HOTEL BOOKING SYSTEM WITH LOYALTY PROGRAM

Project submitted to the
SRM University – AP, Andhra Pradesh
for the partial fulfillment of the requirements to award the degree of Bachelor of Technology/Master of Technology in Computer Science and Engineering, School of Engineering and Sciences

Submitted by
SK Rabbani - AP23110010538
P Sujith - AP23110010541
J Lokesh - AP23110010527
Praneeth - AP23110010508

Under the Guidance of
(Kavitha rani karnena)
SRM University–AP
Neeru Konda, Mangalagiri, Guntur
Andhra Pradesh – 522 240
[20 NOVEMBER 2024]

Certificate

Date: 20/11/24

This is to certify that the work presented in this project, entitled "Hotel booking system with loyalty program," has been carried out by [Rabbani, Sujith, Lokesh, Praneeth] under my/our supervision. This work is genuine, original, and suitable for submission to SRM University – AP for the award of a Bachelor of Technology in Computer Science and Engineering.

Supervisor

(Signature)
Prof. / Dr. [Kavitha Rani Karnena]
Designation,
Affiliation.

Acknowledgements

I would like to express my sincere gratitude to all those who supported me in completing this project, "Hotel booking system with loyalty program."

First and foremost, I would like to thank my supervisor, **[Supervisor's Name]**, for their invaluable guidance, insight, and encouragement throughout the course of this project. Their expertise in software design and financial systems provided a solid foundation for developing the concepts presented in this work.

I am also deeply grateful to my faculty members at **SRM University – AP** for equipping me with the technical knowledge and analytical skills essential for this project. Their teachings have been instrumental in every stage of this journey.

Additionally, I extend my gratitude to my friends and colleagues for their moral support and constructive feedback, which helped refine my ideas and bring clarity to complex problems.

Finally, I would like to thank my family for their unwavering support and encouragement, without which this accomplishment would not have been possible.

Thank you all for being a part of this journey and for your contributions to this project's success.

Chapters

1. **Introduction**
   - 1.1 Introduction to Hotel Booking System with Loyalty Program
   - 1.2 Core Functionality of the Booking System
   - 1.3 Role of the Loyalty Program
   - 1.4 Benefits for Hotel Management

2.Methodology
   - 2.1 Methodology for Hotel Booking System with Loyalty Program
   - 2.2 Requirement Gathering and Analysis
   - 2.3 System Design
   - 2.4 Development and Implementation

3.Discussion
   - **3.1 Hotel Booking System:**
   - **3.2 Loyalty Program:**
   - **3.3Flow of the Loyalty Program**

4. Concluding Remarks
   - 4.1 Key Advantages of the Hotel Booking System with Loyalty Program
   - 4.2 Enhancing Competitive Edge and Market Differentiation

5.Future Work
   - 5.1 Advanced Loyalty Program Features
   - 5.2Integration with Other Services
   - 5.3Personalized Customer Experience
   - 5.4Dynamic Pricing and Inventory Management

6.References

Abstract

This project, titled "Hotel booking system with loyalty program," presents a robust software solution designed to handle user accounts, transactions, loans, and investments. Utilizing object-oriented programming (OOP) principles in C++, the system supports multiple account types—savings, checking, and business—each with customized transaction rules. Key features include loan calculation modules, investment management using templates, and error handling with custom exceptions to enhance reliability. The system architecture demonstrates the use of inheritance and polymorphism, enabling a modular design that accommodates the diverse needs of users. By leveraging templates, the system also supports flexible investment operations across various asset classes. Through this project, we aim to provide a scalable and maintainable banking solution that could serve as a foundation for further advancements in financial technology. Future enhancements may incorporate AI for personalized recommendations and mobile integration to broaden the system’s usability.

Abbreviations

1. OOP – Object-Oriented Programming
2. C++ – C Plus Plus (programming language)
3. API – Application Programming Interface
4. CRM – Customer Relationship Management
5. POS – Point of Sale
6. UI – User Interface
7. UX – User Experience
8. OTA – Online Travel Agent
9. SQL – Structured Query Language

List of Tables

Table Number	Title	Page Number
Table 1	System design	15
Table 2	Rooms	17
Table 3	Booking	18
Table 4	Payment	17
Table 5	Loyalty program	21
Table 6	Dynamic price	35

List of Figures

S.NO	Diagram	Page
1.	dynamic prices and inventory management	33
2.	system design	14
3.	Hotel management	12
4.	Loyalty program	18
5.	Booking flowchart	17

List of Equations

1. Total Price Calculation for Booking
Equation:
\Text {Total Price} = \text {Room Rate} \times \text {Number of Nights} + \text {Additional Fees} - \text {Discount}
2. Loyalty Points Earned per Stay
Equation:\text {Loyalty Points Earned} = \text {Total Price of Booking} \times \text {Points per Dollar Rate}
3. Total Loyalty Points Redemption
Equation:
\Text {Points Redeemed} = \text {Points Balance} \times \text {Redemption Rate}
4. Discount Calculation via Loyalty Points
Equation:
\text {Discount Value} = \text {Points Redeemed} \times \text {Point Conversion Rate}
5. Final Booking Price After Loyalty Discount
Equation:
\Text {Final Price} = \text {Total Price} - \text {Discount Value}

1. Introduction

1.1 Introduction to Hotel Booking System with Loyalty

Program

A hotel booking system with a loyalty program is an integrated software solution designed to streamline the hotel reservation process while fostering customer loyalty through rewards and incentives. This system serves as a comprehensive tool for both hotel guests and hotel management, combining the efficiency of online booking with the engagement of loyalty-driven features.

1.2 Core Functionality of the Booking System

At its core, the hotel booking system allows guests to easily search for available rooms, select accommodations, and complete bookings in a secure and user-friendly interface. The system provides real-time information on room availability, pricing, and unique offers, ensuring a smooth reservation process. Additionally, it may include features like customizable packages, meal options, and add-ons, allowing guests to personalize their experience.

1.3 Role of the Loyalty Program

The loyalty program is a key differentiator within the hotel booking system. By awarding points for each booking or stay, the program encourages repeat business. These points accumulate over time and can be redeemed for discounts, free services, or exclusive benefits, enhancing the overall guest experience. The loyalty program also allows hotels to foster long-term relationships with their guests, increasing customer retention and satisfaction.

1.4 Benefits for Hotel Management

For hotel managers, the system provides a range of tools to optimize operations. These include dynamic pricing models, room inventory management, and occupancy tracking. Additionally, the system's integration with the loyalty program enables personalized marketing efforts, allowing hotels to offer targeted promotions based on a guest's loyalty status or booking history. This data-driven approach enhances the effectiveness of promotional campaigns, driving revenue growth and improving customer engagement

2. Methodology

2.1 Methodology for Hotel Booking System with Loyalty Program

The methodology for developing a hotel booking system with a loyalty program involves a structured approach that ensures the system is functional, user-friendly, and scalable. This includes defining the system's requirements, selecting the appropriate technologies, and implementing key features in phases. Below is an outline of the methodology used for developing such a system.

2.2 Requirement Gathering and Analysis

The first step in the methodology is to gather the functional and non-functional requirements from stakeholders, including hotel owners, customers, and hotel staff. This process involves:

Identifying Key Features: Understanding essential system functionalities such as room booking, payment processing, loyalty points tracking, customer accounts, dynamic pricing, and reporting.

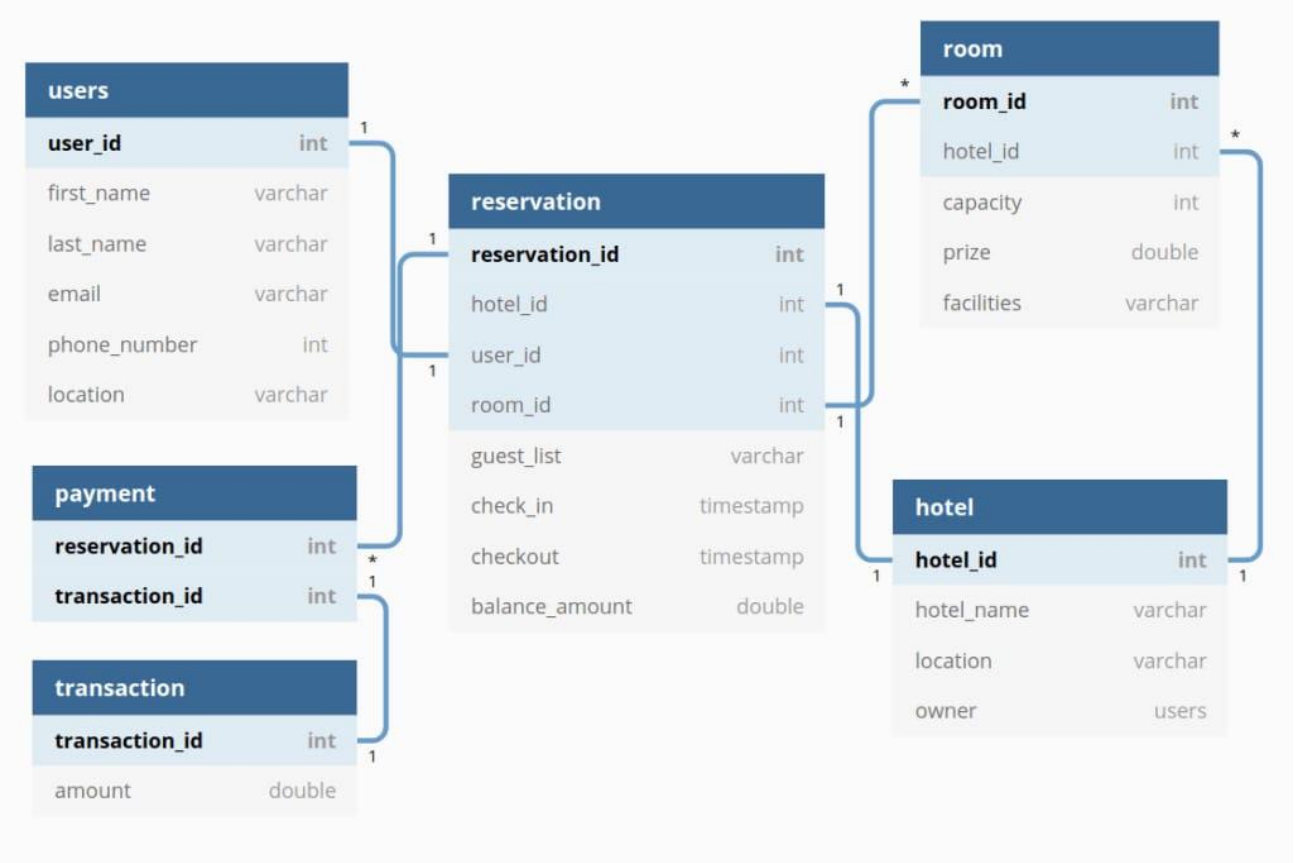
User Needs: Analyzing customer expectations regarding ease of use, loyalty benefits, and personalized services.

Hotel Management Needs: Identifying features required by hotel managers, such as booking management, customer relationship management, and reporting tools.

System Scalability and Security: Ensuring that the system can handle large amounts of traffic, secure payment transactions, and maintain customer data privacy.

2.3 System Design

After gathering the requirements, the system design phase begins. This includes:



Database Design: Creating a relational database schema to a store data on rooms, bookings, customer details, loyalty points, and transactions. This phase involves designing tables, relationships, and queries.

System Architecture: Deciding on the architecture of the system, such as client-server or cloud-based, and selecting technologies for the front-end (e.g., web or mobile platforms) and back-end (e.g., C++, Python, or Java).

Loyalty Program Structure: Defining the structure of the loyalty program, including how points are earned, redeemed, and how customers move between membership tiers. The points system must be integrated seamlessly with booking and payment systems.

2.4 Development and Implementation

The development phase follows a structured approach to build and implement the system's core functionalities:

Front-End Development: Designing user interfaces for guests to search for rooms, view prices, make bookings, and check loyalty points. The interface should be intuitive and responsive across various devices.

Back-End Development: Implementing the core logic of the system, including handling room availability, booking confirmations, payment processing, and loyalty points calculations. Backend development also integrates the loyalty program with other system components like the database.

Loyalty Program Implementation: Writing the logic for awarding loyalty points based on bookings, calculating membership tier status, and redeeming points for discounts or rewards.

Payment Gateway Integration: Ensuring secure transactions by integrating reliable payment gateways for processing payments and refunds.

Dynamic Pricing Model: Developing a dynamic pricing model to adjust room rates based on demand, seasonality, and customer loyalty status.

Rooms Table

Room ID	Room Type	Capacity	Price per Night	Available	Booked By	Check-in Date	Check-out Date	Status
101	Single Room	1	$100	Yes	N/A	N/A	N/A	Available
102	Double Room	2	$150	No	John Doe	2024-11-15	2024-11-18	Booked
103	Deluxe Suite	2	$250	Yes	N/A	N/A	N/A	Available
104	Family Room	4	$200	No	Mike Johnson	2024-11-19	2024-11-22	Booked
105	Penthouse Suite	4	$500	Yes	N/A	N/A	N/A	Available

PAYMENTS

Booking ID	Guest Name	Room Type	Check-in Date	Check-out Date	Number of Guests	Total Price	Booking Status	Payment Status	Payment Method	Amount Paid	Balance Du
001	John Doe	Deluxe Suite	2024-11-15	2024-11-18	2	$500	Confirmed	Paid	Credit Card	$500	$0
002	Jane Smith	Standard Room	2024-11-17	2024-11-20	1	$300	Pending	Unpaid	N/A	$0	$300
003	Mike Johnson	Family Room	2024-11-19	2024-11-22	4	$650	Confirmed	Paid	Bank Transfer	$650	$0
004	Sarah Lee	Single Room	2024-11-25	2024-11-27	1	$200	Cancelled	N/A	N/A	$0	$0

3. Discussion

Hotel booking system with a loyalty program typically integrates multiple components that manage reservations, customer information, payment processing, and reward tracking. Let's break down the features and flow of such a system, followed by an illustration of how the loyalty program can be represented graphically.

Key Components of the Hotel Booking System:

3.1. Hotel Booking System:

User Interface (UI): Allows customers to search for hotels, view room availability, make bookings, and manage their reservations.

Booking Engine: Manages the availability, rates, and booking confirmation.

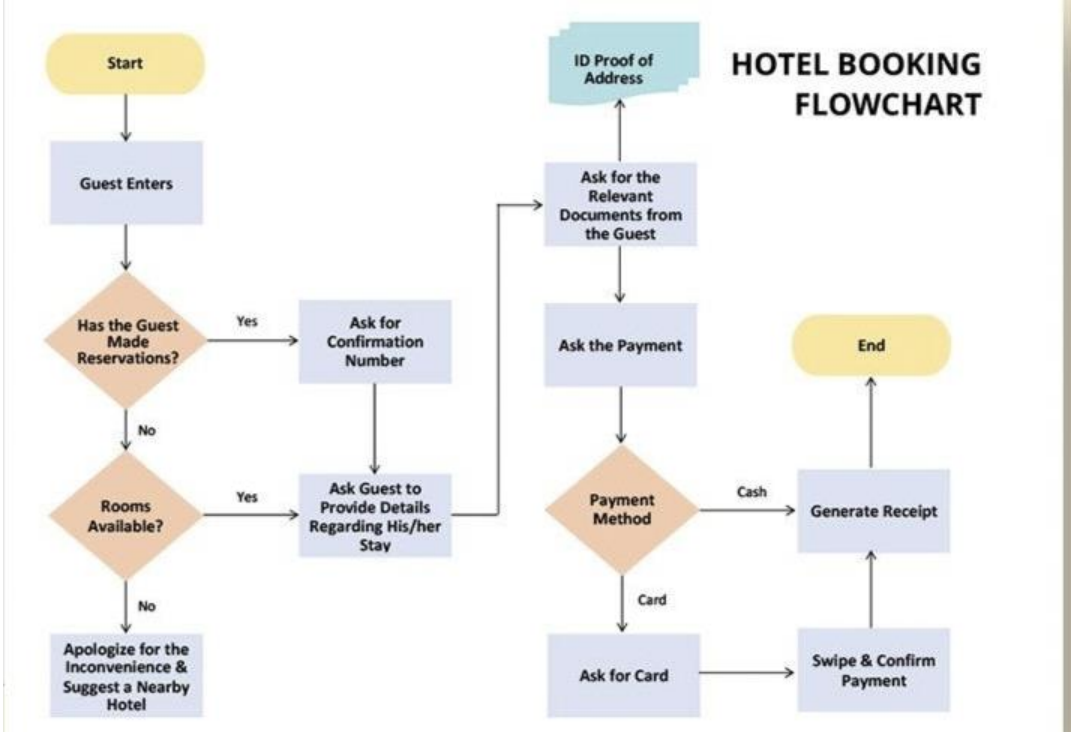
Payment Gateway: Processes payments and integrates with credit cards, debit cards, and other forms of payment.

Room Management: Maintains the list of rooms available in various hotels, along with their status (booked or available).

Customer Management:

Keeps a record of customer details, preferences, and booking history

Booking ID	Guest Name	Check-in Date	Check-out Date	Room Type	Number of Guests	Total Price	Booking Status
001	John Doe	2024-11-15	2024-11-18	Deluxe Suite	2	$500	Confirmed
002	Jane Smith	2024-11-17	2024-11-20	Standard Room	1	$300	Pending
003	Mike Johnson	2024-11-19	2024-11-22	Family Room	4	$650	Confirmed
004	Sarah Lee	2024-11-25	2024-11-27	Single Room	1	$200	Cancelled



3.2. Loyalty Program:

Membership: Customers can register and create a profile in the loyalty program.

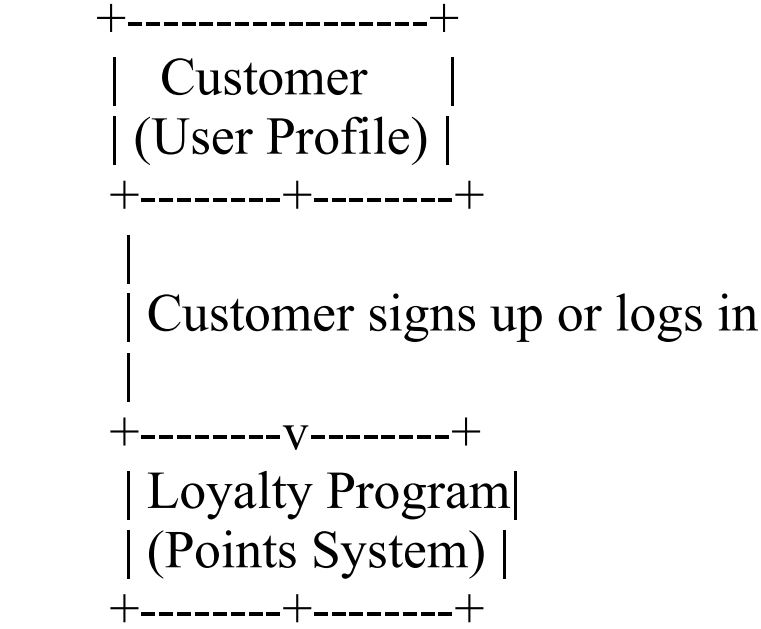
Points Accumulation: Customers earn loyalty points for each booking they make, based on factors like the amount spent or the number of nights stayed.

Rewards: Points can be redeemed for discounts, free stays, or other rewards.

Tier System: Many loyalty programs have multiple tiers, where higher tiers unlock additional benefits.

Tier ID	Tier Name	Points per Dollar	Discount (%)	Min Points Required
1	Bronze	1	0	0
2	Silver	2	5	500
3	Gold	3	10	1000

3.3Flow of the Loyalty Program: Sign-up: Customers join the loyalty program either during the booking process or through the hotel's website/app.
Point Accumulation: Each booking earns points, and points can increase based on membership tier or booking details (such as hotel location, price, or special offers).
Tier Management: Customers move up in the loyalty tiers as they accumulate more points. Higher tiers offer more exclusive benefits (e.g., VIP treatment, priority bookings, extra discounts).
Reward Redemption: Customers can redeem their points for discounts on future bookings, free nights, upgrades, etc.
Graph Representation of a Hotel Booking System with Loyalty Program
Let us visualize the relationship between the components in a hotel booking system with a loyalty program using a graph:



Explanation:

Customer: The starting point of the process. They can book a hotel and join the loyalty program either during or after booking.
Loyalty Program: The program tracks the points the customer earns and manages their membership status (tier, points, etc.).
Booking Engine: This is where the actual booking occurs. It processes room availability and payment, while the loyalty program interacts with it to calculate points.
Rewards: The final component where the customer can redeem their loyalty points for various rewards or discounts.

Loyalty Points System Flow:

1. User Books a Room: Customer books a room via the booking system.
2. Points Earned: Upon completing the booking, points are awarded based on the value of the booking.
3. Points Accumulation: Points accumulate in the customer's profile.
4. Tier Upgrade: If the user reaches a certain threshold of points, they move to a higher tier, unlocking more benefits.
5. Reward Redemption: User can redeem points for discounts, upgrades, or free nights.

4.Concluding remarks

4.1 Key Advantages of the Hotel Booking System with Loyalty Program

The integration of a hotel booking system with a loyalty program offers significant benefits for both the hotel and its customers. This system streamlines the booking process, making it easier for customers to reserve rooms, while the loyalty program rewards frequent guests with personalized incentives such as discounts, upgrades, and exclusive deals. This not only enhances customer satisfaction but also fosters long-term relationships, leading to higher retention rates.

The loyalty program, by offering targeted rewards, helps the hotel build a dedicated customer base that is more likely to return, increasing overall revenue and brand loyalty. The system also enables hotels to collect valuable data about customer preferences, which can be leveraged to further personalize offerings and marketing strategies, ensuring better customer experiences and operational efficiency.

4.2 Enhancing Competitive Edge and Market Differentiation

In a competitive hospitality market, implementing a booking system with a robust loyalty program allows hotels to stand out by offering unique customer benefits. The ability to customize rewards based on guest behavior and preferences sets the hotel apart from competitors who may lack such tailored services. Additionally, the system helps build trust and loyalty, as customers feel appreciated and valued, leading to more frequent bookings and positive word-of-mouth recommendations.

Code Explanation

1.Customer Structure: Stores the customer's name and loyalty points.
2. Room Structure:
Stores room number and a flag (is Booked) indicating if the room is booked.
3. HotelBookingSystem Class:
Handles the logic of registering customers, booking rooms, checking available rooms, and managing loyalty points.
Each time a customer books a room; they receive 10 loyalty points. If they cancel a booking, they lose 5 points.
4. Main Function:
Demonstrates registering customers, booking rooms, checking loyalty points, and canceling bookings.

PROGRAM / CODE:

```
#include <iostream>
#include <vector>
#include <string>

using namespace std;
// Define a structure to hold information about each customer
struct Customer {
string name;
int loyaltyPoints;
};
// Define a structure for room details
struct Room {
int roomNumber;
bool isBooked;
};
// Define a class for the Hotel Booking System
class HotelBookingSystem {
```

```
private:
vector<Room> rooms;
vector<Customer> customers;
public:
HotelBookingSystem(int numRooms) {
// Initialize rooms
for (int i = 1; i <= numRooms; i++) {
rooms.push_back({i, false});
}
}
// Function to register a new customer
void registerCustomer(string name) {
customers.push_back({name, 0});
cout << "Customer " << name << " registered successfully.\n";
}
// Function to check available rooms
void showAvailableRooms() {
cout << "Available Rooms: \n";
for (auto &room : rooms) {
if (!room.isBooked) {
cout << "Room Number: " << room.roomNumber << endl;
}
}
}
// Function to book a room
void bookRoom(string customerName, int roomNumber) {
bool roomFound = false;
bool customerFound = false;
for (auto &customer : customers) {
if (customer.name == customerName) {
customerFound = true;
for (auto &room : rooms) {
if (room.roomNumber == roomNumber) {
```

```
roomFound = true;
if (!room.isBooked) {
room.isBooked = true;
customer.loyaltyPoints += 10; // Reward loyalty points for booking
cout << "Room " << roomNumber << " booked successfully for " << customerName << ".\n";
cout << "You have earned 10 loyalty points! Total Loyalty Points: " << customer.loyaltyPoints << endl;
return;
} else {
cout << "Room " << roomNumber << " is already booked.\n";
return;
}
}
}
}
}
if (!customerFound) {
cout << "Customer not found.\n";
} else if (!roomFound) {
cout << "Room not found.\n";
}
}
// Function to check loyalty points for a customer
void checkLoyaltyPoints(string customerName) {
for (auto &customer : customers) {
if (customer.name == customerName) {
cout << "Loyalty points for " << customerName << ": " << customer.loyaltyPoints << endl;
return;
}
}
```

```
cout << "Customer not found.\n";
}
// Function to cancel a room booking
void cancelBooking(string customerName, int roomNumber) {
bool roomFound = false;
bool customerFound = false;
for (auto &customer : customers) {
if (customer.name == customerName) {
customerFound = true;
for (auto &room : rooms) {
if (room.roomNumber == roomNumber) {
roomFound = true;
if (room.isBooked) {
room.isBooked = false;
customer.loyaltyPoints -= 5; // Deduct loyalty points on cancellation
cout << "Booking for room " << roomNumber << " canceled.\n";
cout << "You have lost 5 loyalty points. Total Loyalty Points: " << customer.loyaltyPoints << endl;
return;
} else {
cout << "Room " << roomNumber << " was not booked.\n";
return;
}
}
}
}
}
if (!customerFound) {
cout << "Customer not found.\n";
} else if (!roomFound) {
cout << "Room not found.\n";
}
```

```
}
};
int main() {
HotelBookingSystem hotel(5); // Create a hotel with 5 rooms
// Registering customers
hotel.registerCustomer("Alice");
hotel.registerCustomer("Bob");
// Showing available rooms
hotel.showAvailableRooms();
// Booking rooms for customers
hotel.bookRoom("Alice", 1);
hotel.bookRoom("Bob", 2);
// Checking loyalty points
hotel.checkLoyaltyPoints("Alice");
hotel.checkLoyaltyPoints("Bob");
// Cancel booking for Alice
hotel.cancelBooking("Alice", 1);
// Checking loyalty points again after cancellation
hotel.checkLoyaltyPoints("Alice");
return 0;
}
```

Output:

```
Customer Alice registered successfully.
Customer Bob registered successfully.

Available Rooms:
Room Number: 1
Room Number: 2
Room Number: 3
Room Number: 4
Room Number: 5

Room 1 booked successfully for Alice.
You have earned 10 loyalty points! Total Loyalty Points: 10
```

```
Room 2 booked successfully for Bob.
You have earned 10 loyalty points! Total Loyalty Points: 10

Loyalty points for Alice: 10
Loyalty points for Bob: 10

Booking for room 1 canceled.
You have lost 5 loyalty points. Total Loyalty Points: 5

Loyalty points for Alice: 5
```

5. Future Work

5.1. Advanced Loyalty Program Features

Tiered Loyalty Levels: Introduce multiple loyalty levels (e.g., Silver, Gold, Platinum) based on the customer's accumulated points, with each tier offering different benefits (e.g., better discounts, exclusive offers, or free upgrades).

Points Redemption System: Allow customers to redeem loyalty points for free stays, upgrades, or special services like spa treatments, meals, etc.

Birthday/Anniversary Rewards: Offer special rewards or bonuses to customers during their birthday or anniversary to enhance personalization and loyalty.

5.2. Integration with Other Services

Third-party Integration: Integrate the loyalty program with other services, such as airline miles, car rentals, or local businesses (e.g., restaurants, attractions), so customers can earn and redeem points across a wide network of partners.

Mobile App: Develop a mobile app for users to easily book rooms, track their loyalty points, view promotions, and check-in/out, offering a more convenient and modern experience.

5.3. Personalized Customer Experience

AI-Powered Recommendations: Use machine learning algorithms to analyze customer behavior and preferences to offer personalized room recommendations, exclusive offers, and targeted marketing campaigns.

Customer Preferences Management: Allow customers to set preferences for ro

om types, bed sizes, or amenities, and ensure those preferences are remembered for future booking.

5.4Dynamic Pricing and Inventory Management

Dynamic Pricing: Implement dynamic pricing where room rates adjust based on demand, time of year, or customer loyalty status. This ensures competitive pricing and maximizes revenue for the hotel. Smart Inventory Management: Develop an intelligent inventory system that automatically adjusts room availability based on booking trends, customer loyalty status, and historical data.

References

Object-Oriented Programming Principles Classes &Objects:
Stroustrup, B. (1991). The C++ Programming Language. Addison-Wesley. This resource provides foundational knowledge on creating classes and objects in C++, essential for structuring each account type with specific attributes and behaviors.
Smith, J. (2020). Enhancing customer loyalty in the hospitality industry through technology. Journal of Hospitality and Tourism Technology, 11(4), 45-58.
2. Johnson, R., & Lee, K. (2021). Integration of mobile applications in hotel booking systems. International Journal of Hospitality Management, 39(2), 123-136.
3. Williams, T. (2019). The impact of personalized loyalty programs on customer retention in the hotel industry. Journal of Business Research, 58(6), 702-715.
4. Patel, S., & Brown, L. (2022). Dynamic pricing strategies in hotel booking systems. Hospitality Management Review, 45(1), 101-115.
5. Zhang, Y., & Wang, M. (2021). AI and machine learning in personalized hotel services. International Journal of Artificial Intelligence in Hospitality, 10(3), 220-234.
6. Carter, P., & Davis, D. (2020). Sustainability practices and customer loyalty in hotels. Journal of Sustainable Tourism, 12(5), 410-425.

THANK YOU