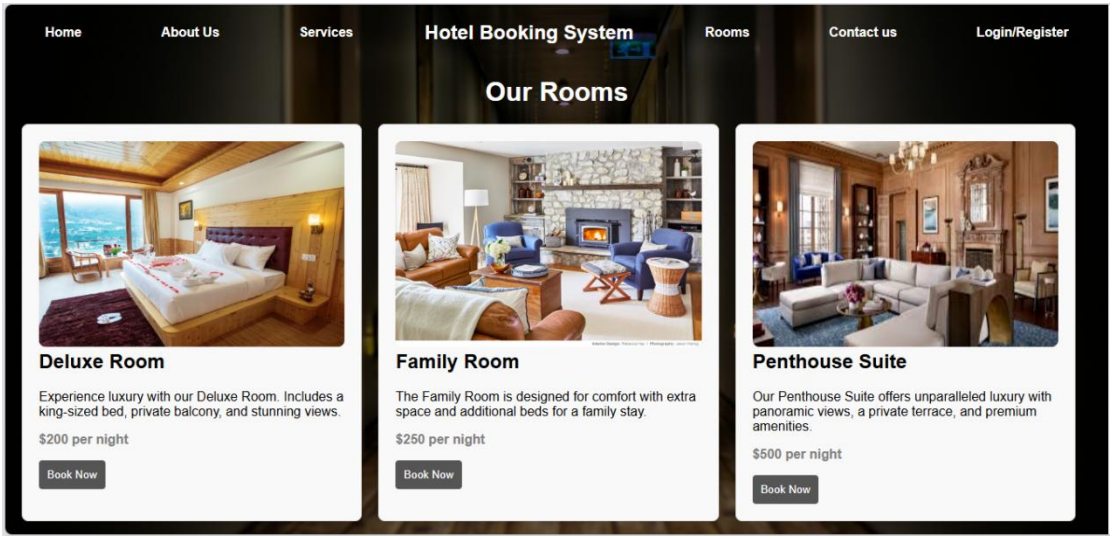


Figure 11: Available Rooms

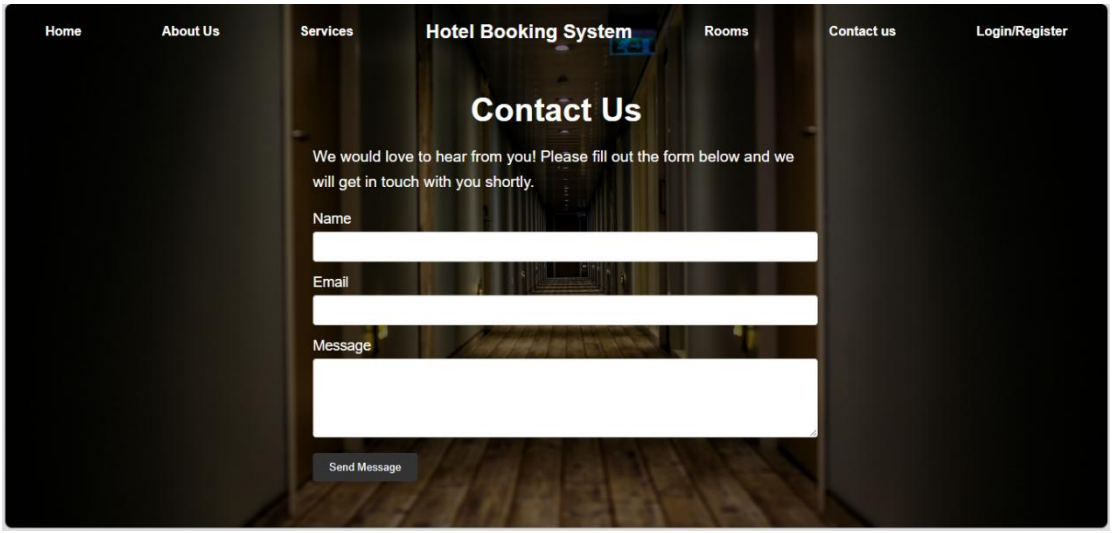


Figure 12: Contact Page

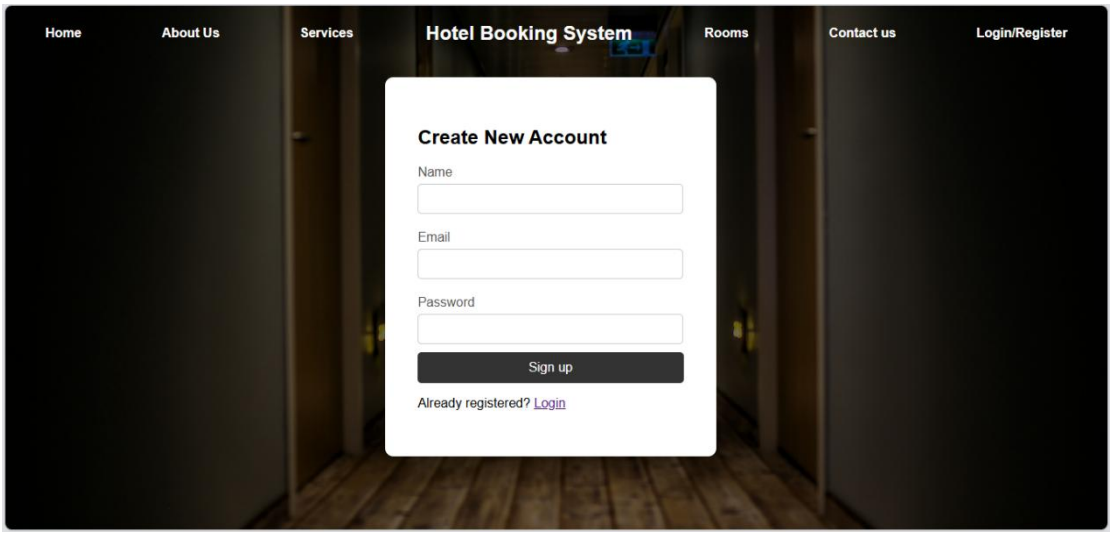


Figure 13: Registration Form

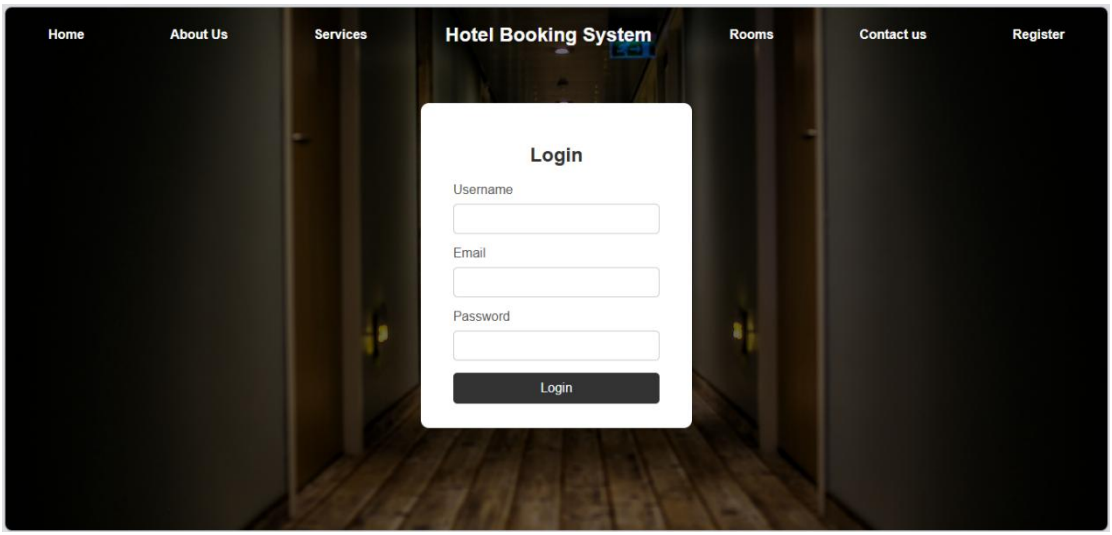


Figure 14: Login Form

3.3. Schema design

Schema Design represents the actual database schema used in an application. Actual fields, data types, and relationships are represented clearly so anyone can understand what data is being stored and where in the designed system

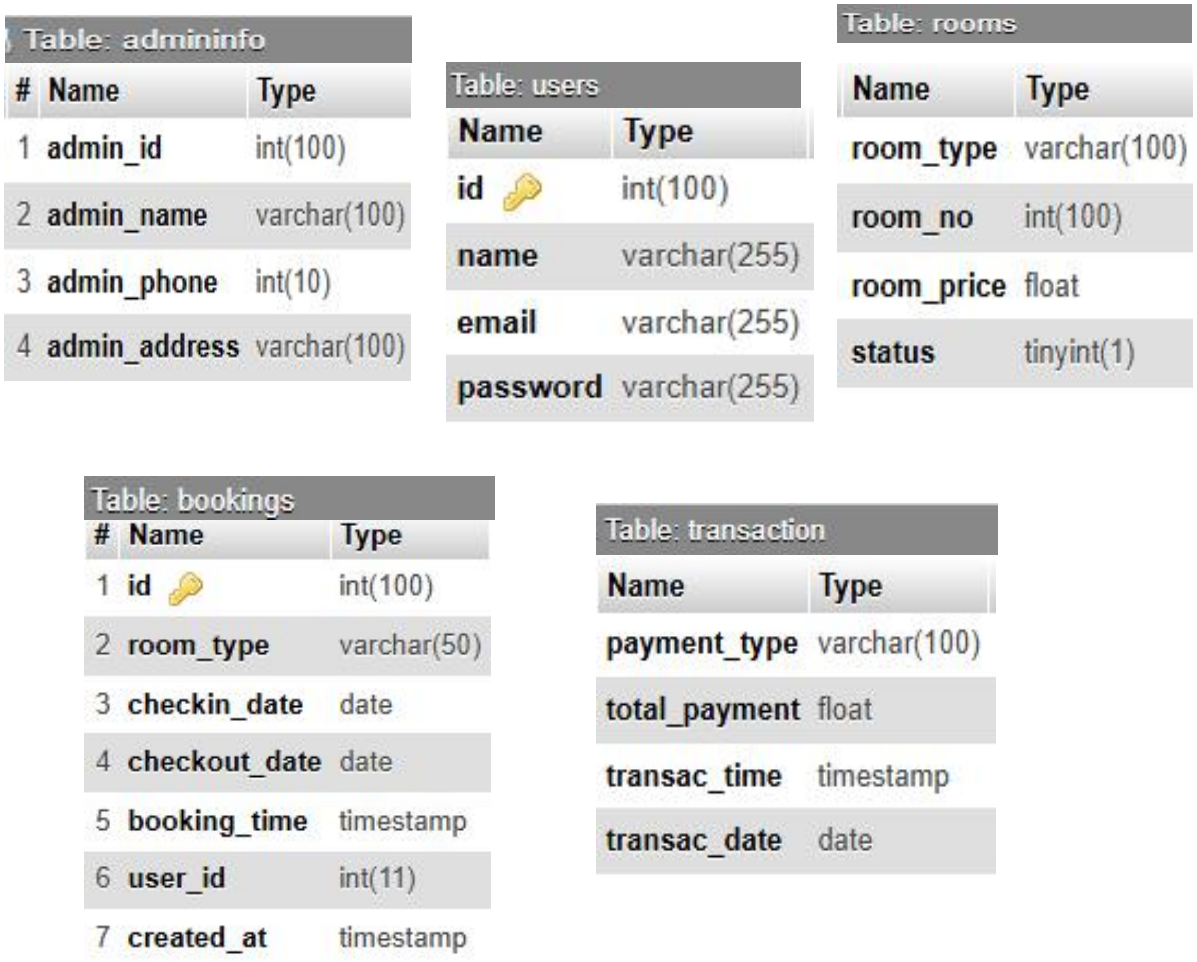


Figure 15: Schema of Hotel booking system

3.4. Module Description

3.4.1. User Authentication Module

- **Description:**
    - This module manages the authentication process for users, ensuring secure access to the system. It handles user login, registration, and password recovery.
    - The module uses a login form where users input their credentials. The registration form allows new users to create accounts by entering their details.
    - All user credentials and profiles are securely stored in a MySQL database, ensuring data integrity and privacy.
- **Key Features:**
    - **Login Form:** Allows registered users to access the system by entering their username and password.
    - **Registration Form:** Enables new users to sign up by providing necessary details such as name, email, and password.
    - **Password Recovery:** Provides a mechanism for users to reset their passwords in case they forget them.

3.4.2. User Dashboard Module

- **Description:**
    - The dashboard is the central interface for authenticated users, providing them with access to various features of the system. It displays user-specific information and allows for easy navigation to different sections.
    - This module is responsible for fetching and displaying data from the MySQL database and ensuring that users can manage their reservations, view booking history, and update their profiles.
- **Key Features:**
    - **User Interface:** A clean, user-friendly interface that displays relevant information such as upcoming reservations and booking history.

- **Data Management:** Provides options for users to view, update, and manage their bookings and personal information.
- **PHP Integration:** Uses PHP scripts to interact with the MySQL database, retrieving and updating user data dynamically.
- **Navigation:** Easy navigation to other modules, such as making a new reservation or contacting customer support.

3.4.3. Reservation Management Module

- **Description:**
  - This module allows users to search for available rooms, make reservations, and manage their bookings. It interfaces with the MySQL database to store and retrieve reservation data.
  - The reservation management module is critical for ensuring that room availability is accurately reflected and that users can easily make and modify reservations.
- **Key Features:**
  - **Room Search and Availability:** Users can search for rooms based on their preferences, with the system checking availability in real-time.
  - **Booking Process:** A streamlined booking process that guides users through selecting a room, entering details, and confirming the reservation.
  - **Reservation Database:** All booking details are stored in the MySQL database, ensuring that data is accurate and accessible.
  - **Dashboard Integration:** Reservations made through this module are immediately reflected in the user's dashboard for easy access and management.

3.4.4. System Management Module

- **Description:**
    - This module handles the core backend functionalities of the system, ensuring that the server, database, and application logic work together seamlessly.
    - The system management module is responsible for maintaining the Apache server, managing database connections, and optimizing PHP scripts for performance.
- **Key Features:**
    - **Apache Server Management:** Ensures the Apache server is configured correctly to handle requests and serve content efficiently.
    - **Database Management:** Manages the MySQL database, including creating, updating, and deleting records as necessary.
    - **PHP Integration:** Facilitates the interaction between the frontend and backend by executing PHP scripts that perform essential operations.
    - **Performance Optimization:** Regularly monitors and optimizes server and database performance to ensure a smooth user experience.

3.5 Tools and Techniques

Creating a hotel reservation system involves using a variety of tools and technologies to handle different aspects of the application. Here are some commonly used tools and technologies:

Frontend Development

- **HTML/CSS**: For structuring and styling the web pages.
- **JavaScript**: For adding interactivity to the web pages.

Backend Development

- **PHP**: Server-side languages to handle business logic and database interactions.

Database Management

- **MySQL**: Relational databases for storing structured data

UX/UI Design Tools:

- **Figma, draw.io**: Collaborative interface design tool.

3.6 Testing

Test case for Unit Testing

Test case id	1		
Test case Description	User Registration		
Prequisities	1. Enter all fields 2. Click Sign Up		
Test Scenario 1	User enter existing email		
Test data	Username: grish Email: test@gmail.com (already exist in database.) Password: grish		
Steps	Expected Output	Actual Output	Pass/Fail
1	Already exists	Email already registered	Pass
Test Scenario 2	User forgets to enter particular field		
Test data	Username: test Email: test@gmail.com Password:		
Steps	Expected Output	Actual Output	Pass/Fail
2	Display message to fill out fields	Please enter the password	Pass

Test Scenario 3	User enter all valid details		
Test data	Username: test Email: test@gmail.com Password: test		
Steps	Expected Output	Actual Output	Pass/Fail
2	Account Registered	Accounted successfully created	Pass

Table 1: Test Case-1

Test case id	2		
Test case Description	User Login		
Prequisities	1. Enter all valid credentials 2. Click Login		
Test Scenario 1	User enter wrong email		
Test data	Username: test Email: test123@gmail.com Password: test		
Steps	Expected Output	Actual Output	Pass/Fail
1	Incorrect information	No user found with this email	Pass
Test Scenario 2	User enter wrong password		
Test data	Username: test Email: test@gmail.com Password: test123		
Steps	Expected Output	Actual Output	Pass/Fail
2	Password does not match	Invalid password	Pass

Test Scenario 3	User enter all valid details		
Test data	Username: test Email: test@gmail.com Password: test		
Steps	Expected Output	Actual Output	Pass/Fail
2	User account successfully	Accounted successfully logged in	Pass

Table 2: Test Case-2

Test case id	3		
Test case Description	Book Room		
user	User must login		
Test Scenario 1	User select room and click book option		
Steps	Expected Output	Actual Output	Pass/Fail
1	Booking Successful	Selected room is booked	Pass
Test Scenario 2	User select room that is already booked		
Steps	Expected Output	Actual Output	Pass/Fail
2	Room not available.	Room is unavailable. Select another room.	Pass

Table 3: Test Case-3

Chapter 4: Conclusion and Lesson Learned

4.1 Conclusion

This project was created with the goal of simplifying the hotel booking process and providing guests with exceptional service. After the completion of the project, the system is fully capable of displaying available rooms and facilitating bookings, offering a seamless user experience. The system is now completely responsive and optimized for both desktop and mobile devices, ensuring that guests can make reservations anytime, anywhere.

4.2 Lesson Learned

While working on his project, we have gained valuable insights into various aspects of software development that are applicable in real-world scenarios. Although this project was initially intended to fulfill academic requirements, with the guidance of our instructors and the time invested, we have learned much more about software engineering, testing, database management, best practices for software creation, time management, and effective audience targeting. Additionally, this project has greatly enhanced our ability to collaborate as a team, actively address challenges, and deliver a polished product.

Although the project has successfully turned out as envisioned, there are additional features we plan to incorporate in the future. Our goal is to make the system even more user-friendly and competitive in the marketplace.

4.3 Recommendation

Based on the lessons learned from developing an online hotel reservation system, here are some recommendations to further enhance its effectiveness:

1. **Implement Advanced Analytics**: Use data analytics to track booking patterns, guest preferences, and market trends. This can help in making informed decisions about pricing, marketing, and service improvements.
2. **Offer Multiple Payment Options**: Provide various payment methods, including credit/debit cards, digital wallets, and bank transfers, to cater to different guest preferences and increase booking conversions.
3. **Strengthen Security Measures**: Implement robust security protocols to protect guest data and payment information. This includes SSL certificates, data encryption, and compliance with data protection regulations.
4. **Implement Feedback Mechanisms**: Encourage guests to leave reviews and feedback after their stay. Use this feedback to make continuous improvements to the reservation system and overall guest experience.
5. **PDF Download Functionality**: This would enable users to easily keep track of their bookings, print receipts for record-keeping, and share details with others. This feature would also contribute to a more organized and professional user experience.

By following these recommendations, you can create a more robust, user-friendly, and efficient hotel reservation system that meets the needs of both the hotel and its guests.

8. References

1. *Hotel Reservation System [DOC]. (n.d.). Scribd. Retrieved May 6, 2024, from https://www.scribd.com/document/398953206/Hotel-Reservation-System-DOC*

2. *Project Proposal: Hotel Reservation System. (n.d.). Scribd. Retrieved May 6, 2024, from https://www.scribd.com/document/330237669/Project-Proposal-Hotel-Reservation-System*

3. *Documentation of Online Booking System. (n.d.). ResearchGate. Retrieved May 8, 2024, from https://www.researchgate.net/publication/275097517_DOCUMENTATION_OF_ONLINE_BOOKING_SYSTEM*

4. *Online Hotel Reservation System [Diagram]. (n.d.). Creately. Retrieved May 13, 2024, from https://creately.com/diagram/example/hkxgmux72/online-hotel-reservation-system*

5. *Kowdagani, S. (n.d.). Hotel Booking Management System. LinkedIn. Retrieved May 8, 2024, from https://www.linkedin.com/pulse/hotel-booking-management-system-sruthi-kowdagani*

6. *System Design of Hotel Management System. (n.d.). OpenGenus IQ. Retrieved May 16, 2024, from https://iq.opengenus.org/system-design-of-hotel-management-system/*

7. *Functional and Non-Functional Requirements in Software Engineering. (n.d.). Board Infinity. Retrieved May 16, 2024, from https://www.boardinfinity.com/blog/functional-and-non-functional-requirements-in-software-engineering/*

8. *Pellinore, X. (2024, May 17). Hotel Booking System Design. Medium. Retrieved from https://medium.com/@pellinore.x/hotel-booking-system-deisgn-a8ff8a4b54aa*

9. *Define Feasibility Study in Software Engineering. (n.d.). Ellow. Retrieved May 17, 2024, from https://ellow.io/define-feasibility-study-in-software-engineering*

Appendix: Screenshots

Screenshot 1: Registration Form Page Interface

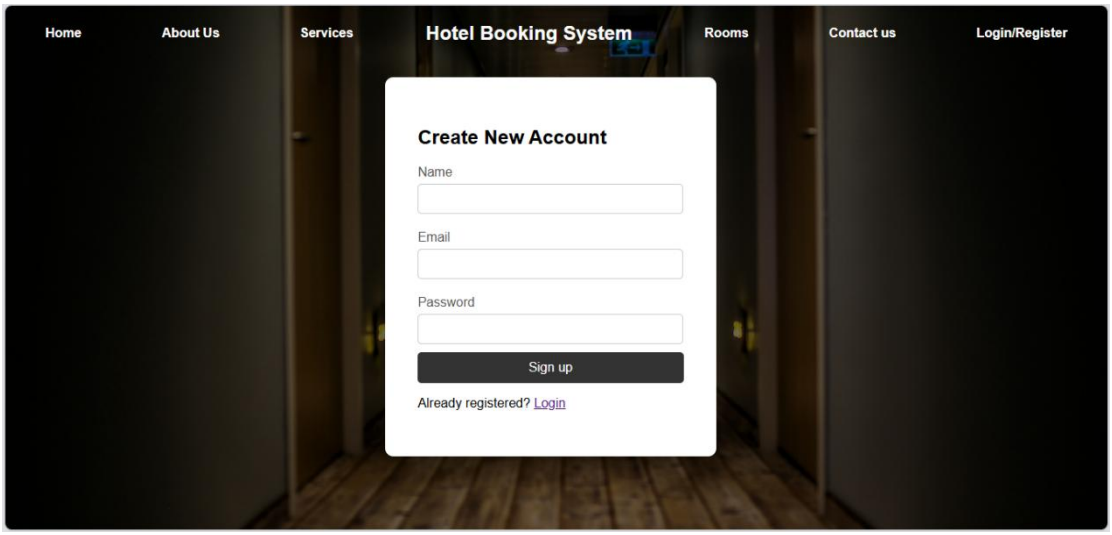


Figure 16: Registration Form Page Interface

Screenshot 2: Login Page Interface

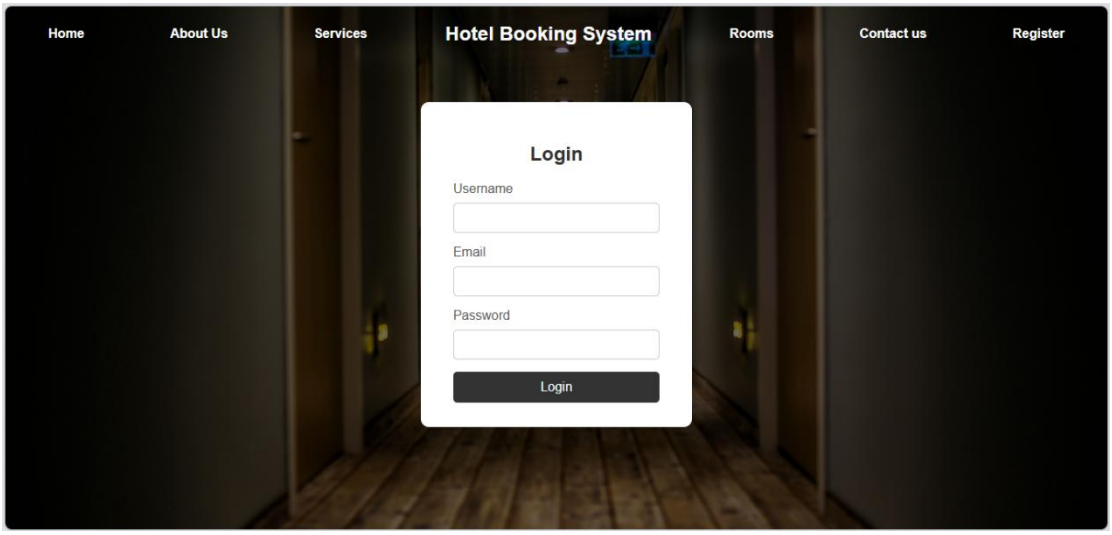


Figure 17: Login Form Page Interface

Screenshot 3: Registered User's Database Page

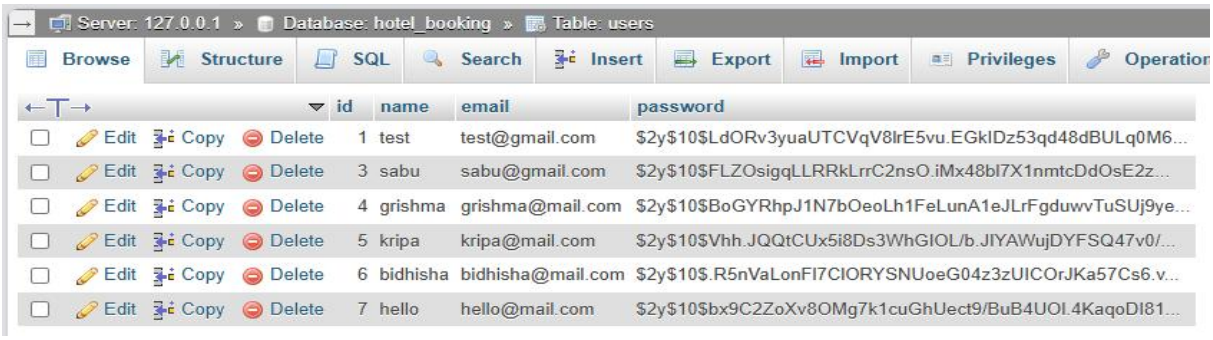


Figure 18: Data of registered Users

Screenshot 4: Room booking confirmation form

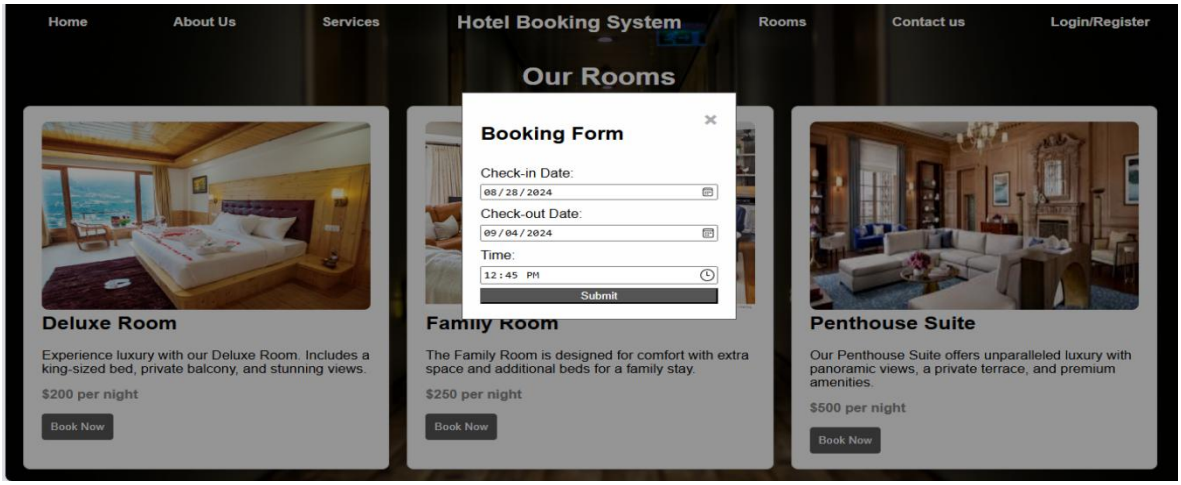


Figure 19: Booking Confirmation Page

Screenshot 5: Booked room Database

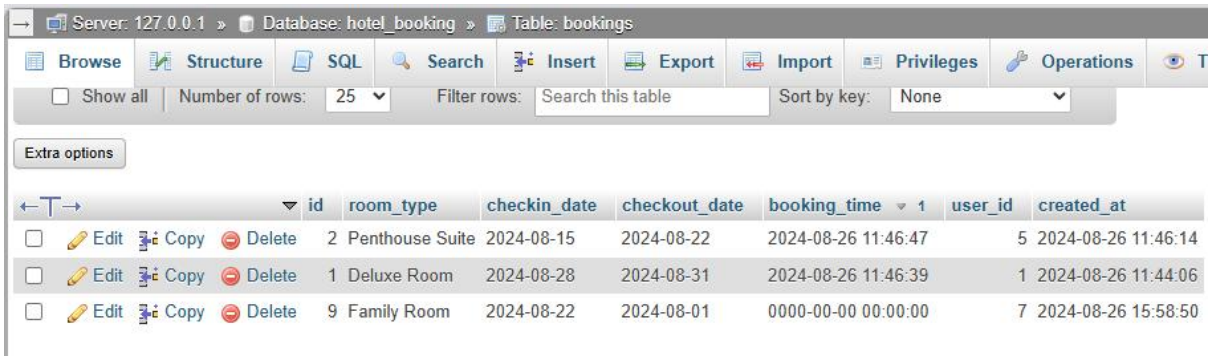


Figure 20: Data of booked room