A

PROJECT REPORT

ON

"HOTEL MANAGEMENT SYSTEM"

SUBMITTED BY:

Bodkhe Gaurav Abasaheb (2124UCEM1041)

SUBJECT:

PROGRAMMING

AND PROBLEM

SOLVING USING

C++

Under the guidance of

Miss. ISHWARI TIRSE MAM

Department of Computer Science and Engineering

Sanjivani Rural Education Society's

SANJIVANI UNIVERSITY

KOPARGAON – 423603, DIST : AHMEDNAGAR

2024-2025

INDEX

INTRODUCTION

A hotel management system is a crucial tool for managing the day-to-day operations of a hotel. Our system, built using C++, will provide a comprehensive solution for managing room bookings, guest information, staff scheduling, and financial transactions. C++ offers a high level of performance, reliability, and flexibility, making it an ideal choice for building a robust and scalable system.

The system will consist of several modules, including room booking, guest information, staff scheduling, financial transactions, and reporting and analytics. Each module will be designed to interact with the others to provide a seamless and integrated system.

We will utilize C++ concepts such as classes and objects, inheritance and polymorphism, file input/output, data structures, and algorithms to build the system. The system will be designed to be user-friendly and intuitive, with a focus on streamlining hotel operations and improving customer satisfaction.

Some of the key features of the system include:

- Room booking management: manage room bookings, availability, and rates
- Guest information management: store and retrieve guest information
- Staff scheduling management: manage staff schedules and shifts
- Financial transaction management: manage financial transactions and generate reports
- Reporting and analytics: generate reports and analyze data to improve hotel operations

By building this hotel management system using C++, we aim to provide a comprehensive and robust solution for hotels to manage their operations efficiently and effectively.

OBJECTIVES

This new hotel management has been developed to meet all the processing requirements which are needed within the hotel industry. This system will allow the hotel reception department to manage all the records of their customers and their payment in easy manner. The person who will use this system will able to get all details on how to use the instruction in offline mode, so no need to take special training before using the system. All proper keywords has been used to define each function of this new system.

To use this system, its users should have a valid password.
Once the correct password will be entered, verification process will be taken place at the backend side and if entered password will be correct, admin will redirected to the next section, where they will able to handle all the business activity using the computer system.

Existing system do not much flexibility and reliability which has been provided by this new hotel management system. In the previous system all records were saved in file under text which can be easily readable by the other person if certain mistakes has been done. Even if the file is in read mode, its contents can be copied and changes can be made to that particular file. Getting information on particular customer for each services which they have taken within their hotel was not possible and much manual processing task was done manually to final bill of a particular customer before they check out from the hotel.

SCOPE OF THE HOTEL MANAGEMENT SYSTEM

The scope of the Hotel Management System using C++ includes:

1. User Authentication: Login and logout functionality for admin and users.

2. Room Management: Adding, editing, and deleting room details, including room type, capacity, and availability.

3. Booking Management: Booking, canceling, and modifying room bookings, including payment processing.

4. Guest Management: Storing and managing guest information, including personal details and booking history.

5. Services Management: Managing hotel services, such as room service, laundry, and amenities.

6. Reporting: Generating reports on room occupancy, bookings, and revenue.

7. Payment Processing: Handling online payments and generating invoices.

8. Room Allocation: Automatically assigning rooms to guests based on availability and preferences.

9. Customer Feedback: Collecting and storing guest feedback and reviews.

10. System Administration: Managing system settings, user roles, and access permissions.

11. Integration: Integrating with third-party systems, such as online travel agencies and payment gateways.

12. Security: Ensuring data security and privacy compliance.

13. Scalability: Designing the system to accommodate future growth and expansion.

14. User Interface: Creating a user-friendly and intuitive interface for users.

15. Error Handling: Implementing error handling and exception handling mechanisms.

METHODOLOGY/IMPLEMENTATION

Requirements gathering and system design are completed, followed by database design and class implementation in C++. Classes for Room, Booking, Guest, Service, Payment, and User are created. Functions for room management, booking management, guest management, service management, payment processing, and user authentication are implemented.

Data validation and error handling ensure data integrity. Testing includes unit testing, integration testing, and system testing. Deployment and maintenance follow.

Implementation details:

- Room management: add, edit, delete rooms
- Booking management: book, cancel, modify bookings
- Guest management: add, edit, delete guest details
- Service management: add, edit, delete services
- Payment processing: handle online payments and generate invoices
- User authentication: login/logout functionality for admin and users

The system provides a user-friendly interface for hotel staff and guests, ensuring efficient room management, booking, and service provision. Data security and privacy are ensured through proper authentication and authorization. The system is scalable and can be easily integrated with third-party systems. Ongoing maintenance and support ensure the system continues to meet the hotel's evolving needs.

EXPECTED OUTCOMES

The system aims to achieve:

- Efficient room management and booking processes
- Improved guest experience through personalized services
- Optimized service provision and automated payment processing
- Reliable user authentication and data security

Benefits:

- Increased productivity and reduced administrative burdens
- Enhanced customer satisfaction and loyalty
- Competitive advantage through technology adoption
- Cost savings through automation and efficient resource allocation
- Data-driven decision making for business growth

Outcomes:

- Streamlined room booking and allocation processes
- Accurate booking tracking and cancellation
- Personalized guest services and preferences
- Effective management of hotel services
- Secure online payment processing and invoicing
- Reliable user authentication and access controls
- Scalable and adaptable system for future growth

By achieving these outcomes, the Hotel Management System using C++ will enhance operational efficiency, improve customer satisfaction, and drive business growth, providing a competitive advantage in the hospitality industry.

CODE

```
#include <iostream>
#include <string>
#include <vector>
using namespace std;
class Hotel {
public:
	string name;
	int rooms;
};
class Booking {
public:
	string guestName;
	int roomNumber;
	string date;
};
vector<Hotel> hotels;
vector<Booking> bookings;
void addHotel() {
	Hotel hotel;
	cout << "Enter hotel name: ";
	cin >> hotel.name;
	cout << "Enter number of rooms: ";
	cin >> hotel.rooms;
	hotels.push_back(hotel);
}
void addBooking() {
	Booking booking;
	cout << "Enter guest name: ";
	cin >> booking.guestName;
	cout << "Enter room number: ";
	cin >> booking.roomNumber;
	cout << "Enter date: ";
	cin >> booking.date;
```

```
        bookings.push_back(booking);
}
void displayHotels() {
        for (auto &hotel : hotels) {
                cout << "Hotel Name: " << hotel.name << endl;
                cout << "Number of Rooms: " << hotel.rooms << endl;
        }
}
void displayBookings() {
        for (auto &booking : bookings) {
                cout << "Guest Name: " << booking.guestName << endl;
                cout << "Room Number: " << booking.roomNumber << endl;
                cout << "Date: " << booking.date << endl;
        }
}
int main() {
        int choice;
        while (true) {
                cout << "1. Add Hotel" << endl;
                cout << "2. Add Booking" << endl;
                cout << "3. Display Hotels" << endl;
                cout << "4. Display Bookings" << endl;
                cout << "5. Exit" << endl;
                cout << "Enter your choice: ";
                cin >> choice;
                switch (choice) {
                        case 1:
                                addHotel();
                                break;
                        case 2:
                                addBooking();
                                break;
                        case 3:
                                displayHotels();
                                break;
```

```
                case 4:
                        displayBookings();
                        break;
                case 5:
                        return 0;
                default:
                        cout << "Invalid choice. Please try again." << endl;
                }
        }
        return 0;
}
```

OUTPUT

```
Output
/tmp/jKUkIbJPrE.o
1. Add Hotel
2. Add Booking
3. Display Hotels
4. Display Bookings
5. Exit
Enter your choice: 1
Enter hotel name: Hotel_Gaurav
Enter number of rooms: 2
1. Add Hotel
2. Add Booking
3. Display Hotels
4. Display Bookings
5. Exit
Enter your choice: 2
Enter guest name: shubham
Enter room number: 3
Enter date: 17/10/2024
1. Add Hotel
2. Add Booking
3. Display Hotels
4. Display Bookings
5. Exit
Enter your choice: 3
Hotel Name: Hotel_Gaurav
Number of Rooms: 2
```

```
1. Add Hotel
2. Add Booking
3. Display Hotels
4. Display Bookings
5. Exit
Enter your choice: 4
Guest Name: shubham
Room Number: 3
Date: 17/10/2024
1. Add Hotel
2. Add Booking
3. Display Hotels
4. Display Bookings
5. Exit
Enter your choice: 5


=== Code Execution Successful ===
```

Conclusion

The Hotel Management System using C++ is a comprehensive solution that streamlines hotel operations, enhancing efficiency, productivity, and customer satisfaction. It automates room booking, allocation, and deallocation processes, improves guest experience through personalized services, optimizes service provision, and reduces operational costs. The system ensures reliable user authentication and data security, and generates comprehensive reports for informed decision-making.

The system's scalability and integration capabilities make it an ideal solution for hotels of various sizes and complexities. C++'s performance, reliability, and flexibility ensure a robust and maintainable solution.

Future developments include:

- Integrating AI and ML for predictive analytics and personalized guest experiences
- Implementing mobile applications for guests and staff
- Expanding reporting and analytics capabilities
- Enhancing security features to ensure compliance with industry standards

The Hotel Management System using C++ is a cutting-edge solution that transforms hotel operations, driving growth and excellence in the hospitality industry. Its comprehensive features, scalability, and integration capabilities make it an ideal choice for hotels seeking to enhance their operational efficiency and customer satisfaction.

By implementing this system, hotels can gain a competitive advantage, increase revenue, and improve customer loyalty. The system's flexibility and customization options ensure that it can be tailored to meet the specific needs of each hotel, making it an invaluable tool for hotel managers and staff.