## A Course Based Project Report On

**Ticket Booking System (Airways)**

## Submitted in partial fulfilment of requirement

**for the completion of**

**Object Oriented Programming Through JAVA**

## LABORATORY

**B.Tech Computer Science and Engineering**

## By

**1. S. Raja Sree (23071A05U6)**

**2. S. Suraj (23071A05U7)**

## 3. V. Shivanvitha (23071A05V0)



### VALLURIPALLI NAGESWARARAO VIGNANA JYOTHI INSTITUTE OF ENGINEERING and

**TECHNOLOGY(AUTONOMOUS INSTITUTE)**

### NAAC ACCREDITED WITH ‘A GRADE

**Vignana Jyothi Nagar, Bachupally, Nizampet (s.o), Hyderabad 500090Phone no: 040-23042758/59/60**

## 2024-2025

**A Course Based Project Report On**

## Ticket Booking System (Airways) Submitted in partial fulfilment of requirement

**For the completion of the**

## Object Oriented Programming Through JAVA Laboratory course

**B.Tech Computer Science and Engineering Under the Guidance of**

## Mrs. A. Madhavi

**CSE Department**

## VNR Vignana Jyothi Institute of Engineering and Technology (Affiliatedto J.N.T.U, Hyderabad)Bachupally(v), Hyderabad, Telangana, India.

**CERTIFICATE**

This is to certify that the project entitled **"Ticket booking System (Airways)"** submitted in partial fulfilment for the course of Object Oriented Programming Through JAVA Laboratory (22PC2IT201) being offered for the award of B.Tech (CSE-D) by VNR VJIET is a result of the bonafide work carried out by **S. Raja Sree(23071A05U6), S. Suraj(23071A05U7) and V. Shivanvitha(23071A05V0)** during the year 2024-2025. This has not been submitted for any other certificate or course. This work is carried out under supervision and has not been submitted to any other University/Institute for award of any degree/diploma.

### Signature of Faculty Signature of

**Head of the Department**

**ACKNOWLEDGEMENT**

An endeavor over a long period can be successful only with the advice and support of many well-wishers. We take this opportunity to express our gratitude and appreciation to all of them.

We wish to express our profound gratitude to our honourable **Principal, Dr. C. D. NAIDU and HOD, Dr. V. Baby, CSE department, VNR Vignana Jyothi Institute of Engineering and Technology** for their constant and dedicated support towards our career moulding and development.

With a great pleasure we express our gratitude to the internal guide **Mrs. A. Madhavi, Assistant Professor, CSE department** for her timely help, constant guidance, cooperation, support and encouragement throughout this project as it has urged us to explore many new things.

Finally, we wish to express our deep sense of gratitude and sincere thanks to our parents, friends and all our well-wishers who have technically and non-technically contributed for the successful completion of this course-based project.

**DECLARATION**

We do declare that the project entitled **“Ticket booking System (Airways)”** submitted to the Department of Computer Science and Engineering (CSE), Vallurupalli Nageswara Rao Vignana Jyothi Institute of Engineering and Technology , Hyderabad, in partial fulfillment of the requirement for the completion of the Object Oriented Programming through JAVA laboratory course of BACHELOR OF TECHNOLOGY in Computer Science and Engineering is the bonafide record of the project report presented under the supervision of Mrs. A, Madhavi, Assistant Professor, CSE Dept, VNR VJIET.

### Signature of the Student:

1. S. Raja Sree (23071A05U6)

2. S. Suraj (23071A05U7)

3. V. Shivanvitha (23071A05V0)

**INDEX**

|  |  |  |
| --- | --- | --- |
| **S.NO** | **TOPIC** | **PAGE NO.** |
| 1. | ABSTRACT | 7 |
| 2. | INTRODUCTION | 8 |
| 3. | CODE | 9 |
| 4. | RESULT | 21 |
| 5. | CONCLUSION | 24 |
| 6. | REFERENCE | 25 |

**ABSTRACT**

The Ticket Managing System(Airways) is a comprehensive desktop application developed to streamline the operational processes of an airline. This system addresses the growing demand for efficient and automated solutions in the aviation sector by integrating a wide range of functionalities tailored for both users and administrators. Key features of the system include customer management, which allows the secure storage and retrieval of passenger details, a booking module for hassle-free ticket reservations, and a flight management system that provides essential information about available flights. The application is built using Java Swing and AWT for an interactive graphical user interface, ensuring user-friendly navigation and intuitive controls. MySQL serves as the database backend, enabling robust data management and real-time updates. To bridge the gap between the interface and data storage, the system employs JDBC (Java Database Connectivity), which facilitates seamless data flow. Special attention has been given to error handling, scalability, and reliability to ensure the system's consistent performance. This document delves into the design, implementation, and practical applications of the system, emphasizing its potential to improve operational efficiency, enhance customer experience, and support decision-making in airline management.

**INTRODUCTION**

The airline industry is a cornerstone of global connectivity, providing essential transportation for millions of passengers daily. However, the complexity and scale of airline operations necessitate efficient and reliable management systems. This project introduces the Ticket Booking System(Airways), a Java-based application designed to address the challenges faced by airlines in managing customer data, flight bookings, journey tracking, and schedule management. By combining modern software development techniques with intuitive user interface design, the system offers a comprehensive solution to streamline operations.

The system is built using Java Swing and AWT for its graphical user interface, providing a visually appealing and interactive platform that ensures ease of navigation. MySQL serves as the backend database, offering secure and efficient data storage for critical information such as passenger records and flight schedules. The integration of JDBC bridges the gap between the application and the database, ensuring real-time data synchronization and a seamless user experience. The modular architecture allows for the addition of new features without disrupting existing functionalities, making the system highly scalable and adaptable to future requirements.

This document explores the rationale behind the system’s development, detailing its objectives, architecture, and core functionalities. By addressing the inefficiencies of traditional systems, this project aims to deliver a robust and user-friendly platform that enhances operational efficiency and customer satisfaction in the airline industry.

**CODE**

**HOME PAGE**

package airlinemanagement;

import javax.swing.\*;

import java.awt.\*;

import java.awt.event.\*;

public class Home extends JFrame implements ActionListener {

public Home() {

setLayout(null);

ImageIcon i1 = new ImageIcon(ClassLoader.getSystemResource("airlinemanagement/icons/front.jpg"));

JLabel image = new JLabel(i1);

image.setBounds(0, 0, 1600, 800);

add(image);

JLabel heading = new JLabel("UNITED AIRLINES WELCOMES YOU");

heading.setBounds(500, 40, 1000, 40);

heading.setForeground(Color.BLUE);

heading.setFont(new Font("Tahoma", Font.PLAIN, 36));

image.add(heading);

JMenuBar menubar = new JMenuBar();

setJMenuBar(menubar);

JMenu details = new JMenu("Details");

menubar.add(details);

JMenuItem flightDetails = new JMenuItem("Flight Details");

flightDetails.addActionListener(this);

details.add(flightDetails);

JMenuItem customerDetails = new JMenuItem("Add Customer Details");

customerDetails.addActionListener(this);

details.add(customerDetails);

JMenuItem bookFlight = new JMenuItem("Book Flight");

bookFlight.addActionListener(this);

details.add(bookFlight);

JMenuItem journeyDetails = new JMenuItem("Journey Details");

journeyDetails.addActionListener(this);

details.add(journeyDetails);

setExtendedState(JFrame.MAXIMIZED\_BOTH);

setVisible(true);

}

public void actionPerformed(ActionEvent ae) {

String text = ae.getActionCommand();

if (text.equals("Add Customer Details")) {

new AddCustomer();

} else if (text.equals("Flight Details")) {

new FlightInfo();

} else if (text.equals("Book Flight")) {

new BookFlight();

} else if (text.equals("Journey Details")) {

new JourneyDetails();

}

}

public static void main(String[] args) {

new Home();

}

}

**CONNECTION JDBC**

import java.sql.\*;

public class Conn {

Connection c;

Statement s;

public Conn(){

try{

Class.forName("com.mysql.cj.jdbc.Driver");

c=DriverManager.getConnection("jdbc:mysql:///airlinemanagementsystem","root","anvitha@13");

s=c.createStatement();

}catch (Exception e){

e.printStackTrace();

}

}

}

**FLIGHT INFO**

import javax.swing.\*;

import java.awt.\*;

import java.sql.\*;

import net.proteanit.sql.DbUtils;

public class FlightInfo extends JFrame{

public FlightInfo() {

getContentPane().setBackground(Color.WHITE);

setLayout(null);

JTable table = new JTable();

try {

Conn conn = new Conn();

ResultSet rs = conn.s.executeQuery("select \* from airlinemanagementsystem.dbmsexl\_csv");

table.setModel(DbUtils.resultSetToTableModel(rs));

} catch(Exception e) {

e.printStackTrace();

}

JScrollPane jsp = new JScrollPane(table);

jsp.setBounds(0, 0, 800, 500);

add(jsp);

setSize(800, 500);

setLocation(400, 200);

setVisible(true);

}

public static void main(String[] args) {

new FlightInfo();

}

}

**JOURNEY DETAILS**

import javax.swing.\*;

import java.awt.\*;

import java.sql.\*;

import java.awt.event.\*;

import net.proteanit.sql.DbUtils;

public class JourneyDetails extends JFrame implements ActionListener{

JTable table;

JTextField pnr;

JButton show;

public JourneyDetails() {

getContentPane().setBackground(Color.WHITE);

setLayout(null);

JLabel lblpnr = new JLabel("PNR Details");

lblpnr.setFont(new Font("Tahoma", Font.PLAIN, 16));

lblpnr.setBounds(50, 50, 100, 25);

add(lblpnr);

pnr = new JTextField();

pnr.setBounds(160, 50, 120, 25);

add(pnr);

show = new JButton("Show Details");

show.setBackground(Color.BLACK);

show.setForeground(Color.WHITE);

show.setBounds(290, 50, 120, 25);

show.addActionListener(this);

add(show);

table = new JTable();

JScrollPane jsp = new JScrollPane(table);

jsp.setBounds(0, 100, 800, 150);

jsp.setBackground(Color.WHITE);

add(jsp);

setSize(800, 600);

setLocation(400, 150);

setVisible(true);

}

public void actionPerformed(ActionEvent ae) {

try {

Conn conn = new Conn();

ResultSet rs = conn.s.executeQuery("select \* from reservation where PNR = '"+pnr.getText()+"'");

if (!rs.isBeforeFirst()) {

JOptionPane.showMessageDialog(null, "No Information Found");

return;

}

table.setModel(DbUtils.resultSetToTableModel(rs));

} catch(Exception e) {

e.printStackTrace();

}

}

public static void main(String[] args) {

new JourneyDetails();

}

}

**ADD CUSTOMER DETAILS**

import javax.swing.\*;

import java.awt.\*;

import java.awt.event.\*;

public class AddCustomer extends JFrame implements ActionListener {

JTextField tfname, tfphone, tfpassport, tfnationality, tfaddress;

JRadioButton rbmale, rbfemale;

public AddCustomer() {

getContentPane().setBackground(Color.WHITE);

setLayout(null);

JLabel heading = new JLabel("ADD CUSTOMER DETAILS");

heading.setBounds(220, 20, 500, 35);

heading.setFont(new Font("Tahoma", Font.PLAIN, 32));

heading.setForeground(Color.BLUE);

add(heading);

JLabel lblname = new JLabel("Name");

lblname.setBounds(60, 80, 150, 25);

lblname.setFont(new Font("Tahoma", Font.PLAIN, 16));

add(lblname);

tfname = new JTextField();

tfname.setBounds(220, 80, 150, 25);

add(tfname);

JLabel lblnationality = new JLabel("Nationality");

lblnationality.setBounds(60, 130, 150, 25);

lblnationality.setFont(new Font("Tahoma", Font.PLAIN, 16));

add(lblnationality);

tfnationality = new JTextField();

tfnationality.setBounds(220, 130, 150, 25);

add(tfnationality);

JLabel lblpassport = new JLabel("Passport No");

lblpassport.setBounds(60, 180, 150, 25);

lblpassport.setFont(new Font("Tahoma", Font.PLAIN, 16));

add(lblpassport);

tfpassport = new JTextField();

tfpassport.setBounds(220, 180, 150, 25);

add(tfpassport);

JLabel lbladdress = new JLabel("Address");

lbladdress.setBounds(60, 230, 150, 25);

lbladdress.setFont(new Font("Tahoma", Font.PLAIN, 16));

add(lbladdress);

tfaddress = new JTextField();

tfaddress.setBounds(220, 230, 150, 25);

add(tfaddress);

JLabel lblgender = new JLabel("Gender");

lblgender.setBounds(60, 280, 150, 25);

lblgender.setFont(new Font("Tahoma", Font.PLAIN, 16));

add(lblgender);

ButtonGroup gendergroup = new ButtonGroup();

rbmale = new JRadioButton("Male");

rbmale.setBounds(220, 280, 70, 25);

rbmale.setBackground(Color.WHITE);

add(rbmale);

rbfemale = new JRadioButton("Female");

rbfemale.setBounds(300, 280, 70, 25);

rbfemale.setBackground(Color.WHITE);

add(rbfemale);

gendergroup.add(rbmale);

gendergroup.add(rbfemale);

JLabel lblphone = new JLabel("Phone");

lblphone.setBounds(60, 330, 150, 25);

lblphone.setFont(new Font("Tahoma", Font.PLAIN, 16));

add(lblphone);

tfphone = new JTextField();

tfphone.setBounds(220, 330, 150, 25);

add(tfphone);

JButton save = new JButton("SAVE");

save.setBackground(Color.BLACK);

save.setForeground(Color.WHITE);

save.setBounds(220, 380, 150, 30);

save.addActionListener(this);

add(save);

ImageIcon image = new ImageIcon(ClassLoader.getSystemResource("airlinemanagement/icons/emp.png"));

JLabel lblimage = new JLabel(image);

lblimage.setBounds(450, 80, 280, 400);

add(lblimage);

setSize(900, 600);

setLocation(300, 150);

setVisible(true);

}

public void actionPerformed(ActionEvent ae) {

String name = tfname.getText();

String nationality = tfnationality.getText();

String phone = tfphone.getText();

String address = tfaddress.getText();

String passportno = tfpassport.getText();

String gender = null;

if (rbmale.isSelected()) {

gender = "Male";

} else {

gender = "Female";

}

try {

Conn conn = new Conn();

String query = "insert into passenger values('" + name + "', '" + nationality + "', '" + phone + "', '" + address + "', '" + passportno + "', '" + gender + "')";

conn.s.executeUpdate(query);

JOptionPane.showMessageDialog(null, "Customer Details Added Successfully");

setVisible(false);

} catch (Exception e) {

e.printStackTrace();

}

}

public static void main(String[] args) {

new AddCustomer();

}

}

**BOOK FLIGHT**

import javax.swing.\*;

import java.awt.\*;

import java.awt.event.\*;

import java.sql.\*;

import com.toedter.calendar.JDateChooser;

import java.util.\*;

public class BookFlight extends JFrame implements ActionListener {

JTextField tfpassport, tfsource, tfdestination;

JLabel tfname, tfnationality, tfaddress, labelgender, labelfcode;

JButton bookflight, fetchButton, flight;

JDateChooser dcdate;

public BookFlight() {

getContentPane().setBackground(Color.WHITE);

setLayout(null);

JLabel heading = new JLabel("Book Flight");

heading.setBounds(420, 20, 500, 35);

heading.setFont(new Font("Tahoma", Font.PLAIN, 32));

heading.setForeground(Color.BLUE);

add(heading);

JLabel lblpassport = new JLabel("Passport No");

lblpassport.setBounds(60, 80, 150, 25);

lblpassport.setFont(new Font("Tahoma", Font.PLAIN, 16));

add(lblpassport);

tfpassport = new JTextField();

tfpassport.setBounds(220, 80, 150, 25);

add(tfpassport);

fetchButton = new JButton("Fetch User");

fetchButton.setBackground(Color.BLACK);

fetchButton.setForeground(Color.WHITE);

fetchButton.setBounds(380, 80, 120, 25);

fetchButton.addActionListener(this);

add(fetchButton);

JLabel lblname = new JLabel("Name");

lblname.setBounds(60, 130, 150, 25);

lblname.setFont(new Font("Tahoma", Font.PLAIN, 16));

add(lblname);

tfname = new JLabel();

tfname.setBounds(220, 130, 150, 25);

add(tfname);

JLabel lblnationality = new JLabel("Nationality");

lblnationality.setBounds(60, 180, 150, 25);

lblnationality.setFont(new Font("Tahoma", Font.PLAIN, 16));

add(lblnationality);

tfnationality = new JLabel();

tfnationality.setBounds(220, 180, 150, 25);

add(tfnationality);

JLabel lbladdress = new JLabel("Address");

lbladdress.setBounds(60, 230, 150, 25);

lbladdress.setFont(new Font("Tahoma", Font.PLAIN, 16));

add(lbladdress);

tfaddress = new JLabel();

tfaddress.setBounds(220, 230, 150, 25);

add(tfaddress);

JLabel lblgender = new JLabel("Gender");

lblgender.setBounds(60, 280, 150, 25);

lblgender.setFont(new Font("Tahoma", Font.PLAIN, 16));

add(lblgender);

labelgender = new JLabel();

labelgender.setBounds(220, 280, 150, 25);

add(labelgender);

JLabel lblsource = new JLabel("Source");

lblsource.setBounds(60, 330, 150, 25);

lblsource.setFont(new Font("Tahoma", Font.PLAIN, 16));

add(lblsource);

tfsource = new JTextField();

tfsource.setBounds(220, 330, 150, 25);

add(tfsource);

JLabel lbldest = new JLabel("Destination");

lbldest.setBounds(60, 380, 150, 25);

lbldest.setFont(new Font("Tahoma", Font.PLAIN, 16));

add(lbldest);

tfdestination = new JTextField();

tfdestination.setBounds(220, 380, 150, 25);

add(tfdestination);

flight = new JButton("Fetch Flights");

flight.setBackground(Color.BLACK);

flight.setForeground(Color.WHITE);

flight.setBounds(380, 380, 120, 25);

flight.addActionListener(this);

add(flight);

JLabel lblfcode = new JLabel("Flight Code");

lblfcode.setBounds(60, 430, 150, 25);

lblfcode.setFont(new Font("Tahoma", Font.PLAIN, 16));

add(lblfcode);

labelfcode = new JLabel();

labelfcode.setBounds(220, 430, 150, 25);

add(labelfcode);

JLabel lbldate = new JLabel("Date of Travel");

lbldate.setBounds(60, 480, 150, 25);

lbldate.setFont(new Font("Tahoma", Font.PLAIN, 16));

add(lbldate);

dcdate = new JDateChooser();

dcdate.setBounds(220, 480, 150, 25);

add(dcdate);

bookflight = new JButton("Book Flight");

bookflight.setBackground(Color.BLACK);

bookflight.setForeground(Color.WHITE);

bookflight.setBounds(220, 530, 150, 25);

bookflight.addActionListener(this);

add(bookflight);

setSize(800, 700);

setLocation(200, 50);

setVisible(true);

}

public void actionPerformed(ActionEvent ae) {

if (ae.getSource() == fetchButton) {

String passportno = tfpassport.getText();

try {

Conn conn = new Conn();

String query = "select \* from passenger where passportno = '" + passportno + "'";

ResultSet rs = conn.s.executeQuery(query);

if (rs.next()) {

tfname.setText(rs.getString("name"));

tfnationality.setText(rs.getString("nationality"));

tfaddress.setText(rs.getString("address"));

labelgender.setText(rs.getString("gender"));

} else {

JOptionPane.showMessageDialog(null, "Invalid Passport Number");

}

} catch (Exception e) {

e.printStackTrace();

}

} else if (ae.getSource() == flight) {

Random random = new Random();

boolean noFlight = random.nextBoolean();

if (noFlight) {

labelfcode.setText("No Flights Found");

} else {

labelfcode.setText(String.valueOf(10000 + random.nextInt(90000)));

}

} else if (ae.getSource() == bookflight) {

String passportno = tfpassport.getText();

String name = tfname.getText();

String nationality = tfnationality.getText();

String src = tfsource.getText();

String dest = tfdestination.getText();

String flightcode = labelfcode.getText();

String ddate = ((JTextField) dcdate.getDateEditor().getUiComponent()).getText();

if (flightcode.equals("No Flights Found")) {

JOptionPane.showMessageDialog(null, "Cannot book a flight. No flights available.");

return;

}

try {

Conn conn = new Conn();

Random random = new Random();

String query = "insert into reservation values('PNR-" + (100000 + random.nextInt(900000)) + "', 'TIC-" + (10000 + random.nextInt(90000)) + "', '" + passportno + "', '" + name + "', '" + nationality + "', '" + src + "', '" + dest + "', '" + flightcode + "', '" + ddate + "')";

conn.s.executeUpdate(query);

JOptionPane.showMessageDialog(null, "Ticket Booked Successfully");

setVisible(false);

} catch (Exception e) {

e.printStackTrace();

}

}

}

public static void main(String[] args) {

new BookFlight();

}

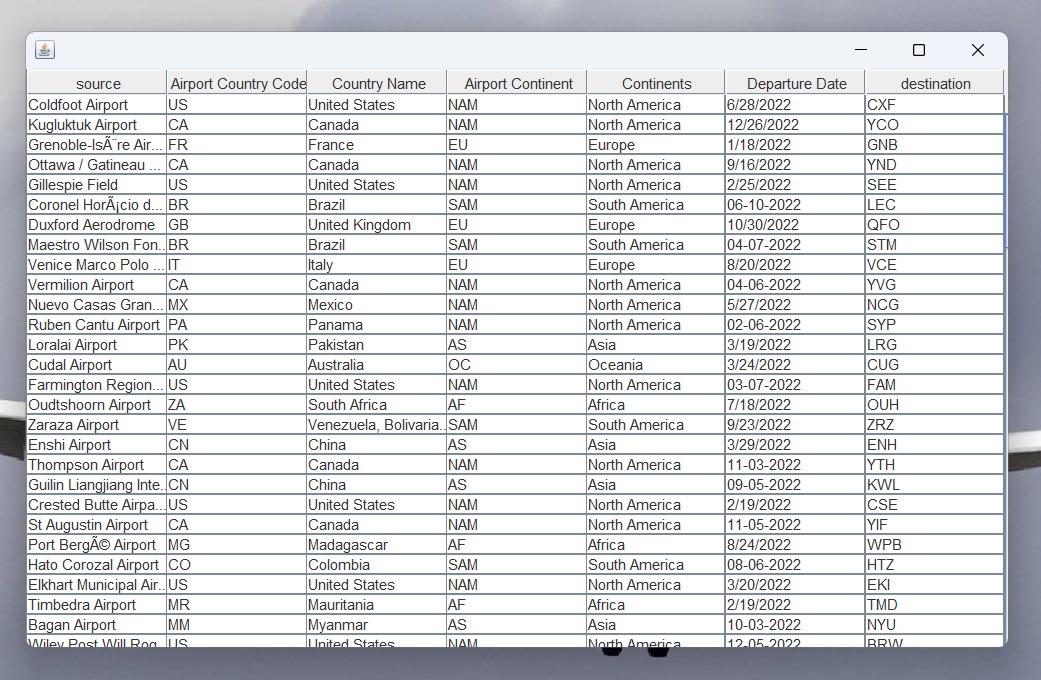
}

**RESULT**

**HOME PAGE:**



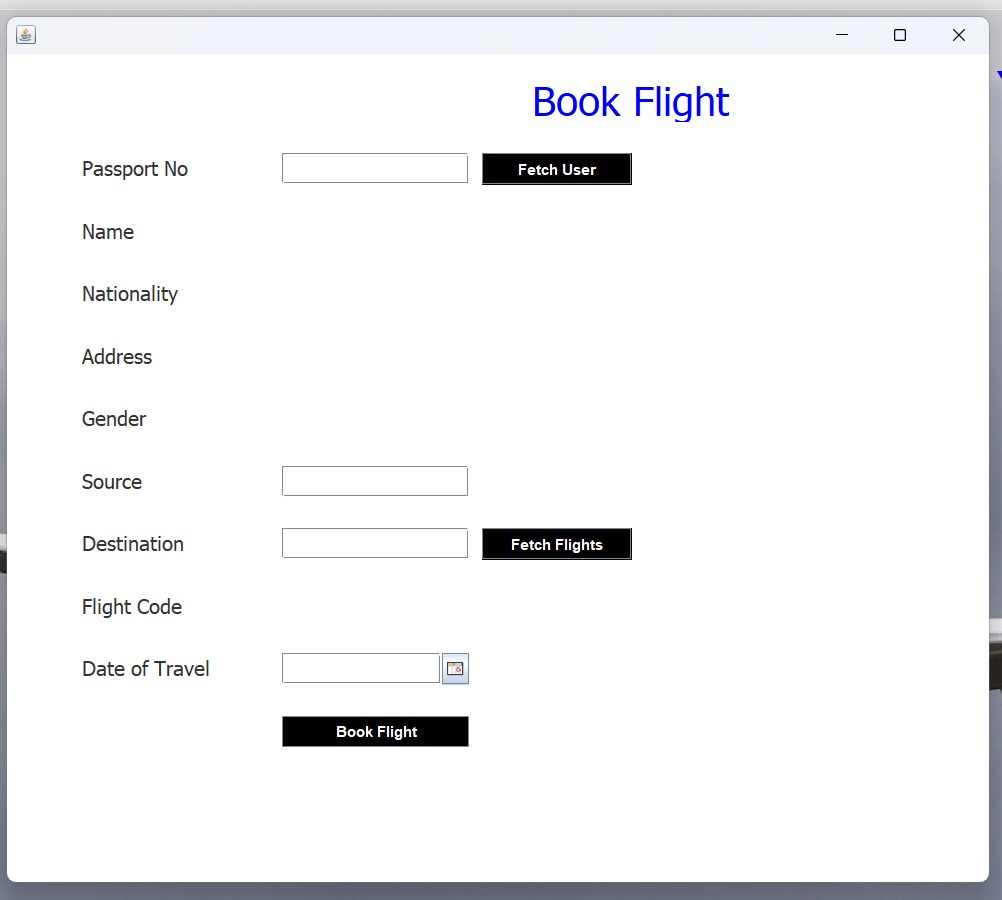
**FLIGHT INFO:**



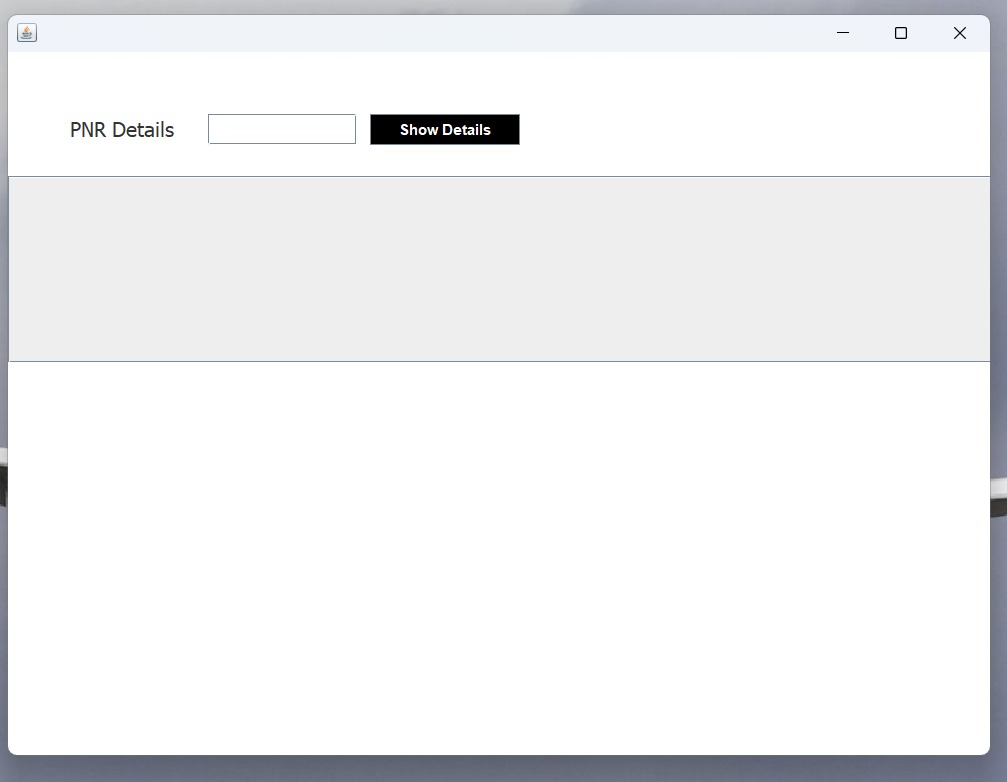
**ADD CUSTOMER DETAILS:**



**BOOK FLIGHT:**



**JOURNEY DETAILS:**



**CONCLUSION**

The Ticket Booking System (Airways) stands as a testament to how technology can revolutionize traditional business operations by automating core airline functions such as passenger data management, flight bookings, and schedule monitoring. By leveraging Java Swing for a visually appealing and intuitive user interface and MySQL for robust and secure data management, the system delivers seamless performance while ensuring data accuracy and reliability.

This project has been thoughtfully designed to address the operational challenges faced by the aviation industry. The modular and scalable architecture ensures that the system remains adaptable to future needs, enabling the easy integration of additional features such as advanced reporting tools, dynamic pricing algorithms, and real-time flight updates. These enhancements would significantly elevate operational efficiency and customer satisfaction.

Looking ahead, the Ticket Booking System (Airways) has the potential to embrace cutting-edge technologies like artificial intelligence for predictive maintenance, machine learning to optimize flight schedules, or blockchain for enhanced security in ticketing and data transactions. The adoption of cloud computing could also provide users with greater accessibility and ensure smooth performance during peak loads. These advancements could make the system even more indispensable in the competitive airline industry.

Overall, this project not only highlights the transformative power of technology but also underscores the importance of innovation and adaptability in creating solutions that can evolve alongside industry demands. It serves as a stepping stone toward a more efficient and customer-focused aviation sector.

**REFERENCES**

1. **Github repository:** **https://github.com/kunaltyagi9/AirlineManagementSystem**
2. [**www.geeksgorgeeks.org**](http://www.geeksgorgeeks.org)
3. **w3schools.com**