

Design and Implementation of Web-Based Hotel Management System

by

MD. Tarek Hossain

ID: CSE2202026146

MD. Rohol Amin

ID: CSE2202026092

Supervised by

Tanjil Mahmud

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)**

May 2026

APPROVAL

The project titled “**Design and Implementation of Web-Based Hotel Management System**” submitted by Md. Tarek Hossain (CSE2202026146), Md. Rohol Amin (CSE2202026092) 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 and approved as to its style and contents.

Board of Examiners

Tanjil Mahmud

Lecturer and Assistant Coordinator

Department of Computer Science and Engineering

Sonargaon University (SU)

Supervisor

(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 **Tanjil Mahmud, Assistant Coordinator and 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

Signature

(Tanjil Mahmud)
Supervisor

Md. Tarek Hossain
ID: CSE2202026146

Md. Rohol Amin
ID: CSE2202026092

ABSTRACT

The Web-Based Hotel Management System is a software application developed using Python to streamline and automate essential hotel operations. This system provides an integrated platform for managing room reservations, customer information, check-in and check-out procedures, and billing processes in an efficient and organized manner. The primary objective of the project is to reduce manual workload, minimize human errors, and improve overall operational efficiency. The system incorporates a structured database to ensure secure storage, quick retrieval, and accurate management of data. It features a user-friendly interface that enables administrators to perform tasks easily without requiring advanced technical knowledge. By digitizing routine activities, the system enhances service quality and ensures better customer satisfaction. This project highlights the importance of adopting modern technological solutions in the hospitality industry. Future enhancements may include online booking capabilities, payment gateway integration, and real-time notification systems to further improve usability and performance.

ACKNOWLEDGMENT

At the very beginning, we would like to express my deepest gratitude to the Almighty Allah for giving us the ability and the strength to finish the task successfully within the schedule time.

We are auspicious that we had the kind association as well as supervision of **Tanjil Mahmud**, Assistant Coordinator and Lecturer, Department of Computer Science and Engineering, Sonargaon University whose hearted and valuable support with best concern and direction acted as necessary recourse to carry out our project.

We are also thankful to all our teachers during our whole education, for exposing us to the beauty of learning.

Finally, our deepest gratitude and love to my parents for their support, encouragement, and endless love.

LIST OF ABBREVIATIONS

API	Application Program Interface
CPU	Central Processing Unit
CRUD	Create, Read, Update, Delete
CSS	Cascading Style Sheets
DBMS	Database Management System
DOB	Date of Birth
GUI	Graphical User Interface
HMS	Hotel Management System
HTML	Hyper Text Markup Language
IDE	Integrated Development Environment
ID	Identification
IP	Internet Protocol
MYSQL	My Structured Query Language
NID	National Identity Card
OOP	Object-Oriented Programming
OS	Operating System
OTP	One-Time Password
PDF	Portable Document Format
RAM	Random Access Memory
REF	Reference Number
SQL	Structured Query Language
TCP	Transmission Control Protocol
TK	Taka
UI	User Interface
URL	Uniform Resource Locator
UX	User Experience

TABLE OF CONTENTS

Title	Page No.
DECLARATION	iii
ABSTRACT	iv
ACKNOWLEDGEMENT	v
LIST OF ABBREVIATION	vi
CHAPTER 1	1 – 2
INTRODUCTION AND OBJECTIVE	
1.1 Introduction	1
1.2 Objectives	1
1.3 Related Works	1
1.4 Problem Statement	2
1.5 Functionalities	2
CHAPTER 2	3 – 5
REQUIREMENT AND ANALYSIS	
2.1 Introduction	3
2.2 Software Requirement Specification	3
2.2.1 Functional Requirements.....	3
2.2.2 Non-Functional Requirements.....	3
2.3 Hardware Requirement	3
2.4 Software Requirement	3
2.5 Justification of Selection Technology	4
2.6 Data Flow Diagram (DFD)	5
CHAPTER 3	6 – 10
SYSTEM DESIGN	
3.1 Introduction	6
3.2 Entity Relationship (ER) Diagram	6
3.3 Data Dictionary	6 – 8
3.4 Database Normalization	8 – 9
3.5 System Architecture	9 – 10

CHAPTER 4	11 – 12
TESTING	
4.1 Introduction	11
4.2 Objectives of Testing	11
4.3 Types of Testing used	11
4.3.1 Unit Testing	11
4.3.2 Integration Testing	11
4.3.3 System Testing	11
4.3.4 User Acceptance Testing	11
4.4 Test Cases	11 – 12
4.5 Testing Tools	12
4.6 Result of Testing	12
CHAPTER 5	13 – 18
SYSTEM IMPLEMENTATION	
5.1 Output Screen	13
5.1.1 Login Module	13
5.1.2 Registration Module.....	13–14
5.1.3 Main Dashboard Room.....	14
5.1.4 Customer Details Screen	15
5.1.5 Room Details Screen.....	15–16
5.1.6 Room Booking Screen	16–17
5.1.7 Billing System	17
5.1.8 Forget Password	17–18
5.2 Database Integration	18
5.3 Conclusion	18
CHAPTER 6	39
CONCLUSION AND FUTURE WORKS	
6.1 Conclusion	19
6.2 Limitation of System	19
6.3 Future Works	19
REFERENCES	20

LIST OF TABLES

<u>Table No.</u>	<u>Title</u>	<u>Page No.</u>
Table 3.1	Data Dictionary	5-8
Table 4.1	Test Cases	11-12

LIST OF FIGURES

<u>Figure No.</u>	<u>Title</u>	<u>Page No.</u>
Fig.2.1	DFD Level 0	4
Fig.2.2	DFD Level 1	5
Fig.2.3	DFD Level 2	5
Fig.3.1	ER Diagram	6
Fig.3.2	Data Normalization	8
Fig.5.1	Login	13
Fig.5.2	Registration	14
Fig.5.3	Main Dashboard	14
Fig.5.4	Customer Details	15
Fig.5.5	Room Details Screen	16
Fig.5.6	Room Booking Screen	16
Fig.5.7	Forget Password	17

CHAPTER 1

INTRODUCTION AND OBJECTIVE

1.1 Introduction

Design And Implementation Of Web-Based Hotel Management System is an integrated software solution designed to manage and automate the core operations of a hotel, including reservations, customer management, billing, and room allocation. With the rapid growth of the hospitality industry, the need for efficient and reliable management systems has become increasingly important. Traditional manual systems are often inefficient, time-consuming, and prone to human error, which can negatively impact customer satisfaction and operational efficiency [1].

The advancement of computer technology has enabled the development of automated systems that streamline hotel operations and improve service quality. A computerized HMS provides real-time access to information, reduces redundancy, and enhances decision-making capabilities. Such systems also ensure better data security and accuracy compared to manual methods [2].

This project focuses on the design and implementation of a Hotel Management System using the Python programming language. Python is widely recognized for its simplicity, readability, and extensive library support, making it an ideal choice for developing scalable and maintainable applications. The proposed system aims to provide a user-friendly interface for managing hotel activities efficiently while minimizing manual intervention[5].

The developed system includes essential features such as room booking, customer record management, billing, and administrative control. By leveraging Python, the system ensures flexibility, ease of maintenance, and cost-effectiveness, making it suitable for small to medium-sized hotels.

1.2 Objectives

The main objectives of this project are:

- To design and develop a hotel management system using Python.
- To automate room booking and reservation processes.
- To maintain organized and accurate customer records.
- To reduce manual errors and workload.
- To generate bills and manage transactions efficiently.
- To provide a simple and user-friendly interface.
- To improve overall hotel service and operational efficiency [3].

1.3 Related Works

Various hotel management systems have been developed using technologies such as Java, PHP, and .NET frameworks. These systems often include advanced features like online booking, payment integration, and cloud-based data storage. However, many of these solutions are complex and costly, making them less suitable for small-scale hotels [4].

Previous research and implementations have demonstrated that Python can be effectively used to develop lightweight and efficient management systems. Python-based applications are easier to develop and maintain due to their simple syntax and modular structure [2]. This project builds upon these approaches by providing a simplified hotel management system that meets essential operational requirements while remaining cost-effective and user-friendly.

1.4 Problem Statement

Manual hotel management systems face several challenges, including data inconsistency, difficulty in record maintenance, and increased chances of human error. Tasks such as room allocation, billing, and customer record management become time-consuming and inefficient when handled manually.

Existing computerized systems may be expensive, complex, and require specialized training, which is not feasible for small and medium-sized hotels. Therefore, there is a need for a simple, reliable, and cost-effective solution. This project addresses these issues by developing a Python-based hotel management system that automates operations and ensures efficient data handling [1], [4].

1.5 Functionalities

The proposed Web-based Hotel Management System provides the following functionalities:

- **Room Management:** Add, update, and view room details.
- **Booking System:** Manage room reservations, check-in, and check-out processes.
- **Customer Management:** Store and retrieve customer information efficiently.
- **Billing System:** Generate invoices and maintain transaction records.
- **Data Management:** Maintain and organize hotel records securely.
- **Admin Control:** Allow administrators to monitor and control system activities.

These functionalities ensure smooth hotel operations and improve service quality while reducing manual workload [3].

CHAPTER 2

REQUIREMENT AND ANALYSIS

2.1 Introduction

This chapter presents the detailed requirements and analysis of the proposed Hotel Management System. It explains the system requirements, software requirement specification (SRS), and justification for selecting Python as the development technology. Proper analysis ensures that the system is efficient, reliable, and capable of meeting user needs.

2.2 Software Requirement Specification (SRS)

The Software Requirement Specification (SRS) describes the functional and non-functional requirements of the system.

2.2.1 Functional Requirements

The system should provide the following functionalities:

- Customer registration and data management
- Room booking and reservation system
- Check-in and check-out management
- Room availability tracking
- Billing and invoice generation
- Admin control panel for managing system operations
- Data storage and retrieval

2.2.2 Non-Functional Requirements

Usability: The system must be user-friendly and easy to operate

Performance: Fast response time for all operations

Reliability: System should run without failure

Security: Protect user and hotel data

Maintainability: Easy to update and modify

Scalability: Ability to handle increasing data

2.3 Hardware Requirements

The system does not require high-end hardware and can run on standard computers:

- Processor: Intel Core i3 or above
- RAM: Minimum 4 GB
- Hard Disk: At least 500 MB free space
- Monitor, Keyboard, Mouse

2.4 Software Requirements

The system is developed using Python and requires the following:

- **Operating System:** Windows / Linux / macOS
- **Programming Language:** Python
- **IDE/Editor:** Visual Studio Code / PyCharm
- **Database:** MySQL or SQLite
- **Libraries/Frameworks:**

- Tkinter (for GUI) OR
- Flask (if web-based system)

2.5 Justification of Selection Technology

Python has been selected as the main development language for this project due to the following reasons:

- **Simplicity:** Easy to learn and write code
- **Rapid Development:** Faster development compared to other languages
- **Extensive Libraries:** Supports GUI (Tkinter) and web development (Flask)
- **Open Source:** No licensing cost
- **Cross-Platform:** Runs on multiple operating systems
- **Maintainability:** Easy to update and debug

Compared to other technologies like Java or .NET, Python requires less code and is more suitable for small to medium-scale applications like this project. Therefore, it is an ideal choice for developing a Hotel Management System.

2.6 Data Flow Diagram (DFD)

Context Level DFD (Level 0)

- **Customer** → makes bookings, payments
- **Admin** → manages rooms, staff, services
- System processes everything and returns confirmation/bills

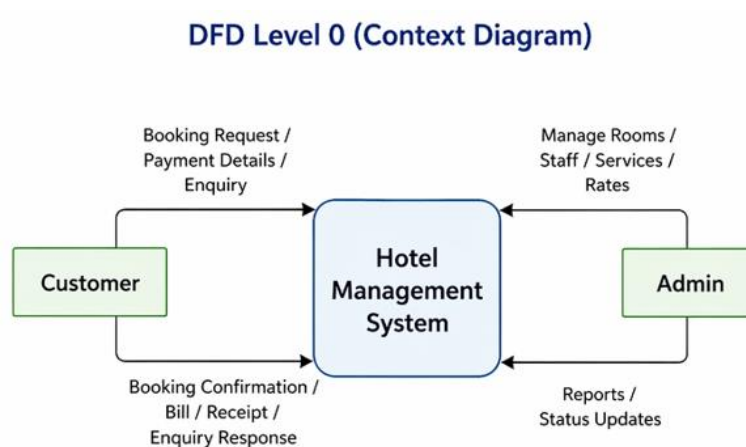


Fig 2.1: DFD level 0

Level 1 (DFD)

Data Stores:

- D1: Customer Database
- D2: Room Database
- D3: Booking Records
- D4: Payment Records

Flow Explanation:

1. Room Booking (1.0)

- Input: Customer details
- Output: Booking info stored in Booking Records

2. Payment System (2.0)

- Handles billing and transactions
- Stores data in Payment Records

3. Check-in/Check-out (3.0)

- Updates room availability
- Interacts with Room Database

4. Admin Control (4.0)

- Manages rooms, staff, and reports

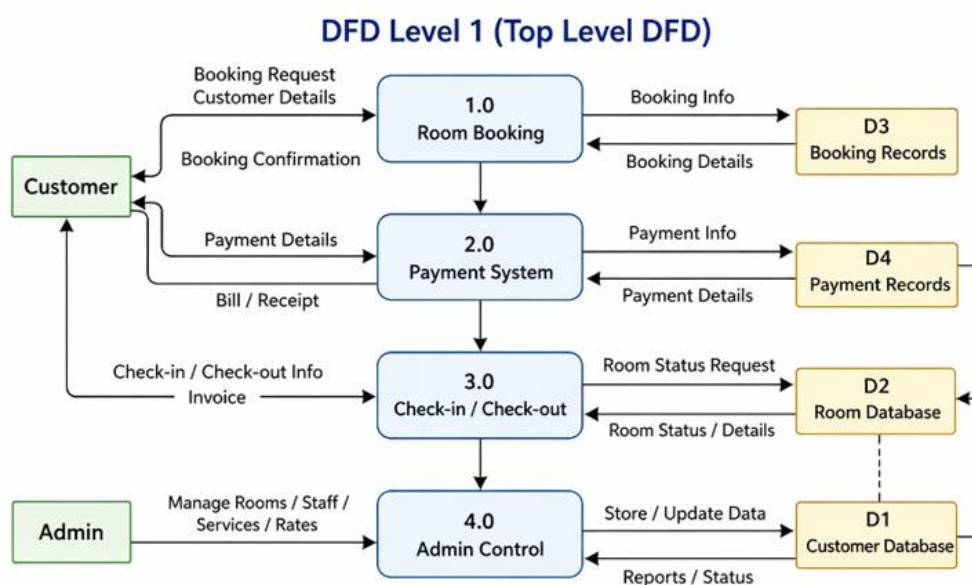


Fig.2.2: DFD Level 1

Level 2: DFD (Detailed – Booking Process)

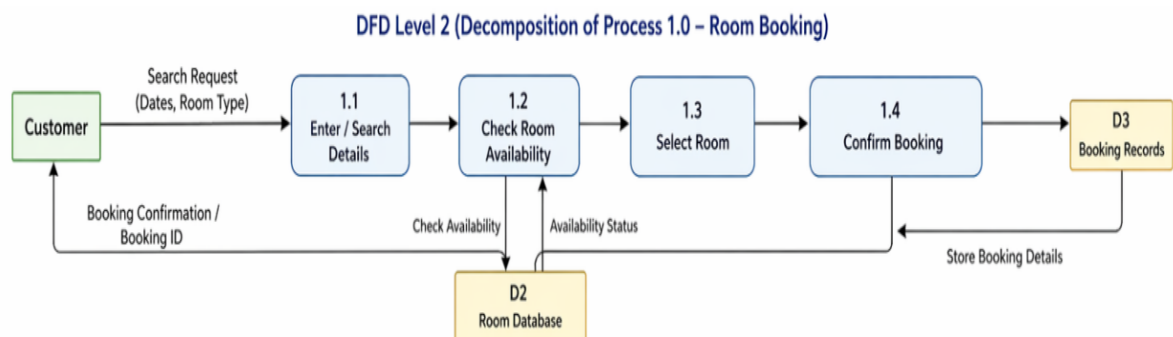


Fig.2.3: DFD Level 2

CHAPTER 3

SYSTEM DESIGN

3.1 Introduction

System design defines the overall structure, architecture, and data organization of the Hotel Management System. It transforms requirements into a blueprint for implementation. This chapter includes **ER diagram, data dictionary, normalization, and system architecture.**

3.2 Entity Relationship (ER) Diagram

Main Entities:

- Customer
- Room
- Booking
- Payment
- Admin

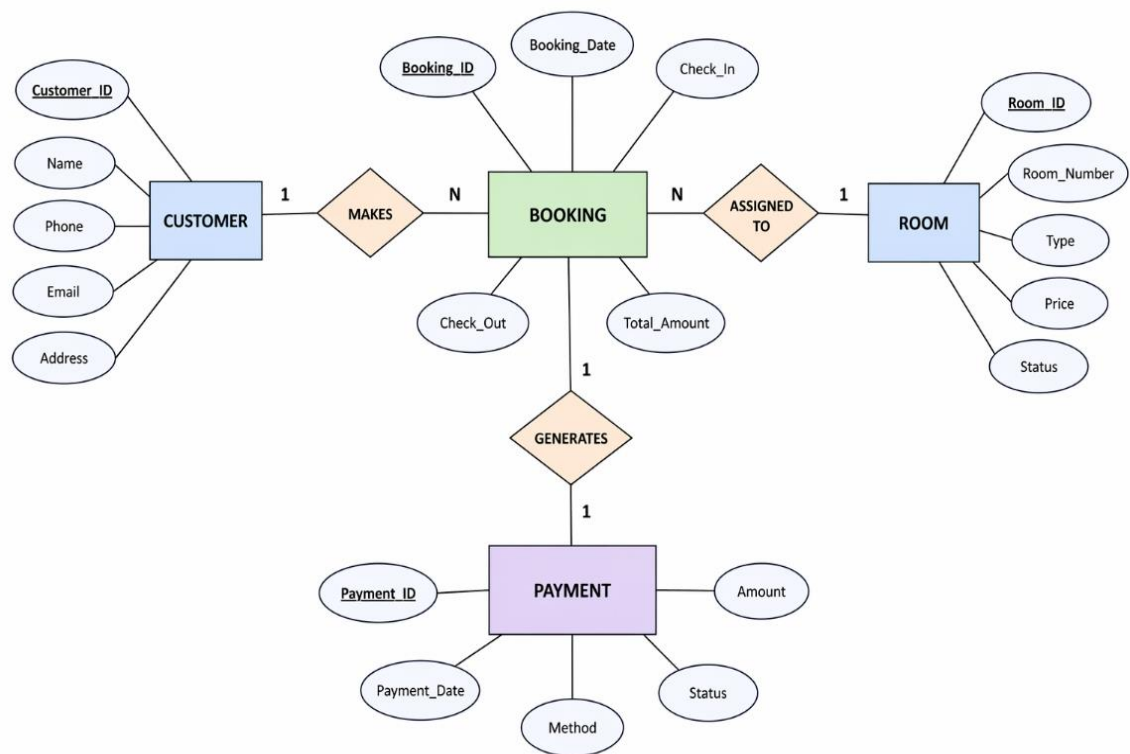


Fig.3.1: ER Diagram

3.3 Data Dictionary

A data dictionary defines all database fields.

Customer Table

Field Name	Type	Description
Customer_ID	INT (PK)	Unique ID
Name	VARCHAR	Customer Name
Phone	VARCHAR	Contact Number
Email	VARCHAR	Email Address

Room Table

Field Name	Type	Description
Room_ID	INT (PK)	Unique Room ID
Type	VARCHAR	Single/Double/Deluxe
Price	FLOAT	Room Price
Status	VARCHAR	Available/Booked

Booking Table

Field Name	Type	Description
Booking_ID	INT (PK)	Unique Booking ID
Customer_ID	INT (FK)	Linked Customer
Room_ID	INT (FK)	Assigned Room
Check_in	DATE	Check-in Date
Check_out	DATE	Check-out Date

Payment Table

Field Name	Type	Description
Payment_ID	INT (PK)	Unique Payment ID
Booking_ID	INT (FK)	Related Booking
Amount	FLOAT	Payment Amount
Date	DATE	Payment Date
Method	VARCHAR	Cash/Card/Online

3.4 Database Normalization

Database normalization is an important process in database design that helps improve the database's efficiency, consistency, and accuracy. It makes it easier to manage and maintain the data and ensures that the database is adaptable to changing business needs.

First Normal Form (1NF)

- No repeating groups
- Each field contains atomic values

Example:

- Separate Customer and Booking tables

Second Normal Form (2NF)

- No partial dependency
- All non-key attributes depend on full primary key

Example:

- Booking depends on Booking_ID, not partially on Customer_ID

Third Normal Form (3NF)

- No transitive dependency

Example:

- Room price stored only in Room table, not in Booking

Normalization in Hotel Management System:

the database is normalized up to **Third Normal Form (3NF)** to ensure efficient data storage and retrieval.

Main tables after normalization:

- Guest Table (Guest_ID, Name, Contact)
- Room Table (Room_ID, Room_Type, Price)
- Booking Table (Booking_ID, Guest_ID, Room_ID, Date)
- Payment Table (Payment_ID, Booking_ID, Amount, Method)

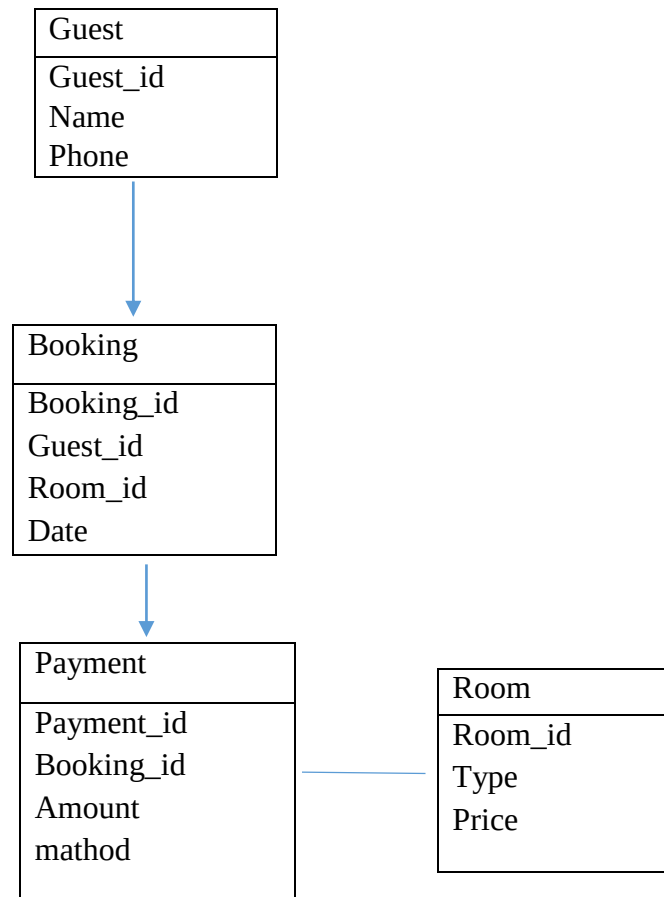


Fig 3.2: database Normalization

3.5 System Architecture

Architecture Type:

Presentation_Layer-(Frontend)



Application_Layer-(Python-Logic)



Database Layer (MySQL / SQLite)

Explanation:

Presentation Layer

- User Interface (UI)
- Built with: Python GUI
- Handles user input/output

Application Layer

- Core logic of system
- Developed using Python
- Handles:
 - Booking processing

- Payment calculation
- Data validation

Database Layer

- Stores all data
- MySQL / SQLite used
- Handles CRUD operations

Workflow:

1. User enters data (Frontend)
2. Python processes logic (Backend)
3. Data stored/retrieved from database
4. Result shown to user

CHAPTER 4

Testing

4.1 Introduction

Testing is the process of evaluating the Web-Based Hotel Management System to ensure that it works correctly, efficiently, and without errors. It helps to identify bugs, verify functionality, and ensure that the system meets user requirements.

4.2 Objectives of testing

- To detect and fix errors (bugs)
- To ensure all features work correctly
- To verify system requirements are fulfilled
- To improve system reliability and performance
- To ensure user satisfaction

4.3 Types of Testing Used

4.3.1 Unit Testing

Each module of the system is tested individually.

Example:

- Login module
- Room booking module
- Payment module

Ensures each part works independently

4.3.2 Integration Testing

Different modules are tested together to check interaction.

Example:

- Booking + Payment module
- Guest + Booking module

Ensures proper data flow between modules

4.3.3 System Testing

The complete system is tested as a whole.

- Checks overall functionality of the system

4.3.4 User Acceptance Testing (UAT)

The system is tested by users to verify it meets requirements.

- Confirms system is ready for real use

4.4 Test Cases

Test Case ID	Test Description	Input	Expected Output	Result
TC01	Login Test	username & password	Login successful	Pass
TC02	Invalid Login	Wrong password	Error message	Pass
TC03	Room Booking	Select room & date	Booking confirmed	Pass

Test Case ID	Test Description	Input	Expected Output	Result
TC04	Payment	Enter payment details	Payment successful	Pass
TC05	Room Availability	Select booked room	Not available	Pass

4.5 Testing Tools

- Python (for backend testing)
- Manual Testing
- Debugging tools

4.6. Result of Testing

All modules of the Hotel Management System were tested successfully. The system performs efficiently with correct output and minimal errors.

CHAPTER 5

SYSTEM IMPLEMENTATION

5.1 Output Screen

The output screen represents the graphical user interface (GUI) of the **Design and Implementation of Hotel Management System** developed using Python (Tkinter). It allows users to interact with the system efficiently. The system contains multiple screens such as the main dashboard, customer details, room booking, and room details.

5.1.1 Login Module

The Login Module ensures secure access to the system by allowing only authorized users to enter.

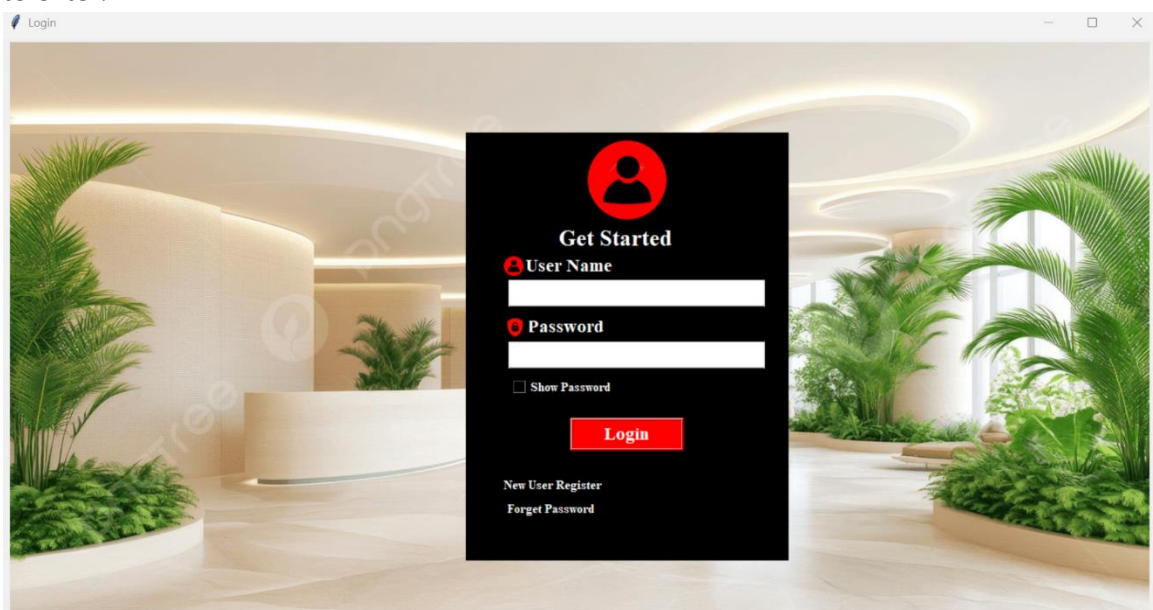


Fig 5.1: Login

Features:

- Username and password input
- Validation of user credentials
- Error message for invalid login
- Access control for system security

Purpose:

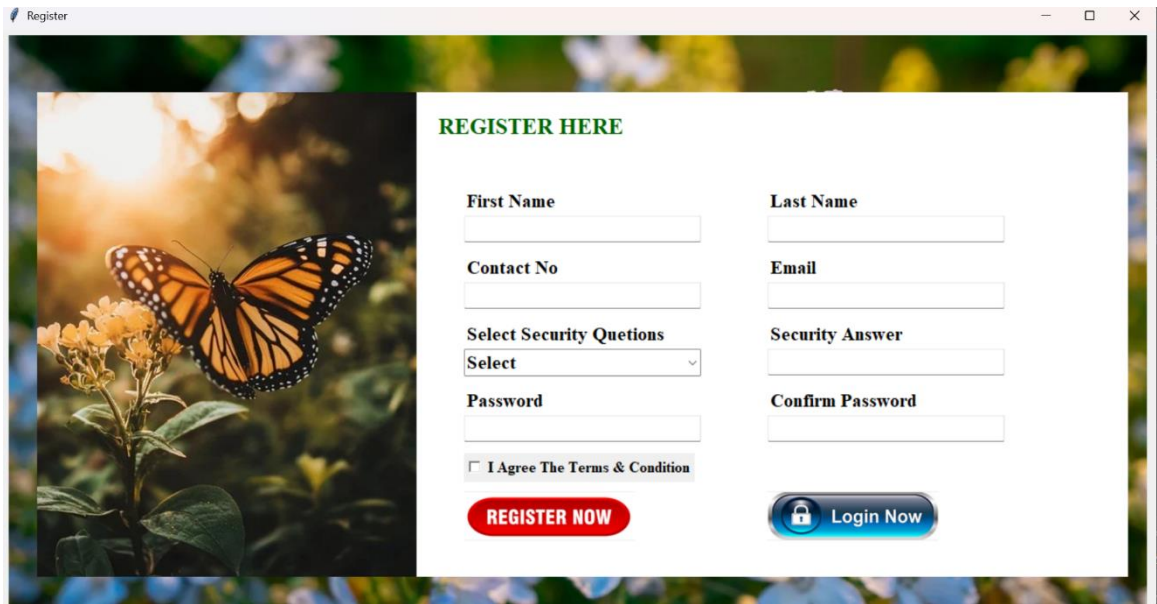
To prevent unauthorized access and protect sensitive hotel data.

5.1.2 Registration Module

The Registration Module allows new users (admin/staff) to create an account in the system.

Features:

- Input fields for user details (username, password, email)
- Password confirmation
- Data stored in database
- Validation for duplicate users



The registration form is titled "REGISTER HERE" in green. It features a large image of a monarch butterfly on the left. The form includes the following fields and elements:

- First Name**: Text input field
- Last Name**: Text input field
- Contact No**: Text input field
- Email**: Text input field
- Select Security Questions**: Dropdown menu with "Select" as the current option
- Security Answer**: Text input field
- Password**: Text input field
- Confirm Password**: Text input field
- ☐ I Agree The Terms & Condition
- REGISTER NOW**: Red button
- Login Now**: Blue button with a lock icon

Fig 5.2: Registration

Purpose:

To manage system users and allow secure account creation.

5.1.3 Main Dashboard

The main screen appears after running the system.

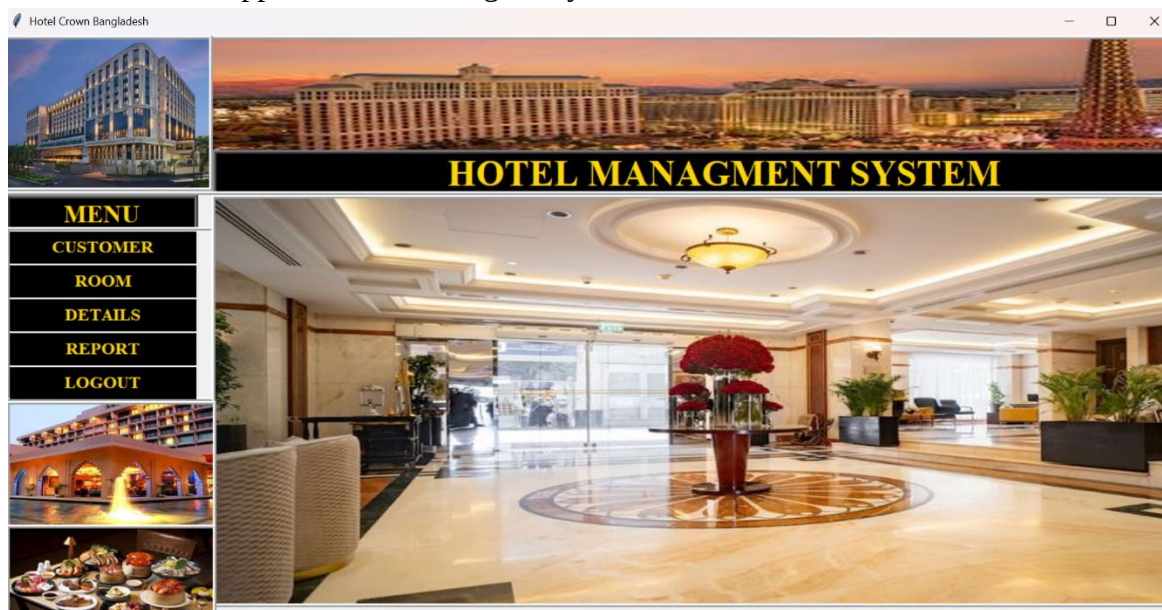


Fig 5.3: Dashboard

Features:

- Displays system title: Design and Implementation of Hotel Management System
- Contains menu buttons:
 - Customer
 - Room
 - Details
 - Report
 - Logout
- Includes images and logo for better UI design

Purpose:

This screen acts as the central navigation panel to access all modules.

5.1.4 Customer Details Screen

This screen is used to manage customer information.

Refer No	Name	Father's Name	Gender	PostCode	Mobile Number	Email	Nationality
4303	alamin	gafur	Other	2310	1715978944	alamin@gmail.co	Bangladeshi
4456	kanom	abul	Female	5630	5466223	kanom@25	American
5466	abir	akkas	Male	1207	123456		Bangladeshi
6355	tarek	monjo	Male	1250	25749631	tarek@120	Bangladeshi
6849	arfin mia	Abdul aziz mia	Male		17236559474		American
8253	Ruhul Amin	abdul gafur	Male	1207	1715482240	ruhul@gmail.co	Bangladeshi
8395	himel	babul	Male	25630	2549635	himel@US	American
9443	Akash	Jabed Ali	Other	1207	1723385576	akash@gmail.com	Bangladeshi

Fig 5.4: Customer details

Features:

- Add new customer details
- Update existing records
- Delete customer information
- Search customer by Mobile or Reference Number
- Display all records in table format

Fields Included:

- Customer Ref
- Name
- Father Name
- Gender
- Mobile Number
- Email
- Nationality
- ID Proof
- Address

Purpose:

To store and manage customer data in the database.

5.1.5 Room Details Screen

This screen is used to add and manage room information.

Features:

- Add new room
- Update room details
- Delete room

- Display all rooms

The screenshot shows a web application window titled 'Hotel Crown Bangladesh'. The main heading is 'ROOM BOOKING DETAILS'. On the left, there is a 'New Room Add' form with input fields for 'Floor', 'Room No', and 'Room Type'. Below these fields are four buttons: 'Add', 'Update', 'Delete', and 'Reset'. On the right, there is a 'Show Room Details' table with columns 'Floor', 'RoomNo', and 'RoomType'. The table contains the following data:

Floor	RoomNo	RoomType
1	1002	Single
5	405	Luxary
6	5002	Duble
5	5003	Double
5	505	Double

Fig 5.5:Room details Screen

Fields Included:

- Floor Number
- Room Number
- Room Type

Purpose:

To maintain room availability and types in the hotel.

5.1.6 Room Booking Screen

This screen handles the booking process.

The screenshot shows a web application window titled 'Hotel Crown Bangladesh'. The main heading is 'ROOM BOOKING DETAILS'. On the left, there is a 'Room Booking Details' form with fields for 'Customer Contact', 'Check In Date', 'Check Out Date', 'Room Type' (set to 'Single'), 'Available Room' (set to '1002'), 'Meal', 'No of Days', 'Paid Tax', 'Sub Total', and 'Total Cost'. There are buttons for 'Fetch Data', 'Bill', 'Add', 'Update', 'Delete', and 'Reset'. On the right, there is a large image of a hotel room. Below the image, there is a 'View Details and Search System' section with a 'Search By:' dropdown set to 'Contact' and buttons for 'Search' and 'Show All'. Below this is a table with booking details:

Contact No.	Check In	Check Out	Room Type	Room No	Meal	No Of
1723385576	20/12/2025	29/12/2025	Laxary	1010	BreakFast	9
25749631	14/12/2025	21/12/2025	Double	1016	Denner	7
2549635	08/12/2025	15/12/2025	Single	303	Lunch	7
017236559474	24/01/2025	29/01/2025	Duble	5003	bercfast	5
17236559474	24/01/2025	29/01/2025	Duble	505	bercfast	5

Fig 5.6: Room booking screen

Features:

- Fetch customer data using mobile number
- Select room type and available room

- Input check-in and check-out date
- Calculate total bill
- Store booking details in database

Fields Included:

- Customer Contact
- Check-In Date
- Check-Out Date
- Room Type
- Room Number
- Meal
- Number of Days
- Total Cost

Purpose:

To manage hotel room booking and billing system.

5.1.7 Billing System

The system automatically calculates total cost based on:

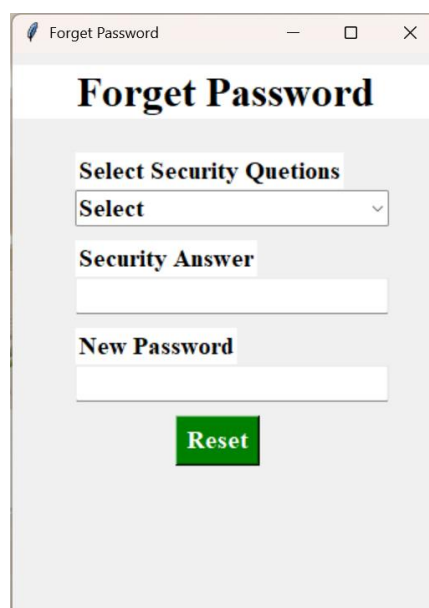
- Room type
- Meal type
- Number of days
- Tax

Output Includes:

- Sub Total
- Tax
- Total Cost

5.1.8 Forgot Password

The Forgot Password Module allows users to recover their account password in case they forget it. This feature improves system usability and ensures continuous access without creating a new account.



The screenshot shows a web browser window titled "Forget Password". The form has a light gray background. At the top, it says "Forget Password" in bold. Below that is a section titled "Select Security Questions" with a dropdown menu showing "Select". Underneath is a "Security Answer" label followed by a text input field. Then, there is a "New Password" label followed by another text input field. At the bottom of the form is a green button with the word "Reset" in white text.

Fig 5.7: Forget Password

Features:

- Input field for registered email or username
- Verification of user identity
- Option to reset or change password
- Error message for invalid or unregistered user
- Secure password update

5.2 Database Integration

The system uses **MySQL database** to store:

- Customer data
- Room details
- Booking records

Database Name:

managment

Tables Used:

- customerdata
- deatails
- room

5.3 Conclusion

The implemented system provides a user-friendly interface for managing hotel operations efficiently. It reduces manual work and ensures accurate data handling through database integration.

CHAPTER 6

CONCLUSION AND FUTURE WORKS

6.1 Conclusion

The **Design and Implementation of Web-Based Hotel Management System** has been successfully designed and implemented using Python with Tkinter for the graphical user interface and MySQL for database management. The system provides an efficient and user-friendly way to manage hotel operations such as customer records, room details, and booking information. This system helps to reduce manual work, minimize errors, and improve data accuracy. The integration of different modules like customer management, room booking, and billing ensures smooth functioning of hotel activities.

The project demonstrates how software solutions can simplify real-world problems by automating repetitive tasks. It also improves data storage, retrieval, and management through a structured database system.

Overall, the system achieves its main objective of providing a simple, effective, and reliable hotel management solution.

6.2 Limitations of the System

Although the system is functional, there are some limitations:

- The system works only on a local machine (not web-based)
- Limited security features (no login authentication)
- UI design is basic and can be improved
- No online booking or payment system
- Report generation feature is not fully implemented

6.3 Future Works

The system can be further improved by adding the following features:

- **Web-Based System:** Convert the system into a web application so it can be accessed from anywhere
- **Login System:** Add admin/user authentication for better security
- **Online Payment Integration:** Allow customers to pay online using cards or mobile banking
- **Advanced Reports:** Generate reports such as daily income, booking history, and customer statistics
- **Mobile App Version:** Develop an Android/iOS version of the system
- **Improved UI Design:** Use modern frameworks for a more attractive interface
- **Cloud Database:** Store data in cloud servers instead of local MySQL
- **Real-time Room Availability:** Show live room status updates

REFERENCES

- [1] G. B. Davis and M. H. Olson, *Management Information Systems: Conceptual Foundations, Structure, and Development*, 2nd ed. New York, NY, USA: McGraw-Hill, 1985.
- [2] E. Matthes, *Python Crash Course: A Hands-On, Project-Based Introduction to Programming*, 2nd ed. San Francisco, CA, USA: No Starch Press, 2019.
- [3] S. Lipschutz and M. Lipson, *Schaum's Outline of Data Structures*, New York, NY, USA: McGraw-Hill, 2014.
- [4] R. Elmasri and S. B. Navathe, *Fundamentals of Database Systems*, 7th ed. Boston, MA, USA: Pearson, 2016.
- [5] G. Kasavana and R. Brooks, *Managing Front Office Operations*, 9th ed. Lansing, MI, USA: Educational Institute, 2013.
- [6] P. O'Connor and A. J. Frew, "The future of hotel electronic distribution: Expert and industry perspectives," *Cornell Hotel and Restaurant Administration Quarterly*, vol. 45, no. 1, pp. 33–45, Feb. 2004.
- [7] M. Kasavana and R. Brooks, *Managing Front Office Operations*, 9th ed. Lansing, MI, USA: American Hotel & Lodging Educational Institute, 2010.
- [8] J. Walker, *Introduction to Hospitality Management*, 4th ed. Upper Saddle River, NJ, USA: Pearson, 2013.