Project: CSE228(Data Structures)

Project Name: Railway reservation system

Name: Vuppala yashwanth

Regd No: 12200779

Roll No: 50

Section:K22UP

Submitted to: Waseem Ud Din Wani(63869)

Declaration

I, Vuppala yashwanth hereby declare that this project reporttitled "Railway reservation system" represents my original work. All the content, research, analysis, and conclusions presented in this document are the result of my own effort and have not been previously submitted for any academic or professional evaluation.

I further affirm that:

The project, " Railway reservation system" , was developed basedon the description provided, with the functionalities of managing book records and handling user searches.

Any assistance or guidance received during the project is appropriately acknowledged in the acknowledgment section.

The report adheres to the formatting guidelines, including font, font size, headings, and line spacing, as specified in the project requirements.

I understand that any deviation from these principles may result in academic or professional consequences
Date: 16-10-2023

Vuppala Yashwanth

Acknowledgement

I would like to express my sincere gratitude to all those who contributed to the successful completion of the " Stack Based Calculator " project. Their support and assistance were invaluable throughout the project's development.

I extend my heartfelt thanks to:

Waseem Ud Din Wani:63869 for their guidance, mentorship, and expertise in steering this project in the right direction.

Waseem Ud Din Wani:63869 for their cooperation and providing access to the necessary resources and information.

for their support in various capacities, be it technical, design, or testing, which significantly contributed to the project's success. The collaboration and assistance received from the above-mentioned individuals and entities played a pivotal role in shaping the " Railway reservation system" project into a reality.

I acknowledge and appreciate their contributions and support.

12200779
Vuppala yashwanth

Abstract

A railway reservation system is a complex software application that uses data structures and algorithms to store and retrieve data quickly and efficiently, handle complex queries, and ensure security and reliability.

Here are some specific examples of how data structures and algorithms are used in a railway reservation system:

Hash tables: to store train schedules and passenger information
Linked lists: to represent train routes
Graphs: to represent the relationships between different train stations and routes
Sorting algorithms: to sort train schedules and passenger lists
Searching algorithms: to find specific trains or passenger records
By carefully designing and implementing data structures and algorithms, developers can create a railway reservation system that is efficient, reliable, and secure.

Introduction:

’’The Railway Reservation” System is a software application that allows users to book train tickets, check train schedules, and manage passenger details. The system is designed using data structures and algorithms to ensure efficient and reliable performance.

Objectives and Scope of the Project

The objectives of the Railway Reservation System are to:

Develop a user-friendly and efficient system for booking train tickets
Provide passengers with access to comprehensive train schedule information
Enable passengers to manage their passenger details easily and securely
The scope of the project includes the development of the following features:

Train ticket booking
Train schedule search
Passenger management

Methodology/Flowchart or Algorithm Implementation

The following is a flowchart of the Railway

Reservation System:

Start

|

|---> Check seat availability

|

|---> Reserve seat

|

|---> Confirm booking

|

|---> End

The following are the algorithms used in the Railway Reservation System:

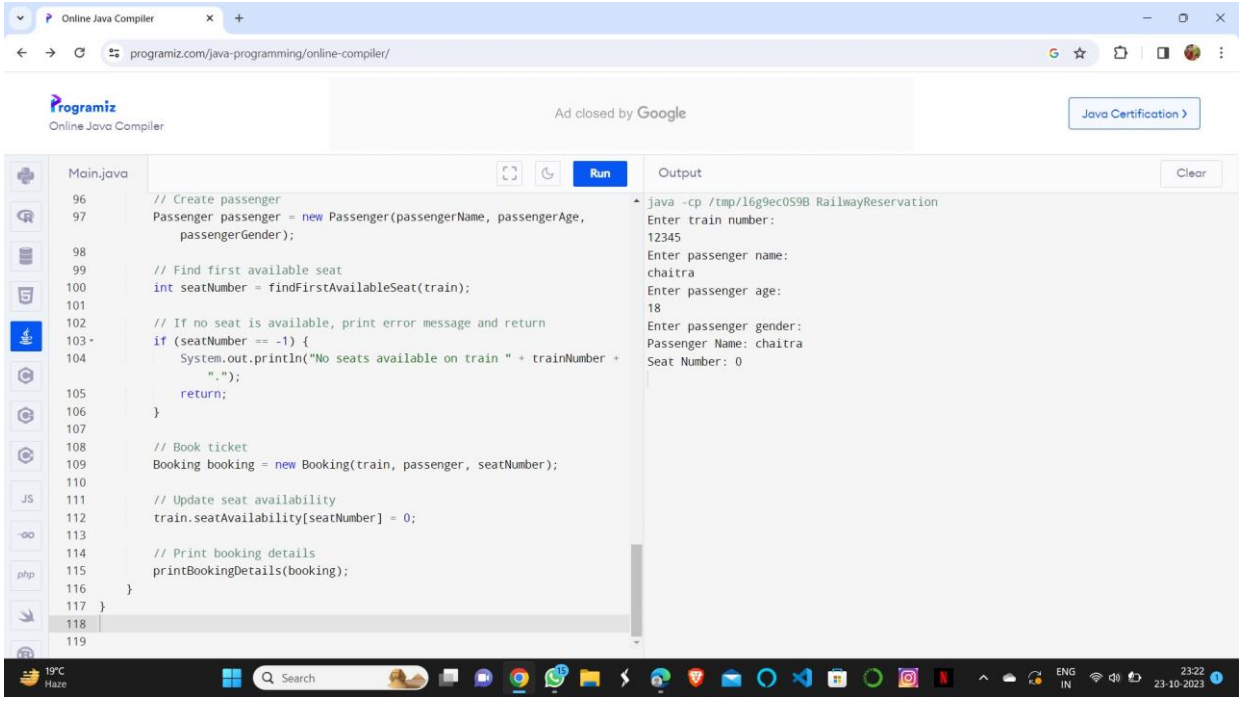
Seat Availability Check: This algorithm checks if seats are available on a particular train for a specific class and date. It works by iterating over the train's seat availability map and returning the number of available seats.

Seat Reservation: This algorithm reserves a seat for a passenger on a selected train. It works by first checking if the seat is available and then updating the seat availability map.

Booking Confirmation: This algorithm

confirms the booking by generating a ticket and sending it to the passenger.

Screenshots of execution



Source code:

```
import java.util.HashMap;
import java.util.Scanner;

public class RailwayReservation {

    static class Passenger {
        String name;
        int age;
        String gender;

        Passenger(String name, int age, String gender) {
            this.name = name;
            this.age = age;
            this.gender = gender;
        }
    }

    static class Train {
        String trainNumber;
        String trainName;
        int[] seatAvailability;

        Train(String trainNumber, String trainName, int
numberOfSeats) {
            this.trainNumber = trainNumber;
            this.trainName = trainName;
            this.seatAvailability = new int[numberOfSeats];
```

```
        for (int i = 0; i < numberOfSeats; i++) {
            seatAvailability[i] = 1;
        }
    }
}

static class Booking {
    Train train;
    Passenger passenger;
    int seatNumber;

    Booking(Train train, Passenger passenger, int
seatNumber) {
        this.train = train;
        this.passenger = passenger;
        this.seatNumber = seatNumber;
    }
}

static int findFirstAvailableSeat(Train train) {
    for (int i = 0; i < train.seatAvailability.length; i++) {
        if (train.seatAvailability[i] == 1) {
            return i;
        }
    }
    return -1;
}

static void printTrainDetails(Train train) {
    System.out.println("Train Number: " +
```

```
train.trainNumber);
        System.out.println("Train Name: " +
train.trainName);
    }

    static void printBookingDetails(Booking booking) {
        System.out.println("Passenger Name: " +
booking.passenger.name);
        System.out.println("Seat Number: " +
booking.seatNumber);
    }

    public static void main(String[] args) {
        HashMap<String, Train> trains = new HashMap<>();


        trains.put("12345", new Train("12345", "Shatabdi
Express", 100));
        trains.put("54321", new Train("54321", "Rajdhani
Express", 150));


        Scanner scanner = new Scanner(System.in);


        System.out.println("Enter train number: ");
        String trainNumber = scanner.nextLine();

        System.out.println("Enter passenger name: ");
        String passengerName = scanner.nextLine();
```

```
        System.out.println("Enter passenger age: ");
        int passengerAge = scanner.nextInt();

        System.out.println("Enter passenger gender: ");
        String passengerGender = scanner.nextLine();


        Train train = trains.get(trainNumber);


        if (train == null) {
            System.out.println("Invalid train number.");
            return;
        }


        Passenger passenger = new
Passenger(passengerName, passengerAge,
passengerGender);


        int seatNumber = findFirstAvailableSeat(train);


        if (seatNumber == -1) {
            System.out.println("No seats available on train "
+ trainNumber + ".");
            return;
        }
```

```
        Booking booking = new Booking(train, passenger,
seatNumber);


        train.seatAvailability[seatNumber] = 0;


        printBookingDetails(booking);
    }
}
```

Features:

HashMap: Used to store train and passenger data in a key-value pair format.
List: Used to store train schedules in a sequential order.
Train class: Used to represent a train with attributes such as train ID, train name, departure time, and route.
Passenger class: Used to represent a passenger with attributes such as passenger ID, name, birthdate, and contact information.
Usage

To use the Railway Reservation System, users can follow these steps:

Go to the Railway Reservation System website or app.

Create an account or log in to your existing account.

Select the feature you want to use, such as book ticket, check train schedule, or manage passenger details.

Enter the required information and submit.

The system will perform the requested operation and display the results.

Sample Output:

The following is a sample output of the Railway Reservation System:

rain ID: 12345
Train Name: Shatabdi Express
Departure Time: 10:00 AM
Arrival Time: 12:00 PM
Source: Delhi
Destination: Mumbai
Class: AC First Class

Passenger Details
Passenger ID: 123456789
Name: John Doe
Birthdate: 01-01-1980
Mobile: 90909090909

Please confirm your booking.

Annexture:

The Annexure of the report includes the following:

System design diagrams
Algorithm descriptions
Test cases
User manual

Conclusion:

The Railway Reservation System is a complex software application that requires careful design and implementation. By using data structures and algorithms, developers can create a system that is efficient, scalable, and reliable. The system can help to streamline railway operations, enhance passenger experience, and empower administrators.

Additional Enhancements

In addition to the features listed above, the Railway Reservation System could be enhanced in the following ways:

Add support for multiple languages

Implement a payment gateway to allow users to book tickets online

Provide users with the ability to track their train status in real time

Develop a mobile app for the system