/\*

\* Click nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-default.txt to change this license

\* Click nbfs://nbhost/SystemFileSystem/Templates/Classes/Class.java to edit this template

\*/

package travelmanagementsystem;

/\*\*

\* Main class that starts the Travel Management System application.

\* It initializes and displays the splash screen.

\*

\* @author Memon enterprise

\*/

public class TravelManagementSystem {

/\*\*

\* Main method that starts the application.

\*

\* @param args the command line arguments

\*/

public static void main(String[] args) {

// Create a new instance of Splash (splash screen)

Splash frame = new Splash();

// Animation effect: gradually increase the size and location of the splash screen

int x = 1;

for (int i = 1; i <= 500; x += 7, i += 6) {

// Adjust the location and size of the splash screen

frame.setLocation(650 - (x + (2 \* i)) / 3, 350 - (i / 2));

frame.setSize(x + i, i);

try {

Thread.sleep(5); // Pause for a short time to create animation effect

} catch (Exception e) {

e.printStackTrace();

}

}

}

}

package travelmanagementsystem;

import java.awt.Color;

import java.awt.Font;

import java.awt.Scrollbar;

import java.awt.TextArea;

import java.awt.event.ActionEvent;

import java.awt.event.ActionListener;

import javax.swing.BorderFactory;

import javax.swing.JButton;

import javax.swing.JFrame;

import javax.swing.JLabel;

/\*\*

\* About class for displaying information about the TravelPerk application.

\*/

public class About extends JFrame implements ActionListener {

private JLabel title; // Title label

private TextArea area; // Text area for displaying information

private JButton back; // Back button to return to the previous screen

public About() {

// Set frame properties

this.setBounds(305, 90, 600, 600);

this.setLayout(null);

this.getContentPane().setBackground(new Color(0x8fbc8f));

this.setTitle("TravelPerk");

this.setResizable(false);

// Initialize and add title label

this.title = new JLabel("About");

this.title.setBounds(240, 30, 300, 50);

this.title.setFont(new Font("Tahoma", Font.BOLD, 35));

this.title.setForeground(Color.BLACK);

this.add(this.title);

// Initialize and add text area with application information

this.area = new TextArea("", 10, 40, Scrollbar.VERTICAL);

this.area.setEditable(false);

this.area.setBounds(20, 100, 540, 350);

this.area.setBackground(new Color(0xcfe8cf));

this.area.setText(

"Application Name: TravelPerk\n\n" +

"Purpose:\n" +

"TravelPerk is a comprehensive Travel and Tourism Management System designed to provide users with " +

"an intuitive and seamless experience for planning and booking their travels. It simplifies travel " +

"arrangements by offering a wide range of functionalities that cater to every aspect of a traveler's needs.\n\n" +

"Features:\n" +

"- Add, update, and view personal details: Maintain your personal information with ease, ensuring " +

"that your travel preferences are always up-to-date.\n" +

"- Explore and book travel packages: Choose from a variety of travel packages tailored to meet " +

"different interests and budgets. From adventure trips to relaxing vacations, TravelPerk has it all.\n" +

"- View details of booked packages and trips: Keep track of your itinerary and booked packages in " +

"one place, making travel planning straightforward and hassle-free.\n" +

"- Check available hotels and destinations: Discover hotels and travel destinations that meet your " +

"preferences and requirements.\n" +

"- Integrated payment system: Enjoy secure and convenient transactions with our integrated payment " +

"system, ensuring that your financial information is protected.\n\n" +

"Technologies Used:\n" +

"- Java: The core programming language for application logic, providing a robust and efficient backend.\n" +

"- Swing: Used for the graphical user interface (GUI), offering a visually appealing and user-friendly " +

"experience.\n" +

"- File handling: For storing and retrieving user details, ensuring data persistence and easy access " +

"to personal information.\n\n" +

"Development Team:\n" +

"TravelPerk is developed by Memon Enterprise, a team committed to enhancing the travel experience by " +

"providing innovative, efficient, and user-friendly solutions for travel management. Our mission is " +

"to make travel planning accessible, enjoyable, and straightforward for everyone.\n\n" +

"Future Enhancements:\n" +

"- Integration with additional payment gateways for more flexibility.\n" +

"- Expansion of travel package options, including international destinations.\n" +

"- Implementation of user reviews and ratings for hotels and destinations to help users make informed decisions.\n" +

"- Mobile application development to provide on-the-go access to TravelPerk features.\n\n" +

"Thank you for choosing TravelPerk as your travel partner. We hope to make your travel experiences " +

"memorable and enjoyable.\n\n" +

"For payment: 03135966880"

);

this.add(this.area);

// Initialize and add back button

this.back = new JButton("Back");

this.back.setBounds(130, 480, 280, 60);

this.back.setBackground(new Color(0xcfe8cf));

this.back.setBorder(BorderFactory.createLineBorder(Color.WHITE));

this.back.setForeground(Color.BLACK);

this.back.setFont(new Font("SAN SERIF", Font.BOLD, 15));

this.back.setFocusPainted(false);

this.back.setBorder(BorderFactory.createLineBorder(Color.BLACK));

this.back.addActionListener(this);

this.back.setActionCommand("Back");

this.add(this.back);

this.setVisible(true);

}

@Override

public void actionPerformed(ActionEvent e) {

// Close the About window on button click

this.setVisible(false);

}

}

package travelmanagementsystem;

import java.awt.Choice;

import java.awt.Color;

import java.awt.Font;

import java.awt.Image;

import java.awt.event.ActionEvent;

import java.awt.event.ActionListener;

import java.io.BufferedWriter;

import java.io.FileWriter;

import java.io.IOException;

import java.util.regex.Matcher;

import java.util.regex.Pattern;

import javax.swing.BorderFactory;

import javax.swing.ImageIcon;

import javax.swing.JButton;

import javax.swing.JFrame;

import javax.swing.JLabel;

import javax.swing.JOptionPane;

import javax.swing.JPanel;

import javax.swing.JTextField;

import javax.swing.border.LineBorder;

/\*\*

\* The AddCustomerDetail class provides a graphical user interface

\* for adding customer details in a travel management system application.

\* It extends JFrame to create the main window and implements ActionListener

\* to handle button clicks and user interactions.

\*/

public class AddCustomerDetail extends JFrame implements ActionListener {

// Components for the add customer details interface

private JTextField Name, Number, Address, Email, Country;

private JPanel p1, p2;

private JLabel lblname, lblid, lblnumber, lblgender, lbladdress, lblcountry, lblemail, p3, p4;

private Choice ID, Gender;

private JButton add, back;

String name, id, number, gender, address, country, email;

private boolean flag;

public AddCustomerDetail() {

this.setBounds(305, 90, 970, 630); // Set size and position of the JFrame

this.setLayout(null); // Use null layout for absolute positioning

this.setResizable(false);

this.setTitle("TravelPerk");

// Set border around the JFrame

this.getRootPane().setBorder(BorderFactory.createLineBorder(new Color(127, 93, 37), 10));

// Load and display background image

ImageIcon i1 = new ImageIcon(getClass().getResource("/Images/image58.jpg"));

Image i2 = i1.getImage().getScaledInstance(970, 630, Image.SCALE\_DEFAULT);

ImageIcon i3 = new ImageIcon(i2);

JLabel image1 = new JLabel(i3);

image1.setBounds(0, 0, 970, 630);

this.flag = false;

// Panel 1 for displaying additional image

p1 = new JPanel();

p1.setBounds(650, 40, 280, 380);

ImageIcon i4 = new ImageIcon(getClass().getResource("/Images/image59.jpg"));

Image i5 = i4.getImage().getScaledInstance(270, 320, Image.SCALE\_AREA\_AVERAGING);

ImageIcon i6 = new ImageIcon(i5);

JLabel image2 = new JLabel(i6);

image2.setBounds(5, -10, 250, 350);

p1.add(image2);

// Label for additional text

p3 = new JLabel("In The Middle of Streets");

p3.setFont(new Font("PLAYFAIR DISPLAY", Font.PLAIN, 15));

p3.setBounds(35, 360, 250, 25);

// Panel 2 for displaying another image

p2 = new JPanel();

p2.setBounds(530, 230, 250, 320);

ImageIcon i7 = new ImageIcon(getClass().getResource("/Images/image1.jpg"));

Image i8 = i7.getImage().getScaledInstance(230, 250, Image.SCALE\_SMOOTH);

ImageIcon i9 = new ImageIcon(i8);

JLabel image3 = new JLabel(i9);

image3.setBounds(10, 0, 230, 250);

p2.add(image3);

// Label for additional text

p4 = new JLabel("Somewhere in Mountains");

p4.setFont(new Font("PLAYFAIR DISPLAY", Font.PLAIN, 15));

p4.setBounds(25, 300, 250, 25);

// Name label and text field

lblname = new JLabel("Name");

lblname.setBounds(30, 65, 100, 20);

lblname.setFont(new Font("SAN SERIF", Font.BOLD, 20));

lblname.setForeground(Color.WHITE);

this.add(lblname);

Name = new JTextField();

Name.setBounds(180, 60, 260, 35);

Name.setBackground(Color.WHITE);

Name.setBorder(BorderFactory.createLineBorder(new Color(127, 93, 37)));

this.add(Name);

// ID label and text field

lblid = new JLabel("ID");

lblid.setBounds(30, 125, 100, 20);

lblid.setFont(new Font("SAN SERIF", Font.BOLD, 20));

lblid.setForeground(Color.WHITE);

this.add(lblid);

ID = new Choice();

ID.add("Select an Option");

ID.add("Debit Card");

ID.add("Credit Card");

ID.setBounds(180, 125, 260, 35);

ID.setBackground(Color.WHITE);

ID.setForeground(Color.BLACK);

this.add(ID);

// Number label and text field

lblnumber = new JLabel("Number");

lblnumber.setBounds(30, 185, 100, 20);

lblnumber.setFont(new Font("SAN SERIF", Font.BOLD, 20));

lblnumber.setForeground(Color.WHITE);

this.add(lblnumber);

Number = new JTextField();

Number.setBounds(180, 180, 260, 35);

Number.setBackground(Color.WHITE);

Number.setBorder(BorderFactory.createLineBorder(new Color(127, 93, 37)));

this.add(Number);

// Gender label and text field

lblgender = new JLabel("Gender");

lblgender.setBounds(30, 245, 100, 20);

lblgender.setFont(new Font("SAN SERIF", Font.BOLD, 20));

lblgender.setForeground(Color.WHITE);

this.add(lblgender);

Gender = new Choice();

Gender.add("Select your Gender");

Gender.add("Male");

Gender.add("Female");

Gender.add("Other");

Gender.setBounds(180, 245, 260, 35);

Gender.setBackground(Color.WHITE);

Gender.setForeground(Color.BLACK);

this.add(Gender);

// Country label and text field

lblcountry = new JLabel("Country");

lblcountry.setBounds(30, 305, 100, 20);

lblcountry.setFont(new Font("SAN SERIF", Font.BOLD, 20));

lblcountry.setForeground(Color.WHITE);

this.add(lblcountry);

Country = new JTextField();

Country.setBounds(180, 300, 260, 35);

Country.setBackground(Color.WHITE);

Country.setBorder(BorderFactory.createLineBorder(new Color(127, 93, 37)));

this.add(Country);

// Address label and text field

lbladdress = new JLabel("Address");

lbladdress.setBounds(30, 365, 100, 20);

lbladdress.setFont(new Font("SAN SERIF", Font.BOLD, 20));

lbladdress.setForeground(Color.WHITE);

this.add(lbladdress);

Address = new JTextField();

Address.setBounds(180, 360, 260, 35);

Address.setBackground(Color.WHITE);

Address.setBorder(BorderFactory.createLineBorder(new Color(127, 93, 37)));

this.add(Address);

// Email label and text field

lblemail = new JLabel("Email");

lblemail.setBounds(30, 425, 100, 20);

lblemail.setFont(new Font("SAN SERIF", Font.BOLD, 20));

lblemail.setForeground(Color.WHITE);

this.add(lblemail);

Email = new JTextField();

Email.setBounds(180, 420, 260, 35);

Email.setBackground(Color.WHITE);

Email.setBorder(BorderFactory.createLineBorder(new Color(127, 93, 37)));

this.add(Email);

// Adding and setting properties for the Add button

add = new JButton("Add");

add.setBounds(30, 485, 185, 45);

add.setBackground(new Color(127, 93, 37));

add.setBorder(new LineBorder(Color.WHITE));

add.setForeground(Color.WHITE);

add.setFont(new Font("SAN SERIF", Font.BOLD, 15));

add.setFocusPainted(false);

add.setBorder(BorderFactory.createLineBorder(Color.BLACK));

add.addActionListener(this);

add.setActionCommand("Add");

this.add(add);

// Back button

back = new JButton("Back");

back.setBounds(250, 485, 185, 45);

back.setBackground(new Color(127, 93, 37));

back.setBorder(new LineBorder(Color.WHITE));

back.setForeground(Color.WHITE);

back.setFont(new Font("SAN SERIF", Font.BOLD, 15));

back.setFocusPainted(false);

back.setBorder(BorderFactory.createLineBorder(Color.BLACK));

back.addActionListener(this);

back.setActionCommand("Back");

this.add(back);

// Adding panels and labels

p1.add(p3);

p2.add(p4);

this.add(p2);

this.add(p1);

this.add(image1);

this.setVisible(true); // Make the JFrame visible

}

// Method to validate email using regular expression

public boolean isValidEmail(String email) {

String emailRegex = "^[a-zA-Z0-9\_+&\*-]+(?:\\.[a-zA-Z0-9\_+&\*-]+)\*@(?:[a-zA-Z0-9-]+\\.)+[a-zA-Z]{2,7}$";

Pattern pattern = Pattern.compile(emailRegex);

Matcher matcher = pattern.matcher(email);

return matcher.matches();

}

// Method to add customer details to a file

public void addCustomerDetailsToFile() {

// Retrieving input values from text fields and choice fields

this.name = Name.getText();

this.id = ID.getSelectedItem();

this.number = Number.getText();

this.gender = Gender.getSelectedItem();

this.country = Country.getText();

this.address = Address.getText();

this.email = Email.getText();

// Validating email format

if (!isValidEmail(email)) {

JOptionPane.showMessageDialog(null, "Please enter a valid email address");

return;

}

// Validating number field to contain only digits

if (!number.matches("\\d+")) {

JOptionPane.showMessageDialog(null, "Please enter a valid number (digits only)");

return;

}

// Checking if any field is empty or not selected

if (name.isEmpty() || id.equals("Select an Option") || number.isEmpty() || gender.equals("Select your Gender") || country.isEmpty() || address.isEmpty() || email.isEmpty()) {

JOptionPane.showMessageDialog(null, "Please fill in all fields!");

return;

}

// Writing the customer details to a file

try (BufferedWriter writer = new BufferedWriter(new FileWriter("customerDetails.txt", true))) {

writer.write("Name: " + name);

writer.newLine();

writer.write("ID: " + id);

writer.newLine();

writer.write("Number: " + number);

writer.newLine();

writer.write("Gender: " + gender);

writer.newLine();

writer.write("Country: " + country);

writer.newLine();

writer.write("Address: " + address);

writer.newLine();

writer.write("Email: " + email);

writer.newLine();

writer.write("-----------------------------");

writer.newLine();

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

this.flag = true; // Set flag to true indicating successful addition

} catch (IOException e) {

e.printStackTrace(); // Print stack trace in case of an IOException

}

}

// Overriding actionPerformed method to handle button actions

@Override

public void actionPerformed(ActionEvent e) {

String command = e.getActionCommand(); // Get the action command from the event

if (command.equals("Add")) {

addCustomerDetailsToFile(); // Call method to add customer details if Add button is clicked

} else if (command.equals("Back")) {

this.setVisible(false); // Hide the frame if Back button is clicked

}

}

// Method to get the flag value indicating whether customer details were added successfully

public boolean getFlag() {

return this.flag;

}

}

package travelmanagementsystem;

import java.awt.Choice;

import java.awt.Color;

import java.awt.Font;

import java.awt.Image;

import java.awt.event.ActionEvent;

import java.awt.event.ActionListener;

import java.io.BufferedWriter;

import java.io.FileWriter;

import java.io.IOException;

import javax.swing.BorderFactory;

import javax.swing.ImageIcon;

import javax.swing.JButton;

import javax.swing.JFrame;

import javax.swing.JLabel;

import javax.swing.JOptionPane;

import javax.swing.JTextField;

import javax.swing.border.LineBorder;

/\*\*

\* This class represents a GUI application for booking trips in a travel management system.

\* It allows users to choose destinations, search hotels, calculate prices, and book trips.

\*/

public class BookDestination extends JFrame implements ActionListener {

private Choice destination, hotels;

private JButton search, book, back, checkPrice;

private JLabel lbldestination, lblhotels, lblpersons, lbltotal, total, lblDay , title;

private JTextField Persons, Days;

private boolean flag;

/\*\*

\* Constructor to initialize the GUI components and set up the frame.

\*/

public BookDestination() {

this.setBounds(305, 90, 970, 630);

this.setLayout(null);

this.getRootPane().setBorder(BorderFactory.createLineBorder(new Color(0x24a6db), 10));

this.setResizable(false);

this.setTitle("TravelPerk");

this.flag = false;

// Background image setup

ImageIcon i1 = new ImageIcon(getClass().getResource("/Images/image8.jpeg"));

Image i2 = i1.getImage().getScaledInstance(970, 630, Image.SCALE\_SMOOTH);

ImageIcon i3 = new ImageIcon(i2);

JLabel image = new JLabel(i3);

image.setBounds(0, 0, 970, 630);

// Title label setup

this.title = new JLabel("Book Trip");

this.title.setBounds(360, 30, 300, 50);

this.title.setFont(new Font("Tahoma", Font.BOLD, 35));

this.title.setForeground(Color.WHITE);

this.add(this.title);

// Destination choice setup

this.lbldestination = new JLabel("Choose Destination");

this.lbldestination.setBounds(30, 115, 200, 20);

this.lbldestination.setFont(new Font("SAN SERIF", Font.PLAIN, 20));

this.lbldestination.setForeground(Color.WHITE);

this.add(this.lbldestination);

this.destination = new Choice();

this.destination.add("London");

this.destination.add("Malaysia");

this.destination.add("Switzerland");

this.destination.add("Los Angeles");

this.destination.add("Turkey");

this.destination.setBounds(240, 115, 260, 35);

this.destination.setBackground(Color.WHITE);

this.destination.setForeground(new Color(0x24a6db));

this.add(this.destination);

// Search hotels button setup

this.search= new JButton("Search Hotels");

this.search.setBounds(540, 110, 185, 45);

this.search.setBackground(new Color(0x24a6db));

this.search.setBorder(new LineBorder(Color.WHITE));

this.search.setForeground(Color.WHITE);

this.search.setFont(new Font("SAN SERIF", Font.BOLD, 15));

this.search.setFocusPainted(false);

this.search.setBorder(BorderFactory.createLineBorder(Color.BLACK));

this.search.addActionListener(this);

this.search.setActionCommand("Search Hotels");

this.add(this.search);

// Hotels label and choice setup

this.lblhotels = new JLabel("Choose Hotel");

this.lblhotels.setBounds(30, 190, 200, 20);

this.lblhotels.setFont(new Font("SAN SERIF", Font.PLAIN, 20));

this.lblhotels.setForeground(Color.WHITE);

this.add(this.lblhotels);

this.hotels = new Choice();

this.hotels.setBounds(240, 190, 260, 35);

this.hotels.setBackground(Color.WHITE);

this.hotels.setForeground(new Color(0x24a6db));

this.add(this.hotels);

// Persons and days labels and text fields setup

this.lblpersons = new JLabel("Select persons");

this.lblpersons.setBounds(30, 265, 200, 20);

this.lblpersons.setFont(new Font("SAN SERIF", Font.PLAIN, 20));

this.lblpersons.setForeground(Color.WHITE);

this.add(this.lblpersons);

this.Persons = new JTextField("1");

this.Persons.setBounds(240, 260, 260, 35);

this.Persons.setBackground(Color.WHITE);

this.Persons.setForeground(Color.BLACK);

this.Persons.setBorder(BorderFactory.createLineBorder(new Color(127, 93, 37)));

this.add(this.Persons);

this.lblDay = new JLabel("Select Days");

this.lblDay.setBounds(30, 340, 200, 20);

this.lblDay.setFont(new Font("SAN SERIF", Font.PLAIN, 20));

this.lblDay.setForeground(Color.WHITE);

this.add(this.lblDay);

this.Days = new JTextField("1");

this.Days.setBounds(240, 335, 260, 35);

this.Days.setBackground(Color.WHITE);

this.Days.setForeground(Color.BLACK);

this.Days.setBorder(BorderFactory.createLineBorder(new Color(127, 93, 37)));

this.add(this.Days);

// Check price button setup

this.checkPrice= new JButton("Check Price");

this.checkPrice.setBounds(540, 335, 185, 45);

this.checkPrice.setBackground(new Color(0x24a6db));

this.checkPrice.setBorder(BorderFactory.createLineBorder(Color.WHITE));

this.checkPrice.setForeground(Color.WHITE);

this.checkPrice.setFont(new Font("SAN SERIF", Font.BOLD, 15));

this.checkPrice.setFocusPainted(false);

this.checkPrice.setBorder(BorderFactory.createLineBorder(Color.BLACK));

this.checkPrice.addActionListener(this);

this.checkPrice.setActionCommand("Check Price");

this.add(this.checkPrice);

// Total price label setup

this.lbltotal = new JLabel("Total Price");

this.lbltotal.setBounds(30, 415, 200, 20);

this.lbltotal.setFont(new Font("SAN SERIF", Font.PLAIN, 20));

this.lbltotal.setForeground(Color.WHITE);

this.add(this.lbltotal);

this.total = new JLabel();

this.total.setBounds(240, 415, 260, 35);

this.total.setFont(new Font("SAN SERIF", Font.PLAIN, 20));

this.total.setForeground(Color.WHITE);

this.add(this.total);

// Book button setup

this.book= new JButton("Book");

this.book.setBounds(135, 480, 280, 60);

this.book.setBackground(new Color(0x24a6db));

this.book.setBorder(BorderFactory.createLineBorder(Color.WHITE));

this.book.setForeground(Color.WHITE);

this.book.setFont(new Font("SAN SERIF", Font.BOLD, 15));

this.book.setFocusPainted(false);

this.book.setBorder(BorderFactory.createLineBorder(Color.BLACK));

this.book.addActionListener(this);

this.book.setActionCommand("Book");

this.add(this.book);

// Back button setup

this.back= new JButton("Back");

this.back.setBounds(505, 480, 280, 60);

this.back.setBackground(new Color(0x24a6db));

this.back.setBorder(BorderFactory.createLineBorder(Color.WHITE));

this.back.setForeground(Color.WHITE);

this.back.setFont(new Font("SAN SERIF", Font.BOLD, 15));

this.back.setFocusPainted(false);

this.back.setBorder(BorderFactory.createLineBorder(Color.BLACK));

this.back.addActionListener(this);

this.back.setActionCommand("Back");

this.add(this.back);

// Adding background image to frame

this.add(image);

this.setVisible(true);

}

/\*\*

\* Action listener method to handle button clicks.

\* @param e ActionEvent object representing the event

\*/

@Override

public void actionPerformed(ActionEvent e) {

String command = e.getActionCommand();

if(command.equals("Search Hotels")){

SelectHotel();

}

else if(command.equals("Check Price")){

if (hotels.getSelectedItem() == null) {

JOptionPane.showMessageDialog(this, "Please select a hotel first.", "Error", JOptionPane.ERROR\_MESSAGE);

} else {

calculatePrice();

}

}

else if(command.equals("Book")){

bookTrip();

}

else{

this.setVisible(false);

}

}

/\*\*

\* Method to populate hotel choices based on selected destination.

\*/

public void SelectHotel() {

hotels.removeAll(); // Clear previous selections

String selectedDestination = destination.getSelectedItem();

switch (selectedDestination) {

case "London":

hotels.add("Accor");

hotels.add("InterContinental Hotel");

hotels.add("Choice Hotel");

break;

case "Malaysia":

hotels.add("Marriott International");

hotels.add("Accor");

hotels.add("InterContinental Hotel");

break;

case "Switzerland":

hotels.add("Wyndham Hotels & Resorts");

hotels.add("Hyatt");

hotels.add("Marriott International");

break;

case "Turkey":

hotels.add("Choice Hotel");

hotels.add("Hilton Hotel and Resorts");

hotels.add("Accor");

break;

default: // Los Angeles

hotels.add("Wyndham Hotels & Resorts");

hotels.add("Hyatt");

hotels.add("Choice Hotel");

break;

}

}

/\*\*

\* Method to calculate and display the total trip cost.

\*/

public void calculatePrice() {

String selectedDestination = destination.getSelectedItem();

int cost = 0;

// Assign cost based on selected destination

switch (selectedDestination) {

case "London":

cost += 80000;

break;

case "Malaysia":

cost += 45000;

break;

case "Turkey":

cost += 25000;

break;

case "Los Angeles":

cost += 70000;

break;

default: // Switzerland

cost += 67000;

break;

}

// Calculate total cost based on number of persons and days

int persons = Integer.parseInt(Persons.getText());

int days = Integer.parseInt(Days.getText());

cost \*= persons;

cost \*= days;

total.setText(cost + " pkr"); // Display total cost in label

}

/\*\*

\* Method to handle booking of the trip, validating inputs and writing to file.

\*/

public void bookTrip() {

String selectedDestination = destination.getSelectedItem();

String selectedHotel = hotels.getSelectedItem();

String personsText = Persons.getText();

String daysText = Days.getText();

// Check if any required field is empty

if (selectedDestination == null || selectedHotel == null || selectedHotel.isEmpty() || personsText.isEmpty() || daysText.isEmpty()) {

JOptionPane.showMessageDialog(this, "Please fill all fields before booking.", "Error", JOptionPane.ERROR\_MESSAGE);

return; // Exit method without booking

}

// Check if total price is 0

String totalPriceStr = total.getText().trim();

if (totalPriceStr.isEmpty() || totalPriceStr.equals("0 pkr")) {

JOptionPane.showMessageDialog(this, "Cannot book trip", "Error", JOptionPane.ERROR\_MESSAGE);

return; // Exit method without booking

}

// Parse persons and days into integers

int persons = Integer.parseInt(personsText);

int days = Integer.parseInt(daysText);

String totalPrice = total.getText();

// Write booking details to file

try (BufferedWriter writer = new BufferedWriter(new FileWriter("bookings.txt", true))) {

writer.write("Destination: " + selectedDestination);

writer.newLine();

writer.write("Hotel: " + selectedHotel);

writer.newLine();

writer.write("Persons: " + persons);

writer.newLine();

writer.write("Days: " + days);

writer.newLine();

writer.write("Total Price: " + totalPrice);

writer.newLine();

writer.write("-------------------------------");

writer.newLine();

this.flag= true;

} catch (IOException e) {

e.printStackTrace();

}

// Show confirmation message

JOptionPane.showMessageDialog(null, "Trip Booked!");

}

public boolean getFlag(){

return flag;

}

}

/\*

\* This class represents the GUI for booking travel packages.

\*/

package travelmanagementsystem;

import java.awt.Choice;

import java.awt.Color;

import java.awt.Font;

import java.awt.Image;

import java.awt.event.ActionEvent;

import java.awt.event.ActionListener;

import java.io.BufferedWriter;

import java.io.FileWriter;

import java.io.IOException;

import javax.swing.BorderFactory;

import javax.swing.ImageIcon;

import javax.swing.JButton;

import javax.swing.JFrame;

import javax.swing.JLabel;

import javax.swing.JOptionPane;

import javax.swing.JPanel;

import javax.swing.JTextField;

import javax.swing.border.LineBorder;

/\*\*

\* GUI for booking packages in the travel management system.

\* Displays user information and allows package selection.

\* Handles booking and price calculation.

\*/

public class BookPackage extends JFrame implements ActionListener {

// UI components

private JTextField Persons;

private Choice Package;

private JButton checkPrice, Book, back;

private JLabel total, title, lblname, lblpackage, lblpersons, lblid, ID, lblnumber, nmbr, lbltotal;

private JPanel p1, p2;

// Customer information

String customerName, customerId, customerNumber;

boolean flag;

// Constructor initializes the booking GUI with customer details.

public BookPackage(String name, String id, String number) {

this.setBounds(305, 90, 970, 630);

this.setLayout(null);

this.setResizable(false);

this.setTitle("TravelPerk");

this.flag = false;

this.customerName = name;

this.customerId = id;

this.customerNumber = number;

// Background image setup

ImageIcon i1 = new ImageIcon(getClass().getResource("/Images/image86.jpg"));

Image i2 = i1.getImage().getScaledInstance(970, 630, Image.SCALE\_SMOOTH);

ImageIcon i3 = new ImageIcon(i2);

JLabel image = new JLabel(i3);

image.setBounds(0, 0, 970, 630);

// Title label

title = new JLabel("Book Package");

title.setBounds(330, 40, 400, 80);

title.setFont(new Font("SAN SERIF", Font.TYPE1\_FONT, 40));

title.setForeground(Color.WHITE);

this.add(title);

// Customer name label

lblname = new JLabel("Name");

lblname.setBounds(30, 120, 150, 20);

lblname.setFont(new Font("SAN SERIF", Font.BOLD, 20));

lblname.setForeground(Color.WHITE);

this.add(lblname);

JLabel Name = new JLabel(name);

Name.setBounds(210, 120, 150, 20);

Name.setFont(new Font("SAN SERIF", Font.BOLD, 20));

Name.setForeground(Color.white);

this.add(Name);

// Package selection label

lblpackage = new JLabel("Select Package");

lblpackage.setBounds(30, 180, 150, 35);

lblpackage.setFont(new Font("SAN SERIF", Font.BOLD, 20));

lblpackage.setForeground(Color.white);

this.add(lblpackage);

// Package options

Package = new Choice();

Package.add("Ultimate Adventure");

Package.add("Luxury Retreat");

Package.add("Cultural Exploration");

Package.setBounds(210, 180, 260, 35);

Package.setBackground(Color.white);

Package.setForeground(Color.BLACK);

this.add(Package);

// Total persons label and text field

lblpersons = new JLabel("Total Persons");

lblpersons.setBounds(30, 240, 150, 35);

lblpersons.setFont(new Font("SAN SERIF", Font.BOLD, 20));

lblpersons.setForeground(Color.WHITE);

this.add(lblpersons);

Persons = new JTextField("1");

Persons.setBounds(210, 235, 260, 35);

Persons.setBackground(Color.WHITE);

Persons.setForeground(Color.BLACK);

Persons.setBorder(BorderFactory.createLineBorder(new Color(127, 93, 37)));

this.add(Persons);

// ID label

lblid = new JLabel("ID");

lblid.setBounds(30, 300, 150, 35);

lblid.setFont(new Font("SAN SERIF", Font.BOLD, 20));

lblid.setForeground(Color.WHITE);

this.add(lblid);

ID = new JLabel(id);

ID.setBounds(210, 300, 150, 35);

ID.setFont(new Font("SAN SERIF", Font.BOLD, 20));

ID.setForeground(Color.WHITE);

this.add(ID);

// Number label

lblnumber = new JLabel("Number");

lblnumber.setBounds(30, 360, 150, 35);

lblnumber.setFont(new Font("SAN SERIF", Font.BOLD, 20));

lblnumber.setForeground(Color.WHITE);

this.add(lblnumber);

nmbr = new JLabel(number);

nmbr.setBounds(210, 360, 150, 35);

nmbr.setFont(new Font("SAN SERIF", Font.BOLD, 20));

nmbr.setForeground(Color.WHITE);

this.add(nmbr);

// Total price label

lbltotal = new JLabel("Total Price");

lbltotal.setBounds(30, 420, 150, 35);

lbltotal.setFont(new Font("SAN SERIF", Font.BOLD, 20));

lbltotal.setForeground(Color.WHITE);

this.add(lbltotal);

total = new JLabel();

total.setBounds(210, 420, 150, 35);

total.setFont(new Font("SAN SERIF", Font.BOLD, 20));

total.setForeground(Color.WHITE);

this.add(total);

// Book button

Book = new JButton("Book");

Book.setBounds(30, 485, 190, 45);

Book.setBackground(new Color(0x609d9f));

Book.setBorder(new LineBorder(Color.WHITE));

Book.setForeground(Color.WHITE);

Book.setFont(new Font("SAN SERIF", Font.BOLD, 15));

Book.setFocusPainted(false);

Book.setBorder(BorderFactory.createLineBorder(Color.BLACK));

Book.addActionListener(this);

Book.setActionCommand("Book");

this.add(Book);

// Check price button

checkPrice = new JButton("Check Price");

checkPrice.setBounds(265, 485, 190, 45);

checkPrice.setBackground(new Color(0x609d9f));

checkPrice.setBorder(new LineBorder(Color.WHITE));

checkPrice.setForeground(Color.WHITE);

checkPrice.setFont(new Font("SAN SERIF", Font.BOLD, 15));

checkPrice.setFocusPainted(false);

checkPrice.setBorder(BorderFactory.createLineBorder(Color.BLACK));

checkPrice.addActionListener(this);

checkPrice.setActionCommand("Check Price");

this.add(checkPrice);

// Back button

back = new JButton("Back");

back.setBounds(700, 485, 185, 45);

back.setBackground(new Color(0x609d9f));

back.setBorder(new LineBorder(Color.WHITE));

back.setForeground(Color.WHITE);

back.setFont(new Font("SAN SERIF", Font.BOLD, 15));

back.setFocusPainted(false);

back.setBorder(BorderFactory.createLineBorder(Color.BLACK));

back.addActionListener(this);

back.setActionCommand("Back");

this.add(back);

// Image panels

p1 = new JPanel();

p1.setBounds(580, 130, 280, 180);

p1.setBackground(Color.red);

p1.setLayout(null);

ImageIcon i4 = new ImageIcon(getClass().getResource("/Images/image83.jpg"));

Image i5 = i4.getImage().getScaledInstance(280, 180, Image.SCALE\_SMOOTH);

ImageIcon i6 = new ImageIcon(i5);

JLabel image1 = new JLabel(i6);

image1.setBounds(0, 0, 280, 180);

p1.add(image1);

p2 = new JPanel();

p2.setBounds(540, 260, 280, 180);

p2.setBackground(Color.red);

p2.setLayout(null);

ImageIcon i7 = new ImageIcon(getClass().getResource("/Images/image56.jpg"));

Image i8 = i7.getImage().getScaledInstance(280, 180, Image.SCALE\_SMOOTH);

ImageIcon i9 = new ImageIcon(i8);

JLabel image2 = new JLabel(i9);

image2.setBounds(0, 0, 280, 180);

p2.add(image2);

// Adding components to the frame

this.add(p2);

this.add(p1);

this.add(image);

this.setVisible(true);

}

/\*\*

\* Saves the booking details to a file.

\*/

private void saveBookingDetailsToFile() {

String packageName = Package.getSelectedItem();

String totalPersons = Persons.getText();

String totalPrice = total.getText();

try (BufferedWriter writer = new BufferedWriter(new FileWriter("bookedPackages.txt", true))) {

writer.write("Customer Name: " + customerName);

writer.newLine();

writer.write("Customer ID: " + customerId);

writer.newLine();

writer.write("Customer Number: " + customerNumber);

writer.newLine();

writer.write("Package: " + packageName);

writer.newLine();

writer.write("Total Persons: " + totalPersons);

writer.newLine();

writer.write("Total Price: " + totalPrice);

writer.newLine();

writer.newLine();

writer.flush();

this.flag = true; // Indicates booking was saved

} catch (IOException e) {

e.printStackTrace(); // Handle exceptions

}

}

/\*\*

\* Handles button actions.

\*

\* @param e Action event

\*/

@Override

public void actionPerformed(ActionEvent e) {

String command = e.getActionCommand();

if (command.equals("Book")) {

String packageName = Package.getSelectedItem();

String totalPersons = Persons.getText();

String totalPrice = total.getText();

if (packageName.isEmpty() || totalPersons.isEmpty() || totalPrice.isEmpty()) {

JOptionPane.showMessageDialog(null, "Please fill in all fields."); // Error message

} else {

saveBookingDetailsToFile();

JOptionPane.showMessageDialog(null, "Package Booked!"); // Confirmation

}

} else if (command.equals("Check Price")) {

String pack = Package.getSelectedItem();

int cost = 0;

if (pack.equals("Ultimate Adventure")) {

cost += 120000; // Price for Ultimate Adventure

} else if (pack.equals("Luxury Retreat")) {

cost += 340000; // Price for Luxury Retreat

} else {

cost += 240000; // Price for Cultural Exploration

}

int persons = Integer.parseInt(Persons.getText());

cost \*= persons;

total.setText(cost + " pkr"); // Display total price

} else {

this.setVisible(false); // Close the window

}

}

// Returns the flag indicating booking status.

public boolean getFlag() {

return this.flag;

}

}

/\*

\* Click nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-default.txt to change this license

\* Click nbfs://nbhost/SystemFileSystem/Templates/Classes/Class.java to edit this template

\*/

package travelmanagementsystem;

import java.awt.Color;

import java.awt.Font;

import java.awt.Image;

import javax.swing.ImageIcon;

import javax.swing.JFrame;

import javax.swing.JLabel;

/\*\*

\* This class represents a JFrame for checking hotels with a slideshow of hotel images and names.

\* It implements Runnable to run a thread that cycles through hotel names and displays corresponding images.

\* Each image is displayed for a set duration before switching to the next.

\*

\* @author Memon enterprise

\*/

public class CheckHotels extends JFrame implements Runnable {

private Thread t;

private JLabel l1, l2, l3, l4, l5, l6, l7;

private JLabel[] label = new JLabel[]{l1, l2, l3, l4, l5, l6, l7};

private JLabel captions;

private ImageIcon i1 = null, i2 = null, i3 = null, i4 = null, i5 = null, i6 = null, i7 = null;

private Image j1 = null, j2 = null, j3 = null, j4 = null, j5 = null, j6 = null, j7 = null;

private ImageIcon k1 = null, k2 = null, k3 = null, k4 = null, k5 = null, k6 = null, k7 = null;

/\*\*

\* Constructor to initialize the JFrame and setup hotel images and captions.

\* It sets the initial bounds, layout, and starts a thread for the slideshow.

\*/

public CheckHotels() {

//set frame

this.setBounds(305, 90, 970, 630);

this.setLayout(null);

this.setResizable(false);

this.setTitle("TravelPerk");

//Displaythe name of hotels

captions = new JLabel();

captions.setBounds(30, 500, 600, 60);

captions.setFont(new Font("Tahoma", Font.BOLD, 40));

captions.setForeground(Color.WHITE);

this.add(captions);

ImageIcon[] image = new ImageIcon[]{i1, i2, i3, i4, i5, i6, i7};

Image[] jimage = new Image[]{j1, j2, j3, j4, j5, j6, j7};

ImageIcon[] kimage = new ImageIcon[]{k1, k2, k3, k4, k5, k6, k7};

// Load images and scale them for display

for (int i = 0; i < 7; i++) {

image[i] = new ImageIcon(getClass().getResource("/Images/image" + (i + 87) + ".jpg"));

jimage[i] = image[i].getImage().getScaledInstance(970, 630, Image.SCALE\_SMOOTH);

kimage[i] = new ImageIcon(jimage[i]);

label[i] = new JLabel(kimage[i]);

label[i].setBounds(0, 0, 970, 630);

label[i].setVisible(false);

this.add(label[i]);

}

label[0].setVisible(true); // Show the first image

t = new Thread(this); // Initialize thread for slideshow

t.start(); // Start the thread

this.setVisible(true); // Make the frame visible

}

/\*\*

\* Runnable interface method implementation.

\* Runs a thread that cycles through hotel names and displays corresponding images in a slideshow manner.

\*/

@Override

public void run() {

String[] hotelNames = new String[]{"Hyatt", "Accor", "Wyndham Hotels & Resorts",

"InterContinental Hotel", "Choice Hotel",

"Marriott International", "Hilton Hotel and Resorts"};

try {

for (int i = 0; i < 6; i++) {

label[i].setVisible(true);

captions.setText(hotelNames[i]);

Thread.sleep(2500); // Display each image for 2.5 seconds

label[i].setVisible(false);

if (i == 5) {

i = -1; // Reset index to loop through hotel names again

}

}

} catch (InterruptedException e) {

e.printStackTrace();

}

}

}

/\*

\* Click nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-default.txt to change this license

\* Click nbfs://nbhost/SystemFileSystem/Templates/Classes/Class.java to edit this template

\*/

package travelmanagementsystem;

import java.awt.Color;

import java.awt.Font;

import java.awt.Image;

import javax.swing.ImageIcon;

import javax.swing.JFrame;

import javax.swing.JLabel;

import javax.swing.JPanel;

import javax.swing.JTabbedPane;

/\*\*

\* This class represents a window that displays different travel packages

\* in a tabbed pane format.

\* Each tab contains information about a specific package, including

\* activities, duration, location, and price.

\* Users can browse packages and view images associated with each package.

\*/

public class CheckPackages extends JFrame {

private JTabbedPane tab;

private JPanel p1, p2, p3, p4, package2, p5, p6, p7, package3, p9, p10, p11;

private JLabel l1, l2, l3, l5, l6, l7, l8, l9, l10, l11, l12, l13, l14, l15, l16, l18, l19, lbl1, lbl2, l20, l21, l22, lbl3, l30, l31, l32, l33, l34, l35, l36, l37, l38;

/\*\*

\* Constructor for CheckPackages.

\* Initializes the GUI components and layout for each travel package.

\*/

public CheckPackages() {

// Set frame size and position

this.setBounds(305, 90, 970, 630);

this.setResizable(false);

this.setTitle("TravelPerk");

tab = new JTabbedPane(); // Create tabbed pane

// Package 1 - Ultimate Adventure Package

p1 = new JPanel();

p1.setLayout(null);

p1.setBackground(new Color(0x587935));

// Label for package title

l1 = new JLabel("Ultimate Adventure Package");

l1.setForeground(Color.WHITE);

l1.setBounds(40, -50, 500, 200);

l1.setFont(new Font("Tahoma", Font.BOLD, 27));

p1.add(l1);

// Additional labels for package details

l2 = new JLabel("3 Days and 4 Nights");

l2.setForeground(Color.white);

l2.setBounds(40, 55, 400, 100);

l2.setFont(new Font("Tahoma", Font.BOLD, 18));

p1.add(l2);

l3 = new JLabel("Package for Thrill-seekers and Adventure");

l3.setForeground(Color.white);

l3.setBounds(40, 100, 500, 100);

l3.setFont(new Font("Tahoma", Font.BOLD, 18));

p1.add(l3);

l12 = new JLabel("Enthusiasts");

l12.setForeground(Color.white);

l12.setBounds(40, 120, 400, 100);

l12.setFont(new Font("Tahoma", Font.BOLD, 18));

p1.add(l12);

l5 = new JLabel("Hiking, Zip Lining");

l5.setForeground(Color.white);

l5.setBounds(40, 164, 400, 100);

l5.setFont(new Font("Tahoma", Font.BOLD, 18));

p1.add(l5);

l6 = new JLabel("White Water Rafting");

l6.setForeground(Color.white);

l6.setBounds(40, 210, 400, 100);

l6.setFont(new Font("Tahoma", Font.BOLD, 18));

p1.add(l6);

l9 = new JLabel("Camping");

l9.setForeground(Color.white);

l9.setBounds(40, 255, 400, 100);

l9.setFont(new Font("Tahoma", Font.BOLD, 18));

p1.add(l9);

l10 = new JLabel("Great Smoky Mountains National Park,");

l10.setForeground(Color.white);

l10.setBounds(40, 300, 400, 100);

l10.setFont(new Font("Tahoma", Font.BOLD, 18));

p1.add(l10);

l11 = new JLabel("Tennessee/North Carolina");

l11.setForeground(Color.white);

l11.setBounds(40, 320, 400, 100);

l11.setFont(new Font("Tahoma", Font.BOLD, 18));

p1.add(l11);

l8 = new JLabel("120000 PKR per person");

l8.setForeground(Color.white);

l8.setBounds(40, 365, 400, 100);

l8.setFont(new Font("Tahoma", Font.BOLD, 18));

p1.add(l8);

// Panel for package images

p2 = new JPanel();

p2.setBounds(580, 80, 270, 190);

p2.setBackground(Color.red);

p2.setLayout(null);

// Load and scale image for display

ImageIcon i1 = new ImageIcon(getClass().getResource("/Images/image65.jpg"));

Image i2 = i1.getImage().getScaledInstance(270, 190, Image.SCALE\_SMOOTH);

ImageIcon i3 = new ImageIcon(i2);

JLabel image1 = new JLabel(i3);

image1.setBounds(0, 0, 270, 190);

p2.add(image1);

p1.add(p2);

// Additional image panels with similar configurations

p3 = new JPanel();

p3.setBounds(450, 240, 270, 190);

p3.setBackground(Color.red);

p3.setLayout(null);

ImageIcon i4 = new ImageIcon(getClass().getResource("/Images/image66.jpg"));

Image i5 = i4.getImage().getScaledInstance(270, 190, Image.SCALE\_AREA\_AVERAGING);

ImageIcon i6 = new ImageIcon(i5);

JLabel image2 = new JLabel(i6);

image2.setBounds(0, 0, 270, 190);

p3.add(image2);

p1.add(p3);

p4 = new JPanel();

p4.setBounds(630, 330, 270, 190);

p4.setBackground(Color.red);

p4.setLayout(null);

ImageIcon i7 = new ImageIcon(getClass().getResource("/Images/image67.jpg"));

Image i8 = i7.getImage().getScaledInstance(270, 190, Image.SCALE\_SMOOTH);

ImageIcon i9 = new ImageIcon(i8);

JLabel image3 = new JLabel(i9);

image3.setBounds(0, 0, 270, 190);

p4.add(image3);

p1.add(p4);

// Label for booking call-to-action

lbl1 = new JLabel("BOOK NOW");

lbl1.setBounds(80, 390, 400, 200);

lbl1.setForeground(Color.WHITE);

lbl1.setFont(new Font("SAN SERIF", Font.BOLD, 30));

p1.add(lbl1);

// Background image for the first package

ImageIcon i25 = new ImageIcon(getClass().getResource("/Images/image69.jpg"));

Image i26 = i25.getImage().getScaledInstance(970, 580, Image.SCALE\_SMOOTH);

ImageIcon i27 = new ImageIcon(i26);

JLabel image7 = new JLabel(i27);

image7.setBounds(0, 0, 970, 580);

p1.add(image7);

// Initialize other packages (Package 2 and 3) with similar configurations

// Package 2 - Luxury Retreat Package

package2 = new JPanel();

package2.setLayout(null);

package2.setBackground(Color.red);

l13 = new JLabel("Luxury Retreat Package");

l13.setForeground(Color.black);

l13.setBounds(40, -50, 500, 200);

l13.setFont(new Font("Tahoma", Font.BOLD, 27));

package2.add(l13);

l14 = new JLabel("5 Days and 6 Nights");

l14.setForeground(Color.black);

l14.setBounds(40, 55, 400, 100);

l14.setFont(new Font("Tahoma", Font.BOLD, 18));

package2.add(l14);

l15 = new JLabel("Package offers a luxurious and relaxing");

l15.setForeground(Color.black);

l15.setBounds(40, 100, 500, 100);

l15.setFont(new Font("Tahoma", Font.BOLD, 18));

package2.add(l15);

l16 = new JLabel("experience");

l16.setForeground(Color.black);

l16.setBounds(40, 120, 400, 100);

l16.setFont(new Font("Tahoma", Font.BOLD, 18));

package2.add(l16);

l7 = new JLabel("Spa Treatments, Gourmet Dining");

l7.setForeground(Color.black);

l7.setBounds(40, 164, 400, 100);

l7.setFont(new Font("Tahoma", Font.BOLD, 18));

package2.add(l7);

l18 = new JLabel("Private Yacht Tour");

l18.setForeground(Color.black);

l18.setBounds(40, 210, 400, 100);

l18.setFont(new Font("Tahoma", Font.BOLD, 18));

package2.add(l18);

l19 = new JLabel("Premium accommodations and indulgent");

l19.setForeground(Color.black);

l19.setBounds(40 , 255 , 400 , 100);

l19.setFont(new Font("Tahoma" , Font.BOLD , 18));

package2.add(l19);

l20 = new JLabel("activities");

l20.setForeground(Color.black);

l20.setBounds(40 , 275 , 400 , 100);

l20.setFont(new Font("Tahoma" , Font.BOLD , 18));

package2.add(l20);

l21 = new JLabel("Maldives");

l21.setForeground(Color.black);

l21.setBounds(40 , 320 , 400 , 100);

l21.setFont(new Font("Tahoma" , Font.BOLD , 18));

package2.add(l21);

l22 = new JLabel("340000 per person");

l22.setForeground(Color.black);

l22.setBounds(40 , 365 , 400 , 100);

l22.setFont(new Font("Tahoma" , Font.BOLD , 18));

package2.add(l22);

// Panel for package 2 images

p5 = new JPanel();

p5.setBounds(580 , 80 , 270 , 190);

p5.setBackground(Color.red);

p5.setLayout(null);

ImageIcon i10 = new ImageIcon(getClass().getResource("/Images/image71.jpg"));

Image i11 = i10.getImage().getScaledInstance(270, 190, Image.SCALE\_SMOOTH);

ImageIcon i12 = new ImageIcon(i11);

JLabel image4 = new JLabel(i12);

image4.setBounds(0 , 0 , 270 , 190);

p5.add(image4);

package2.add(p5);

p6 = new JPanel();

p6.setBounds(450 , 240 , 270 , 190);

p6.setBackground(Color.red);

p6.setLayout(null);

ImageIcon i13 = new ImageIcon(getClass().getResource("/Images/image72.jpg"));

Image i14 = i13.getImage().getScaledInstance(270, 190, Image.SCALE\_AREA\_AVERAGING);

ImageIcon i15 = new ImageIcon(i14);

JLabel image5 = new JLabel(i15);

image5.setBounds(0 , 0 , 270 , 190);

p6.add(image5);

package2.add(p6);

p7 = new JPanel();

p7.setBounds(630 , 330 , 270 , 190);

p7.setBackground(Color.red);

p7.setLayout(null);

ImageIcon i16 = new ImageIcon(getClass().getResource("/Images/image73.jpg"));

Image i17 = i16.getImage().getScaledInstance(270, 190, Image.SCALE\_SMOOTH);

ImageIcon i18 = new ImageIcon(i17);

JLabel image6 = new JLabel(i18);

image6.setBounds(0 , 0 , 270 , 190);

p7.add(image6);

package2.add(p7);

// Label for booking call-to-action in package 2

lbl2 = new JLabel("BOOK NOW");

lbl2.setBounds(80 , 390 , 400 , 200);

lbl2.setForeground(Color.black);

lbl2.setFont(new Font("SAN SERIF" , Font.BOLD , 30));

package2.add(lbl2);

// Background image for package 2

ImageIcon i28 = new ImageIcon(getClass().getResource("/Images/image74.jpg"));

Image i29 = i28.getImage().getScaledInstance(970, 580, Image.SCALE\_AREA\_AVERAGING);

ImageIcon i30 = new ImageIcon(i29);

JLabel image8 = new JLabel(i30);

image8.setBounds(0 , 0 , 970 , 580);

package2.add(image8);

// Package 3 - Cultural Exploration Package

package3 = new JPanel();

package3.setLayout(null);

package3.setBackground(Color.red);

l30 = new JLabel("Cultural Exploration Package");

l30.setForeground(Color.WHITE);

l30.setBounds(40 , -50 , 500 , 200);

l30.setFont(new Font("Tahoma" , Font.BOLD , 27));

package3.add(l30);

l31 = new JLabel("7 Days and 8 Nights");

l31.setForeground(Color.white);

l31.setBounds(40 , 55 , 400 , 100);

l31.setFont(new Font("Tahoma" , Font.BOLD , 18));

package3.add(l31);

l32 = new JLabel("Ideal for travelers interested in immersing themselves");

l32.setForeground(Color.white);

l32.setBounds(40 , 100 , 500 , 100);

l32.setFont(new Font("Tahoma" , Font.BOLD , 18));

package3.add(l32);

l33 = new JLabel("in the history, culture, and traditions");

l33.setForeground(Color.white);

l33.setBounds(40 , 120 , 400 , 100);

l33.setFont(new Font("Tahoma" , Font.BOLD , 18));

package3.add(l33);

l34 = new JLabel("Historical Tours, Museum Visits");

l34.setForeground(Color.white);

l34.setBounds(40 , 164 , 400 , 100);

l34.setFont(new Font("Tahoma" , Font.BOLD , 18));

package3.add(l34);

l35 = new JLabel("Cultural Shows, Cooking Classes");

l35.setForeground(Color.white);

l35.setBounds(40 , 210 , 400 , 100);

l35.setFont(new Font("Tahoma" , Font.BOLD , 18));

package3.add(l35);

l36 = new JLabel("Premium accommodations");

l36.setForeground(Color.white);

l36.setBounds(40 , 255 , 400 , 100);

l36.setFont(new Font("Tahoma" , Font.BOLD , 18));

package3.add(l36);

l37 = new JLabel("Kyoto, Japan");

l37.setForeground(Color.white);

l37.setBounds(40 , 300 , 400 , 100);

l37.setFont(new Font("Tahoma" , Font.BOLD , 18));

package3.add(l37);

l38 = new JLabel("240000 PKR per person");

l38.setForeground(Color.white);

l38.setBounds(40 , 345 , 400 , 100);

l38.setFont(new Font("Tahoma" , Font.BOLD , 18));

package3.add(l38);

// Panel for package 3 images

p9 = new JPanel();

p9.setBounds(580 , 90 , 270 , 190);

p9.setBackground(Color.red);

p9.setLayout(null);

ImageIcon i20 = new ImageIcon(getClass().getResource("/Images/image76.jpg"));

Image i21 = i20.getImage().getScaledInstance(270, 190, Image.SCALE\_SMOOTH);

ImageIcon i22 = new ImageIcon(i21);

JLabel image9 = new JLabel(i22);

image9.setBounds(0 , 0 , 270 , 190);

p9.add(image9);

package3.add(p9);

p10 = new JPanel();

p10.setBounds(450 , 240 , 270 , 190);

p10.setBackground(Color.red);

p10.setLayout(null);

ImageIcon i23 = new ImageIcon(getClass().getResource("/Images/image79.jpg"));

Image i24 = i23.getImage().getScaledInstance(270, 190, Image.SCALE\_AREA\_AVERAGING);

ImageIcon i31 = new ImageIcon(i24);

JLabel image10 = new JLabel(i31);

image10.setBounds(0 , 0 , 270 , 190);

p10.add(image10);

package3.add(p10);

p11 = new JPanel();

p11.setBounds(630 , 330 , 270 , 190);

p11.setBackground(Color.red);

p11.setLayout(null);

ImageIcon i32 = new ImageIcon(getClass().getResource("/Images/image77.jpg"));

Image i33 = i32.getImage().getScaledInstance(270, 190, Image.SCALE\_SMOOTH);

ImageIcon i34 = new ImageIcon(i33);

JLabel image11 = new JLabel(i34);

image11.setBounds(0 , 0 , 270 , 190);

p11.add(image11);

package3.add(p11);

// Label for booking call-to-action in package 3

lbl3 = new JLabel("BOOK NOW");

lbl3.setBounds(80 , 370 , 400 , 200);

lbl3.setForeground(Color.WHITE);

lbl3.setFont(new Font("SAN SERIF" , Font.BOLD , 30));

package3.add(lbl3);

// Background image for package 3

ImageIcon i35 = new ImageIcon(getClass().getResource("/Images/image78.jpg"));

Image i36 = i35.getImage().getScaledInstance(970, 580, Image.SCALE\_DEFAULT);

ImageIcon i37 = new ImageIcon(i36);

JLabel image12 = new JLabel(i37);

image12.setBounds(0 , 0 , 970 , 580);

package3.add(image12);

// Add tabs for each package to the tabbed pane

tab.addTab("Package 1", null , p1);

tab.addTab("Package 2", null , package2);

tab.addTab("Package 3", null , package3);

// Add the tabbed pane to the frame and set it visible

this.add(tab);

this.setVisible(true);

}

}

/\*

\* Click nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-default.txt to change this license

\* Click nbfs://nbhost/SystemFileSystem/Templates/Classes/Class.java to edit this template

\*/

package travelmanagementsystem;

import java.awt.Color;

import java.awt.Font;

import java.awt.Frame;

import java.awt.Image;

import java.awt.event.ActionEvent;

import java.awt.event.ActionListener;

import java.io.BufferedReader;

import java.io.FileReader;

import java.io.IOException;

import javax.swing.BorderFactory;

import javax.swing.ImageIcon;

import javax.swing.JButton;

import javax.swing.JFrame;

import javax.swing.JLabel;

import javax.swing.JLayeredPane;

import javax.swing.JOptionPane;

import javax.swing.JPanel;

/\*\*

\*

\* @author Memon enterprise

\*/

public class Dashboard extends JFrame implements ActionListener{

// Fields to store customer details

private String name , id , number, email, address, gender , country;

// Buttons for various actions

private JButton viewDetails , viewTripDetails , personalDetails , checkPackage, bookPackages , viewPackage , viewHotels , checkDestinations , bookTrip , Payments , about , signout;

// Flags for tracking user actions

private boolean viewDestination , add , check;

// Instances of other classes used in the dashboard

private BookDestination bookDestination;

private AddCustomerDetail addCustomer;

private BookPackage bookpackage;

// Panels for organizing UI components

private JPanel p1 , p2;

private JLayeredPane p3;

private JLabel titleH;

/\*\*

\* Constructor for the Dashboard class.

\* Initializes the UI components and layout of the dashboard.

\*/

public Dashboard(){

this.setExtendedState(JFrame.MAXIMIZED\_BOTH);

this.setLayout(null);

this.setResizable(false);

this.setTitle("TravelPerk");

this.viewDestination = false;

this.add = false;

this.check =false;

// Panel p1 configuration

this.p1 = new JPanel();

this.p1.setLayout(null);

this.p1.setLayout(null);

this.p1.setBounds(0 , 0 , 1600 , 65);

this.p1.setBackground(new Color(55 , 119 , 131));

//Logo image in panel p1

ImageIcon img1 = new ImageIcon(getClass().getResource("/Images/image51.png"));

Image img2 = img1.getImage().getScaledInstance(40, 40, Image.SCALE\_AREA\_AVERAGING);

ImageIcon img3 = new ImageIcon(img2);

JLabel image2 = new JLabel(img3);

image2.setBounds(35 , 10 , 40 , 40);

p1.add(image2);

// Title label in panel p1

JLabel title = new JLabel("Dashboard");

title.setBounds(85 , 10 , 200 , 55);

title.setFont(new Font("SAN SERIK" , Font.BOLD , 28));

title.setForeground(Color.WHITE);

p1.add(title);

this.add(this.p1);

// Set window border

this.getRootPane().setBorder(BorderFactory.createLineBorder(new Color(35 , 99 , 111) , 10));

// Panel p2 configuration

this.p2 = new JPanel();

this.p2.setLayout(null);

this.p2.setLayout(null);

this.p2.setBounds(0 , 0 , 300 , 800);

this.p2.setBackground(new Color(55 , 119 , 131));

this.add(this.p2);

// Buttons in panel p2 for various functionalities

// Add personal details buttons

this.personalDetails = new JButton("Add Personal Details");

this.personalDetails.setBounds(0 , 65 , 300 , 51);

this.personalDetails.setBackground(new Color(55 , 119 , 131));

this.personalDetails.setFont(new Font("SAN SERIK" , Font.PLAIN , 20));

this.personalDetails.setFocusPainted(false);

this.personalDetails.setBorder(BorderFactory.createLineBorder(Color.BLACK));

this.personalDetails.addActionListener(this);

this.personalDetails.setActionCommand("Add Personal Details");

this.p2.add(this.personalDetails);

//View personal details button

this.viewDetails = new JButton("View Personal Details");

this.viewDetails.setBounds(0 , 116 , 300 , 51);

this.viewDetails.setBackground(new Color(55 , 119 , 131));

this.viewDetails.setFont(new Font("SAN SERIK" , Font.PLAIN , 20));

this.viewDetails.setFocusPainted(false);

this.viewDetails.setBorder(BorderFactory.createLineBorder(Color.BLACK));

this.viewDetails.addActionListener(this);

this.viewDetails.setActionCommand("View Personal Details");

this.p2.add(this.viewDetails);

// Check available packages button

this.checkPackage = new JButton("Check Packages");

this.checkPackage.setBounds(0 , 167 , 300 , 51);

this.checkPackage.setBackground(new Color(55 , 119 , 131));

this.checkPackage.setFont(new Font("SAN SERIK" , Font.PLAIN , 20));

this.checkPackage.setFocusPainted(false);

this.checkPackage.setBorder(BorderFactory.createLineBorder(Color.BLACK));

this.checkPackage.addActionListener(this);

this.checkPackage.setActionCommand("Check Packages");

this.p2.add(this.checkPackage);

//Book package button

this.bookPackages = new JButton("Book Package");

this.bookPackages.setBounds(0 , 218 , 300 , 51);

this.bookPackages.setBackground(new Color(55 , 119 , 131));

this.bookPackages.setFont(new Font("SAN SERIK" , Font.PLAIN , 20));

this.bookPackages.setFocusPainted(false);

this.bookPackages.setBorder(BorderFactory.createLineBorder(Color.BLACK));

this.bookPackages.addActionListener(this);

this.bookPackages.setActionCommand("Book Package");

this.p2.add(this.bookPackages);

//View booked package button

this.viewPackage = new JButton("View Booked Package");

this.viewPackage.setBounds(0 , 269 , 300 , 51);

this.viewPackage.setBackground(new Color(55 , 119 , 131));

this.viewPackage.setFont(new Font("SAN SERIK" , Font.PLAIN , 20));

this.viewPackage.setFocusPainted(false);

this.viewPackage.setBorder(BorderFactory.createLineBorder(Color.BLACK));

this.viewPackage.addActionListener(this);

this.viewPackage.setActionCommand("View Booked Package");

this.p2.add(this.viewPackage);

// View available hotels button

this.viewHotels = new JButton("View Hotels");

this.viewHotels.setBounds(0 , 320 , 300 , 51);

this.viewHotels.setBackground(new Color(55 , 119 , 131));

this.viewHotels.setFont(new Font("SAN SERIK" , Font.PLAIN , 20));

this.viewHotels.setFocusPainted(false);

this.viewHotels.setBorder(BorderFactory.createLineBorder(Color.BLACK));

this.viewHotels.addActionListener(this);

this.viewHotels.setActionCommand("View Hotels");

this.p2.add(this.viewHotels);

//Check available destinations button

this.checkDestinations = new JButton("View Destinations");

this.checkDestinations.setBounds(0 , 371 , 300 , 51);

this.checkDestinations.setBackground(new Color(55 , 119 , 131));

this.checkDestinations.setFont(new Font("SAN SERIK" , Font.PLAIN , 20));

this.checkDestinations.setFocusPainted(false);

this.checkDestinations.setBorder(BorderFactory.createLineBorder(Color.BLACK));

this.checkDestinations.addActionListener(this);

this.checkDestinations.setActionCommand("View Destinations");

this.p2.add(this.checkDestinations);

// Book trip button

this.bookTrip = new JButton("Book Trip");

this.bookTrip.setBounds(0 , 422 , 300 , 51);

this.bookTrip.setBackground(new Color(55 , 119 , 131));

this.bookTrip.setFont(new Font("SAN SERIK" , Font.PLAIN , 20));

this.bookTrip.setFocusPainted(false);

this.bookTrip.setBorder(BorderFactory.createLineBorder(Color.BLACK));

this.bookTrip.addActionListener(this);

this.bookTrip.setActionCommand("Book Trip");

this.p2.add(this.bookTrip);

// View booked trip details button

this.viewTripDetails = new JButton("View Trip Details");

this.viewTripDetails.setBounds(0 , 473 , 300 , 51);

this.viewTripDetails.setBackground(new Color(55 , 119 , 131));

this.viewTripDetails.setFont(new Font("SAN SERIK" , Font.PLAIN , 20));

this.viewTripDetails.setFocusPainted(false);

this.viewTripDetails.setBorder(BorderFactory.createLineBorder(Color.BLACK));

this.viewTripDetails.addActionListener(this);

this.viewTripDetails.setActionCommand("View Trip Details");

this.p2.add(this.viewTripDetails);

//Make payment button

this.Payments = new JButton("Payments");

this.Payments.setBounds(0 , 524 , 300 , 51);

this.Payments.setBackground(new Color(55 , 119 , 131));

this.Payments.setFont(new Font("SAN SERIK" , Font.PLAIN , 20));

this.Payments.setFocusPainted(false);

this.Payments.setBorder(BorderFactory.createLineBorder(Color.BLACK));

this.Payments.addActionListener(this);

this.Payments.setActionCommand("Payments");

this.p2.add(this.Payments);

// About button

this.about = new JButton("About");

this.about.setBounds(0 , 575 , 300 , 51);

this.about.setBackground(new Color(55 , 119 , 131));

this.about.setFont(new Font("SAN SERIK" , Font.PLAIN , 20));

this.about.setFocusPainted(false);

this.about.setBorder(BorderFactory.createLineBorder(Color.BLACK));

this.about.addActionListener(this);

this.about.setActionCommand("About");

this.p2.add(this.about);

// Update personal details button

this.signout = new JButton("Sign out");

this.signout.setBounds(0 , 626 , 300 , 51);

this.signout.setBackground(new Color(55 , 119 , 131));

this.signout.setFont(new Font("SAN SERIK" , Font.PLAIN , 20));

this.signout.setFocusPainted(false);

this.signout.setBorder(BorderFactory.createLineBorder(Color.BLACK));

this.signout.addActionListener(this);

this.signout.setActionCommand("Update Personal Details");

this.p2.add(this.signout);

// Panel p3 configuration

this.p3 = new JLayeredPane();

this.p3.setLayout(null);

this.p3.setBounds(300 , 65 , 1000 , 800);

this.p3.setBackground(Color.PINK);

// Background image in panel p3

ImageIcon i1 = new ImageIcon(getClass().getResource("/Images/image50.jpg"));

Image i2 = i1.getImage().getScaledInstance(1000, 700, Image.SCALE\_DEFAULT);

ImageIcon i3 = new ImageIcon(i2);

JLabel image = new JLabel(i3);

image.setBounds(0, 0, 1000, 700);

p3.add(image);

// Title label in panel p3

titleH = new JLabel("TravelPerk");

titleH.setFont(new Font("SAN SERIF", Font.BOLD, 50));

titleH.setForeground(new Color(33 , 84 , 99));

titleH.setBounds(330 , 20 , 300 , 50);

p3.add(titleH , JLayeredPane.PALETTE\_LAYER);

this.add(p3);

this.setVisible(true);

}

/\*\*

\* Loads customer details from a file and sets them to corresponding fields.

\* Handles file reading errors.

\*/

public void loadCustomerDetailsFromFile() {

try (BufferedReader reader = new BufferedReader(new FileReader("customerDetails.txt"))) {

String line;

while ((line = reader.readLine()) != null) {

if (line.startsWith("Name: ")) {

name = line.substring(6);

} else if (line.startsWith("ID: ")) {

id = line.substring(4);

} else if (line.startsWith("Number: ")) {

number = line.substring(8);

} else if (line.startsWith("Gender: ")) {

gender = line.substring(8);

} else if (line.startsWith("Country: ")) {

country = line.substring(9);

} else if (line.startsWith("Address: ")) {

address = line.substring(9);

} else if (line.startsWith("Email: ")) {

email = line.substring(7);

}

}

} catch (IOException e) {

e.printStackTrace();

}

}

/\*\*

\* Handles button click events based on action commands.

\* Performs actions like adding, updating, and viewing customer details.

\* Also handles booking packages, trips, and viewing destinations.

\*/

@Override

public void actionPerformed(ActionEvent e){

String command = e.getActionCommand();

// Close all other frames except the Dashboard

for (Frame frame : Frame.getFrames()) {

if (frame != this) {

frame.dispose();

}

}

// Then, based on the command, show the corresponding frame or window

if(command.equals("Add Personal Details")){

this.addCustomer = new AddCustomerDetail();

}

else if(command.equals("View Personal Details")){

try {

this.add = addCustomer.getFlag();

} catch(Exception ex) {

// Handle exception

}

if(this.add){

loadCustomerDetailsFromFile();

new ViewCustomerDetails(name, id, number, gender, country, address, email);

} else {

JOptionPane.showMessageDialog(this, "You haven't added information yet.", "Error", JOptionPane.ERROR\_MESSAGE);

}

}

else if(command.equals("Check Packages")){

new CheckPackages();

}

else if(command.equals("Book Package")){

try {

this.add = addCustomer.getFlag();

} catch(Exception ex) {

// Handle exception

}

if(this.add){

loadCustomerDetailsFromFile();

this.bookpackage = new BookPackage(name, id, number);

} else {

JOptionPane.showMessageDialog(this, "You need to add personal information to book package", "Error", JOptionPane.ERROR\_MESSAGE);

}

}

else if(command.equals("View Booked Package")){

try {

this.check = bookpackage.getFlag();

} catch(Exception ex) {

// Handle exception

}

if(this.check){

new ViewPackageDetails();

} else {

JOptionPane.showMessageDialog(this, "You haven't booked a package yet!", "Error", JOptionPane.ERROR\_MESSAGE);

}

}

else if(command.equals("View Hotels")){

new CheckHotels();

}

else if(command.equals("View Destinations")){

new ViewDestinations();

}

else if(command.equals("Book Trip")){

try {

this.add = addCustomer.getFlag();

} catch(Exception ex) {

// Handle exception

}

if(this.add){

this.bookDestination = new BookDestination();

} else {

JOptionPane.showMessageDialog(this, "You need to add personal information to book package", "Error", JOptionPane.ERROR\_MESSAGE);

}

}

else if(command.equals("View Trip Details")){

try {

this.viewDestination = bookDestination.getFlag();

} catch(Exception ex) {

// Handle exception

}

if (this.viewDestination){

loadCustomerDetailsFromFile();

new ViewTripDetails(name);

} else {

JOptionPane.showMessageDialog(this, "You haven't booked a trip yet.", "Error", JOptionPane.ERROR\_MESSAGE);

}

}

else if(command.equals("Payments")){

new Payment();

}

else if(command.equals("About")){

new About();

}

else{

this.setVisible(false);

new Login();

}

}

}

/\*

\* Click nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-default.txt to change this license

\* Click nbfs://nbhost/SystemFileSystem/Templates/Classes/Class.java to edit this template

\*/

package travelmanagementsystem;

import java.awt.Choice;

import java.awt.Color;

import java.awt.Font;

import java.awt.Image;

import java.awt.event.ActionEvent;

import java.awt.event.ActionListener;

import java.io.BufferedReader;

import java.io.BufferedWriter;

import java.io.File;

import java.io.FileNotFoundException;

import java.io.FileReader;

import java.io.FileWriter;

import java.io.IOException;

import java.util.ArrayList;

import javax.swing.BorderFactory;

import javax.swing.ImageIcon;

import javax.swing.JButton;

import javax.swing.JFrame;

import javax.swing.JLabel;

import javax.swing.JOptionPane;

import javax.swing.JPanel;

import javax.swing.JPasswordField;

import javax.swing.JTextField;

import javax.swing.border.LineBorder;

/\*\*

\* This class represents a JFrame for handling password recovery.

\* It allows users to search for their username, answer a security question,

\* and set a new password if the answer is correct.

\* Implements ActionListener for button actions.

\*

\* The password update mechanism reads from and writes to a file 'users.txt'.

\*

\* @author Memon enterprise

\*/

public class ForgotPassword extends JFrame implements ActionListener {

private JTextField Username, Name, Answer;

private JPasswordField Password;

private Choice security;

private JButton set, search, back;

private JPanel p1, p2;

private JLabel title, lblname, lblusername, lblsecurity, ans, lblpassword;

/\*\*

\* Constructor initializes the JFrame layout and components for password recovery.

\* Sets initial UI elements and button actions.

\*/

public ForgotPassword() {

this.setSize(900, 400);

this.setLocation(250, 150);

this.setLayout(null);

this.setResizable(false);

this.setTitle("TravelPerk");

this.getRootPane().setBorder(BorderFactory.createLineBorder(new Color(59, 89, 152), 10));

p1 = new JPanel();

p1.setBounds(0, 0, 500, 400);

p1.setBackground(new Color(59, 89, 152));

p1.setLayout(null);

p2 = new JPanel();

p2.setBounds(500, 0, 400, 400);

p2.setBackground(Color.WHITE);

this.add(p2);

ImageIcon i1 = new ImageIcon(getClass().getResource("/Images/image35.jpg"));

Image i2 = i1.getImage().getScaledInstance(350, 400, Image.SCALE\_SMOOTH);

ImageIcon i3 = new ImageIcon(i2);

JLabel image = new JLabel(i3);

image.setBounds(40, 15, 300, 330);

p2.add(image);

title = new JLabel("Forgot Password?");

title.setBounds(110, -10, 300, 100);

title.setFont(new Font("CALIBRI", Font.BOLD, 30));

title.setForeground(Color.WHITE);

p1.add(title);

lblusername = new JLabel("Username");

lblusername.setBounds(20, 70, 100, 25);

lblusername.setFont(new Font("SAN SERIF", Font.PLAIN, 15));

lblusername.setForeground(Color.WHITE);

p1.add(lblusername);

Username = new JTextField();

Username.setBounds(130, 70, 200, 25);

Username.setBackground(Color.WHITE);

Username.setBorder(BorderFactory.createEmptyBorder());

p1.add(Username);

search = new JButton("Search");

search.setBounds(350, 65, 130, 35);

search.setBackground(new Color(66, 103, 178));

search.setBorder(new LineBorder(Color.WHITE));

search.setForeground(Color.WHITE);

search.setFocusPainted(false);

search.addActionListener(this);

search.setActionCommand("Search");

p1.add(search);

lblname = new JLabel("Name");

lblname.setBounds(20, 110, 100, 25);

lblname.setFont(new Font("SAN SERIF", Font.PLAIN, 15));

lblname.setForeground(Color.WHITE);

p1.add(lblname);

Name = new JTextField();

Name.setBounds(130, 110, 200, 25);

Name.setBorder(BorderFactory.createEmptyBorder());

Name.setBackground(Color.WHITE);

Name.setEditable(false);

p1.add(Name);

lblsecurity = new JLabel("Security Qs");

lblsecurity.setBounds(20, 150, 100, 25);

lblsecurity.setFont(new Font("SAN SERIF", Font.PLAIN, 15));

lblsecurity.setForeground(Color.WHITE);

p1.add(lblsecurity);

security = new Choice();

security.add("Favorite");

security.add("Super Hero");

security.add("Animal");

security.add("Car");

security.add("Cartoon");

security.setBounds(130, 150, 200, 25);

security.setBackground(Color.WHITE);

security.setForeground(new Color(95, 95, 192));

security.setEnabled(false);

p1.add(security);

ans = new JLabel("Answer");

ans.setBounds(20, 190, 100, 25);

ans.setFont(new Font("SAN SERIF", Font.PLAIN, 15));

ans.setForeground(Color.WHITE);

p1.add(ans);

Answer = new JTextField();

Answer.setBounds(130, 190, 200, 25);

Answer.setBorder(BorderFactory.createEmptyBorder());

Answer.setBackground(Color.WHITE);

Answer.setEditable(false);

p1.add(Answer);

lblpassword = new JLabel("Password");

lblpassword.setBounds(20, 230, 100, 25);

lblpassword.setFont(new Font("SAN SERIF", Font.PLAIN, 15));

lblpassword.setForeground(Color.WHITE);

p1.add(lblpassword);

Password = new JPasswordField();

Password.setBounds(130, 230, 200, 25);

Password.setBorder(BorderFactory.createEmptyBorder());

Password.setBackground(Color.WHITE);

Password.setEnabled(false);

p1.add(Password);

set = new JButton("Set Password");

set.setBounds(80, 270, 130, 35);

set.setBackground(new Color(66, 103, 178));

set.setBorder(new LineBorder(Color.WHITE));

set.setForeground(Color.WHITE);

set.setFocusPainted(false);

set.addActionListener(this);

set.setActionCommand("Set Password");

set.setEnabled(false);

p1.add(set);

back = new JButton("Back");

back.setBounds(250, 270, 130, 35);

back.setBackground(new Color(66, 103, 178));

back.setBorder(new LineBorder(Color.WHITE));

back.setForeground(Color.WHITE);

back.setFocusPainted(false);

back.addActionListener(this);

back.setActionCommand("Back");

p1.add(back);

this.add(p1);

this.setVisible(true);

}

/\*\*

\* ActionListener implementation to handle button actions for searching users,

\* setting passwords, and navigating back.

\* @param e ActionEvent triggered by user interaction with buttons.

\*/

@Override

public void actionPerformed(ActionEvent e) {

String command = e.getActionCommand();

if (command.equals("Search")) {

String username = Username.getText().trim();

String[] userDetails = getUserDetails(username);

if (userDetails != null) {

Name.setText(userDetails[1]);

Answer.setText("");

Password.setText("");

Answer.setEditable(true);

Password.setEnabled(true);

security.setEnabled(true);

set.setEnabled(true);

} else {

Name.setText("");

security.select("Favorite");

Answer.setText("");

Password.setText("");

Answer.setEditable(false);

Password.setEnabled(false);

security.setEnabled(false);

set.setEnabled(false);

JOptionPane.showMessageDialog(null, "Username not found!");

}

} else if (command.equals("Set Password")) {

String username = Username.getText().trim();

String selectedSecurityQuestion = security.getSelectedItem();

String answer = Answer.getText().trim();

String newPassword = new String(Password.getPassword());

if (username.isEmpty() || selectedSecurityQuestion.isEmpty() || answer.isEmpty() || newPassword.isEmpty()) {

JOptionPane.showMessageDialog(null, "Please fill in all fields!");

} else if (validateAnswer(username, selectedSecurityQuestion, answer)) {

try {

if (updatePassword(username, newPassword)) {

JOptionPane.showMessageDialog(null, "Password Updated Successfully!");

clearFields();

this.setVisible(false);

new Login().setVisible(true);

} else {

JOptionPane.showMessageDialog(null, "Error Updating Password");

}

} catch (IOException ex) {

ex.printStackTrace();

JOptionPane.showMessageDialog(null, "Error " + ex.getMessage());

}

} else {

JOptionPane.showMessageDialog(null, "Incorrect Answer or Security Question!");

}

} else if (command.equals(("Back"))) {

this.setVisible(false);

new Login().setVisible(true);

}

}

/\*\*

\* Retrieves user details from a file based on the provided username.

\* @param username Username of the user to search.

\* @return String array of user details if found, null otherwise.

\*/

private String[] getUserDetails(String username) {

try (BufferedReader reader = new BufferedReader(new FileReader("users.txt"))) {

String line;

while ((line = reader.readLine()) != null) {

String[] userDetails = line.split(",");

if (userDetails[0].equals(username)) {

return userDetails;

}

}

} catch (IOException e) {

e.printStackTrace();

}

return null;

}

/\*\*

\* Validates the answer provided by the user against the stored security answer and question.

\* @param username Username of the user for validation.

\* @param securityQuestion Security question selected by the user.

\* @param answer Answer provided by the user.

\* @return True if the answer matches the stored security answer, false otherwise.

\*/

private boolean validateAnswer(String username, String securityQuestion, String answer) {

String[] userDetails = getUserDetails(username);

if (userDetails != null && userDetails.length >= 5) {

return userDetails[4].equalsIgnoreCase(securityQuestion) && userDetails[3].equalsIgnoreCase(answer);

}

return false;

}

/\*\*

\* Updates the password for the specified username in the 'users.txt' file.

\* @param username Username of the user to update password.

\* @param newPassword New password to set for the user.

\* @return True if the password was updated successfully, false otherwise.

\* @throws FileNotFoundException If the 'users.txt' file is not found.

\* @throws IOException If an I/O error occurs while updating the file.

\*/

private boolean updatePassword(String username, String newPassword) throws FileNotFoundException, IOException {

File inputFile = new File("users.txt");

ArrayList<String> lines = new ArrayList<>();

boolean isUpdated = false;

try (BufferedReader reader = new BufferedReader(new FileReader(inputFile))) {

String line;

while ((line = reader.readLine()) != null) {

String[] userDetails = line.split(",");

if (userDetails[0].equals(username)) {

// Update the password

userDetails[2] = newPassword;

isUpdated = true;

}

// Store the updated or unmodified line

lines.add(String.join(",", userDetails));

}

}

if (!isUpdated) {

return false; // No update made

}

// Write the updated contents back to the file

try (BufferedWriter writer = new BufferedWriter(new FileWriter(inputFile))) {

for (String updatedLine : lines) {

writer.write(updatedLine);

writer.newLine();

}

}

return true; // Password updated successfully

}

/\*\*

\* Clears all input fields and resets UI elements to their initial states.

\*/

private void clearFields() {

Username.setText("");

Name.setText("");

Answer.setText("");

Password.setText("");

security.select("Favorite");

Answer.setEditable(false);

Password.setEnabled(false);

security.setEnabled(false);

set.setEnabled(false);

}

}

/\*

\* Click nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-default.txt to change this license

\* Click nbfs://nbhost/SystemFileSystem/Templates/Classes/Class.java to edit this template

\*/

package travelmanagementsystem;

import java.awt.Color;

import java.awt.Font;

import javax.swing.ImageIcon;

import javax.swing.JFrame;

import javax.swing.JLabel;

import javax.swing.JProgressBar;

/\*\*

\* The Loading class displays a loading screen with a progress bar and text,

\* simulating the loading process of a travel and tourism management application.

\* It extends JFrame and implements Runnable to run the loading process in a separate thread.

\* When the loading is complete, it hides itself and shows the Dashboard.

\*/

public class Loading extends JFrame implements Runnable {

private Thread t;

private JProgressBar bar;

private JLabel title, subTitle, text;

/\*\*

\* Constructs a Loading object which initializes the JFrame and sets up the loading screen components.

\*/

public Loading() {

// Create the label with the GIF

ImageIcon icon = new ImageIcon("C:\\Users\\Memon enterprise\\Downloads\\trvl.gif");

JLabel bg = new JLabel(icon);

bg.setSize(650, 450);

// Add the label to the content pane as background

this.setContentPane(bg);

t = new Thread(this);

this.setSize(650 , 450);

this.setLocation(300 , 120);

this.getContentPane().setBackground(new Color(200 , 162 , 200));

this.setTitle("TravelPerk");

this.setLayout(null);

this.setResizable(false);

//shows the loading progress bar

bar = new JProgressBar();

bar.setBounds(0 , 393 , 650 , 20);

bar.setStringPainted(true);

bar.setForeground(Color.ORANGE);

bg.add(bar);

//loading label

text = new JLabel("Loading...");

text.setBounds(270 , 360, 200 , 30);

text.setFont(new Font("SAN SERIF" , Font.BOLD , 22));

text.setForeground(Color.BLACK);

bg.add(text);

t.start();

this.setVisible(true);

}

/\*\*

\* Implements the run method from Runnable interface to simulate the loading process.

\* When the loading is complete (progress reaches 100%), it hides the loading screen and shows the Dashboard.

\*/

@Override

public void run() {

try {

for(int i = 0; i <= 100 ; i++) {

int max = bar.getMaximum();

int value = bar.getValue();

if(value < max) {

bar.setValue(bar.getValue() + 1);

} else {

this.setVisible(false);

new Dashboard().setVisible(true);

}

Thread.sleep(50);

}

} catch(Exception e) {

e.printStackTrace();

}

}

}

/\*

\* Click nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-default.txt to change this license

\* Click nbfs://nbhost/SystemFileSystem/Templates/Classes/Class.java to edit this template

\*/

package travelmanagementsystem;

import java.awt.Color;

import java.awt.Font;

import java.awt.Image;

import java.awt.event.ActionEvent;

import java.awt.event.ActionListener;

import java.io.BufferedReader;

import java.io.FileReader;

import java.io.IOException;

import javax.swing.BorderFactory;

import javax.swing.ImageIcon;

import javax.swing.JButton;

import javax.swing.JFrame;

import javax.swing.JLabel;

import javax.swing.JOptionPane;

import javax.swing.JPanel;

import javax.swing.JPasswordField;

import javax.swing.JTextField;

import javax.swing.border.LineBorder;

/\*\*

\* The Login class provides a user interface for user authentication.

\* It extends JFrame and implements ActionListener to handle user interactions.

\* Users can log in, sign up, or recover their password.

\*

\* @author Memon enterprise

\*/

public class Login extends JFrame implements ActionListener{

private JPasswordField Password;

private JTextField Username;

private JPanel p1 , p2;

private JLabel title , lblusername , lblpassword;

private JButton login , signup , forget;

/\*\*

\* Constructs a Login object which initializes the JFrame and sets up the login interface.

\*/

public Login(){

this.setSize(900 , 400);

this.setLocation(250 , 150);

this.setLayout(null);

this.setResizable(false);

this.setTitle("TravelPerk");

//create border for the frame

this.getRootPane().setBorder(BorderFactory.createLineBorder(new Color(139 , 69 , 19) , 10));

//Panel to add the image

p1 = new JPanel();

p1.setBackground(Color.WHITE);

p1.setBounds(0 , 0 , 400 , 400);

p1.setLayout(null);

//Image added in panel

ImageIcon i1 = new ImageIcon(getClass().getResource("/Images/image31.jpg"));

Image i2 = i1.getImage().getScaledInstance(350, 400, Image.SCALE\_DEFAULT);

ImageIcon i3 = new ImageIcon(i2);

JLabel image = new JLabel(i3);

image.setBounds(0, 0, 400, 400);

p1.add(image);

this.add(p1);

//Panel to create login

p2 = new JPanel();

p2.setLayout(null);

p2.setBounds(400 , 0 , 500 , 400);

p2.setBackground(new Color(177 , 109 , 82));

this.add(p2);

//Login title

title = new JLabel("Login");

title.setBounds(200, 20, 100, 60);

title.setFont(new Font("CALIBRI" , Font.BOLD , 30));

p2.add(title);

//Username label

lblusername = new JLabel("Username");

lblusername.setBounds(80, 70, 100, 25);

lblusername.setFont(new Font("SAN SERIF" , Font.PLAIN , 20));

p2.add(lblusername);

//Textfield to get username

Username = new JTextField();

Username.setBounds(80 , 100 , 300 , 30);

Username.setBorder(BorderFactory.createEmptyBorder());

Username.setBackground(new Color(210 , 180 , 140));

p2.add(Username);

//Password label

lblpassword = new JLabel("Password");

lblpassword.setBounds(80, 145, 100, 25);

lblpassword.setFont(new Font("SAN SERIF" , Font.PLAIN , 20));

p2.add(lblpassword);

//Textfield to get password

Password = new JPasswordField();

Password.setBounds(80 , 175 , 300 , 30);

Password.setBorder(BorderFactory.createEmptyBorder());

Password.setBackground(new Color(210 , 180 , 140));

p2.add(Password);

//Login to your account

login = new JButton("Log in");

login.setBounds(80 , 230 , 130 , 35);

login.setBackground(new Color(30 , 116 , 93));

login.setBorder(new LineBorder(new Color(75 , 75 , 172)));

login.setFocusPainted(false);

login.addActionListener(this);

login.setActionCommand("Login");

p2.add(login);

// Signup button

signup = new JButton("Sign Up");

signup.setBounds(250 , 230 , 130 , 35);

signup.setBackground(new Color(30 , 116 , 93));

signup.setBorder(new LineBorder(new Color(75 , 75 , 172)));

signup.setFocusPainted(false);

signup.addActionListener(this);

signup.setActionCommand("Sign Up");

p2.add(signup);

//Forgot password button

forget = new JButton("Forgot Password?");

forget.setBounds(170 , 280 , 130 , 35);

forget.setBackground(new Color(177 , 109 , 82));

forget.setBorder(BorderFactory.createEmptyBorder());

forget.setFocusPainted(false);

forget.addActionListener(this);

forget.setActionCommand("Forgot Password");

p2.add(forget);

//Set visibility of the frame

this.setVisible(true);

}

/\*\*

\* Handles action events for the login buttons.

\* @param e The ActionEvent triggered by user interaction.

\*/

@Override

public void actionPerformed(ActionEvent e){

String command = e.getActionCommand();

if(command.equals("Login")){

String username = Username.getText();

String password = new String(Password.getPassword());

if(authenticate(username , password)){

this.setVisible(false);

new Loading().setVisible(true);

}else{

JOptionPane.showMessageDialog(null, "Invalid username or password!");

}

}

else if(command.equals("Sign Up")){

this.setVisible(false);

new Signup().setVisible(true);

}

else if(command.equals("Forgot Password")){

this.setVisible(false);

new ForgotPassword().setVisible(true);

}

}

/\*\*

\* Authenticates user credentials against a stored database of users.

\* @param username The username entered by the user.

\* @param password The password entered by the user.

\* @return true if authentication succeeds, false otherwise.

\*/

private boolean authenticate(String username, String password){

try(BufferedReader reader = new BufferedReader(new FileReader("users.txt"))){

String line;

while((line = reader.readLine()) != null){

String[] userDetails = line.split(",");

if(userDetails[0].equals(username) && userDetails[2].equals(password)){

return true;

}

}

}

catch(IOException e){

e.printStackTrace();

}

return false;

}

}

/\*

\* Click nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-default.txt to change this license

\* Click nbfs://nbhost/SystemFileSystem/Templates/Classes/Class.java to edit this template

\*/

package travelmanagementsystem;

import java.awt.Color;

import java.awt.Desktop;

import java.awt.Font;

import java.awt.Image;

import java.awt.event.ActionEvent;

import java.awt.event.ActionListener;

import java.io.IOException;

import java.net.URI;

import java.net.URISyntaxException;

import javax.swing.BorderFactory;

import javax.swing.ImageIcon;

import javax.swing.JButton;

import javax.swing.JFrame;

import javax.swing.JLabel;

import javax.swing.border.LineBorder;

/\*\*

\* This class handles the payment functionality for the travel management system.

\* It extends JFrame and implements ActionListener to handle user interactions.

\*/

public class Payment extends JFrame implements ActionListener {

private JButton pay, back;

/\*\*

\* Constructor to initialize the frame and its components.

\*/

public Payment() {

this.setBounds(305, 90, 970, 630);

this.setLayout(null);

this.setResizable(false);

this.getContentPane().setBackground(Color.WHITE);

this.setTitle("TravelPerk");

// Set the border of the root pane

this.getRootPane().setBorder(BorderFactory.createLineBorder(new Color(0x27b05e), 20));

// Load and scale the image

ImageIcon i1 = new ImageIcon(getClass().getResource("/Images/images.png"));

Image i2 = i1.getImage().getScaledInstance(412, 412, Image.SCALE\_AREA\_AVERAGING);

ImageIcon i3 = new ImageIcon(i2);

JLabel image = new JLabel(i3);

image.setBounds(240, 0, 412, 412);

// Back button

back = new JButton("Back");

back.setBounds(650, 430, 200, 55);

back.setBackground(new Color(0x27b05e));

back.setBorder(new LineBorder(new Color(127, 93, 37)));

back.setForeground(Color.WHITE);

back.setFont(new Font("SAN SERIF", Font.BOLD, 15));

back.setFocusPainted(false);

back.setBorder(BorderFactory.createLineBorder(Color.BLACK));

back.addActionListener(this);

back.setActionCommand("Back");

this.add(back);

// Pay button

pay = new JButton("Pay");

pay.setBounds(80, 430, 200, 55);

pay.setBackground(new Color(0x27b05e));

pay.setBorder(new LineBorder(new Color(127, 93, 37)));

pay.setForeground(Color.WHITE);

pay.setFont(new Font("SAN SERIF", Font.BOLD, 15));

pay.setFocusPainted(false);

pay.addActionListener(e -> {

try {

// Open the Easypaisa website in the default browser

Desktop.getDesktop().browse(new URI("https://easypaisa.com.pk"));

} catch (IOException | URISyntaxException ex) {

ex.printStackTrace();

}

});

this.add(pay);

this.add(image);

this.setVisible(true);

}

/\*\*

\* Placeholder method for making a payment.

\*/

public void makePayment() {

// Add payment logic here

}

/\*\*

\* Action handler for the buttons.

\*/

@Override

public void actionPerformed(ActionEvent e) {

String command = e.getActionCommand();

if (command.equals("Back")) {

this.setVisible(false);

}

}

}

/\*

\* Click nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-default.txt to change this license

\* Click nbfs://nbhost/SystemFileSystem/Templates/Classes/Class.java to edit this template

\*/

package travelmanagementsystem;

import java.awt.Choice;

import java.awt.Color;

import java.awt.Font;

import java.awt.Image;

import java.awt.event.ActionEvent;

import java.awt.event.ActionListener;

import java.io.BufferedReader;

import java.io.BufferedWriter;

import java.io.FileReader;

import java.io.FileWriter;

import java.io.IOException;

import javax.swing.BorderFactory;

import javax.swing.ImageIcon;

import javax.swing.JButton;

import javax.swing.JFrame;

import javax.swing.JLabel;

import javax.swing.JOptionPane;

import javax.swing.JPanel;

import javax.swing.JPasswordField;

import javax.swing.JTextField;

import javax.swing.border.LineBorder;

/\*\*

\* This class handles the sign-up functionality for the travel management system.

\* It extends JFrame and implements ActionListener to handle user interactions.

\*/

public class Signup extends JFrame implements ActionListener {

private JPasswordField Password;

private JTextField Name, Username, Answer;

private Choice security;

private JPanel p2, p1;

private JLabel title, lblusername, lblname, lblpassword, lblsecurity, ans;

private JButton signup, back;

/\*\*

\* Constructor to initialize the frame and its components.

\*/

public Signup() {

this.setSize(900, 400);

this.setLocation(250, 150);

this.setLayout(null);

this.setResizable(false);

this.setTitle("TravelPerk");

this.getRootPane().setBorder(BorderFactory.createLineBorder(new Color(139 , 69 , 19), 10));

// Left panel for form input

p2 = new JPanel();

p2.setLayout(null);

p2.setBounds(0, 0, 500, 400);

p2.setBackground(new Color(177 , 109 , 82));

this.add(p2);

// Right panel for the image

p1 = new JPanel();

p1.setBackground(Color.WHITE);

p1.setBounds(500, 0, 400, 400);

p1.setLayout(null);

// Load and scale the image

ImageIcon i1 = new ImageIcon(getClass().getResource("/Images/image31.jpg"));

Image i2 = i1.getImage().getScaledInstance(350, 400, Image.SCALE\_SMOOTH);

ImageIcon i3 = new ImageIcon(i2);

JLabel image = new JLabel(i3);

image.setBounds(40, 10, 300, 330);

p1.add(image);

this.add(p1);

// Title label

title = new JLabel("Sign Up");

title.setBounds(200, 20, 100, 60);

title.setFont(new Font("CALIBRI", Font.BOLD, 30));

title.setForeground(Color.BLACK);

p2.add(title);

// Username label and text field

lblusername = new JLabel("Username");

lblusername.setBounds(80, 70, 100, 25);

lblusername.setFont(new Font("SAN SERIF", Font.PLAIN, 15));

lblusername.setForeground(Color.BLACK);

p2.add(lblusername);

Username = new JTextField();

Username.setBounds(190, 70, 200, 25);

Username.setBorder(BorderFactory.createEmptyBorder());

Username.setBackground(new Color(210 , 180 , 140));

p2.add(Username);

// Name label and text field

lblname = new JLabel("Name");

lblname.setBounds(80, 110, 200, 25);

lblname.setFont(new Font("SAN SERIF", Font.PLAIN, 15));

lblname.setForeground(Color.BLACK);

p2.add(lblname);

Name = new JTextField();

Name.setBounds(190, 110, 200, 25);

Name.setBorder(BorderFactory.createEmptyBorder());

Name.setBackground(new Color(210 , 180 , 140));

p2.add(Name);

// Password label and password field

lblpassword = new JLabel("Password");

lblpassword.setBounds(80, 150, 100, 25);

lblpassword.setFont(new Font("SAN SERIF", Font.PLAIN, 15));

lblpassword.setForeground(Color.BLACK);

p2.add(lblpassword);

Password = new JPasswordField();

Password.setBounds(190, 150, 200, 25);

Password.setBorder(BorderFactory.createEmptyBorder());

Password.setBackground(new Color(210 , 180 , 140));

p2.add(Password);

// Security question label and choice

lblsecurity = new JLabel("Security Qs");

lblsecurity.setBounds(80, 190, 100, 25);

lblsecurity.setFont(new Font("SAN SERIF", Font.PLAIN, 15));

lblsecurity.setForeground(Color.BLACK);

p2.add(lblsecurity);

security = new Choice();

security.add("Favorite");

security.add("Super Hero");

security.add("Animal");

security.add("Car");

security.add("Cartoon");

security.setBounds(190, 190, 200, 25);

security.setBackground(new Color(210 , 180 , 140));

security.setForeground(Color.BLACK);

p2.add(security);

// Answer label and text field

ans = new JLabel("Answer");

ans.setBounds(80, 230, 100, 25);

ans.setFont(new Font("SAN SERIF", Font.PLAIN, 15));

ans.setForeground(Color.BLACK);

p2.add(ans);

Answer = new JTextField();

Answer.setBounds(190, 230, 200, 25);

Answer.setBorder(BorderFactory.createEmptyBorder());

Answer.setBackground(new Color(210 , 180 , 140));

p2.add(Answer);

// Signup button

signup = new JButton("Sign up");

signup.setBounds(80, 270, 130, 35);

signup.setBackground(new Color(30 , 116 , 93));

signup.setBorder(new LineBorder(new Color(75 , 75 , 172)));

signup.setForeground(Color.BLACK);

signup.setFocusPainted(false);

signup.addActionListener(this);

signup.setActionCommand("Signup");

p2.add(signup);

// Back button

back = new JButton("Back");

back.setBounds(250, 270, 130, 35);

back.setBackground(new Color(30 , 116 , 93));

back.setBorder(new LineBorder(new Color(75 , 75 , 172)));

back.setForeground(Color.BLACK);

back.setFocusPainted(false);

back.addActionListener(this);

back.setActionCommand("Back");

p2.add(back);

this.setVisible(true);

}

/\*\*

\* Action handler for the buttons.

\*/

@Override

public void actionPerformed(ActionEvent e) {

String command = e.getActionCommand();

if (command.equals("Signup")) {

String username = Username.getText();

String name = Name.getText();

String answer = Answer.getText();

String password = new String(Password.getPassword());

String securityQuestion = security.getSelectedItem();

// Store user data if the username is not taken

if (storeUserData(username, name, password, answer, securityQuestion)) {

JOptionPane.showMessageDialog(null, "USER REGISTERED SUCCESSFULLY!");

this.setVisible(false);

new Login().setVisible(true);

} else {

JOptionPane.showMessageDialog(null, "Username already exists! Please choose a different username.");

}

} else if (command.equals("Back")) {

this.setVisible(false);

new Login().setVisible(true);

}

}

/\*\*

\* Checks if the given username is already taken.

\*

\* @param username The username to check.

\* @return true if the username is taken, false otherwise.

\*/

private boolean isUsernameTaken(String username) {

try (BufferedReader reader = new BufferedReader(new FileReader("users.txt"))) {

String line;

while ((line = reader.readLine()) != null) {

String[] userDetails = line.split(",");

if (userDetails[0].equals(username)) {

return true;

}

}

} catch (IOException e) {

e.printStackTrace();

}

return false;

}

/\*\*

\* Stores the user data in a file.

\*

\* @param username The username.

\* @param name The name.

\* @param password The password.

\* @param answer The security question answer.

\* @param securityQuestion The security question.

\* @return true if the data is stored successfully, false otherwise.

\*/

private boolean storeUserData(String username, String name, String password, String answer, String securityQuestion) {

if (isUsernameTaken(username)) {

return false;

}

try (BufferedWriter writer = new BufferedWriter(new FileWriter("users.txt", true))) {

writer.write(username + "," + name + "," + password + "," + answer + "," + securityQuestion);

writer.newLine();

writer.flush();

return true;

} catch (IOException e) {

e.printStackTrace();

return false;

}

}

}

/\*

\* Click nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-default.txt to change this license

\* Click nbfs://nbhost/SystemFileSystem/Templates/Classes/Class.java to edit this template

\*/

package travelmanagementsystem;

import java.awt.Color;

import java.awt.Font;

import java.awt.Image;

import javax.swing.ImageIcon;

import javax.swing.JFrame;

import javax.swing.JLabel;

import javax.swing.JLayeredPane;

import javax.swing.SwingConstants;

/\*\*

\* This class displays a splash screen for the travel management system.

\* It extends JFrame and implements Runnable to handle the splash screen thread.

\*/

public class Splash extends JFrame implements Runnable {

Thread thread;

JLayeredPane layeredPane;

ImageIcon i1, i3;

Image i2;

JLabel image;

JLabel text;

/\*\*

\* Constructor to initialize the frame and its components.

\*/

public Splash() {

this.setResizable(false);

this.setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

this.setTitle("TravelPerk");

layeredPane = new JLayeredPane();

layeredPane.setSize(1200, 600);

// Load and scale the splash image

i1 = new ImageIcon(getClass().getResource("/Images/image27.jpg"));

i2 = i1.getImage().getScaledInstance(1200, 600, Image.SCALE\_SMOOTH);

i3 = new ImageIcon(i2);

image = new JLabel(i3);

image.setBounds(0, 0, 1200, 600);

// Add splash screen text

text = new JLabel("TravelPerk", SwingConstants.CENTER);

text.setFont(new Font("Serif", Font.BOLD, 70));

text.setForeground(new Color(255, 10, 0));

text.setBounds(-70, -70, 1200, 600);

// Add components to layered pane

layeredPane.add(image, JLayeredPane.DEFAULT\_LAYER);

layeredPane.add(text, JLayeredPane.PALETTE\_LAYER);

this.add(layeredPane);

this.setVisible(true); // Make the frame visible

thread = new Thread(this);

thread.start(); // Start the thread for splash screen

}

/\*\*

\* Run method to handle the splash screen delay and transition.

\*/

public void run() {

try {

Thread.sleep(3000); // Show splash screen for 3 seconds

new Login(); // Transition to the login screen

setVisible(false); // Hide the splash screen

} catch (Exception e) {

e.printStackTrace();

}

}

}

/\*

\* Click nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-default.txt to change this license

\* Click nbfs://nbhost/SystemFileSystem/Templates/Classes/Class.java to edit this template

\*/

package travelmanagementsystem;

import java.awt.Color;

import java.awt.Font;

import java.awt.Image;

import java.awt.event.ActionEvent;

import java.awt.event.ActionListener;

import javax.swing.BorderFactory;

import javax.swing.ImageIcon;

import javax.swing.JButton;

import javax.swing.JFrame;

import javax.swing.JLabel;

import javax.swing.border.LineBorder;

/\*\*

\* This class displays detailed personal information of a customer.

\* It extends JFrame and implements ActionListener to handle button actions.

\*

\* @autor Memon enterprise

\*/

public class ViewCustomerDetails extends JFrame implements ActionListener {

private JLabel title, lblname, personName, lblid, personID, lblnumber, personNumber, lblgender, personGender, lblcountry, personCountry, lbladdress, personAddress, lblemail, personEmail;

private JButton back;

/\*\*

\* Constructor to initialize the frame and its components.

\* @param name Customer's name

\* @param id Customer's ID

\* @param number Customer's contact number

\* @param gender Customer's gender

\* @param country Customer's country

\* @param address Customer's address

\* @param email Customer's email

\*/

public ViewCustomerDetails(String name, String id, String number, String gender, String country, String address, String email) {

this.setBounds(305, 90, 970, 630);

this.setLayout(null);

this.setResizable(false);

this.setTitle("TravelPerk");

this.getRootPane().setBorder(BorderFactory.createLineBorder(Color.BLACK, 10));

ImageIcon i1 = new ImageIcon(getClass().getResource("/Images/image64.jpg"));

Image i2 = i1.getImage().getScaledInstance(970, 630, Image.SCALE\_SMOOTH);

ImageIcon i3 = new ImageIcon(i2);

JLabel image = new JLabel(i3);

image.setBounds(0, 0, 970, 630);

// Title label

title = new JLabel("Personal Details");

title.setBounds(330, 30, 300, 60);

title.setFont(new Font("SAN SERIF", Font.BOLD, 35));

title.setForeground(Color.BLACK);

this.add(title);

// Name label

lblname = new JLabel("Name");

lblname.setBounds(115, 120, 100, 20);

lblname.setFont(new Font("SAN SERIF", Font.BOLD, 20));

lblname.setForeground(Color.BLACK);

this.add(lblname);

personName = new JLabel(name);

personName.setBounds(245, 120, 400, 20);

personName.setFont(new Font("SAN SERIF", Font.BOLD, 15));

personName.setForeground(Color.BLACK);

this.add(personName);

// ID label

lblid = new JLabel("ID");

lblid.setBounds(115, 180, 100, 20);

lblid.setFont(new Font("SAN SERIF", Font.BOLD, 20));

lblid.setForeground(Color.BLACK);

this.add(lblid);

personID = new JLabel(id);

personID.setBounds(245, 180, 400, 20);

personID.setFont(new Font("SAN SERIF", Font.BOLD, 15));

personID.setForeground(Color.BLACK);

this.add(personID);

// Number label

lblnumber = new JLabel("Number");

lblnumber.setBounds(115, 240, 100, 20);

lblnumber.setFont(new Font("SAN SERIF", Font.BOLD, 20));

lblnumber.setForeground(Color.BLACK);

this.add(lblnumber);

personNumber = new JLabel(number);

personNumber.setBounds(245, 240, 400, 20);

personNumber.setFont(new Font("SAN SERIF", Font.BOLD, 15));

personNumber.setForeground(Color.BLACK);

this.add(personNumber);

// Gender label

lblgender = new JLabel("Gender");

lblgender.setBounds(505, 120, 100, 20);

lblgender.setFont(new Font("SAN SERIF", Font.BOLD, 20));

lblgender.setForeground(Color.BLACK);

this.add(lblgender);

personGender = new JLabel(gender);

personGender.setBounds(635, 120, 400, 20);

personGender.setFont(new Font("SAN SERIF", Font.BOLD, 15));

personGender.setForeground(Color.BLACK);

this.add(personGender);

// Country label

lblcountry = new JLabel("Country");

lblcountry.setBounds(505, 180, 100, 20);

lblcountry.setFont(new Font("SAN SERIF", Font.BOLD, 20));

lblcountry.setForeground(Color.BLACK);

this.add(lblcountry);

personCountry = new JLabel(country);

personCountry.setBounds(635, 180, 400, 20);

personCountry.setFont(new Font("SAN SERIF", Font.BOLD, 15));

personCountry.setForeground(Color.BLACK);

this.add(personCountry);

// Address label

lbladdress = new JLabel("Address");

lbladdress.setBounds(505, 240, 100, 20);

lbladdress.setFont(new Font("SAN SERIF", Font.BOLD, 20));

lbladdress.setForeground(Color.BLACK);

this.add(lbladdress);

personAddress = new JLabel(address);

personAddress.setBounds(635, 240, 400, 20);

personAddress.setFont(new Font("SAN SERIF", Font.BOLD, 15));

personAddress.setForeground(Color.BLACK);

this.add(personAddress);

// Email label

lblemail = new JLabel("Email");

lblemail.setBounds(320, 300, 100, 20);

lblemail.setFont(new Font("SAN SERIF", Font.BOLD, 20));

lblemail.setForeground(Color.BLACK);

this.add(lblemail);

personEmail = new JLabel(email);

personEmail.setBounds(430, 300, 400, 20);

personEmail.setFont(new Font("SAN SERIF", Font.BOLD, 15));

personEmail.setForeground(Color.BLACK);

this.add(personEmail);

// Back button

back = new JButton("Back");

back.setBounds(720, 500, 200, 55);

back.setBackground(new Color(176, 159, 124));

back.setBorder(new LineBorder(new Color(127, 93, 37)));

back.setForeground(Color.BLACK);

back.setFont(new Font("SAN SERIF", Font.BOLD, 15));

back.setFocusPainted(false);

back.setBorder(BorderFactory.createLineBorder(Color.BLACK));

back.addActionListener(this);

back.setActionCommand("Back");

this.add(back);

this.add(image);

this.setVisible(true); // Make the frame visible

}

// Overriding the actionPerformed method to handle button actions

@Override

public void actionPerformed(ActionEvent e) {

String command = e.getActionCommand();

if (command.equals("Back")) {

this.setVisible(false); // Hide the frame when the back button is clicked

}

}

}

/\*

\* Click nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-default.txt to change this license

\* Click nbfs://nbhost/SystemFileSystem/Templates/Classes/Class.java to edit this template

\*/

package travelmanagementsystem;

import java.awt.Color;

import java.awt.Font;

import java.awt.Image;

import javax.swing.ImageIcon;

import javax.swing.JButton;

import javax.swing.JFrame;

import javax.swing.JLabel;

/\*\*

\* This class displays a slideshow of travel destinations.

\* It extends JFrame and implements Runnable to handle the slideshow functionality.

\*/

public class ViewDestinations extends JFrame implements Runnable {

private Thread t; // Thread to handle the slideshow

private JLabel[] label = new JLabel[5]; // Array to hold JLabel components for images

private JLabel captions; // JLabel to display the captions

private ImageIcon i1 = null, i2 = null, i3 = null, i4 = null, i5 = null;

private Image j1 = null, j2 = null, j3 = null, j4 = null, j5 = null;

private ImageIcon k1 = null, k2 = null, k3 = null, k4 = null, k5 = null;

private JButton back;

// Constructor to initialize the frame and its components

public ViewDestinations() {

this.setBounds(305, 90, 970, 630);

this.setLayout(null);

this.setResizable(false);

this.setTitle("TravelPerk");

// Initialize the caption label

captions = new JLabel();

captions.setBounds(30, 500, 600, 60);

captions.setFont(new Font("Tahoma", Font.BOLD, 40));

captions.setForeground(Color.WHITE);

this.add(captions);

// Arrays to hold the image icons and scaled images

ImageIcon[] image = new ImageIcon[5];

Image[] jimage = new Image[5];

ImageIcon[] kimage = new ImageIcon[5];

// Loop to load and set the images

for (int i = 0; i < 5; i++) {

image[i] = new ImageIcon(getClass().getResource("/Images/image" + (i + 94) + ".jpg"));

jimage[i] = image[i].getImage().getScaledInstance(970, 630, Image.SCALE\_SMOOTH);

kimage[i] = new ImageIcon(jimage[i]);

label[i] = new JLabel(kimage[i]);

label[i].setBounds(0, 0, 970, 630);

label[i].setVisible(false);

this.add(label[i]);

}

label[0].setVisible(true); // Set the first image to be visible initially

t = new Thread(this); // Initialize and start the thread

t.start();

this.setVisible(true); // Make the frame visible

}

// Overriding the run method to handle the slideshow

@Override

public void run() {

String[] destinationsNames = new String[]{"London", "Turkey", "Switzerland", "Malaysia", "Los Angeles"};

try {

int i = 0;

while (true) {

label[i].setVisible(true); // Show the current image

captions.setText(destinationsNames[i]); // Set the caption for the current image

Thread.sleep(2500); // Pause for 2.5 seconds

label[i].setVisible(false); // Hide the current image

i++;

if (i == 5) { // Reset the index if it reaches the end of the array

i = 0;

}

}

} catch (Exception e) {

e.printStackTrace(); // Print stack trace in case of an exception

}

}

}

/\*

\* Click nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-default.txt to change this license

\* Click nbfs://nbhost/SystemFileSystem/Templates/Classes/Class.java to edit this template

\*/

package travelmanagementsystem;

import java.awt.Color;

import java.awt.Font;

import java.awt.Image;

import java.awt.event.ActionEvent;

import java.awt.event.ActionListener;

import java.io.BufferedReader;

import java.io.FileReader;

import java.io.IOException;

import javax.swing.BorderFactory;

import javax.swing.ImageIcon;

import javax.swing.JButton;

import javax.swing.JFrame;

import javax.swing.JLabel;

/\*\*

\* This class displays the details of a booked package in a travel management system.

\* It extends JFrame and implements ActionListener to handle button actions.

\* The details are read from a file and displayed in a user-friendly format.

\*

\* @author Memon enterprise

\*/

public class ViewPackageDetails extends JFrame implements ActionListener{

// Declare GUI components

private JLabel Name, pack, ID, nmbr, persons, total, title, lblname, lblpackage, lblid, lblnumber, lblpersons, lbltotal;

private JButton back;

// Constructor to initialize the frame and its components

public ViewPackageDetails(){

this.setBounds(305, 90, 970, 630);

this.setLayout(null);

this.setResizable(false);

this.setTitle("TravelPerk");

this.getRootPane().setBorder(BorderFactory.createLineBorder(new Color(0x21252e), 10));

// Load and set background image

ImageIcon i1 = new ImageIcon(getClass().getResource("/Images/image17.jpeg"));

Image i2 = i1.getImage().getScaledInstance(970, 630, Image.SCALE\_SMOOTH);

ImageIcon i3 = new ImageIcon(i2);

JLabel image = new JLabel(i3);

image.setBounds(0, 0, 970, 630);

// Set the title label

title = new JLabel("Booked Package Details");

title.setBounds(260, 40, 600, 80);

title.setFont(new Font("SAN SERIF", Font.TYPE1\_FONT, 40));

title.setForeground(Color.WHITE);

this.add(title);

// Set the name label

lblname = new JLabel("Name");

lblname.setBounds(50, 200, 200, 30);

lblname.setFont(new Font("SAN SERIF", Font.BOLD, 25));

lblname.setForeground(Color.WHITE);

this.add(lblname);

// Set the label to display the name

Name = new JLabel();

Name.setBounds(300, 200, 200, 30);

Name.setFont(new Font("SAN SERIF", Font.BOLD, 20));

Name.setForeground(Color.white);

this.add(Name);

// Set the selected package label

lblpackage = new JLabel("Selected Package");

lblpackage.setBounds(50, 290, 300, 30);

lblpackage.setFont(new Font("SAN SERIF", Font.BOLD, 25));

lblpackage.setForeground(Color.white);

this.add(lblpackage);

// Set the label to display the package

pack = new JLabel();

pack.setBounds(300, 290, 200, 30);

pack.setFont(new Font("SAN SERIF", Font.BOLD, 20));

pack.setForeground(Color.white);

this.add(pack);

// Set the persons label

lblpersons = new JLabel("Total Persons");

lblpersons.setBounds(50, 390, 200, 30);

lblpersons.setFont(new Font("SAN SERIF", Font.BOLD, 25));

lblpersons.setForeground(Color.WHITE);

this.add(lblpersons);

// Set the label to display the number of persons

persons = new JLabel();

persons.setBounds(300, 390, 200, 30);

persons.setFont(new Font("SAN SERIF", Font.BOLD, 20));

persons.setForeground(Color.WHITE);

this.add(persons);

// Set the ID label

lblid = new JLabel("ID");

lblid.setBounds(550, 200, 200, 30);

lblid.setFont(new Font("SAN SERIF", Font.BOLD, 25));

lblid.setForeground(Color.WHITE);

this.add(lblid);

// Set the label to display the ID

ID = new JLabel();

ID.setBounds(720, 200, 200, 30);

ID.setFont(new Font("SAN SERIF", Font.BOLD, 20));

ID.setForeground(Color.WHITE);

this.add(ID);

// Set the number label

lblnumber = new JLabel("Number");

lblnumber.setBounds(550, 290, 200, 30);

lblnumber.setFont(new Font("SAN SERIF", Font.BOLD, 25));

lblnumber.setForeground(Color.WHITE);

this.add(lblnumber);

// Set the label to display the number

nmbr = new JLabel();

nmbr.setBounds(720, 290, 200, 30);

nmbr.setFont(new Font("SAN SERIF", Font.BOLD, 20));

nmbr.setForeground(Color.WHITE);

this.add(nmbr);

// Set the total price label

lbltotal = new JLabel("Total Price");

lbltotal.setBounds(550, 390, 200, 30);

lbltotal.setFont(new Font("SAN SERIF", Font.BOLD, 25));

lbltotal.setForeground(Color.WHITE);

this.add(lbltotal);

// Set the label to display the total price

total = new JLabel();

total.setBounds(720, 390, 200, 30);

total.setFont(new Font("SAN SERIF", Font.BOLD, 20));

total.setForeground(Color.WHITE);

this.add(total);

// Set the back button

back = new JButton("Back");

back.setBounds(350, 460, 250, 60);

back.setBackground(new Color(0x073f65));

back.setBorder(BorderFactory.createLineBorder(Color.WHITE));

back.setForeground(Color.WHITE);

back.setFont(new Font("SAN SERIF", Font.BOLD, 15));

back.setFocusPainted(false);

back.setBorder(BorderFactory.createLineBorder(Color.BLACK));

back.addActionListener(this);

back.setActionCommand("Back");

this.add(back);

// Load the booking details from the file

loadBookingDetails();

// Add the background image

this.add(image);

this.setVisible(true);

}

// Method to load booking details from a file

private void loadBookingDetails() {

try (BufferedReader reader = new BufferedReader(new FileReader("bookedPackages.txt"))) {

String line;

StringBuilder lastBooking = new StringBuilder();

String currentBooking = null;

// Read the file line by line and extract details

while ((line = reader.readLine()) != null) {

if (line.trim().isEmpty()) {

currentBooking = lastBooking.toString();

lastBooking.setLength(0); // Reset for next booking block

} else {

lastBooking.append(line).append("\n");

}

}

// Assign the last booking block if available

if (lastBooking.length() > 0) {

currentBooking = lastBooking.toString();

}

// If booking details are found, set them to the respective labels

if (currentBooking != null) {

String[] details = currentBooking.split("\n");

if (details.length >= 6) {

Name.setText(details[0].split(": ")[1]);

ID.setText(details[1].split(": ")[1]);

nmbr.setText(details[2].split(": ")[1]);

pack.setText(details[3].split(": ")[1]);

persons.setText(details[4].split(": ")[1]);

total.setText(details[5].split(": ")[1]);

}

} else {

System.out.println("No bookings found.");

}

} catch (IOException e) {

e.printStackTrace(); // Print stack trace in case of an IOException

}

}

// Overriding actionPerformed method to handle button actions

@Override

public void actionPerformed(ActionEvent e) {

String command = e.getActionCommand(); // Get the action command from the event

if (command.equals("Back")) {

this.setVisible(false); // Hide the frame if Back button is clicked

}

}

}

/\*

\* Click nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-default.txt to change this license

\* Click nbfs://nbhost/SystemFileSystem/Templates/Classes/Class.java to edit this template

\*/

package travelmanagementsystem;

import java.awt.Color;

import java.awt.Font;

import java.awt.Image;

import java.awt.event.ActionEvent;

import java.awt.event.ActionListener;

import java.io.BufferedReader;

import java.io.FileReader;

import java.io.IOException;

import javax.swing.BorderFactory;

import javax.swing.ImageIcon;

import javax.swing.JButton;

import javax.swing.JFrame;

import javax.swing.JLabel;

/\*\*

\*

\* @author Memon enterprise

\*/

public class ViewTripDetails extends JFrame implements ActionListener{

// Declare GUI components

private JLabel lblname, title, Name, Destination, lbldestination, lblhotel, Hotel, lblpersons, Persons, lbldays, Days, lblprice, Price;

private JButton back;

// Constructor to initialize the frame and its components

public ViewTripDetails(String name){

this.setBounds(305, 90, 970, 630);

this.setLayout(null);

this.setResizable(false);

this.setTitle("TravelPerk");

// Load and set background image

ImageIcon i1 = new ImageIcon(getClass().getResource("/Images/image99.jpg"));

Image i2 = i1.getImage().getScaledInstance(970, 630, Image.SCALE\_SMOOTH);

ImageIcon i3 = new ImageIcon(i2);

JLabel image = new JLabel(i3);

image.setBounds(0, 0, 970, 630);

// Set the title label

title = new JLabel("Trip Details");

title.setBounds(330, 40, 400, 80);

title.setFont(new Font("SAN SERIF", Font.TYPE1\_FONT, 40));

title.setForeground(Color.WHITE);

this.add(title);

// Set the name label

lblname = new JLabel("Name");

lblname.setBounds(50, 200, 200, 30);

lblname.setFont(new Font("SAN SERIF", Font.BOLD, 20));

lblname.setForeground(Color.WHITE);

this.add(lblname);

// Set the label to display the name

Name = new JLabel(name);

Name.setBounds(230, 200, 200, 30);

Name.setFont(new Font("SAN SERIF", Font.BOLD, 20));

Name.setForeground(Color.white);

this.add(Name);

// Set the destination label

lbldestination = new JLabel("Destination");

lbldestination.setBounds(50, 290, 300, 30);

lbldestination.setFont(new Font("SAN SERIF", Font.BOLD, 25));

lbldestination.setForeground(Color.WHITE);

this.add(lbldestination);

// Set the label to display the destination

Destination = new JLabel();

Destination.setBounds(230, 290, 200, 30);

Destination.setFont(new Font("SAN SERIF", Font.BOLD, 20));

Destination.setForeground(Color.WHITE);

this.add(Destination);

// Set the hotel label

lblhotel = new JLabel("Hotel");

lblhotel.setBounds(50, 390, 200, 30);

lblhotel.setFont(new Font("SAN SERIF", Font.BOLD, 25));

lblhotel.setForeground(Color.WHITE);

this.add(lblhotel);

// Set the label to display the hotel

Hotel = new JLabel();

Hotel.setBounds(230, 390, 300, 30);

Hotel.setFont(new Font("SAN SERIF", Font.BOLD, 18));

Hotel.setForeground(Color.WHITE);

this.add(Hotel);

// Set the persons label

lblpersons = new JLabel("Total Persons");

lblpersons.setBounds(530, 200, 200, 30);

lblpersons.setFont(new Font("SAN SERIF", Font.BOLD, 25));

lblpersons.setForeground(Color.WHITE);

this.add(lblpersons);

// Set the label to display the number of persons

Persons = new JLabel();

Persons.setBounds(750, 200, 200, 30);

Persons.setFont(new Font("SAN SERIF", Font.BOLD, 20));

Persons.setForeground(Color.WHITE);

this.add(Persons);

// Set the days label

lbldays = new JLabel("Days");

lbldays.setBounds(530, 290, 200, 30);

lbldays.setFont(new Font("SAN SERIF", Font.BOLD, 25));

lbldays.setForeground(Color.WHITE);

this.add(lbldays);

// Set the label to display the number of days

Days = new JLabel();

Days.setBounds(750, 290, 200, 30);

Days.setFont(new Font("SAN SERIF", Font.BOLD, 20));

Days.setForeground(Color.WHITE);

this.add(Days);

// Set the price label

lblprice = new JLabel("Total Amount");

lblprice.setBounds(530, 390, 200, 30);

lblprice.setFont(new Font("SAN SERIF", Font.BOLD, 25));

lblprice.setForeground(Color.WHITE);

this.add(lblprice);

// Set the label to display the total price

Price = new JLabel();

Price.setBounds(750, 390, 200, 30);

Price.setFont(new Font("SAN SERIF", Font.BOLD, 20));

Price.setForeground(Color.WHITE);

this.add(Price);

// Set the back button

back = new JButton("Back");

back.setBounds(350, 460, 250, 60);

back.setBackground(new Color(0x3ab094));

back.setBorder(BorderFactory.createLineBorder(Color.WHITE));

back.setForeground(Color.WHITE);

back.setFont(new Font("SAN SERIF", Font.BOLD, 15));

back.setFocusPainted(false);

back.setBorder(BorderFactory.createLineBorder(Color.BLACK));

back.addActionListener(this);

back.setActionCommand("Back");

this.add(back);

// Load the booking details

loadBookingDetails();

// Add the background image

this.add(image);

this.setVisible(true);

}

// Overriding actionPerformed method to handle button actions

@Override

public void actionPerformed(ActionEvent e){

String command = e.getActionCommand(); // Get the action command from the event

if(command.equals("Back")){

this.setVisible(false); // Hide the frame if Back button is clicked

}

}

// Method to load booking details from a file

public void loadBookingDetails() {

try (BufferedReader reader = new BufferedReader(new FileReader("bookings.txt"))) {

String line;

String lastDestination = "";

String lastHotel = "";

String lastPersons = "";

String lastDays = "";

String lastPrice = "";

// Read the file line by line and extract details

while ((line = reader.readLine()) != null) {

String[] parts = line.split(": ");

if (parts.length < 2) {

continue; // Skip lines that don't have the expected format

}

// Assign values based on the label

switch (parts[0]) {

case "Destination":

lastDestination = parts[1];

break;

case "Hotel":

lastHotel = parts[1];

break;

case "Persons":

lastPersons = parts[1];

break;

case "Days":

lastDays = parts[1];

break;

case "Total Price":

lastPrice = parts[1];

break;

}

}

// Set the extracted details to the respective labels

Destination.setText(lastDestination);

Hotel.setText(lastHotel);

Persons.setText(lastPersons);

Days.setText(lastDays);

Price.setText(lastPrice);

} catch (IOException e) {

e.printStackTrace(); // Print stack trace in case of an IOException

}

}

}