UNIVERSITI TEKNOLOGI MALAYSIA

OBJECT ORIENTED PROGRAMMING (SECJ2154)

SEMESTER 1 2023/2024

GROUP PROJECT

HOTEL BOOKING SYSTEM

AHMAD NABIL BIN AHMAD NAZRIL (B23CS0020)

AQMAR ILHAN BIN MOHAMAD FADZILAH (B23CS0027)

EIZKHAN BIN SAHARIZA (B23CS0035)

MUHAMMAD HANIF AZRI BIN AZIZAN (B23CS0057)

2 SECRH

SECTION 01

Lecturer:

MADAM LIZAWATI MI YUSUF

17rd JANUARY 2024

Table of Contents

SECTION A: PROJECT DESCRIPTION

Synopsis:

Our group have to develop a hotel booking system named Moyai Hotel Booking System, which manages the booking process for the hotel. The system allows guests to make reservations, and it automatically updates the availability of rooms in real-time. The system also manages the payment process, ensuring that the hotel receives payment for the booking and that the guest's payment information is securely stored.

Moyai Hotel Booking System is an application that allows staff to book the room for the guest. The system automates the booking process, syncing up with the hotel's booking information to facilitate a smoother booking process. It also allowing the staff of the hotels to distribute the room availability in real time. The system is essential for hotels to have global appeal and increased occupancy rates, and it improves efficiency by automatically syncing across the system.

This system likely used to manage individual bookings within this system. It may provide methods to create, update, and delete bookings, as well as to retrieve information about a specific booking. The class may also interact with other components of the system, such as the payment gateway, to process payments and update the booking status accordingly.

Overall, the Moyai Hotel Booking Information Report provides a snapshot of a hotel reservation, while the key component of the hotel reservation system that manages the booking process and enables hotels to manage their bookings efficiently and effectively.

Objective and Scope:

i)   Automate Processes:
- Reduce manual work: Automate tasks like reservation confirmation emails, guest communication, and report generation.
- Minimize errors: Eliminate human error through automated data entry and validation.
- Improve data accuracy: Use automated systems to track room availability, guest information, and booking details.
- Enhance staff productivity: Free up staff time for guest interaction and value-added services.

ii)   Enhance Customer Experience:
- User-friendly interface: Design a clean, intuitive booking system with clear navigation and concise instructions.
- Mobile accessibility: Develop a responsive booking system optimized for mobile devices and tablets.
- Personalized recommendations: Suggest customized room options and amenities based on guest preferences.
- Seamless booking process: Allow quick and easy reservations with minimal steps and interruptions.
- Real-time updates: Inform guests about room availability, check-in times, and service schedules in real-time.

iii)   Optimize Resource Management:

- Dynamic pricing: Adjust room rates based on demand, seasonality, and other factors to maximize revenue.
- Real-time inventory updates: Track room availability in real-time to avoid overbooking and lost revenue.
- Efficient staff scheduling: Optimize staff schedules based on guest occupancy and anticipated workload.
- Automated maintenance: Schedule housekeeping and maintenance services automatically to optimize labor usage.
- Data-driven decision making: Utilize reservation data to analyze guest trends and inform future business strategies.

Purpose:

The Moyai Hotel Booking Information system is a tool used by hotels to allow guests to make easy, secure, direct reservations online. It can also be known as a central reservation system and helps hoteliers get more bookings through their most profitable channels. This might include powerful online travel agents like Booking.com and Expedia, the hotel's own website and Facebook page, or metasearch sites like Google Hotel Ads. The system integrates with other key tools such as a channel manager and booking engine to ensure the hotel's rates and availability are always up-to-date and that valuable direct reservations can be accepted. It also allows hoteliers to access lots of valuable performance data to help with decision making.

SECTION B: FLOW CHART AND CLASS DIAGRAMS

Flow Chart:
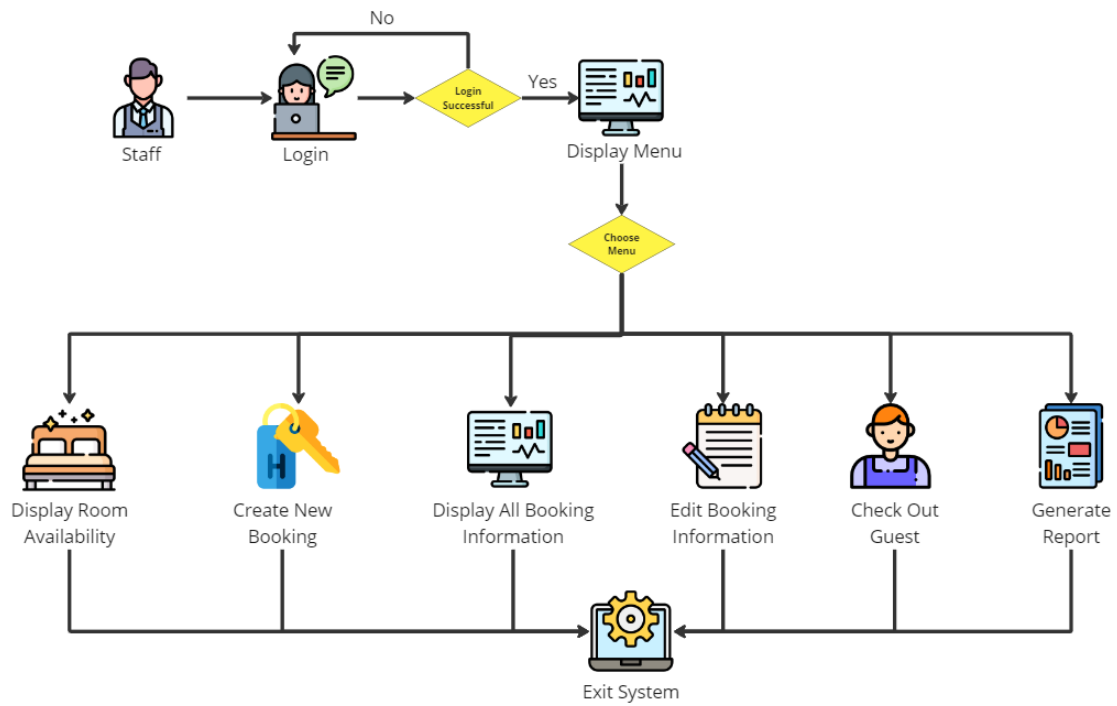
Class Diagrams:
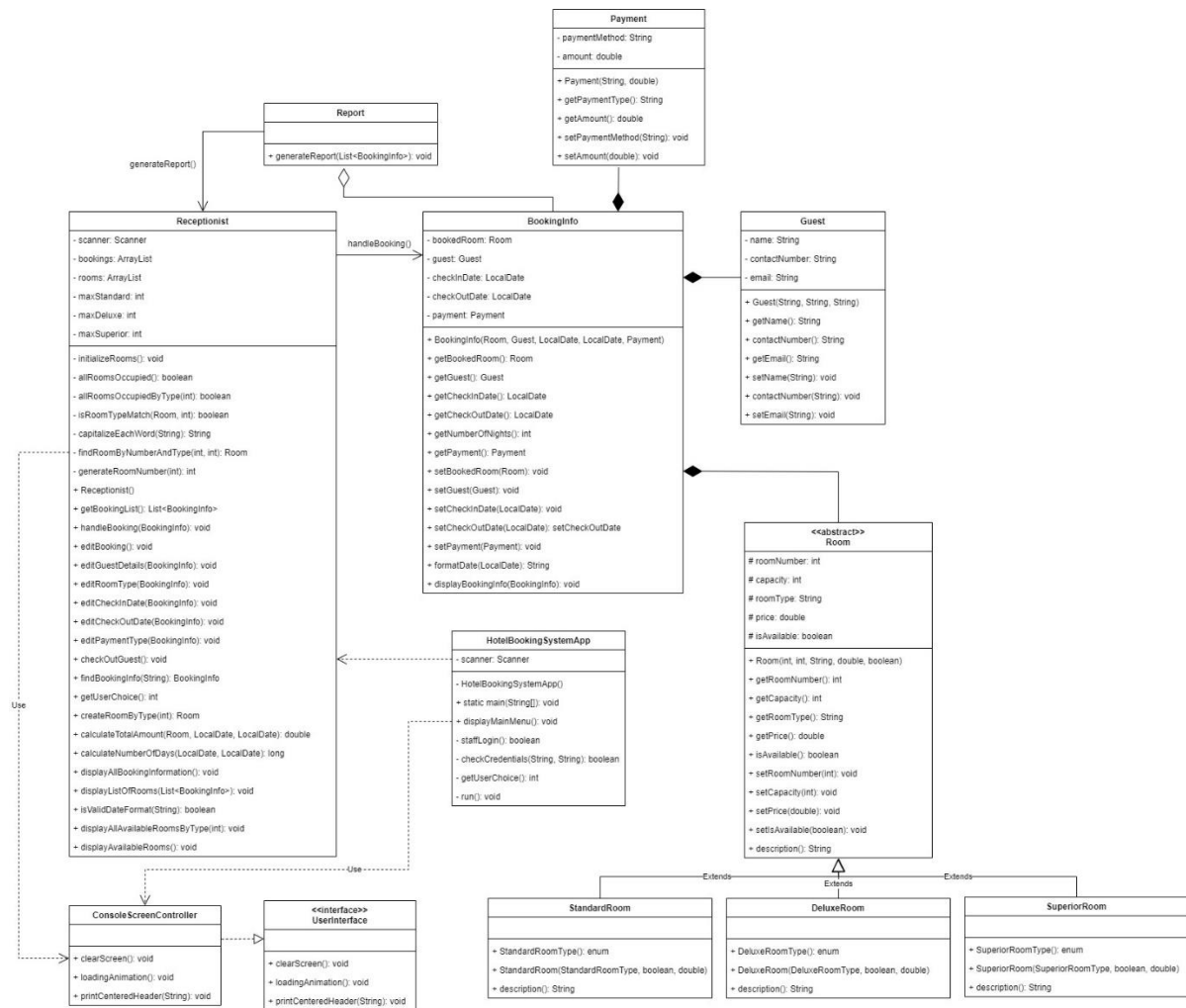
Class BookingInfo:

Attributes	Description
bookedRoom: Room	Represents the booked room
guest: Guest	Represents the guest
checkIn:Date	Represents the check-in date
checkout:Date	Represents the check-out date
payment: Payment	Represents the payment
Methods	**Description**
BookingInfo(Room, Guest, LocalDate, Local Date, Payment)	This seems like a constructor or a method to create a booking
getBookedRoom( ): Room	Returns the bookedRoom attribute.
getGuest( ): Guest	Returns the guest attribute.
getCheckInDate( ): LocalDate	Returns the checkInDate attribute.
getCheckOutDate( ): LocalDate	Returns the checkOutDate attribute.
getNumberOfNights( ):int	Returns the number of nights stay

getPayment( ): Payment	Returns the payment attribute.
setBookedRoom(Room): void	Sets the bookedRoom attribute to the provided Room object.
setGuest(Guest): void	Sets the guest attribute to the provided Guest object.
setCheckInDate(LocalDate): void	Sets the checkInDate attribute to the provided LocalDate object.
setCheckOutDate(LocalDate): void	Sets the checkOutDate attribute to the provided LocalDate object.
setPayment(Payment): void	Sets the checkOutDate attribute to the provided LocalDate object.
displayBookingInfo(BookingInfo)	Returns the check-out date
formatDate(LocalDate):String	Returns the name of the guest

Class HotelBoookingSystemApp:

Attributes	Description
scanner: Scanner	Represents scanner to read input from the user.
Operations	**Description**
HotelBookingSystemApp( )	This method used to initialize the scanner
static main(String[ ]):void	This seems to be the main method, which is the entry point of a Java program.
displayMainMenu( ):void	This method appears to be responsible for displaying the main menu of the program
staffLogin( ): boolean	This method authenticates the user by checking their credentials against a text file.
checkCredentials(String, String): boolean	This helper method for staffLogin() reads the credentials file that have username and password stored in it
getUserChoice( ): int	This method reads an integer input from the user and returns it.
run( ): void	The main method displays the main menu and handles user input.

Class Guest:

Attributes	Description
name:String	Represents the name of the guest
contactNumber:String	Represents the contact number of the guest
email:String	Represents the email address of the guest
Methods	**Description**
Guest(String,String,String)	The constructor for the Guest class, taking three parameters (name, email, and contactNumber) to initialize the

	guest object.
getName( ): String	Returns the value of the name
contactNumber( ): String	Returns the value of the contactNumber.
getEmail( ): String	Returns the value of the email
setName(String): void	Sets the value of the name
contactNumber(String): void	Sets the value of the contactNumber
setEmail(String): void	Sets the value of the email

Class Room:

Attributes	Description
Protected roomNumber: int	Represents the room number. The protected keyword indicates that this attribute is accessible within the class and its subclasses.
Protected roomType: String	Represents the type of the room. Also protected for access within the class and its subclasses.
Protected capacity: int	Represents the capacity of the room. Also protected for access within the class and its subclasses.
Protected price: double	Represents the price of the room. Also protected for access within the class and its subclasses.
Protected isAvailable: boolean	Represents the availability of the room. Also protected for access within the class and its subclasses.
Methods	**Description**
Room (int, int, String, double, boolean)	This seems to be a constructor for the Room class, taking five parameters (roomNumber, capacity, roomType, price, and isAvalable) to initialize the room object.
getRoomNumber( ): int	Returns the room number represented as integer.
getCapacity( ): int	
getRoomType( ): String	Returns the room type, likely represented as a String
getPrice( ): double	Returns the price of the room, likely represented as a double.
getIsAvailable( ): boolean	Returns a boolean value indicating whether the room is available.
setRoomNumber(int)	Sets the value of RoomNumber
setCapacity (int)	Sets the value of Capacity
setPrice (int)	Sets the value of price
setIsAvailable (boolean)	Sets the value of IsAvailable

description ( ): String	This method is intended to provide a textual description of the room.

Class StandardRoom:

Attributes	Description
Operations	Description
StandardRoomType( ): enum	The StandardRoomType enumeration used to define the different types of standard rooms and their associated base prices.
StandardRoom(StandardRoomType, boolean, double)	The constructor takes a StandardRoomType, a boolean, and a double as arguments and initializes the room type, availability, and room price for the StandardRoom object.
description( ): String	This method is intended to provide a textual description of the room.

Class DeluxeRoom:

Attributes	Description
Operations	Description
DeluxeRoomType( ): enum	The DeluxeRoomType enumeration used to define the different types of standard rooms and their associated base prices.
DeluxeRoom (DeluxeRoom Type, boolean, double)	The constructor takes a DeluxeRoomType, a boolean, and a double as arguments and initializes the room type, availability, and room price for the DeluxeRoom object.
description( ): String	This method is intended to provide a textual description of the room.

Class SuperiorRoom:

Attributes	Description
Operations	Description
SuperiorRoomType( ): enum	The SuperiorRoomType enumeration used to define the different types of standard rooms and their associated base prices.
SuperiorRoom(SuperiorRoomType, boolean, double)	The constructor takes a SuperiorRoomType, a boolean, and a double as arguments and initializes the room type, availability, and room price for the SuperiorRoom object.
description( ): String	This method is intended to provide a textual description of the room.

Class Report:

Attributes	Description
Operations	**Description**
generateReport( ):void	This operation represents a method named generateReport. It doesn't take any parameters (void), and its purpose is to generate a report. The implementation details of this method would depend on the specific requirements of your program.

Class Receptionist:

Attributes	Description
private scanner: Scanner	Represents a Scanner object used to read input from the user.
private bookings: ArrayList	Represents to store a collection of Booking objects
private rooms: ArrayList	Represents to store a collection of Booking objects
private maxStandard: int	Represents the maximum number of Standard rooms.
private maxDeluxe: int	Represents the maximum number of Deluxe rooms.
private maxSuperior: int	Represents the maximum number of Superior rooms.
Methods	**Description**
Receptionist():	The constructor method that initializes the scanner object and creates and initializes the rooms list with the specified number of standard, deluxe, and superior rooms.
getBookingList( ):BookingInfo	This method is named getBookingList and likely should return a list or collection of BookingInfo objects.
initializeRooms( ):void	This method that initializes the rooms list with the specified number of standard, deluxe, and superior rooms.
handleBooking (BookingInfo): void	This method that handles the booking process.
capitalizeEachWord (String): String	This method is to capitalize the first letter of each word in the given input string.
findRoomByNumberAndType (int, int):Room	This method used to search for a specific room in a list of rooms based on its room number and type.
editBooking( ): void	This method allows the user to edit an existing booking.

editGuestDetails (BookingInfo): void	This method used to update the details of a guest in a booking system.
editRoomType (BookingInfo): void	This method used to change the type of room that is associated with a given BookingInfo object.
editCheckInDate (BookingInfo): void	This method takes a BookingInfo object as an argument and parses a new check-in date in the format 'yyyy-MM-dd' using the 'date' command in bash.
editCheckOutDate (BookingInfo): void	This method used to update the check-out date of a BookingInfo object.
editPaymentType (BookingInfo): void	This method used to update the payment type for a given BookingInfo object.
deleteBooking ( ): void	This method is named deleteBooking and is likely responsible for deleting or canceling an existing booking.
findBookingInfo (String): BookingInfo	This method is to search through a collection of BookingInfo objects.
getUserChoice ( ): int	This method is to handles invalid user input and clearing the buffer.
createRoomByType (int): Room	This method used to create a room of a specific type, mark it as booked, and assign a room number to it.
generateRoomNumber (int): int	The method uses a switch statement to determine the room number based on the value of roomTypeChoice.
calculateTotalAmount (Room, LocalDate, LocalDate) double) :	This method calculates the total amount to be charged for a room rental, given the room, check-in date, and check-out date.
calculateNumberOfDays (LocalDate, LocalDate): long	This method can be used to calculate the number of days between any two LocalDate objects
displayAllBookingInformation( ): void	The method used to display all the booking information currently stored in the bookings list.
isValidDateFormat(String): boolean	This method checks if a given string is in the format of 'dd-MM-yyyy'.
displayAvailableRoomsByType (int): void	This method is responsible for printing a list of available rooms that match the selected room type.
displayAvailableRooms( ): void	This method takes an array of objects as an argument, where each object represents a room with properties 'roomNumber' and 'roomStatus'.
isRoomTypeMatch(Room, int): boolean	This method is to check if the type of the given Room object matches the

	roomTypeChoice parameter.

Class Payment:

Attributes	Description
private paymentMethod: String	Represents the type of payment method
private amount: double	Represents the amount of payment
Operations	**Description**
Payment (String, double)	The constructor takes in two parameters, paymentMethod and amount, and initializes the corresponding attributes with these values.
getPaymentType ( ): String	Returns the value of the paymentMethod
getAmount ( ): double	Returns the value of the amount
setPaymentMethod (String)	Sets the value of the paymentMethod
setAmount (double): void	Sets the value of the amount

Interface UserInterface:

Attributes	Description
Operations	**Description**
clearScreen( ): void	This method intended to clear any previous output.
loadingAnimation ( ): void	This method intended to show loading animation.
printCenteredHeader ( ): void	This method used for print a centered header on the screen.

Class ConsoleScreenController:

Attributes	Description
Operations	**Description**
clearScreen( ): void	This method intended to clear any previous output.
loadingAnimation ( ): void	This method intended to show loading animation.
printCenteredHeader ( ): void	This method used for print a centered header on the screen.

SECTION C: OO CONCEPTS AND USER MANUAL

OO Concepts

.1.1   Association

```java
public void handleBooking(BookingInfo bookingInfo) {                          //Association - BookingInfo Class
    String fullName, contactNumber, email, checkInDateString, checkOutDateString;
    LocalDate checkInDate = null;
    LocalDate checkOutDate = null;
    double totalAmount;

    if (allRoomsOccupied()) {
        System.out.println(x:"Sorry, all rooms are currently occupied. No new bookings can be made at the moment.");
        return;
    }

    System.out.println(x:"----- Make New Booking -----");
    System.out.print(s:"\nChoose a room type \n(1 for Standard, 2 for Deluxe, 3 for Superior): ");

    int roomTypeChoice = getUserChoice();

    if (roomTypeChoice == 1 && allRoomsOccupiedByType(roomType:1)) {
        System.out.println(x:"Sorry, all Standard rooms are currently occupied. Please choose another room type.\n");
        return;
    } else if (roomTypeChoice == 2 && allRoomsOccupiedByType(roomType:2)) {
        System.out.println(x:"Sorry, all Deluxe rooms are currently occupied. Please choose another room type.\n");
        return;
    } else if (roomTypeChoice == 3 && allRoomsOccupiedByType(roomType:3)) {
        System.out.println(x:"Sorry, all Superior rooms are currently occupied. Please choose another room type.\n");
        return;
    }

    displayAvailableRoomsByType(roomTypeChoice);

    // Assign room number manually
    System.out.print(s:"Enter the desired room number: ");
    int roomNumber = scanner.nextInt();

    userInterface.clearScreen();
```

.1.2 Composition

```java
public class BookingInfo {

    // Composition: The class has member variables representing other classes, forming a composition.
    private Room bookedRoom;
    private Guest guest;
    private LocalDate checkInDate;
    private LocalDate checkOutDate;
    private Payment payment;

    // Constructor: Initializes the BookingInfo object with provided values during object creation.
    public BookingInfo(Room bookedRoom, Guest guest, LocalDate checkInDate, LocalDate checkOutDate, Payment payment) {
        // Encapsulation: Setting private fields using constructor parameters.
        this.bookedRoom = bookedRoom;
        this.guest = guest;
        this.checkInDate = checkInDate;
        this.checkOutDate = checkOutDate;
        this.payment = payment;
    }

    // Getter methods: Provide access to the private fields, following encapsulation principles.
    public Room getBookedRoom() {
        return bookedRoom;
    }

    public Guest getGuest() {
        return guest;
    }

    public LocalDate getCheckInDate() {
        return checkInDate;
    }

    public LocalDate getCheckOutDate() {
        return checkOutDate;
    }
```

.1.3 Inheritance, Polymorphism and Encapsulation

```java
// Inheritance: DeluxeRoom extends the Room class, inheriting its attributes and methods.
public class DeluxeRoom extends Room {

    // Enum for DeluxeRoomType with associated base price.
    public enum DeluxeRoomType {
        DELUXE(basePrice:200.0); // The start room number is now managed separately

        private final double basePrice;

        DeluxeRoomType(double basePrice) {
            this.basePrice = basePrice;
        }

        // Encapsulation: Provides access to the private basePrice field.
        public double getBasePrice() {
            return basePrice;
        }
    }

    // Constructor: Initializes DeluxeRoom with specific DeluxeRoomType, availability, and room price.
    public DeluxeRoom(DeluxeRoomType roomType, boolean availability, double roomPrice) {
        // Constructor chaining: Calls the superclass constructor to initialize common attributes.
        super(roomNumber:0, capacity:6, roomType:"Deluxe", roomPrice, availability);
    }

    // Polymorphism and Method Override: Overrides the description method in the base class Room.
    @Override
    public String description() {
        return "Deluxe Room with capacity of " + getCapacity() + " guest.";
    }
}
```

14

.1.4   Abstraction

```java
// Abstraction: Declares an abstract class named Room, serving as a base class for specific types of rooms.
abstract class Room {

    // Encapsulation: Protected fields to encapsulate the internal state, accessible to subclasses.
    protected int roomNumber;
    protected int capacity;
    protected String roomType;  // Added roomType
    protected double price;
    protected boolean isAvailable;

    // Encapsulation and Constructor: Initializes the common attributes of a room during object creation.
    public Room(int roomNumber, int capacity, String roomType, double price, boolean isAvailable) {
        this.roomNumber = roomNumber;
        this.capacity = capacity;
        this.roomType = roomType;   // Set roomType in the constructor
        this.price = price;
        this.isAvailable = isAvailable;
    }
```

.1.5   Text file operation

Create new file:

```java
public class Report {
    UserInterface userInterface = new ConsoleScreenController();

    public void generateReport(List<BookingInfo> bookings) throws IOException {
        if (bookings.isEmpty()) {
            System.out.print(s:"No bookings to generate. Please create a new booking! \nNavigating to the main menu");
            userInterface.loadingAnimation();
            userInterface.clearScreen();
            return;
        }

        FileWriter fw = new FileWriter(fileName:"./report/Report.txt"); //Create new folder with Report.txt in the folder
        BufferedWriter bw = new BufferedWriter(fw);
        PrintWriter pw = new PrintWriter(bw);

        LocalDateTime now = LocalDateTime.now();
        DateTimeFormatter dateFormatter = DateTimeFormatter.ofPattern(pattern:"dd-MM-yyyy");
        String currentDate = now.format(dateFormatter);

        // Calculate the center alignment for the hotel title
        int titleLength = "Moyai Hotel Booking Information Report".length();
        int padding = (130 - titleLength) / 2;
        String titlePadding = " ".repeat(Math.max(a:0, padding));

        // Print the report header
        pw.printf(format:"%s%s%s[ %3s ]%n", titlePadding, "Moyai Hotel Booking Information Report", titlePadding, currentDa
        pw.println(x:"----------------------------------------------------------------------------------------------------
        pw.printf(format:"%-2s | %-12s | %-25s | %-10s | %-16s | %-16s | %-8s | %-15s | %-17s%n",
                ...args:"#", "Room Number", "Guest Name", "Room Type", "Check-in Date", "Check-out Date","Night(s)", "Booki
        pw.println(x:"---+--------------+-------------------------+------------+------------------+------------------+---

        // Sort bookings by room number
        Collections.sort(bookings, new Comparator<BookingInfo>() {
            //@Override
            public int compare(BookingInfo b1, BookingInfo b2) {
                Integer roomNumber1 = b1.getBookedRoom().getRoomNumber();
                Integer roomNumber2 = b2.getBookedRoom().getRoomNumber();
                return roomNumber1.compareTo(roomNumber2);
```

Read input file:

```java
private boolean checkCredentials(String username, String password) {
    try {
        File file = new File(pathname:"C:\\Users\\ASUS\\Desktop\\Degree UTM\\SEMESTER 1\\OOP\\Projek\\HotelBookingSystem
        Scanner fileScanner = new Scanner(file);

        while (fileScanner.hasNextLine()) {
            String line = fileScanner.nextLine();
            String[] credentials = line.split(regex:",");

            if (credentials.length == 2) {
                String storedUsername = credentials[0];
                String storedPassword = credentials[1];

                if (username.equals(storedUsername) && password.equals(storedPassword)) {
                    fileScanner.close();
                    return true;
                }
            } else {
                System.out.println(x:"Error: Invalid credentials format in the file.");
            }
        }

        fileScanner.close();
    } catch (FileNotFoundException e) {
        System.out.println(x:"Error: Credentials file not found. Please create new credential file");
        System.exit(status:0);
    }

    return false;
}
```

User Manual

1. Staff need to login before they can use this system.

2. After staff input their username and password, staff can access the Moyai Hotel Booking System.

Staff Login

Username: admin
Password: admin

----- Moyai Hotel Booking System -----

[1] Display Room Availability
[2] Create Booking
[3] Display All Booking Information
[4] Edit Booking
[5] Check-Out Guest
[6] Generate Report
[7] Exit

Enter your choice: █

---
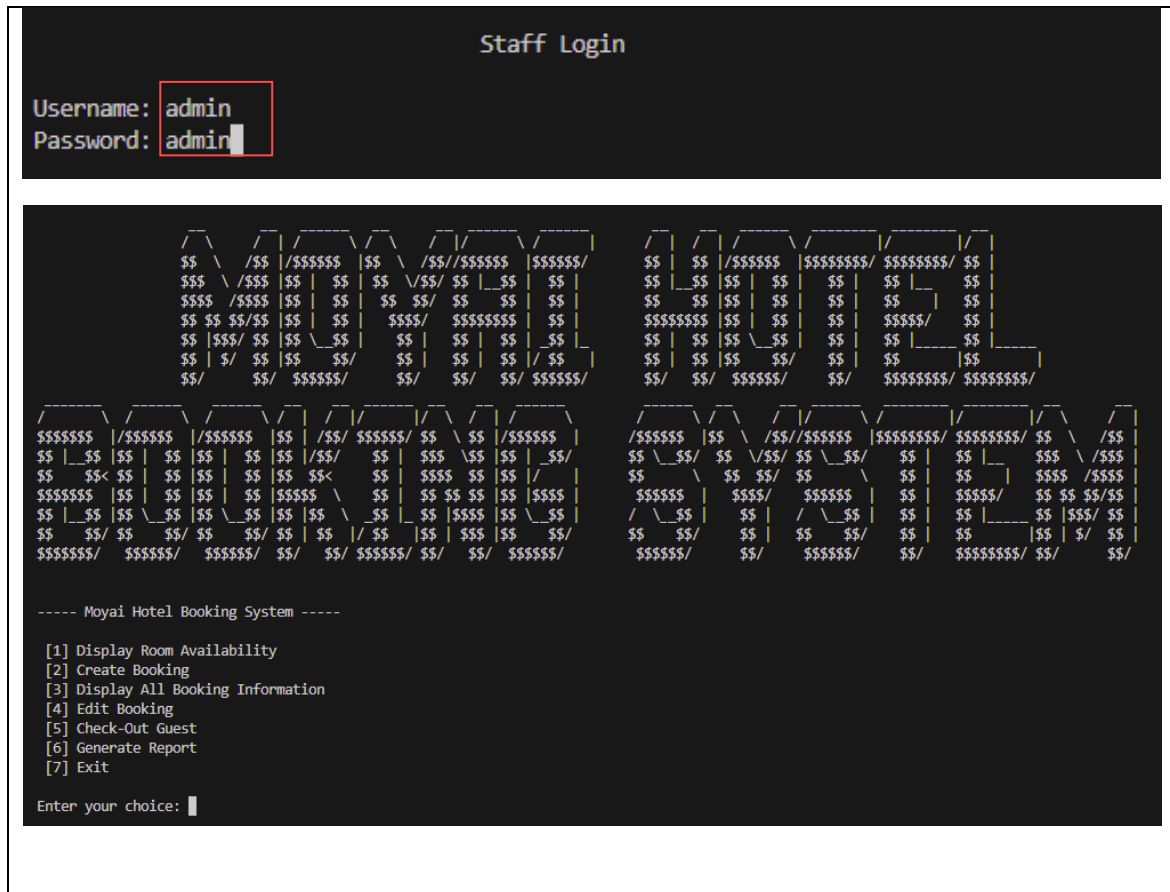
3. Staff need to enter choice 1 to check the Display Room Availability.

4. After that system show the Display All Type of Room Availabilty. Press Enter to go back to main menu.

----- Moyai Hotel Booking System -----

[1] Display Room Availability
[2] Create Booking
[3] Display All Booking Information
[4] Edit Booking
[5] Check-Out Guest
[6] Generate Report
[7] Exit

Enter your choice: 1

```
                      ----- Room Availability -----

--- Available Standard Rooms ---
[100] - Available
[101] - Available
[102] - Available
[103] - Available
[104] - Available

--- Available Deluxe Rooms ---
[200] - Available
[201] - Available
[202] - Available
[203] - Available
[204] - Available

--- Available Superior Rooms ---
[300] - Available
[301] - Available
[302] - Available
[303] - Available
[304] - Available

--- Booked Standard Rooms ---

--- Booked Deluxe Rooms ---

--- Booked Superior Rooms ---

----------------------------

Press Enter to Continue...
```

5. When staff want to create booking for the customers, staff need to enter choice 2 in main menu Create Booking.

6. Staff can choose what type of room which are 1 for Standard Room, 2 for Deluxe, 3 for Superior for new booking

7. Staff can choose the available room for the Standard Room by enter the room number.

8. Staff need to enter all the information for the customer, after that booking successful.

9. Staff can choose whether want to view the booking details by enter 1 for Yes 2 for No.

```
Enter guest full name: Eizkhan
Enter guest contact number: 012345678987
Enter guest email: eizkhan@gmail.com
Enter check-in date (dd-MM-yyyy): 16-01-2024
Enter check-out date (dd-MM-yyyy): 18-01-2024
Select payment method:
1. Cash
2. Debit/Credit Card
3. eWallet
Enter your choice: 1
Booking successful!
Do you want to view the booking details? (1 - Yes/ 0 - No) : 1
```

```
----- Guest: Eizkhan Booking Information -----
        Room Number        : 100
        Guest              : Eizkhan
        Contact Number     : 012345678987
        Email              : eizkhan@gmail.com
        Check-in Date      : Tue, 16 Jan 2024
        Check-out Date     : Thu, 18 Jan 2024
        Number of Night(s) : 2
        Total Amount       : RM200.00
        Payment Type       : Cash
        Room Type          : Standard
        Description        : Standard Room with capacity of 2 guest.
------------------------------------------------------------------

Navigating to main menu..
```

10. When staff want to see all booking information, staff need to enter choice 3

11. Then, all the booking information pop up and press Enter to continue to main menu.

12. Staff can edit booking bye enter choice 4

13. Staff can choose what booked room number to edit the booking info for that room.

14. Staff select what information to edit, if staff want to edit guest, enter choice 1

15. After done editing all guest details, staff need to press enter to continue

```
----- Guest: Eizkhan Booking Information -----
        Room Number         : 100
        Guest               : Eizkhan
        Contact Number      : 012345678987
        Email               : eizkhan@gmail.com
        Check-in Date       : Tue, 16 Jan 2024
        Check-out Date      : Thu, 18 Jan 2024
        Number of Night(s)  : 2
        Total Amount        : RM200.00
        Payment Type        : Cash
        Room Type           : Standard
        Description         : Standard Room with capacity of 2 guest.
-----------------------------------------------------------------

Select what to edit:
1. Guest Details
2. Check-in Date
3. Check-out Date
4. Payment Type
5. Cancel
Enter your choice: 1
```

```
 ----- Guest: Eizkhan Booking Information -----
        Room Number         : 100
        Guest               : Eizkhan
        Contact Number      : 012345678987
        Email               : eizkhan@gmail.com
        Check-in Date       : Tue, 16 Jan 2024
        Check-out Date      : Thu, 18 Jan 2024
        Number of Night(s)  : 2
        Total Amount        : RM200.00
        Payment Type        : Cash
        Room Type           : Standard
        Description         : Standard Room with capacity of 2 guest.
-----------------------------------------------------------------

Select what to edit:
1. Guest Details
2. Check-in Date
3. Check-out Date
4. Payment Type
5. Cancel
Enter your choice: 1
Enter new guest full name: Ahmad Nabil
Enter new guest contact number: 01754893281
Enter new guest email: nabil@gmail.com
Guest details updated successfully. <Press enter to continue>
```

16. When staff wants to edit Check-in Date, staff need to enter choice 2 and then edit the new date.

17. When staff wants to edit Check-out Date, staff need to enter choice 3 and then edit the new date.

```
----- Guest: Ahmad Nabil Booking Information -----
        Room Number       : 100
        Guest             : Ahmad Nabil
        Contact Number    : 01754893281
        Email             : nabil@gmail.com
        Check-in Date     : Tue, 16 Jan 2024
        Check-out Date    : Thu, 18 Jan 2024
        Number of Night(s) : 2
        Total Amount      : RM200.00
        Payment Type      : Debit/Credit Card
        Room Type         : Standard
        Description       : Standard Room with capacity of 2 guest.
----------------------------------------------------------------

Select what to edit:
1. Guest Details
2. Check-in Date
3. Check-out Date
4. Payment Type
5. Cancel
Enter your choice: 2
Enter new check-in date (dd-MM-yyyy): 20-01-2024
```

```
----- Guest: Ahmad Nabil Booking Information -----
        Room Number          : 100
        Guest                : Ahmad Nabil
        Contact Number       : 01754893281
        Email                : nabil@gmail.com
        Check-in Date        : Sat, 20 Jan 2024
        Check-out Date       : Thu, 18 Jan 2024
        Number of Night(s)  : -2
        Total Amount         : RM200.00
        Payment Type         : Debit/Credit Card
        Room Type            : Standard
        Description          : Standard Room with capacity of 2 guest.
------------------------------------------------------------------------


Select what to edit:
1. Guest Details
2. Check-in Date
3. Check-out Date
4. Payment Type
5. Cancel
Enter your choice: 3
Enter new check-out date (dd-MM-yyyy): 23-01-2024
```

18. When staff want to edit customer payment type, enter choice 4 and choose new payment method.

```
----- Guest: Ahmad Nabil Booking Information -----
        Room Number         : 100
        Guest               : Ahmad Nabil
        Contact Number      : 01754893281
        Email               : nabil@gmail.com
        Check-in Date       : Sat, 20 Jan 2024
        Check-out Date      : Tue, 23 Jan 2024
        Number of Night(s)  : 3
        Total Amount        : RM200.00
        Payment Type        : Cash
        Room Type           : Standard
        Description         : Standard Room with capacity of 2 guest.
--------------------------------------------------------------------

Select what to edit:
1. Guest Details
2. Check-in Date
3. Check-out Date
4. Payment Type
5. Cancel
Enter your choice: 4

Select new payment method:
1. Cash
2. Debit/Credit Card
3. eWallet
Enter your choice: 2
Payment type updated successfully. <Press enter to continue>
```

19. Staff can check the latest booking information by enter choice 3 in main menu to see whether the booking information are changed or not.

```
  __   _  ___ ___  _   ___  ___       __  _  ____  __ ___  _____  __
 / \  / | /___ \/ \  / |/___ \ ____|   / |/ | /___ \ /___/ |___/ |/ |
$$ \  /$$ |/$$$$$ |$$  \ /$$//$$$$$$ |$$$$$$/   $$ |  $$ |/$$$$$ |$$$$$$$/ $$$$$$$$/ $$ |
$$$ \ /$$$ |$$ |  $$ | $$ \/$$/ $$ |_$$ |  $$ |   $$ |_$$ |$$ |  $$ |  $$ |  $$ |__   $$ |
$$$$ /$$$$ |$$ |  $$ |  $$ $$/  $$ |   $$ |  $$ |   $$ | $$ |$$ |  $$ |  $$ |  $$ |   $$ |
$$ $$ $$/$$ |$$ |  $$ |   $$$$/  $$$$$$$$ |  $$ |   $$$$$$$$ |$$ |  $$ |  $$ |  $$$$$/  $$ |
$$ |$$$/ $$ |$$ \__$$ |   $$ |  $$ |  $$ |_  $$ |   $$ |  $$ |$$ \__$$ |  $$ |  $$ |____ $$ |_____
$$ | $/ $$ |$$  $$/   $$ |  $$ |  $$ |/ $$  |  $$/   $$ |  $$ |$$  $$/   $$ |  $$     |$$       |
$$/     $$/ $$$$$$/   $$/   $$/   $$/ $$$$$$/   $$/    $$/  $$/ $$$$$$/   $$/   $$$$$$$$/ $$$$$$$$/

 _____   _____   _____    __  ___  __  __    ___  __ ___   _____   ____    __ ___ __  __  __   _  __
/      \ /      \ /      \ | /  |/ | |  |/ | /      \ | /      \ /$$$$$$$ |$$$$$$$$/ $$$$$$$$/ $$ \  / $$ |
$$$$$$$ |/$$$$$$ |/$$$$$$ | $$ | /$$/ $$$$$$/ $$ \ $$ |/$$$$$$$ | /$$$$$$ |$$ \  /$$//$$//$$$$$$ $$ \  /$$$ |
$$ |__$$ | $$ |  $$ |$$ |  $$ |/$$ |  $$ |  $$$$ \$$ |$$ |__$$ | $$ \__$$/ $$  \/$$ $$ \__$$ |  $$ |__ $$$$ \ /$$$ |
$$    $$< $$ |  $$ |$$ |  $$ |$$ |  $$ |$$$$   $$  $$ |$$    $$< $$      \  $$ $$/ $$    \ $$ |   $$$$ /$$$$ |
$$$$$$$ |$$ |  $$ |$$ |  $$ |$$$$$   \  $$ |  $$ |$$$$ |$$$$$$$ | $$$$$$  |  $$$$/  $$$$$$  |$$ |   $$ $$ $$/$$ |
$$ |__$$ |$$ \__$$ |$$ \__$$ |$$ |$$$ \ $$ \ $$ | $$ |$$ |__$$ | / \__$$ |   $$ |  / \__$$ |  $$ |_____$$ |$$$/ $$ |
$$    $$/ $$    $$/ $$    $$/ $$ | $$  |$$ |$$$ |$$  | $$ |$$    $$/ $$    $$/   $$ |  $$    $$/  $$ |    $$ | $/ $$ |
$$$$$$$/  $$$$$$/   $$$$$$/  $$/   $$/ $$/  $$/ $$/   $$/ $$$$$$$/   $$$$$$/    $$/    $$$$$$$$/ $$/     $$/

----- Moyai Hotel Booking System -----

 [1] Display Room Availability
 [2] Create Booking
 [3] Display All Booking Information
 [4] Edit Booking
 [5] Check-Out Guest
 [6] Generate Report
 [7] Exit

Enter your choice: 3
```

```
                ----- All Booking Information -----

----- Guest: Ahmad Nabil Booking Information -----
        Room Number       : 100
        Guest             : Ahmad Nabil
        Contact Number    : 01754893281
        Email             : nabil@gmail.com
        Check-in Date     : Sat, 20 Jan 2024
        Check-out Date    : Tue, 23 Jan 2024
        Number of Night(s) : 3
        Total Amount      : RM200.00
        Payment Type      : Debit/Credit Card
        Room Type         : Standard
        Description       : Standard Room with capacity of 2 guest.
-----------------------------------------------------------------
Press Enter to Continue...█
```

20. Staff can enter choice 5 if staff want to cancel edit.

```
----- Guest: Ahmad Nabil Booking Information -----
        Room Number       : 100
        Guest             : Ahmad Nabil
        Contact Number    : 01754893281
        Email             : nabil@gmail.com
        Check-in Date     : Sat, 20 Jan 2024
        Check-out Date    : Tue, 23 Jan 2024
        Number of Night(s) : 3
        Total Amount      : RM200.00
        Payment Type      : Debit/Credit Card
        Room Type         : Standard
        Description       : Standard Room with capacity of 2 guest.
-----------------------------------------------------------------

Select what to edit:
1. Guest Details
2. Check-in Date
3. Check-out Date
4. Payment Type
5. Cancel
Enter your choice: 5█
```

21. Staff can generate report for the booking information by enter choice 6 in main menu.

22. System tells the report has been successfully generated in local files and go back to main menu.

23. Staff can find the report on their local files in the computer to see the report in txt file.


```
----- Moyai Hotel Booking System -----

[1] Display Room Availability
[2] Create Booking
[3] Display All Booking Information
[4] Edit Booking
[5] Check-Out Guest
[6] Generate Report
[7] Exit

Enter your choice: 6
```

```
The report has been successfully generated!
Navigating to the main menu.
```

Report - Notepad
File Edit Format View Help

```
|                          Moyai Hotel Booking Information Report                          [ 16-01-2024 ]
---------------------------------------------------------------------------------------------------------
# | Room Number  | Guest Name            | Room Type | Check-in Date   | Check-out Date  | Night(s) | Booking Amount | Payment Type
--+--------------+-----------------------+-----------+-----------------+-----------------+----------+----------------+-------------
1 | 100          | Ahmad Nabil           | Standard  | Sat, 20 Jan 2024 | Tue, 23 Jan 2024 |    3     | RM 200.0       | Debit/Credit Card
```

24. Staff can update when the guest check-out from the hotel by enter choice 5.

25. Staff can choose which room number have to check-out and staff can refer the room number from the list.


```
----- Moyai Hotel Booking System -----

 [1] Display Room Availability
 [2] Create Booking
 [3] Display All Booking Information
 [4] Edit Booking
 [5] Check-Out Guest
 [6] Generate Report
 [7] Exit
Enter your choice: 5
```

```
                          ----- Check Out Guest -----

----- List of Booked Room Number -----
[100]


Choose the room number to check out: 100
```

26. Staff can exit the system by enter choice 7.

GITHUB

Group 2 Github link:

https://github.com/jjn7702/SECJ2154-
OOP/tree/main/Submission/sec01_perdana/Group2

CONCLUSION

In the age of instant gratification and global travel, hoteliers face a constant challenge: how to streamline booking processes, elevate guest experiences, and stand out in a crowded market. The Moyai Hotel Booking System emerges as a beacon of innovation, addressing these challenges head-on and offering a paradigm shift in hospitality management.

For hotels, Moyai is a game-changer. The system acts as a central nervous system, syncing booking information automatically and ensuring global distribution channels are always kept up-to-date. This translates to improved efficiency, reduced overbooking, and maximized occupancy rates. Real-time data insights empower hoteliers to make informed decisions on pricing, resource allocation, and guest preferences, optimizing operations and driving profitability.

But Moyai's power lies not just in technology, but in its focus on human connection. Staff are equipped with user-friendly tools to manage bookings efficiently, while guests benefit from a smooth, intuitive booking experience. This frees up both parties to focus on what truly matters: building relationships and fostering genuine hospitality.

The implications of Moyai are far-reaching. Increased efficiency allows hotels to expand their reach, attracting guests from all corners of the globe. The system's user-friendly interface and automation features cater to today's tech-savvy traveler, enhancing guest satisfaction and loyalty. By empowering both guests and hoteliers, Moyai fosters a positive, dynamic ecosystem that benefits everyone involved.

In conclusion, the Moyai Hotel Booking System is not simply a software solution; it's a revolution in hospitality management. It empowers hotels to thrive in a competitive landscape, offering guests a seamless booking experience and personalized service. As the travel industry continues to evolve, Moyai stands poised to play a pivotal role, shaping the future of hospitality with its focus on automation, efficiency, and genuine guest interaction.