Design and Implementation of a Hotel Management System

by

Mst Tahmina Akter
ID: CSE2101022059

Md Musfiqur Rahaman
ID: CSE2103024026

Md Asib Hoshen
ID: CSE2103024121

Supervised by
Salma Tabashum

Submitted in partial fulfillment of the requirements for the degree of
Bachelor of Science in Computer Science and Engineering



DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
SONARGAON UNIVERSITY (SU)
September 2025

APPROVAL

The project titled **"Design and Implementation of a Hotel Management System"**, submitted by Md Asib Hoshen (CSE2103024121), Mst Tahmina Akter (CSE2101022059), and Md Musfiqur Rahaman (CSE2101022059) to the Department of Computer Science and Engineering, Sonargaon University (SU), has been accepted as satisfactory for the partial fulfillment of the requirements for the degree of Bachelor of Science in Computer Science and Engineering (B.Sc. in CSE) and approved as to its style and contents.

Board of Examiners

Salma Tabashum
Lecturer

Supervisor

Department of Computer Science and Engineering
Sonargaon University (SU)

(Examiner Name and Signature)
Department of Computer Science and Engineering
Sonargaon University (SU)

Examiner 1

(Examiner Name and Signature)
Department of Computer Science and Engineering
Sonargaon University (SU)

Examiner 2

(Examiner Name and Signature)
Department of Computer Science and Engineering
Sonargaon University (SU)

Examiner 3

DECLARATION

We, hereby, declare that the work presented in this report is the outcome of the investigation performed by us under the supervision of Salma Tabashum, Lecturer, Department of Computer Science and Engineering, Sonargaon University, Dhaka, Bangladesh. We reaffirm that no part of this project has been or is being submitted elsewhere for the award of any degree or diploma.

Countersigned

Salma Tabashum
Lecturer
Supervisor

Signature

Md Asib Hoshen
ID: CSE2103024121

Mst Tahmina Akter
ID: CSE2101022059

Md. Musfiqur Rahaman
ID: CSE2101022059

ABSTRACT

Management is a necessary step in the success of any software project, and discipline is required by standards, regulations, and quality improvement initiatives. Defining and managing requirements is one of the most important challenges in IT and software development projects. Organizations must ensure that their systems meet customer needs, provide compliance, and stay on schedule and within budget. Poorly expressed requirements can create inefficiencies and lead to failure, while a well-defined system ensures long-term success and efficiency.

The **Design and Implementation of a Hotel Management System (HMS)** has been undertaken as a project using modern web technologies such as PHP, HTML, CSS, JavaScript, and MySQL. The main aim of this project is to develop a web-based system that simplifies hotel operations like room booking, guest registration, billing, and reporting. This system has been developed to carry out hotel processes quickly and efficiently, replacing the limitations of manual systems. By automating routine tasks, the system ensures accuracy, saves time, and enhances guest satisfaction. Hence, it provides a complete solution for managing hotel operations in a reliable and user-friendly manner.

ACKNOWLEDGMENT

At the very beginning, we would like to express our deepest gratitude to the Almighty Allah for giving us the strength, patience, and wisdom to complete this project successfully within the given schedule.We are truly grateful to our respected supervisor **Salma Tabashum**, Lecturer, Department of Computer Science and Engineering, Sonargaon University, for his valuable guidance, continuous support, and constructive suggestions throughout the development of this project. His encouragement and advice have been essential in shaping the outcome of this work.

We would also like to extend our sincere thanks to **Prof. Bulbul Ahamed**, Head of the Department of Computer Science and Engineering, and Brig. Gen. (Retd) **Prof. Habibur Rahman Kamal**, NDC, PSC, Dean, Faculty of Science and Engineering, Sonargaon University, for their kind concern, guidance, and encouragement.

Our heartfelt appreciation also goes to all the faculty members of the Department of Computer Science and Engineering, whose teaching and support inspired us throughout our academic journey.

Finally, we dedicate our deepest gratitude and love to our parents and families for their constant support, encouragement, and sacrifices, which gave us the strength to complete this work.

LIST OF ABBREVIATIONS

CSS	Cascading Style Sheet
DB	Database
JS	JavaScript
PHP	Hypertext Preprocessor
SQL	Structured Query Language
API	Application Programming Interface
AJAX	Asynchronous JavaScript and XML
Apache	Apache HTTP Server
HTML	Hypertext Markup Language
HTTP	Hypertext Transfer Protocol
BDT	Bangladeshi Taka (currency in payments)
MVT	Model View Template s
PC	Personal Computer
RAM	Random Access Memory
RDBMS	Relational Database Management System
ROM	Read Only Memory
SDLC	Software Development Life Cycle
UAT	User Acceptance Testing
UI	User Interface
UML	Unified Modeling Language
URL	Uniform Resource Locator
UX	User Experience
VS	Visual Studio
MS Office	Microsoft Office

TABLE OF CONTENTS

Title			Page No.
DECLARATION			I
ABSTRACT			II
ACKNOWLEDGEMENT			III
LIST OF ABBREVIATION			IV
CHAPTER 1			1 – 3
INTRODUCTION TO HOTEL MANAGEMENT SYSTEM			
1.1	Introduction		1
1.2	Project Objective		1
1.3	Project Planning		1
1.4	Problem Statement		2
1.5	Feasibility Study		2
1.6	Requirement Specification		3
	1.6.1	Hardware Requirements	3
	1.6.2	Software Requirements	3
CHAPTER 2			4-7
BACKGROUND OF HOTEL MANAGEMENT SYSTEM			
2.1	Background		4
2.2	Introduction to Web Development		4
2.3	Technologies Used		4
	2.3.1	Hypertext Markup Language	4
	2.3.2	Cascading Style Sheets	4
	2.3.3	JavaScript	5
	2.3.4	PHP	5
	2.3.5	MySQL Database	5
2.4	Role of Each Technology in the Project		5
2.5	Why These Technologies Were Chosen		5
2.6	Related Work		6

CHAPTER 3 7-9
IMPORTANCE OF HOTEL MANAGEMENT SYSTEM
3.1 Importance of Hotel Management System.…….. 7
3.2 Hotell Management System Website……. 7
3.2.1 Key Features of the Home Page……. 7
3.2.2 Billing section…… 8
3.2.3 Room Browsing and Details…….....…. 8
3.2.4 Complaint Management System…… 8
3.2.5 Admin Management System ……. 8
3.3 Importance of These Features in the Hotel Context…….……. 9

CHAPTER 4 10-15
SYSTEM ANALYSIS OF THE PROJECT
4.1 Software Development Life Cycle…… 10
4.2 Implementing the SDLC ……....…. 10
4.2.1 Planning/Requirement Gathering……... 10
4.2.2 System Design…… 11
4.2.3 Implementation/Development…… 11
4.2.4 Testing…… 11
4.2.5 Deployment…… 11
4.2.6 Maintenance and Updates…….. 11
4.2.7 Evaluation and Retirement……. 11
4.3 Introduction to Front End Tool..……..... 12
4.4 Need for Visual Programming…….. 12
4.5 Advantages of Visual Programming…….. 12
4.6 Introduction to Back End Tool…….. 12
4.7 Research Methodology…….. 12
4.8 Data Collection Methods……... 12
4.9 Questionnaire……. 13
4.10 Data Analysis Method……... 13
4.11 Software/product development methods……... 13
I Inception Phase ……. 14
II Elaboration Phase……... 14
III Construction Phase …… 14
IV Transition Phase…….…… 14

CHAPTER 5 16-19
GRAPHICAL REPRESENT OF PROJECT
5.1 Data Flow Diagram……..……..... 16
5.2 Waterfall Model……..... 16
5.3 Data flow of Hotel Management …….. 16
5.4 Use case Diagram…….. 17
5.5 E-R Diagram…….. 18

CHAPTER 6 20-21
TESTING AND DEBUGGING
6.1 Introduction……..…..... 20
6.2 Testing and Debugging…….……..... 20
6.3 Performance Testing Report…….. 21

CHAPTER 7 22-30
PROJECT OVERVIEW
7.1 Introduction…… 22
7.2 System Design & Architecture…….. 22
7.3 Technology Stack…….. 22
7.4 Implementation Strategy…… 23
7.5 Overview…… 23-30

CHAPTER 8 31-49
FUTURE SCOPE & CONCLUSION
8.1 Limitations of Hotel System….....…… 31
8.2 Future scope of the project ……. 31
8.3 Challenges in Implementing HMS…… 32
8.4 Conclusion……. 32

REFERENCES ……..…….. 33

LIST OF TABLES

Figure No.	Title	Page No.
Table 1.1	Project Plan	1
Table 1.2	Hardware requirements	3
Table 1.3	Software Requirements	3
Table 4.1	Functional requirements for user	15
Table 4.2	Functional requirements for admin	15

LIST OF FIGURES

Figure No.	Title	Page No.
Fig 1.1	Project Planning	2
Fig 4.1	Software Development Life cycle	10
Fig 4.2	Unified software development process	13
Fig 5.2	Waterfall mode	16
Fig 5.3	Data flow diagram	17
Fig 5.4	Use case diagram	18
Fig 5.5	ER- Diagram	19
Fig 6.2	Summary of all testing types with results	20
Fig 6.3	Active Users vs Time and Response Time vs Time.	21

CHAPTER 1

INTRODUCTION TO HOTEL MANAGEMENT SYSTEM

1.1 Introduction

Introduction to the Exchange Hotel Management System Website Project: This project focuses on the design and development of a Hotel Management System (HMS) website aimed at improving hotel operations, guest experience, and administrative efficiency. The web-based platform provides a comprehensive solution for managing various aspects of a hotel, including room booking, guest registration, billing, payment processing, and communication between hotel staff and guests. By automating these processes, the system ensures smooth hotel operations while enhancing overall service quality.

1.2 Project Objective

Hotels businesses worldwide face increasing pressure to improve efficiency and provide better services to their guests. Hotel managers must ensure that resources are utilized effectively, while also ensuring that guests receive timely and high-quality services. A comprehensive, easy-to-navigate Hotel Management System can drastically reduce administrative burdens, streamline room booking and check-in/check-out processes, and help hotel staff deliver a higher standard of service.

The objectives of Hotel Management System (HMS) are:

a) To provide an online room booking and reservation system.
b) To provide an oTo maintain a systematic way of storing guest and booking information.nline room booking and reservation system.
c) To enable secure and efficient billing and payment processing.
d) To enableTo provide administrative tools for managing rooms, guests, and reports. secure and efficient billing and payment processing.

1.3 Project planning

Once we examine that the project is feasible, we undertake project planning. The table below describes how we planned our project.

No	Task Name	Duration	Start	End
1	Planning and Analysis	3 Month	06-09-2024	06-11-2024
2	Design & Implementation	3 Month	07-11-2024	16-02-2025
3	Development	3 Month	05-03-2025	02-06-2025
4	Testing and Maintenance	3 Month	10-06-2025	10-09-2025

Table 1.1 Project Plan

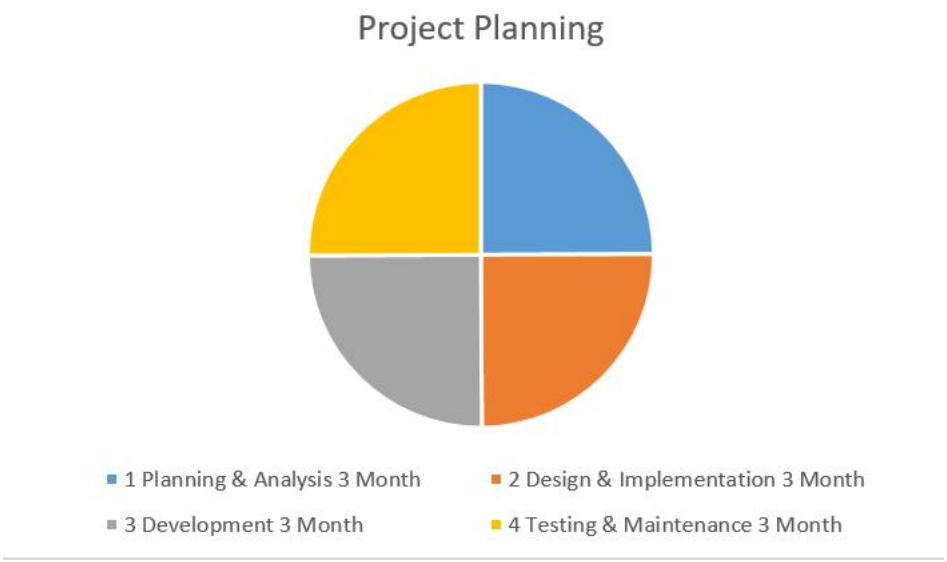


Fig 1.1 Project Planning

1.4 Problem Statement

The current manual system for hotel management requires guests to either call the hotel or physically visit the reception desk to make reservations or check room availability. This process is time-consuming, prone to errors, and often results in double bookings or customer dissatisfaction due to lack of real-time updates. Additionally, manual billing and record-keeping make financial management inefficient and increase the risk of inaccuracies. By implementing an online Hotel Management System, these issues can be eliminated. Online booking systems not only save time but also serve as revenue-generating tools that enhance operational efficiency and strengthen relationships with guests.

1.5 Feasibility Study for a Hotel Management System (HMS)

A Feasibility Study is a critical part of any project to evaluate its practicality, potential costs, and the likelihood of success. When developing a Hotel Management System (HMS), it is important to assess whether the proposed solution can meet both the technical and business requirements. Below is a comprehensive feasibility study, including technical, operational, legal, and financial aspects.

- **Project Overview**

A Hotel Management System (HMS) is a software application that helps manage hotel operations such as room booking, guest check-in and check-out, billing, and staff tasks. The system brings all important activities into one platform, which makes the work faster, reduces mistakes, and improves the guest experience.

- **Technical Feasibility**

This part checks if the Hotel Management System can be built with available technology. The project uses common tools like HTML, CSS, JavaScript, Python (Django), and SQL. These technologies are reliable, secure, and able to handle hotel data such as bookings, payments, and reports

- **Operational Feasibility**

The HMS will fit daily hotel work by helping staff manage rooms, bills, and reports more easily. Guests can book rooms online and get faster service, saving time for both staff and guests.

1.6 Requirement Specification

To successfully implement a Hotel Management System (HMS), it's essential to define both the hardware and software requirements. These requirements ensure the system is scalable, reliable, secure, and can handle the operational needs of a hotel, from managing usert records to booking, control and billing. Below are the detailed hardware and software requirements for a Hospital Management System:

1.6.1 Hardware requirements

To develop this system, hardware requirement is important things to be required. Table 1.2 shows the hardware requirements to develop Hotel Management System (HMS).

Hardware	Description
Laptop with 4GB of RAM	Workstation for the project
Hard disk space (100GB+)	To save data
Others	Keyboard and Microsoft Mouse or compatible pointing device and Android mobile etc.
Printer	A printer to test the report generating function.

Table 1.2 Hardware requirements

1.6.2 Software requirements

After hardware requirement, software requirement is the important things to be install. Table 1.3 shows the software requirement to develop Hotel Management System (HMS)

Software	Description	Cost (RM)
Apache/XAMPP Server	Local web server	Free
PHP	Custom Admin Interface	Free
HTML, CSS, JavaScript	Frontend design & interactivityn	Free
Microsoft Office	Documentation Tools.	Free
Notepad++ / VS Code	Code editor	Free
MySQL	Database system.	Free
Microsoft Visio	To draw Gantt chart	Free
Web Hosting (optional)	Platform to get hosting	Free
Google Chrome, Internet Explorer	Web Browser.	Free

Table 1.3 Software Requirements

CHAPTER 2

BACKGROUND OF HOTEL MANAGEMENT SYSTEM

2.1 Background

The Hotel Management System (HMS) is designed to make hotel operations easier and more efficient. It helps manage room reservations, guest records, billing, and staff tasks through a web-based platform. The system uses web technologies like HTML, CSS, JavaScript, PHP, and MySQL to create a complete and user-friendly solution.

2.2 Introduction to Web Development

- Web development is the process of creating websites and web applications that are interactive and dynamic. Today, almost every business relies on web applications to improve efficiency and provide better services. Hotels also benefit from web-based systems that allow online booking, easy data management, and faster customer service.
- In this project, we use a mix of frontend (HTML, CSS, JavaScript) and backend (PHP, MySQL) technologies to build a full-stack web application..

2.3 Technologies Used

In our Hotel Management System project, we used HTML, CSS, JavaScript, PHP, and MySQL.

HTML was used to create the main structure of the hotel website.
CSS was used to style the website, making it attractive and user-friendly.
JavaScript was used to add interactivity, such as form validation and dynamic updates.
PHP was used as the backend language to manage bookings, billing, and communication between the system and the database.
MySQL was used as the database to store all guest, room, and booking details.

2.3.1 Hypertext Markup Language (HTML)

HTML is the standard language for building web pages. It defines elements like headings, paragraphs, forms, links, and images. In this project, HTML is used to create pages for hotel operations such as room booking, guest registration, and billing.

2.3.2 Cascading Style Sheets (CSS)

CSS is used to design and style web pages, including colors, fonts, and layouts. In this project, CSS makes the hotel website visually appealing and easy to use for guests and staff

2.3.3 JavaScript (JS)

JavaScript is a scripting language that makes web pages interactive. In this project, it is used for form validation (e.g., checking booking details before submission) and other dynamic features.

2.3.4 PHP

PHP is a server-side scripting language used for backend logic. In this project, PHP handles room bookings, guest records, payment processing, and the admin dashboard.

2.3.5 MySQL Database

MySQL is a relational database system used to store and manage data. In this project, MySQL keeps records of rooms, guests, bookings, and payments, and connects with PHP to update and retrieve information.

2.4 Role of Each Technology in the Project

- **HTML**
  Defines the structure of the hotel management web pages, including forms for room booking, guest registration, and check-in/check-out details.
- **CSS**
  Provides the design and layout for the system, making the interface user-friendly, visually appealing, and accessible on both desktop and mobile devices.
- **JavaScript (JS)**
  Adds interactivity to the system, such as form validation, dynamic content updates, and improving the overall user experience.
- **PHP**
  Handles the backend logic of the project. It processes bookings, manages guest data, generates bills, and connects the frontend with the database.
- **MySQL**
  Works as the database for storing and managing data such as guest records, room availability, bookings, and payments.

2.6 Why These Technologies Were Chosen

Simplicity and Efficiency
PHP, HTML, CSS, and JavaScript are widely used and easy to understand. They allow faster development and make the system simple to maintain. MySQL is also easy to set up and reliable for storing hotel data.

Scalability
PHP and MySQL can handle a growing number of users and data as the hotel expands. The system can be upgraded with new features like online payments, mobile access, or reporting tools without affecting existing functions.

Web Compatibility
HTML, CSS, and JavaScript ensure that the system works well on all web browsers and devices, including desktops, tablets, and smartphones. This makes it convenient for both hotel staff and guests to access the system anytime.

2.7 Related Work

Many hotel systems like Opera PMS and Cloudbeds already exist to manage bookings and billing. We also studied some simple PHP projects online. Our system is designed to be easier, low-cost, and focused on the main hotel needs.

Example Background Section for a Hospital Management System Project

With the increasing demand for efficient management systems in the hospitality industry, a Hotel Management System (HMS) has become a necessity. The primary goal of an HMS is to streamline hotel operations, improve guest services, and reduce administrative workload. This project aims to develop a web-based hotel management system that integrates multiple functions such as room booking, guest registration, check-in/check-out, billing, and reporting using modern web technologies like HTML, CSS, JavaScript, PHP, and MySQL.

- **HTML** is used for structuring the web pages. It defines the layout and content, allowing easy interaction with users through forms, tables, and buttons for booking and guest details.
- **CSS** is used to style the web pages, providing a clean, professional look and making the system responsive to different screen sizes.
- **JavaScript** adds interactivity to the system. It is used for validating forms, updating content dynamically, and improving the user experience during room booking or billing.
- **PHP** is employed for writing the back-end logic of the system. It processes user requests, communicates with the database, and handles tasks like managing reservations, updating room status, generating bills, and preparing reports.
- **MySQL**, a reliable relational database system, is used to store and manage hotel data such as guest records, booking details, room availability, and payment history. It ensures data security and easy retrieval for reporting.

The combination of these technologies ensures that the system is simple to develop, secure, scalable, and user-friendly. By using PHP with MySQL, the project benefits from strong database connectivity, smooth data management, and an efficient workflow for hotel operations.

CHAPTER 3

IMPORTANCE OF HOTEL MANAGEMENT SYSTEM

3.1 Importance of Hotel Management System

In the modern hospitality industry, efficiency, accuracy, and customer satisfaction are paramount. The growing number of guests, the increasing variety of services offered, and the need for real-time data access have made traditional, paper-based hotel management practices inadequate.

A Hotel Management System (HMS) addresses these challenges by integrating various hotel operations into a unified, digital platform. It enables hotel administrators to manage reservations, room availability, payments, complaints, and staff assignments seamlessly. Guests benefit from quick booking processes, transparent billing, and improved communication with hotel staff.

3.2 Hotel Management System Website - Home Page

The Hotel Management System (HMS) website is designed to streamline hotel operations through an online interface. The website provides guests with a simple and intuitive platform to access hotel services, such as room booking, reservation history, and lodging details. Meanwhile, administrators can manage bookings, track payments, handle complaints, and oversee staff operations.The home page serves as the first point of interaction for guests, offering quick access to essential hotel services, ensuring convenience, and establishing a professional digital presence.

3.2.1 Key Features of the Home Page

1. **Header Section:**
    - **Logo and Hotel Name**: The hotel logo and brand name are prominently displayed at the top for quick recognition
    - **Navigation Menu**: Clear and concise navigation links (e.g., Home, Rooms, Booking, About Us, Contact) are provided for easy access to the website's main sections.
    - **Search Bar**: Guests can quickly search for room types, availability, or services directly from the header.
2. **Main Banner:**
    - The home page features a large, visually appealing banner showcasing hotel services, special offers, or welcoming text such as "Welcome to [Hotel Name] – Comfort, Luxury, and Hospitality Await You."
    - Background images (e.g., premium rooms, dining area, or hotel exterior) highlight the hotel's quality.
3. **Introduction Section:**
    - A brief description of the hotel and its core services is displayed.
    - This section highlights the hotel's mission and values, emphasizing hospitality, comfort, and quality service.
    - Users are introduced to the main services such as room booking, dining, free WiFi, 24/7 customer support, and event hosting.

3.2.2 Reservation & Billing Section

The booking and billing features are central to the HMS. Guests can reserve rooms online and pay securely, while administrators can manage and track payments.

Key Features:

- **Automated Booking**: The system instantly confirms reservations based on available rooms.
- **Transparent Billing**: Each booking automatically generates a bill detailing room charges and services.t.
- **Online Payment**: Integration allows guests to pay online, reducing dependency on manual transactions.
- **Booking History:** Guests can view their past and upcoming reservations through the booking history page.

3.2.3 Room Browsing and Details

Guests can browse through all available rooms with detailed descriptions and images.

Key Features:

- **Room Categories**: Single, double, deluxe, or suite rooms with descriptions and images.
- **Room Availability**: Real-time availability status ensures accurate booking.
- **Room View Page**: Each room has its own detail page, showcasing price, amenities, and photos to help guests make decisions..

3.2.4 Complaint Management System

The HMS also provides a complaint management module where guests can submit issues or feedback online.

Key Features:

- **Complaint Submission**: Guests can lodge complaints or feedback directly on the website.
- **Admin Monitoring**: Administrators view and manage complaints from the dashboard.
- **Faster Resolution**: Ensures timely responses, improving customer trust and satisfaction.

3.2.5 Admin Management System

The admin dashboard centralizes hotel management tasks. It allows administrators to:

Key Features:

- Manage room records (add, update, or remove rooms).
- Handle reservations and check-ins/check-outs.
- Track staff assignments and maintain employee history.
- Monitor revenue, available rooms, and pending payments through real-time counters.
- Access complaints and ensure quick resolutions.

3.3 Importance of These Features in the Hotel Context

- **Guest Convenience**: Online booking, room browsing, and billing history improve customer experience.
- **Operational Efficiency**: Automation reduces staff workload and minimizes errors.
- **Transparency**: Clear billing and complaint tracking build guest trust.
- **Scalability**: The system is expandable to include advanced features such as payment gateways, mobile apps, or loyalty programs in the future.

CHAPTER 4

SYSTEM ANALYSIS OF THE PROJECT

4.1 Software Development Life Cycle (SDLC)

A Hotel Management System (HMS) is developed by following the Software Development Life Cycle (SDLC) to ensure the system meets both guest and administrative needs. SDLC provides a structured approach consisting of planning, design, development, testing, deployment, and maintenance.

The system development life cycle can be divided into seven phases as shown in figure

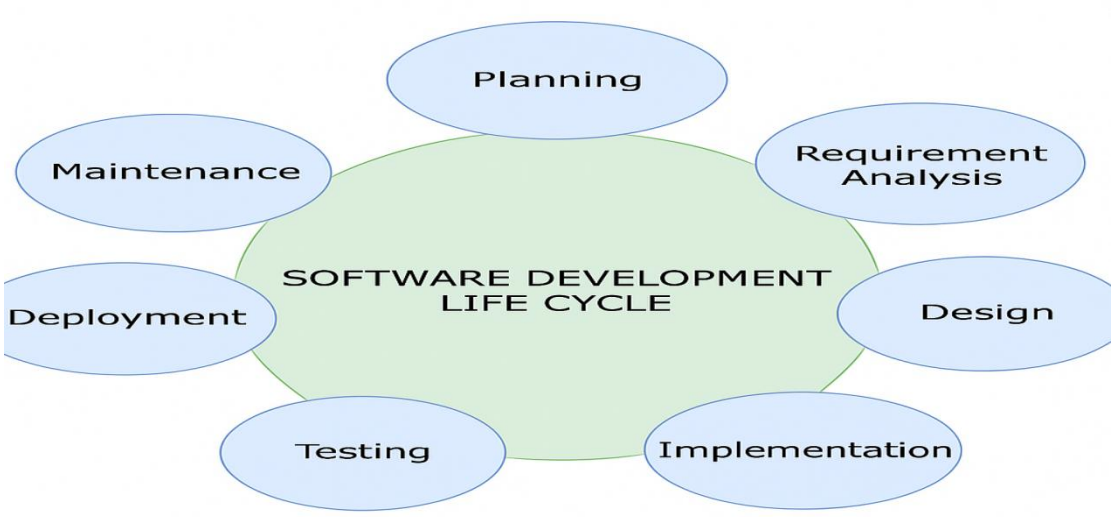


Figure 4.1: Software Development Life Cycle.

4.2 Implementing the SDLC for Hotel Management System (HMS)

4.2.1 Planning/Requirement Gathering

The goal of this phase was to identify hotel requirements and define the scope of the system.

- **Functional Needs**: Room booking, room browsing,complaints handling, staff management, admin dashboard with reports, billing, booking history.
- **Non-Functional Needs**: Security of guest data, ease of use, scalability, and system reliability.
- **Stakeholders**: Guests (users), hotel staff, administrators.

4.2.2 System Design

This phase produced blueprints for the system:

- **Architecture**: A client-server model where the website (PHP/HTML/CSS/JS frontend) interacts with a MySQL database.
- **Database Design**: Tables for rooms, reservations, users, staff, complaints, and payments.
- **User Interface**: Guest-facing pages (index.php, booking.php, complaints.php) and admin pages (dashboard.php, reservation.php, staff_mang.php).
- **Modules**: Guest booking, booking history, complaint submission, room browsing, admin reservation management, and staff assignment.

4.2.3 Implementation / Development

- The system was coded in PHP for server-side logic.
- MySQL was used for the database.
- HTML, CSS, Bootstrap, and jQuery were used for the frontend.
- Separate modules were developed for guest and admin sides.
- File upload handling was added for room images.

4.2.4 Testing

- **Unit Testing**: Checked room booking, complaints submission, and login modules.
- **Integration Testing**: Verified guest and admin modules work together.
- **System Testing**: Validated overall booking workflow from guest booking to admin confirmation.
- **User Acceptance Testing (UAT)**: Conducted with sample users to ensure usability.

4.2.5 Deployment

The system can be deployed on a web server (e.g., Apache with PHP & MySQL).

- Deployment involved configuring the database (hotelms.sql) and connecting through db.php.
- Guests access via the hotel's domain, while admins log in through the /admin/ portal.

4.2.6 Maintenance and Updates

- Bugs, errors, and usability issues are tracked and fixed
- Regular updates include security patches and new features (e.g., online payment gateway, reporting tools).

4.2.7 Evaluation and Retirement

- The system effectiveness is evaluated by monitoring booking efficiency, complaint resolution, and guest satisfaction.
- Eventually, the system can be upgraded with a mobile app or migrated to cloud-based hosting for scalability.

4.3 Introduction to Front-End Tools

The front end is built using:

- HTML/CSS for structure and styling.
- Bootstrap for responsive layouts.
- JavaScript/jQuery for interactivity (form validation, AJAX requests).

These tools create a clean and responsive design accessible on desktop and mobile.

4.4 Need for Visual ProgrammingThe front end is built using:

Visual programming via Bootstrap and CSS provides intuitive navigation with menus, banners, and room galleries. It reduces complexity for developers and ensures a user-friendly design for guests.

4.5 Advantages of Visual ProgrammingThe front end is built using:

- Predefined UI components (buttons, forms, tables) make development faster.
- Responsive design ensures compatibility with multiple devices.
- Easy maintenance since styles are centralized.

4.6 Introduction to Back-End Tools

The back end is developed in **PHP** and connected to a **MySQL database**.

Key SQL Operations:

- INSERT for new bookings, complaints, staff records.
- SELECT for fetching room availability and booking history.
- UPDATE for modifying reservations or staff details.
- DELETE for removing outdated data.

The database schema ensures proper relationships between users, rooms, and reservations.

4.7 Research Methodology

The project followed an applied research methodology. Requirements were gathered by studying existing hotel booking systems and identifying their gaps. Questionnaires and user feedback helped finalize the system design.

4.8 Data Collection Methods

Data was collected through:

- **Observation**: Studying existing hotel booking workflows.
- **Questionnaires**: Getting feedback from potential users about their expectations.
- **Interviews**: Discussions with hotel staff and administrators.

4.9 Questionnaire

The questionnaire included:

- Guest preferences (online booking vs. in-person).
- Importance of features like booking history, complaints, and online payments.
- Staff/admin expectations (easy room management,complaint tracking, payment monitoring).

4.10 Data Analysis Method

Responses were analyzed statistically to prioritize features. Most users preferred:

- Fast room booking.
- Clear billing history.
- Complaint management.

This analysis shaped the functional requirements of the HMS.

4.11 Software/product development methods

To ensure a structured and systematic development process for the Hotel Management System (HMS), the Unified Software Development Process (USDP) was chosen as the guiding methodology.USDP is an iterative and incremental software development process that is use-case driven, architecture-centric, and risk-focused. It ensures that requirements are well defined, designs are clearly modeled, and implementations are carried out in manageable phases.

USDP is divided into four phases' inception, elaboration, construction and transition.

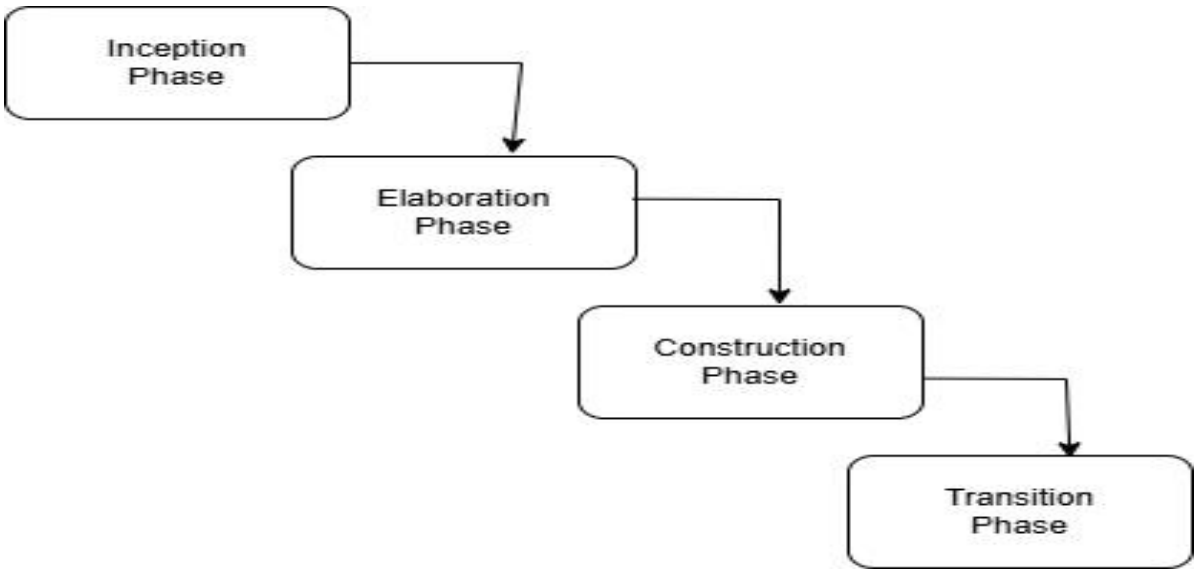


Fig 4.2: Unified Software Development Process

I. Inception Phase

In this phase, the scope of the project was defined and requirements were collected.

- Problems with manual hotel management were identified (e.g., difficulty in tracking reservations, lack of real-time room availability, manual billing errors).
- Objectives were established: allow online room booking, track guest history, manage staff, and provide an admin dashboard with reports.
- Stakeholders (guests, hotel staff, and administrators) were identified.

II. Elaboration Phase

During this phase, detailed analysis was conducted, and system requirements were finalized.

- Functional requirements were gathered for both guests (room browsing, booking, complaint submission) and administrators (room management, staff records, billing, and complaint resolution).
- System design was prepared, including database schema (hotelms.sql) and architectural diagrams (ERD, Use Case).
- Risks such as data security, user authentication, and file upload safety were considered.

III. Construction Phase

The system was implemented in this phase based on the requirements and design.

- **Frontend**: HTML, CSS, Bootstrap, and jQuery were used for guest and admin interfaces.
- **Backend**: PHP handled reservation logic, complaints, billing, and staff management.
- **Database**: MySQL stored user details, bookings, payments, and complaints.

- **Modules Implemented**:

- Guest side: room browsing, booking, booking history, complaints
- Admin side: room management, staff management, reservation handling, complaints, and dashboard counters (income, available rooms, pending payments).

IV. Transition Phase

In this phase, the system was deployed and tested with real scenarios.

- **Testing**: Unit, integration, and user acceptance testing were performed.
- **Deployment**: The system was deployed on a PHP-MySQL environment (e.g., XAMPP server).
- **Feedback**: Guest and admin trial usage helped identify areas for improvement (e.g., future integration of payment gateways and mobile support).
- **Maintenance**: Provisions were made for fixing bugs, updating features, and scaling to handle more hotel operations.

❖ Functional requirements User

No.	Requirement	Description
1	View Room Information	The system should allow guests to view room details such as type, price, and availability.
2	Make a Booking	Guests should be able to book available rooms by providing personal details and dates of stay.
3	View Booking History	Guests can check their past and upcoming reservations through their account.
4	Submit Complaint	Guests can lodge complaints or provide feedback directly through the website.
5	Payment & Billing	Guests can view detailed billing information and make payments online.

Table 4.1 Functional requirements for user

❖ Functional requirements Admin

No.	Requirement	Description
1	Login Authentication	Admins should log in securely using a username and password.
2	Manage Rooms	Admins can add, edit, update, or delete room details, including images and pricing..
3	Manage Reservations	Admins can confirm, modify, or cancel reservations.
4	Manage Stafft	Admins can add, edit, and monitor staff records and history.
5	Handle Complaints	Admins can view, track, and respond to guest complaints.
6	Generate Reports	Admins can monitor income, pending payments, room availability.

Table 4.2 Functional requirements for admin

CHAPTER 5

GRAPHICAL REPRESENT OF PROJECT

5.1 Data Flow Diagram

The Data Flow Diagram (DFD), also known as a bubble chart, is used to clarify system requirements and identify major transformations that will become part of the system design. It is the starting point for specifications, breaking down processes into the lowest level of detail.A DFD consists of a series of bubbles connected by lines. The bubbles represent processes or transformations of data, while the lines represent the flow of data within the system.

DFD Symbols:
- A square/rectangle defines a source or destination of data (external entity).
- An arrow represents the flow of data (data in motion).
- A circle (process) shows transformation of incoming data into outgoing data.
- An open-ended rectangle represents a data store (data at rest or repository).

5.2 Waterfall Model

The Waterfall Model is a linear and sequential software development methodology. It is one of the earliest and most widely used models in software engineering. The model is divided into distinct phases, where each phase depends on the deliverables of the previous one and must be completed before moving to the next.For the Hotel Management System, we are following the Waterfall Model to design and implement the project as a real-life software solution.

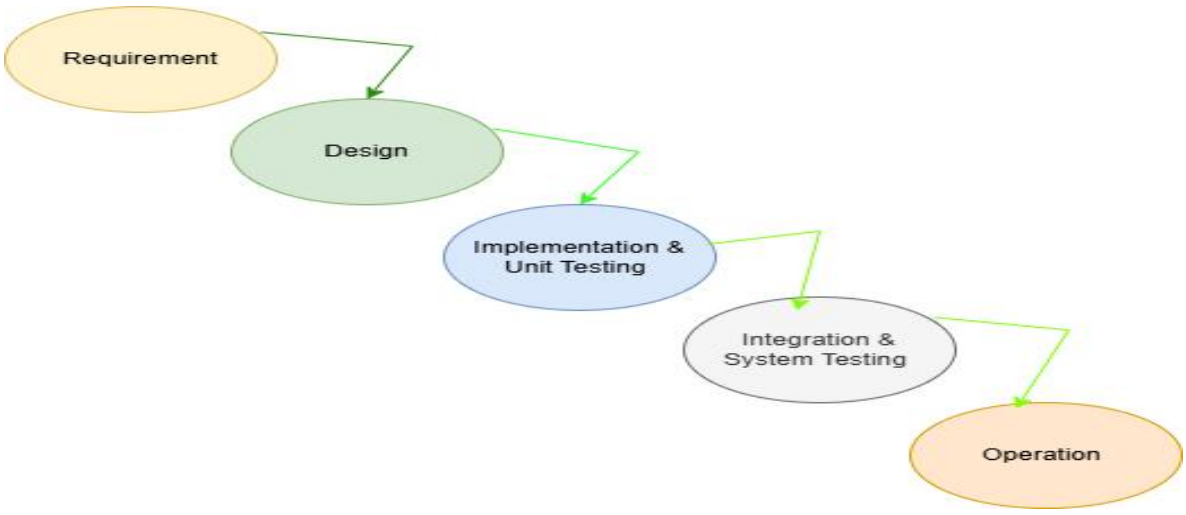


Fig-5.2: Waterfall model

5.3 Data Flow of Hotel Management System

Data flow refers to the movement of data between different components, processes, or modules of the system. It shows how data is input, processed, stored, and output within the Hotel Management System.

In this system:
- Guests can search and book rooms.
- Payments are processed and stored.
- Staff manage check-in, check-out, and complaints.
- Admin manages rooms, staff, and can perform emergency bookings.

This flow is represented through diagrams that show the interaction of users, processes, and data stores.

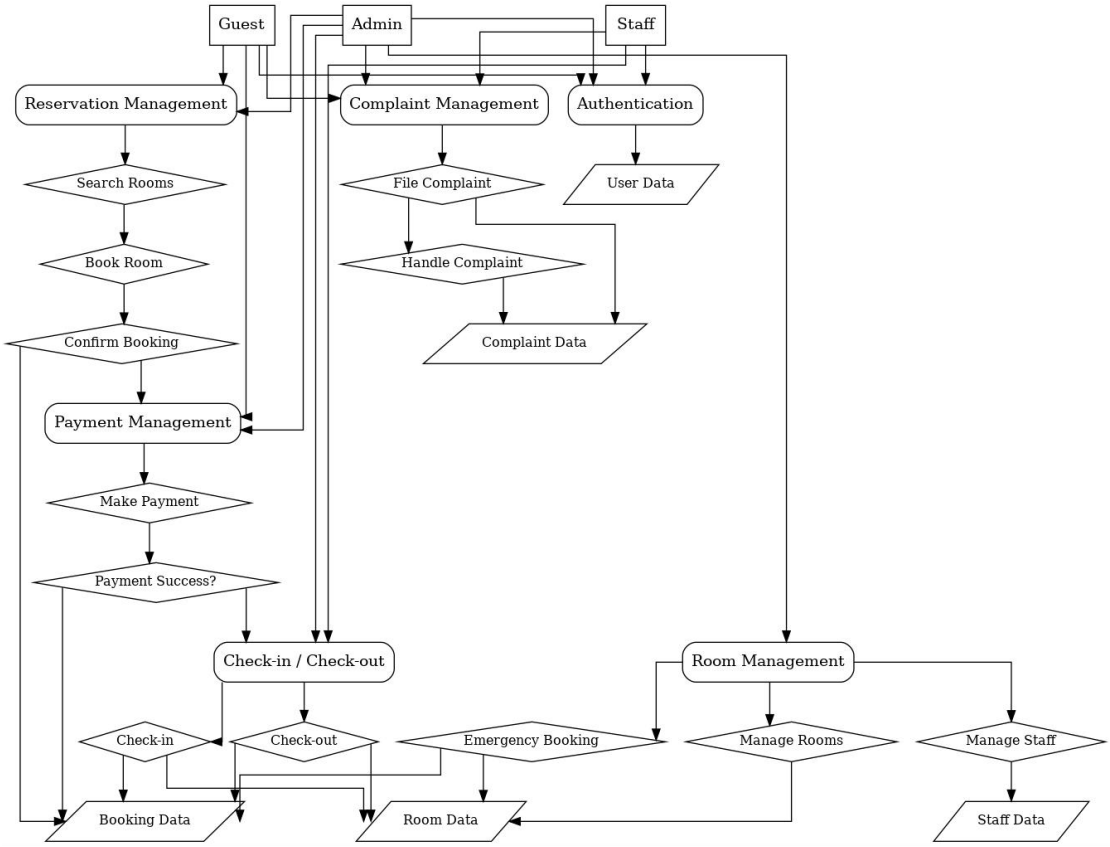
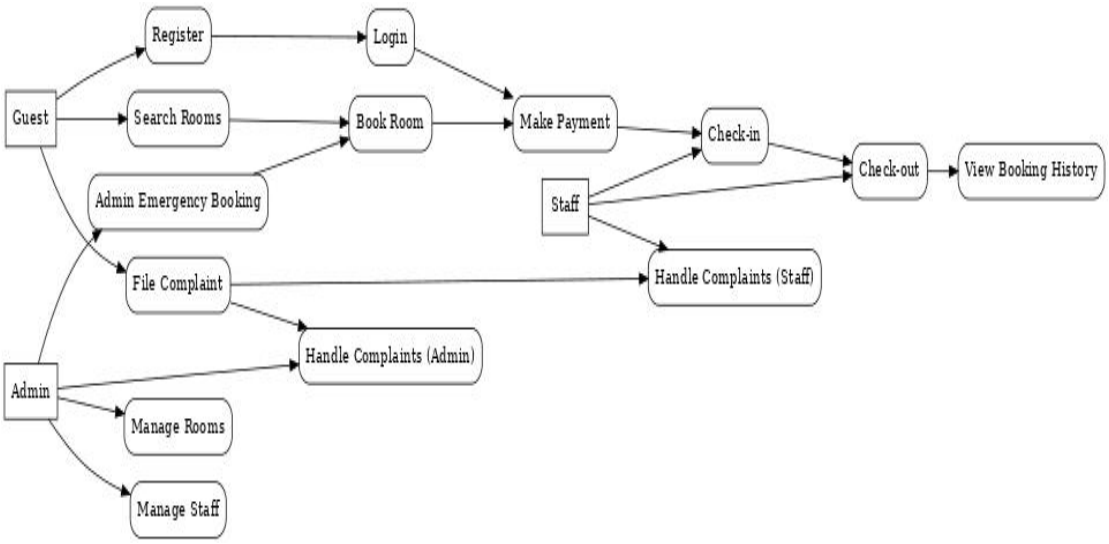


Fig 5.3: Data Flow Diagram

5.4 Use Case Diagram

A Use Case Diagram is a type of behavioral UML diagram that shows the interactions between system actors (Guest, Staff, Admin) and the functionalities of the system.

- Guest: Register, Login, Search Rooms, Book Room, Make Payment, View History, File Complaint.
- Staff: Check-in, Check-out, Handle Complaints.
- Admin: Manage Rooms, Manage Staff, Handle Complaints, Perform Emergency Booking.



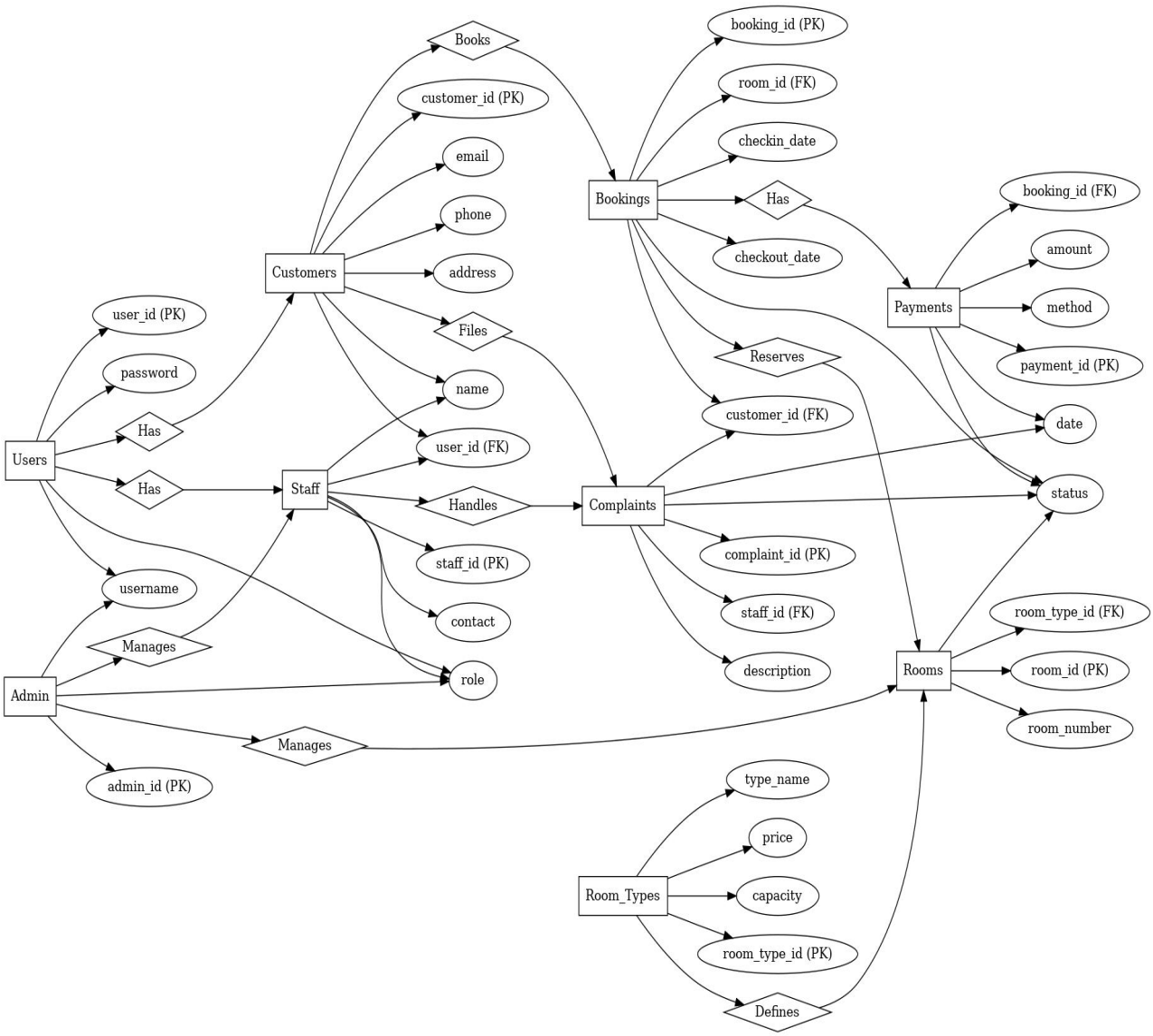
[Fig 5.4: Use Case Diagram]

5.5 Entity Relationship Diagram (ERD)

The Entity Relationship Diagram (ERD) represents the database structure of the Hotel Management System. It shows the entities, their attributes, and the relationships between them.

Key entities include:
1. Customer
2. Room
3. Booking
4. Payment
5. Staff
6. Admin
7. Complaint

In the ERD:
- Entities are represented by rectangles.
- Attributes are represented by ovals.
- Relationships are represented by diamonds.



[Fig 5.5: ER-Diagram of Hotel Management System]

CHAPTER 6

TESTING AND DEBUGGING OF HOSPITAL MANAGEMENT HOSPITAL

6.1 Introduction

The implementation phase of software development translates design specifications into working source code. Once the system is developed, testing and debugging are necessary to ensure that it performs correctly, is reliable, and meets user requirements. Testing helps detect errors, while debugging removes them to make the system stable.

6.2 Testing and Debugging

For the Hotel Management System (developed with PHP, MySQL, HTML, and CSS), different types of testing were performed:

1. Unit Testing – Tested individual modules like room booking, login, and payment to ensure they work correctly.
2. Integration Testing – Verified that modules such as booking and payment work together without issues.
3. Functional Testing – Ensured that the system features (room reservation, check-in, complaints, etc.) meet requirements.
4. Regression Testing – Checked that new updates (like Admin Emergency Booking) did not break existing features.
5. Security Testing – Verified that the system is protected from unauthorized access and common attacks.
6. Performance Testing – Measured how the system behaves with multiple users booking rooms and making payments simultaneously.
7. User Interface Testing – Confirmed that the system is easy to use and responsive across devices.

Test Type	Description	Result
Unit Testing	Testing individual modules (Login, Booking, Payment).	Pass
Integration Testing	Verifying interactions between modules (Booking ↔ Payment).	Pass
Functional Testing	Checking features meet requirements (Reservation, Complaints).	Pass
Regression Testing	Ensuring new updates do not break existing features.	Pass
Security Testing	Testing system protection against unauthorized access.	Pass
Performance Testing	Measuring system performance under concurrent usage.	Pass
User Interface Testing	Ensuring UI is user-friendly and responsive.	Pass

Fig 6.2: Summary of all testing types with results.

6.3 Performance Testing Report

Performance testing was carried out using tools like JMeter to simulate multiple users accessing the system.

- Average concurrent users supported: 200–250 users per 10 seconds.
- Future scalability goal: 500–1000 users per 10 seconds.

The report generated includes:

- Request summary graph (success vs failed requests).
- Error summary table.
- Response time percentiles.
- Active user threads over time.

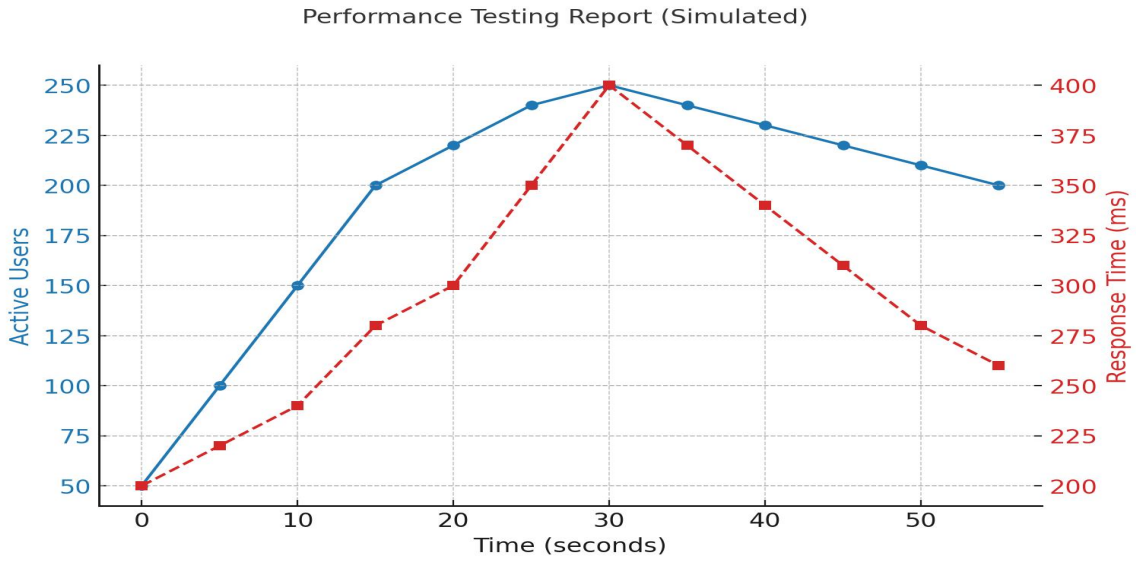


Fig 6.3: **Active Users vs Time** and **Response Time vs Time**.

6.4 Result

The testing results show that the Hotel Management System is functional, stable, and can handle concurrent users effectively. Future improvements will focus on enhancing scalability and security.

CHAPTER 7

PROJECT OVERVIEW

7.1 Introduction

In today's hospitality industry, effective management of hotel operations is essential for customer satisfaction and business growth. This project focuses on the development of a Hotel Management System (HMS) using PHP, MySQL, HTML, and CSS. The system simplifies room booking, customer registration, billing, and administrative control, while also providing features like complaints handling and payment processing. The primary goal is to reduce manual work, improve accuracy, and ensure seamless interaction between guests and hotel staff.

7.2 System Design and Architecture

- **Front-End**:
  - Developed using HTML and CSS for structure and presentation.
  - Provides user-friendly pages for hotel services, room booking, contact, complaints, and customer account management.
  - Responsive design ensures accessibility across desktop and mobile devices.
- **Back-End**:
  - Built using PHP as the server-side scripting language.
  - Handles business logic including bookings, complaints, user authentication, and payment validation.
  - Includes modules for customer, admin, and staff functionalities
- **Database**:
  - MySQL is used as the relational database.
  - Stores information about customers, rooms, bookings, payments, complaints, and staff.
  - Ensures data integrity and supports efficient queries for real-time booking updates.
- **Security**:
  - Implements user authentication (login and registration system).
  - Passwords are hashed before storage.
  - Admin section is separated from the user interface to ensure role-based access.
  - Payment options (e.g., bKash, Nagad, Rocket) are integrated for secure transaction handling.

7.3 Technology Stack

- **HTML & CSS**: For the structure and styling of the web pages, creating an interactive user interface.
- **Bootstrap** (optional): To ensure responsiveness and a modern look for the hospital management system.
- **JavaScript** (optional): To add interactivity to the pages and enhance the user experience (e.g., form validation, dynamic content).

- **MySQL**: A relational database management system (RDBMS) to store hospital records and data related to patients, doctors, appointments, and billing.
- **P**HP: Server-side scripting for dynamic functionality.

7.4 Implementation Strategy

- **Phase 1:** Requirement Gathering & Analysis – Identify hotel requirements, user roles (customer, admin, staff), and key system features.
- **Phase 2:** System Design – Create wireframes, design the database schema, and plan navigation flow.
- **Phase 3:** Front-End Development – Build hotel pages including home, rooms, booking, contact, complaints, and payment interfaces.
- **Phase 4:** Back-End Development – Implement PHP scripts for booking management, login/registration, complaints, and payments.
- **Phase 5:** Integration & Testing – Connect PHP logic with MySQL database and test all workflows.
- **Phase 6:** Deployment & Documentation – Deploy the system on a server and prepare documentation for end users and administrators.

7.5 Overview

User Side

1. Home page : Introduction to hotel services.

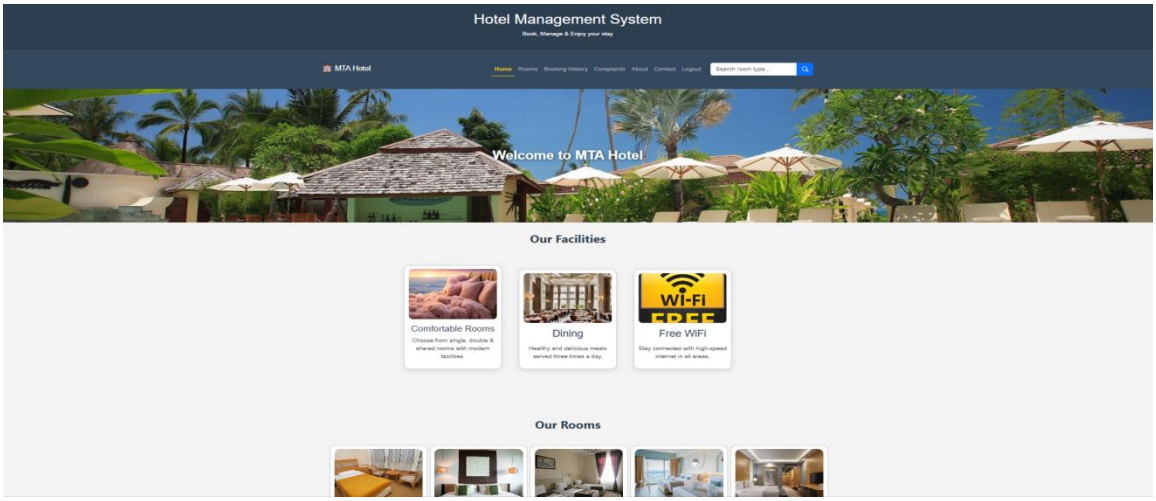


Fig 7.1: Home Page

2.Rooms & Room Rates

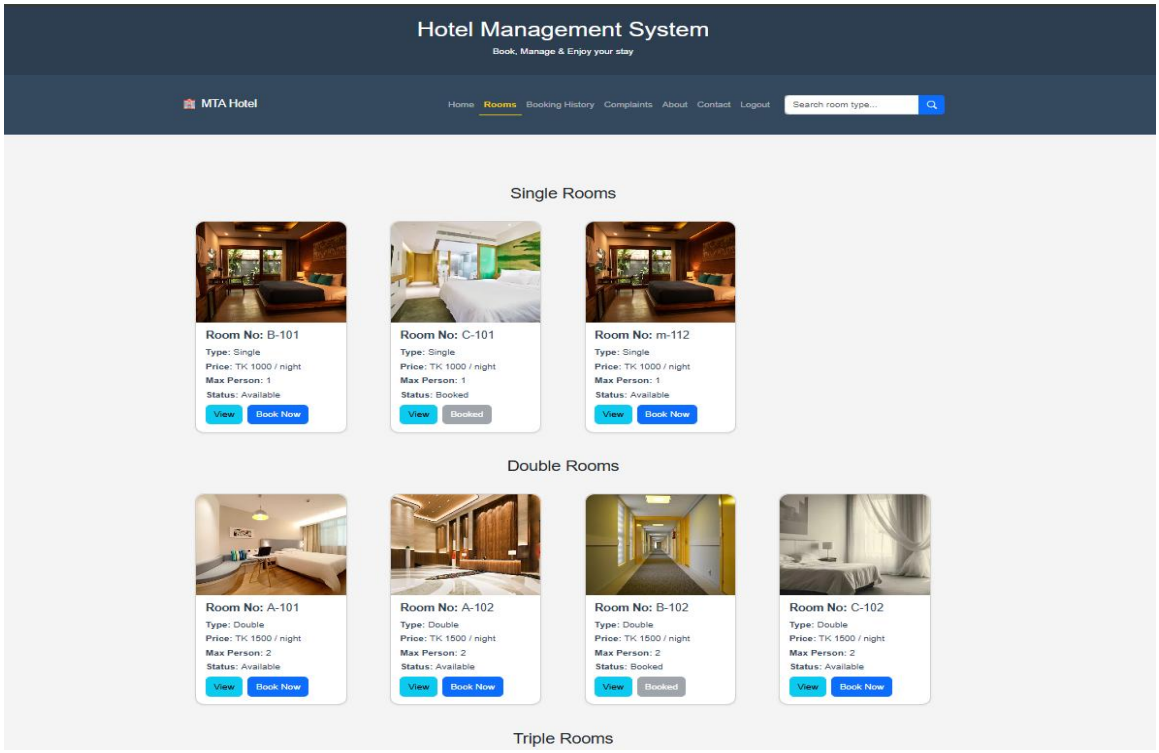


Fig 7.2: Room page Section wise

3.Room Booking

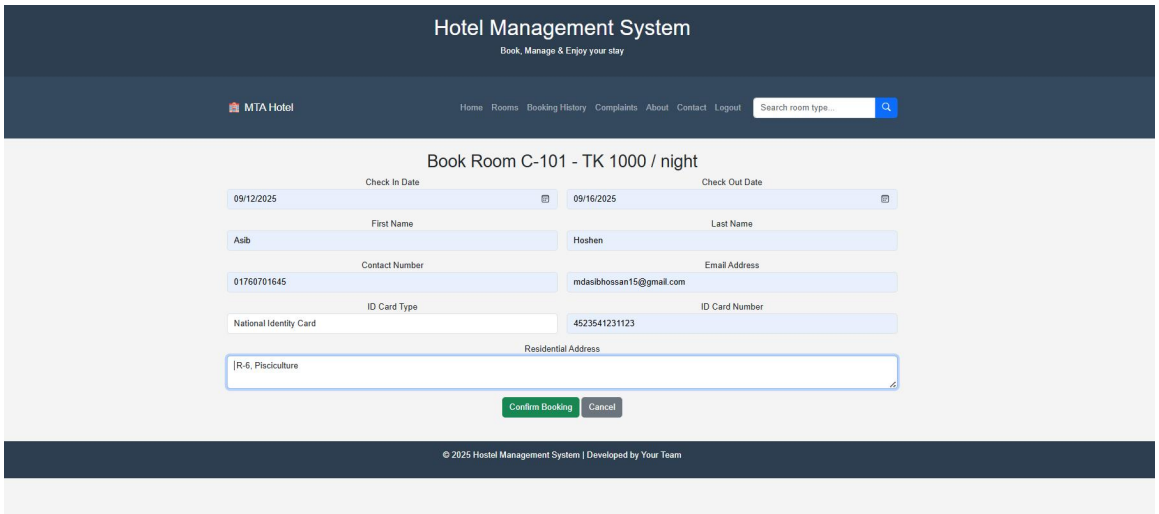


Fig 7.3: Room Bookin

4.Booking History

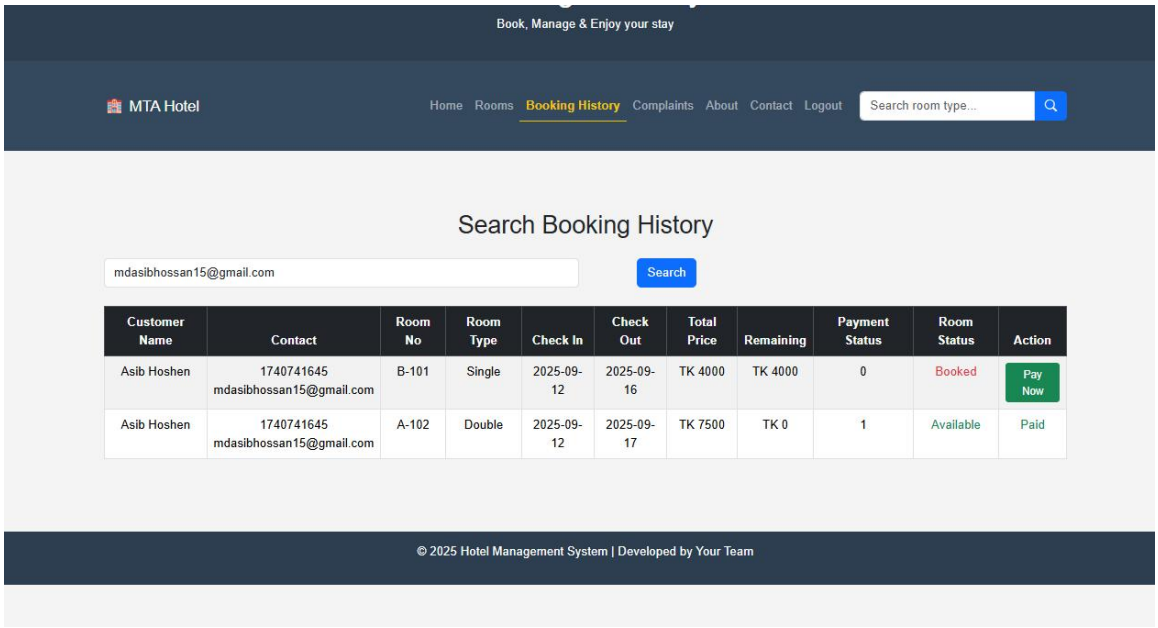


Fig 7.4: Booking History

5.Complaints Page

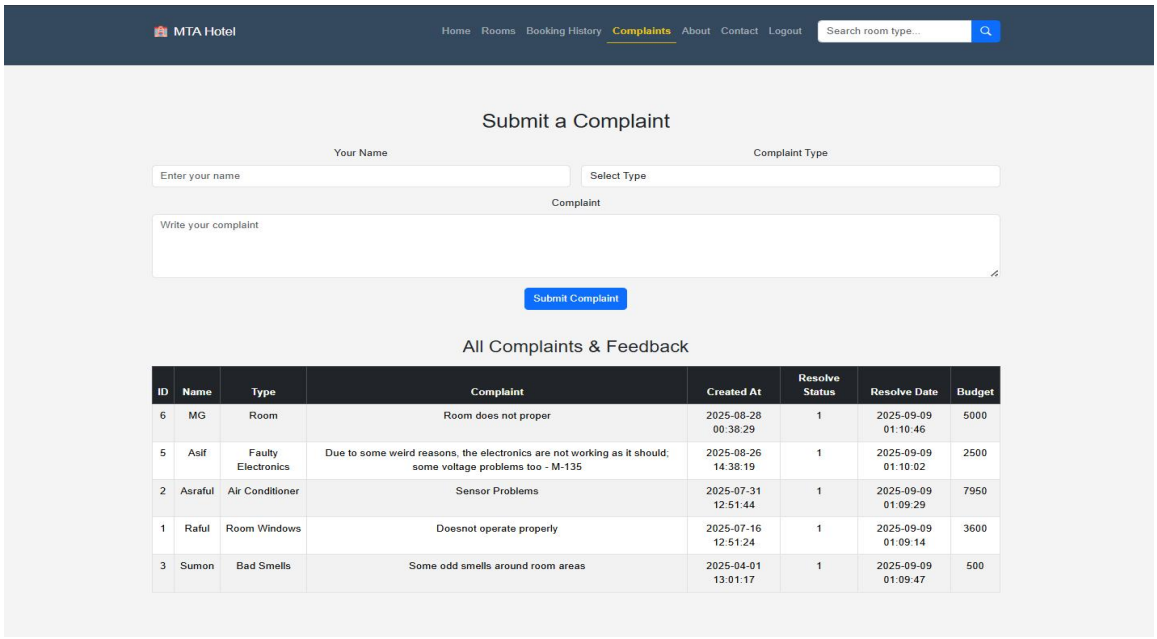


Fig 7.5: Complaints Page

6. Payment Page

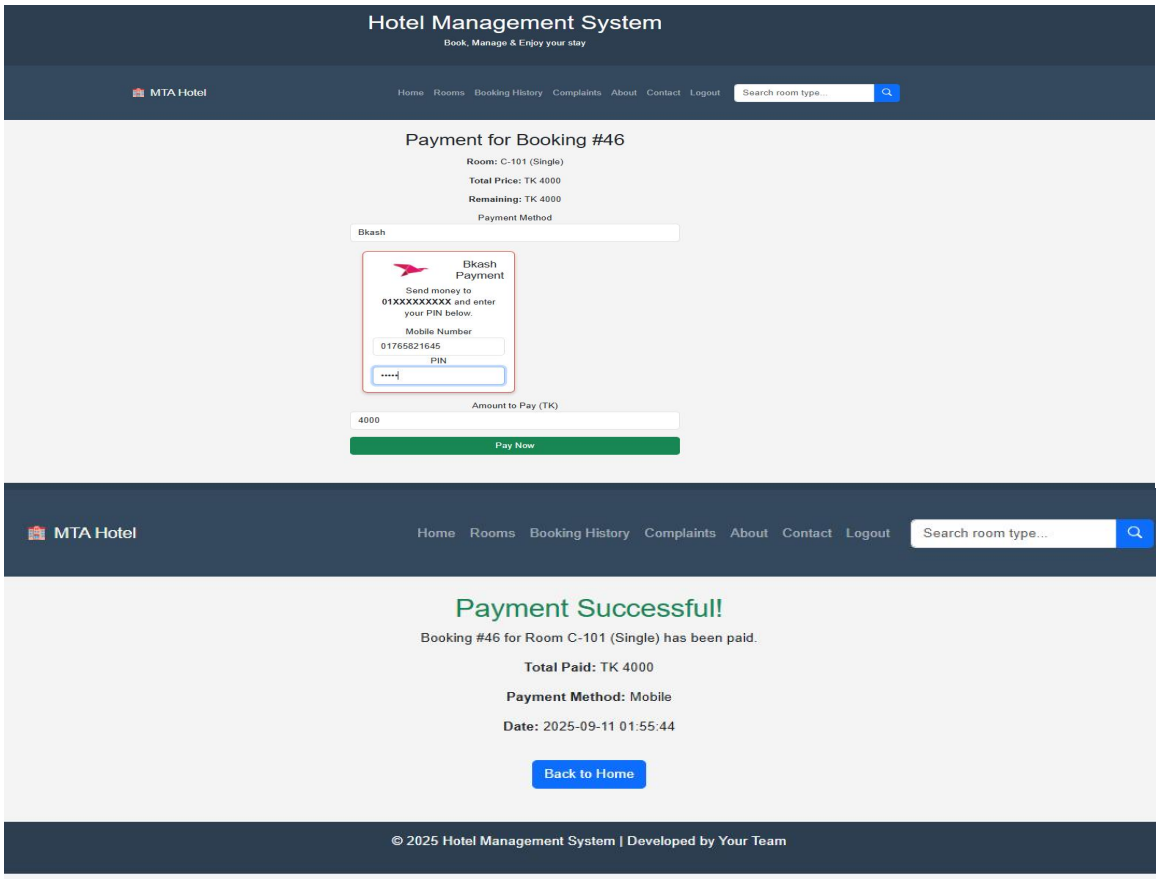


Fig 7.6:Payment Page

7. About Us Page

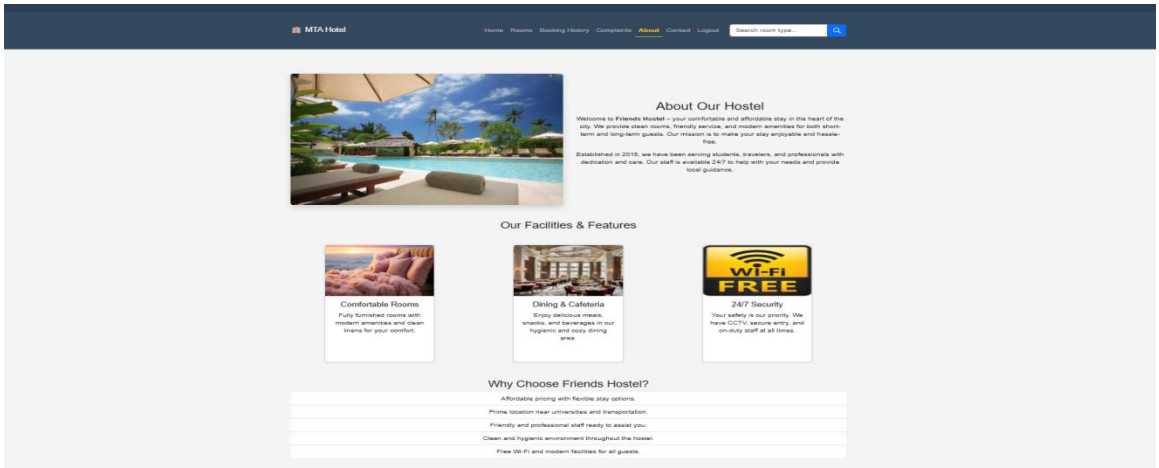


Fig 7.7: About page

8. Contact

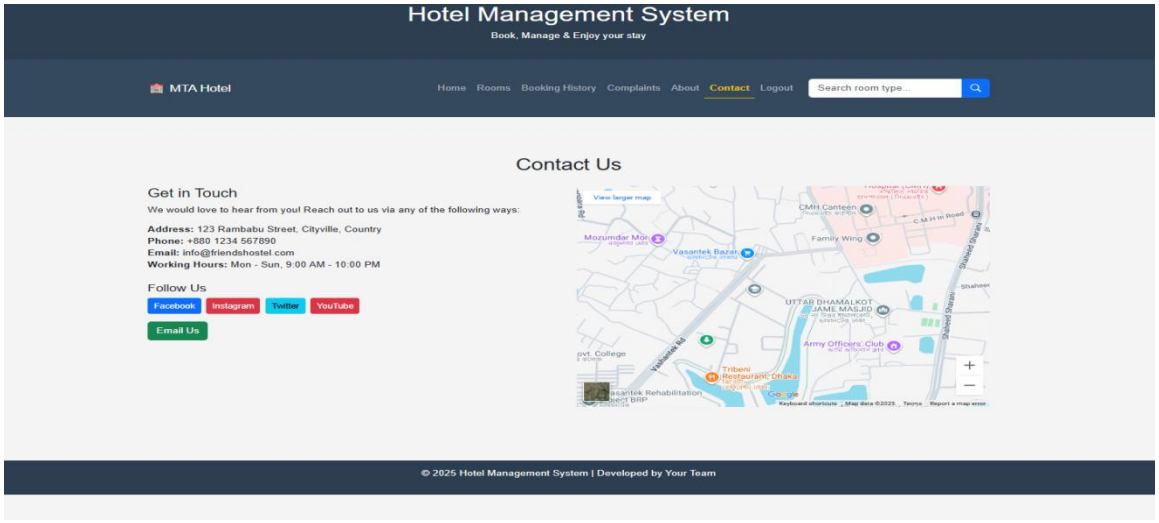


Fig 7.8: Contact page

Admin Side

9. Admin Login Page

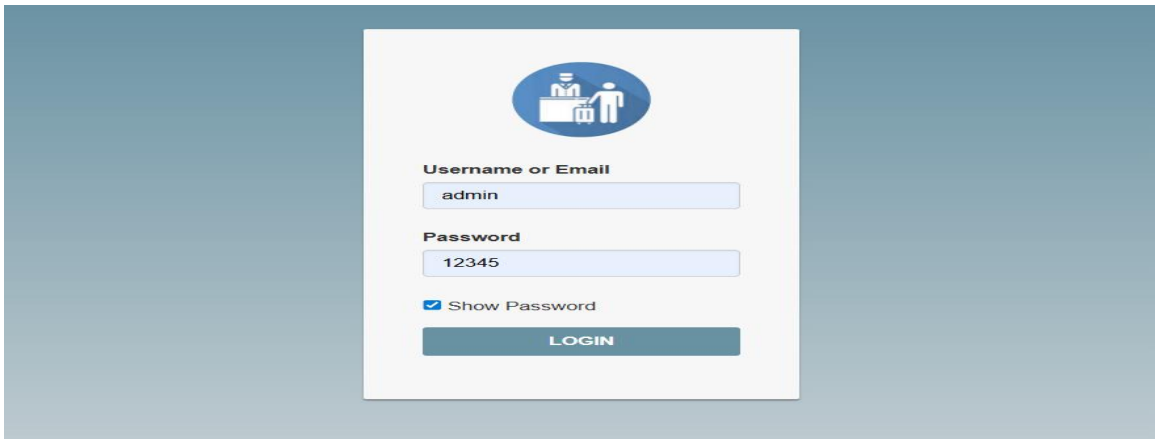


Fig 7.9: Admin Login Page

10. Admin Panel Dashboard

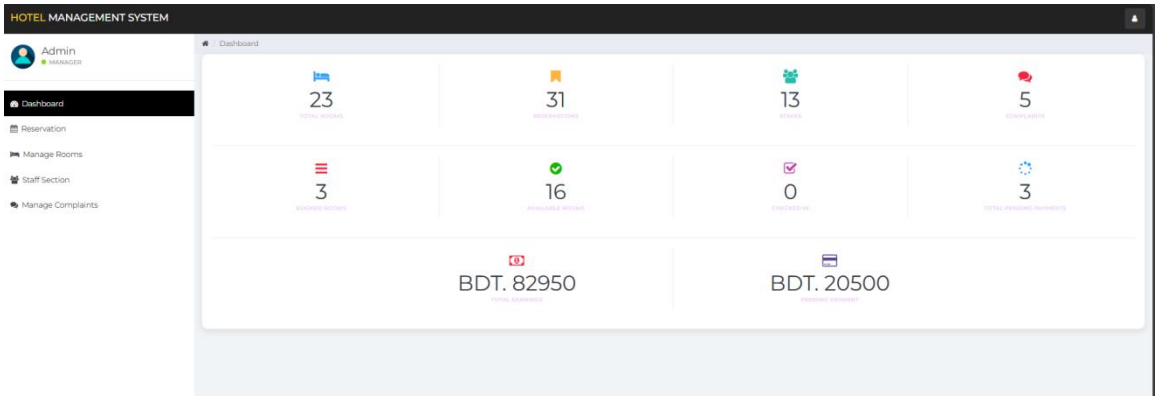


Fig 7.10: Admin Panel Dashboard

10. Reservation Management

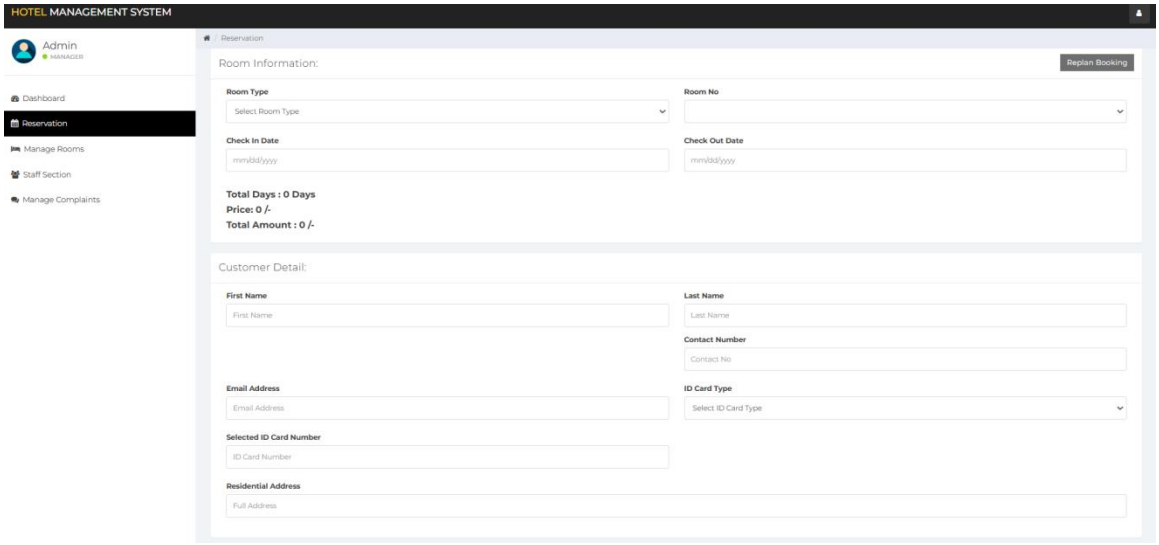


Fig 7.11: Reservation Management page

12. Room Management

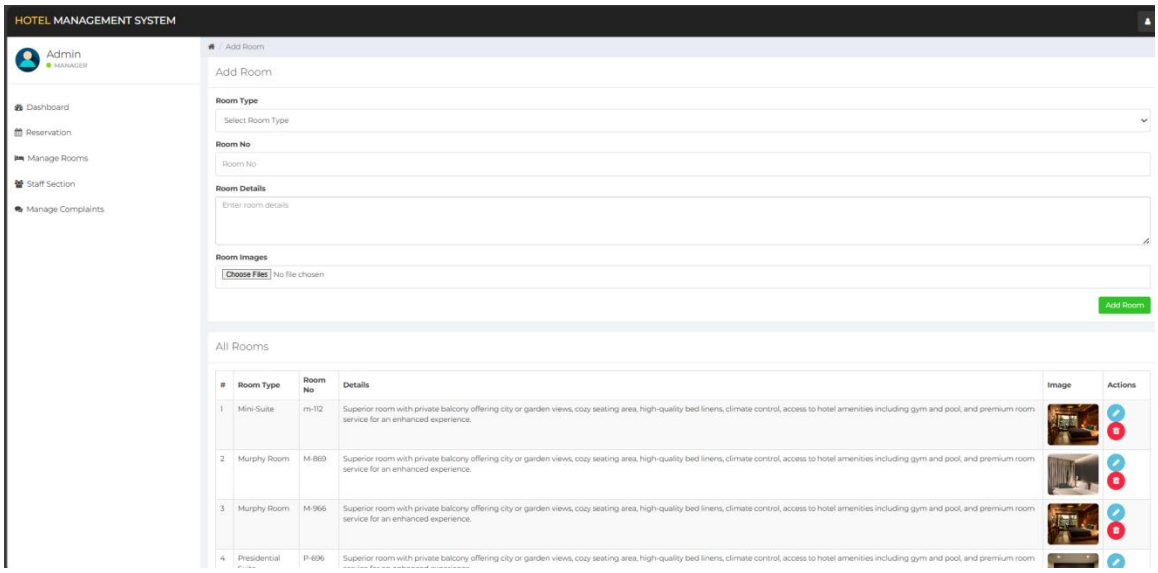


Fig 7.12: Room Management page

13. Booking Management

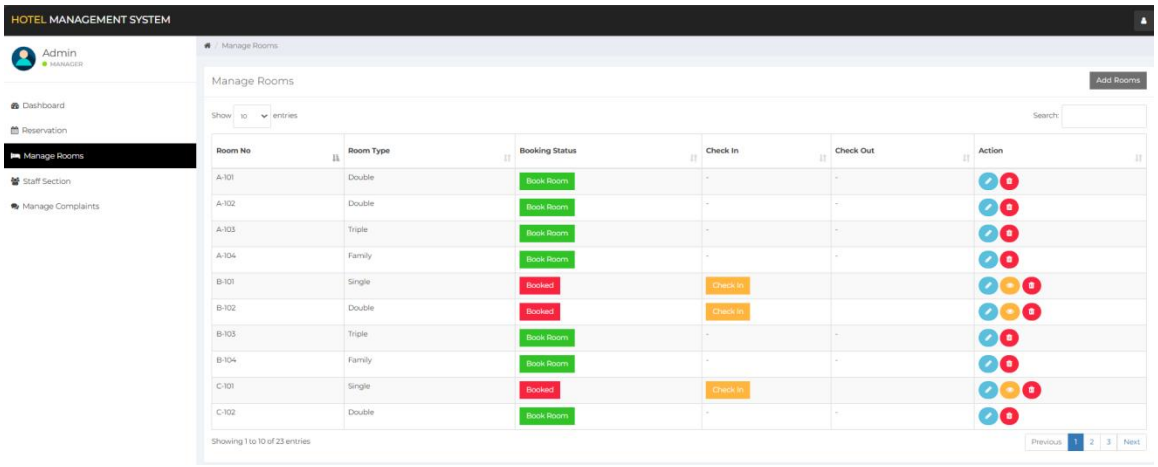


Fig 7.13: Booking Management page

14.Complaint Management

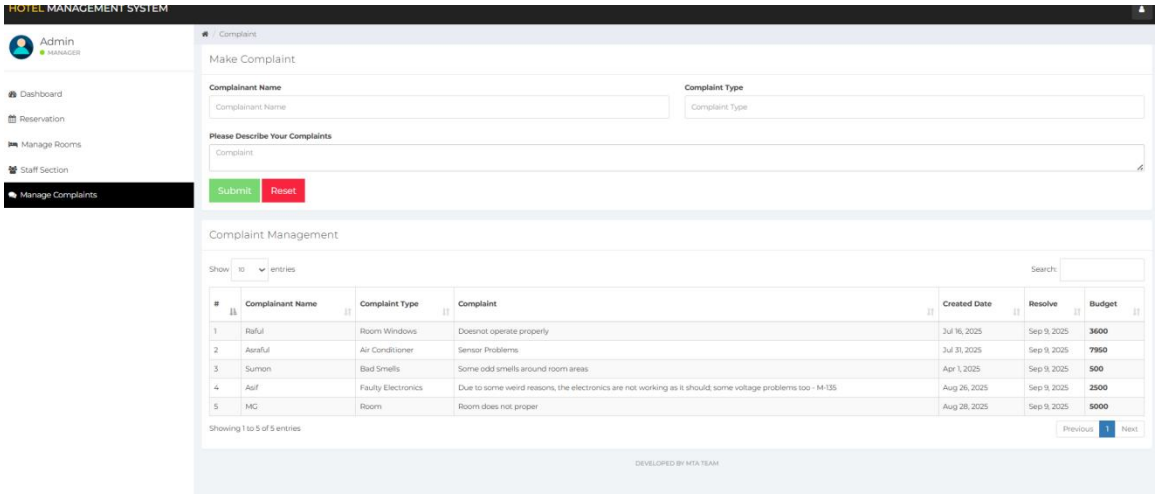


Fig 7.14: Complaint Management page

15. Staff Management

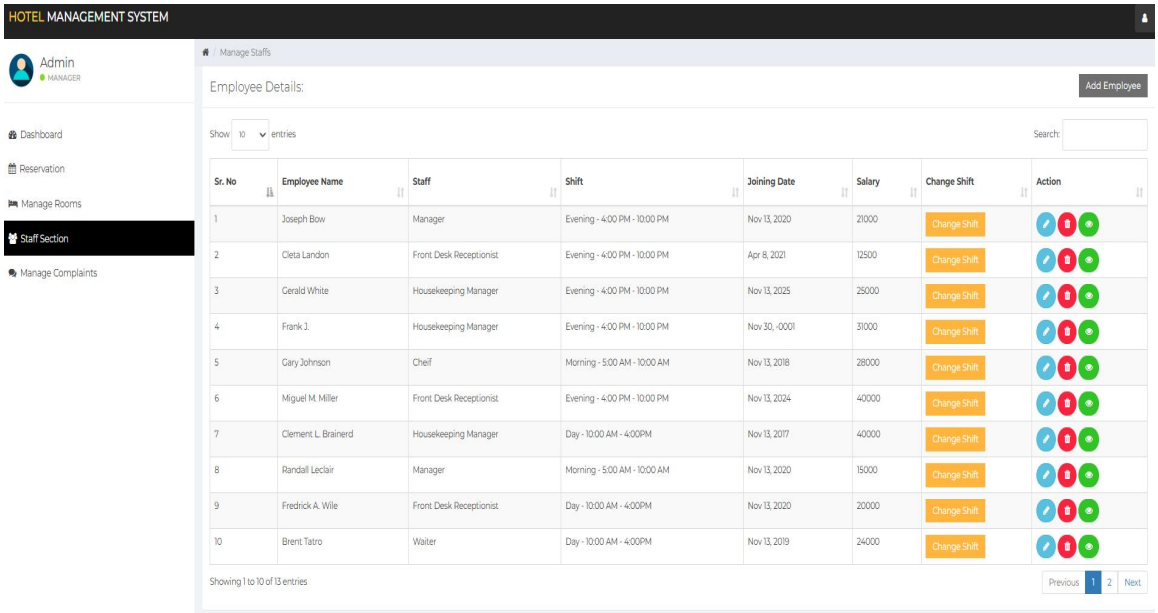


Fig 7.15: Staff Management page

CHAPTER 8

FUTURE SCOPE AND CONCLUSION

8.1 Limitation OF Project

Although the Hotel Management System offers significant improvements in automating booking, payments, staff, and room management, there are some limitations observed during development:

- Limited Online Payment Integration: The current system supports basic payments but lacks integration with multiple secure gateways.
- No Mobile Application: The system is web-based and does not include Android/iOS apps for hotel guests.
- Lack of AI Features: No AI-based recommendation system for room suggestions or customer service is included.

8.2 Future Scope OF The Project

1. Integration with Advanced Payment Systems

Adding multiple secure online payment gateways (PayPal, Stripe, Mobile Banking) for smooth transactions.

2. Mobile Application Support

Developing Android and iOS apps so that guests can book rooms, make payments, and check booking history from mobile devices.

3. Artificial Intelligence and Chatbots

AI-powered chatbots can be introduced for handling guest queries and complaints.
Recommendation engines can suggest rooms or services based on user history.

4. Cloud Deployment

Hosting the Hotel Management System on cloud servers for better scalability and real-time synchronization.

5. IoT and Smart Hotel Features

Integration with IoT devices (smart keys, automated check-in/check-out, room temperature controls) for an enhanced guest experience.

6. Multi-Hotel Support

Expanding the system to manage multiple hotels under the same chain, with centralized reporting and administration.

8.3 Challenges in Implementation

During development, the following challenges were encountered:

Integration of Modules: Connecting booking, payments, staff, and complaints modules required careful design to ensure smooth data flow.

Security Concerns: Ensuring safe handling of guest data and preventing unauthorized access was a major priority.

Database Management: Designing a scalable database schema to handle large volumes of bookings and guest information.

User Experience: Balancing between staff/admin functionality and guest usability in a single system required detailed planning.

8.4 Conclusion

The Hotel Management System (HMS) developed is a web-based application designed to automate essential hotel operations such as room booking, check-in/check-out, payments, staff management, and complaint handling.

The system successfully integrates:

Guest Features: Registration, room search, booking, payments, and viewing booking history.

Staff Features: Managing check-in, check-out, and handling complaints.

Admin Features: Managing rooms, staff, complaints, and emergency bookings.

Achievements:

- Improved efficiency in booking and payment workflows.
- Role-based access for Admin, Staff, and Guests with secure login.
- Automation of manual processes, reducing human error.
- Easy scalability for future improvements such as analytics, AI, and mobile apps.

Final Thoughts:

This project provided valuable experience in building a real-world full-stack system using PHP, MySQL, HTML, CSS, and JavaScript. It demonstrated how hotel operations can be streamlined with technology. With further enhancements such as mobile support, AI features, and IoT integration, the system can evolve into a complete smart hotel management solution.

REFERENCES

[1] Marijn Haverbeke, (2019). “Eloquent JavaScript: A Modern Introduction to Programming”(3rd edition)

[2] Ben Frain, (2021). “Responsive Web Design with HTML5 and CSS4” (1st edition).

[3] Mr. Daniel Foreman, Mr. Daniel Charles Foreman, (May 6, 2021). “Bootstrap 5 Foundations: The Random Knowledge Enthusiast” (1st edition).

[4] Mr. Daniel Foreman, Mr. Daniel Charles Foreman, (September 24, 2010). “Beginning PHP
and MySQL : From Novice to Professional” (4th edition).

[5] Letkowski, J., 2015. Doing database design with MySQL. Journal of Technology Research, 6, p.1.

[6] Bhattacharyya, D.K., 2006. Research methodology. Excel Books India.

[7] O’Fallon, M.J. & Rutherford, D.G., (2010). Hotel Management and Operations (5th edition). John Wiley & Sons.

[8] Walker, J.R., (2021). Introduction to Hospitality (8th edition). Pearson Education

[9] Kasavana, M.L. & Cahill, J.J., (2007). Managing Technology in the Hospitality Industry (6th edition). American Hotel & Lodging Educational Institute.

[10] Laudon, K.C. & Laudon, J.P., (2022). Management Information Systems: Managing the Digital Firm (17th edition). Pearson Education.

[11] Law, R. & Jogaratnam, G., (2005). A study of hotel information technology applications. International Journal of Contemporary Hospitality Management, 17(2), pp.170–180.

[12] Chathoth, P.K. & Olsen, M.D., (2003). Information technology in hospitality organizations: A strategic management perspective. Tourism and Hospitality Research, 4(1), pp.73–93.