**AIRLINE RESERVATION SYSTEM**

Project submitted to the

SRM University – AP, Andhra Pradesh

for the partial fulfilment of the requirements to award the degree of

**Bachelor of Technology**

In

**Computer Science and Engineering**

**School of Engineering and Sciences**

Submitted by

**Candidate Name**

**Lakshmi Nikhitha Dodda (AP21110011270)**

**A picture containing text

Description automatically generated**

Under the Guidance of

**(V. Veda Sri)**

**SRM University–AP**

**Neerukonda, Mangalagiri, Guntur**

**Andhra Pradesh – 522 240**

**[MAY,2023]**

**CERTIFICATE**

**Date: 18-May-2023**

This is to certify that the work present in this Project entitled “**Airline Reservation System**” has been carried out by **Lakshmi Nikhitha Dodda** under my/our supervision. The work is genuine, original, and suitable for submission to the SRM University – AP for the award of Bachelor of Technology/Master of Technology in **School of Engineering and Sciences**.

**Supervisor**

(Signature)

Mrs V. Veda Sri

**Co-supervisor**

(Signature)

Dr. Jatindra Kumar Dash

**ACKNOWLEDGEMENT**

The satisfaction that accompanies the successful completion of any task would be incomplete without introducing the people who made it possible and whose constant guidance and encouragement crowns all efforts with success.

We are extremely grateful and express our profound gratitude and indebtedness to our project guide, Veda Sri, Lecturer, Department of Computer Science & Engineering, SRM University, Andhra Pradesh, for his kind help and for giving us the necessary guidance and valuable suggestions in completing this project work.

**Lakshmi Nikhitha Dodda (AP21110011270)**

**Table of Contents**

[Certificate](https://docs.google.com/document/d/175MECKwMKJFIc82PoU15SL8H35G4VBQqWC8UYDCAda4/edit#heading=h.gjdgxs) 2

[Acknowledgements](https://docs.google.com/document/d/175MECKwMKJFIc82PoU15SL8H35G4VBQqWC8UYDCAda4/edit#heading=h.30j0zll) 3

Table of Contents 4

[Abstract](https://docs.google.com/document/d/175MECKwMKJFIc82PoU15SL8H35G4VBQqWC8UYDCAda4/edit#heading=h.3znysh7) 5

[1. Introduction](https://docs.google.com/document/d/175MECKwMKJFIc82PoU15SL8H35G4VBQqWC8UYDCAda4/edit#heading=h.17dp8vu) 6

[2. Objective 7](https://docs.google.com/document/d/175MECKwMKJFIc82PoU15SL8H35G4VBQqWC8UYDCAda4/edit#heading=h.lnxbz9)

[3. Source Code](https://docs.google.com/document/d/175MECKwMKJFIc82PoU15SL8H35G4VBQqWC8UYDCAda4/edit#heading=h.44sinio) 8

4. Conclusion 15

**ABSTRACT**

  The airline reservation system is an online flight booking engine developed to help passengers in reserving their tickets online, access information about the cancellation of a booked ticket, its fare tariffs, rescheduling a ticket, Airline schedules, ticket costs, and PNRs (Passenger Name Records). The airline reservation system has become a critical component of the airline industry, allowing airlines to efficiently manage their operations and provide customers with a convenient and seamless booking experience. The main purpose of the Airline Reservation system is to allow customers to interact with some basic information such as all flight information, availability of accommodation in flights and information about the inventory (i.e. economy, premium economy, business) using java development. This component provides a graphical user interface (GUI) and also Database which is used to store and retrieve data. This has been divided into two modules: booking module and reporting module which are further classified into reservation, cancellation, Confirmed Passenger List & Waiting list. However, with the rapid advancements in technology and the increasing demand for online services, the airline reservation system is expected to continue evolving and improving.

**Keywords:**  Fare tariffs, Accommodation, Inventory, Java development, GUI, Database, Airline schedules, Booking Module, Reports Module.

**INTROCUTION:**

An airline reservation system involves the development of a computerized system for managing flight schedules and passenger bookings for an airline. The project typically includes several key components, including seat availability, booking and ticketing, passenger check-in, and boarding. The project will involve the development of software that can be accessed by both customers and airline staff. Customers will be able to use the system to search for flights, book and purchase tickets, and manage their bookings, while airline staff will be able to use the system to manage flight schedules, seat availability, and passenger information, among other things. This typically involves analyzing the existing system, identifying areas for improvement, and designing and implementing new features to enhance the system's functionality and performance. It requires careful planning, analysis, design, development, implementation, and maintenance to ensure the system meets the airline's needs and provides a positive customer experience. It involves actions such as determining airport strategy and gathering and disseminating information on airline commercial and operational priorities.

**OBJECTIVE:**

The main goal is to provide a user-friendly interface for customers to book and manage their flights and provide efficient management tools for airline staff to manage flight schedules, inventory, and pricing.

* Develop a scalable and robust system.
* Provide an intuitive user interface.
* Implement efficient algorithms.
* Ensure system security.
* Integrate with other systems.

The purpose of this system project in Java is to provide a modern and efficient system that improves the customer experience, streamlines airline operations, increases efficiency and productivity, improves revenue management, and enhances data management while ensuring system security and integrating with other systems.

Overall, the objective is to provide an automated system that replaces the traditional manual processes involved in managing and booking flights.

**SOURCE CODE**

import java.util.InputMismatchException;

import java.util.Scanner;

class Flight {

    private String flightNumber;

    private String source;

    private String destination;

    private String flightType; // Domestic or International

    private int availableSeats;

    public Flight(String flightNumber, String source, String destination, String flightType, int availableSeats) {

        this.flightNumber = flightNumber;

        this.source = source;

        this.destination = destination;

        this.flightType = flightType;

        this.availableSeats = availableSeats;

    }

    public String getFlightNumber() {

        return flightNumber;

    }

    public String getSource() {

        return source;

    }

    public String getDestination() {

        return destination;

    }

    public String getFlightType() {

        return flightType;

    }

    public int getAvailableSeats() {

        return availableSeats;

    }

    public void bookSeats(int numSeats) {

        if (availableSeats >= numSeats) {

            availableSeats -= numSeats;

        }

    }

}

class PaymentModule {

    private static final String DESIRED\_UPI\_ID = "ns@upi";

    private static final String DESIRED\_USERNAME = "ap0916";

    private static final String DESIRED\_PASSWORD = "161834";

    private static final long DESIRED\_CARD\_NUMBER = 8284895969161834L;

    private static final String DESIRED\_EXPIRY\_DATE = "11/25";

    public static boolean processPayment() {

        System.out.println("--------------------------------Payment Portal--------------------------------\n\n");

        System.out.println("1. Credit/Debit Card\n");

        System.out.println("2. UPI\n");

        System.out.println("3. Net Banking\n");

        System.out.println("4. Cancel Payment\n");

        Scanner scanner = new Scanner(System.in);

        int paymentOption = scanner.nextInt();

        switch (paymentOption) {

            case 1:

                System.out.println("CREDIT/DEBIT CARD\n");

                System.out.print("Enter the Card Number:\n ");

                long cardNumber = scanner.nextLong();

                System.out.print("Enter the Expiry date (MM/YY):\n ");

                String expiryDate = scanner.next();

                System.out.print("Enter the CVV:\n");

                int cvv = scanner.nextInt();

                // Check if the entered card number and expiry date are valid

                if (isValidCardDetails(cardNumber, expiryDate)) {

                    // Process the card payment

                    System.out.println("Payment Successful!\n");

                    return true;

                } else {

                    System.out.println("Invalid card number or expiry date. Payment Failed.\n");

                    return false;

                }

            case 2:

                System.out.println("UPI");

                System.out.print("Enter the UPI ID:\n ");

                String upiId = scanner.next();

                // Check if the entered UPI ID is valid

                if (isValidUpiId(upiId)) {

                    // Process the UPI payment

                    System.out.println("Payment Successful!\n");

                    return true;

                } else {

                    System.out.println("Invalid UPI ID. Payment Failed.\n");

                    return false;

                }

            case 3:

                System.out.println("NET BANKING");

                System.out.print("Enter the username:\n ");

                String username = scanner.next();

                System.out.print("Enter the password:\n ");

                String password = scanner.next();

                // Check if the entered username and password are valid

                if (isValidCredentials(username, password)) {

                    // Process the net banking payment

                    System.out.println("Payment Successful!\n");

                    return true;

                } else {

                    System.out.println("Invalid username or password. Payment Failed.\n");

                    return false;

                }

            case 4:

                System.out.println("Payment Cancelled.\n");

                return false;

            default:

                System.out.println("Invalid payment option. Payment Failed.\n");

                return false;

        }

    }

    private static boolean isValidCardDetails(long cardNumber, String expiryDate) {

        // Validate card number and expiry date against desired values

        return cardNumber == DESIRED\_CARD\_NUMBER && expiryDate.equals(DESIRED\_EXPIRY\_DATE);

    }

    private static boolean isValidUpiId(String upiId) {

        // Validate UPI ID against desired value

        return upiId.equals(DESIRED\_UPI\_ID);

    }

    private static boolean isValidCredentials(String username, String password) {

        // Validate username and password against desired values

        return username.equals(DESIRED\_USERNAME) && password.equals(DESIRED\_PASSWORD);

    }

}

class FlightBookingSystem {

    private Flight[] flights;

    public FlightBookingSystem() {

        flights = new Flight[10];

        flights[0] = new Flight("SN1618", "Delhi", "Mumbai", "Domestic", 50);

        flights[1] = new Flight("NS1816", "Delhi", "New York", "International", 100);

        flights[2] = new Flight("AP0916", "Delhi", "London", "International", 75);

        flights[3] = new Flight("PA1609", "Mumbai", "Kolkata", "Domestic", 80);

        flights[4] = new Flight("DLN113", "Mumbai", "Singapore", "International", 120);

        flights[5] = new Flight("DLN054", "Chennai", "Delhi", "Domestic", 60);

        flights[6] = new Flight("N34511", "Chennai", "Dubai", "International", 90);

        flights[7] = new Flight("S11543", "Kolkata", "Bangalore", "Domestic", 70);

        flights[8] = new Flight("NPM169", "Kolkata", "Sydney", "International", 110);

        flights[9] = new Flight("S18054", "Bangalore", "Hyderabad", "Domestic", 65);

    }

    public void viewAvailableFlights() {

        System.out.println("Available Flights:\n");

        System.out.println("Flight Number   Source          Destination     Type              Available Seats");

        System.out.println("---------------------------------------------------------------------------------");

        for (Flight flight : flights) {

            System.out.printf("%-15s %-15s %-15s %-20s %d\n", flight.getFlightNumber(), flight.getSource(),

                    flight.getDestination(), flight.getFlightType(), flight.getAvailableSeats());

        }

        System.out.println();

    }

    public void bookDomesticFlight(String flightNumber, int numSeats) {

        for (Flight flight : flights) {

            if (flight.getFlightNumber().equals(flightNumber) && flight.getFlightType().equals("Domestic")) {

                if (flight.getAvailableSeats() >= numSeats) {

                    boolean paymentSuccessful = PaymentModule.processPayment(); // Process payment and check if it is successful

                    if (paymentSuccessful) {

                        flight.bookSeats(numSeats); // Decrease available seats

                        System.out.println("Booking Successful! Enjoy your flight.\n");

                    } else {

                        System.out.println("Payment cancelled. Booking failed. Seats remain unchanged.\n");

                        return; // Return without decreasing the available seats

                    }

                } else {

                    System.out.println("Not enough seats available on the selected flight.\n");

                }

                return;

            }

        }

        System.out.println("Invalid flight number or flight type.\n");

    }

    public void bookInternationalFlight(String flightNumber, int numSeats) {

        for (Flight flight : flights) {

            if (flight.getFlightNumber().equals(flightNumber) && flight.getFlightType().equals("International")) {

                if (flight.getAvailableSeats() >= numSeats) {

                    boolean paymentSuccessful = PaymentModule.processPayment(); // Process payment and check if it is successful

                    if (paymentSuccessful) {

                        flight.bookSeats(numSeats); // Decrease available seats

                        System.out.println("Booking Successful! Enjoy your flight.\n");

                    } else {

                        System.out.println("Payment cancelled. Booking failed. Seats remain unchanged.\n");

                        return; // Return without decreasing the available seats

                    }

                } else {

                    System.out.println("Not enough seats available on the selected flight.\n");

                }

                return;

            }

        }

        System.out.println("Invalid flight number or flight type.\n");

    }

}

public class Main {

    public static void main(String[] args) {

        Scanner scanner = new Scanner(System.in);

        FlightBookingSystem bookingSystem = new FlightBookingSystem();

        System.out.println("-----------------------------------Welcome-----------------------------------\n");

        System.out.println("1. View Available Flights\n");

        System.out.println("2. Book Domestic Flight\n");

        System.out.println("3. Book International Flight\n");

        System.out.println("4. Exit\n");

        while (true) {

            try {

                System.out.print("Enter your choice: ");

                int choice = scanner.nextInt();

                switch (choice) {

                    case 1:

                        bookingSystem.viewAvailableFlights();

                        break;

                    case 2:

                        System.out.print("Enter the flight number: ");

                        String domesticFlightNumber = scanner.next();

                        System.out.print("Enter the number of seats to book: ");

                        int numSeatsDomestic = scanner.nextInt();

                        bookingSystem.bookDomesticFlight(domesticFlightNumber, numSeatsDomestic);

                        break;

                    case 3:

                        System.out.print("Enter the flight number: ");

                        String internationalFlightNumber = scanner.next();

                        System.out.print("Enter the number of seats to book: ");

                        int numSeatsInternational = scanner.nextInt();

                        bookingSystem.bookInternationalFlight(internationalFlightNumber, numSeatsInternational);

                        break;

                    case 4:

                        System.out.println("Thank you for using the Flight Booking System. Goodbye!");

                        System.exit(0);

                    default:

                        System.out.println("Invalid choice. Please enter a valid option.\n");

                }

            } catch (InputMismatchException e) {

                System.out.println("Invalid input. Please enter a valid choice.\n");

                scanner.nextLine(); // Clear the input buffer

            }

        }

    }

}

**CONCLUSION:**

The satisfaction that accompanies the successful completion of any task would be incomplete without introducing the people who made it possible and whose constant guidance and encouragement crowns all efforts with success.

We are extremely grateful and express our profound gratitude and indebtedness to our project guide, **V. Veda Sri**, Lecturer, Department of Computer Science & Engineering, SRM University, Andhra Pradesh, for his kind help and for giving us the necessary guidance and valuable suggestions in completing this project work.