

TOURMASTER

A MICRO PROJECT REPORT

Submitted by

NAVADEEP T	2303717610421103
PRAVEEN KUMAR B	2303717610421108
UGESH K	2303717610421128
YESHWANTHRAJ G	2303717610421132

in partial fulfillment for the award of the degree of

BACHELOR OF ENGINEERING

in

COMPUTER SCIENCE AND ENGINEERING



COIMBATORE INSTITUTE OF TECHNOLOGY

(Government Aided Autonomous Institution Affiliated to Anna University)

COIMBATORE-641014

ANNA UNIVERSITY: CHENNAI 600 025

October 2024

COIMBATORE INSTITUTE OF TECHNOLOGY
(Government Aided Autonomous Institution Affiliated to Anna University)
COIMBATORE-641014



**COURSE CODE/NAME: 23CS323- JAVA PROGRAMMING
LABORATORY**

ACADEMIC YEAR: 2024-25

CERTIFIED BONAFIDE RECORD OF WORK DONE BY

S.No	Register No	Name	Signature
1	2303717610421103	NAVADEEP T	
2	2303717610421108	PRAVEEN KUMAR B	
3	2303717610421128	UGESH K	
4	2303717610421132	YESHWANTHRAJ G	

INTERNAL EXAMINER

EXTERNAL EXAMINER

COIMBATORE INSTITUTE OF TECHNOLOGY, COIMBATORE-641014
ACADEMIC YEAR: 2024 - 2025 SEMESTER: ODD
DEPARTMENT: COMPUTER SCIENCE AND ENGINEERING

Project Rubric - Java Programming Laboratory

Criteria	Excellent (5)	Good (4)	Fair (3)	Needs Improvement (2)	Unsatisfactory (1)
Design and Architecture Application architecture follows OOP principles					
Implementation - Implementation of classes and objects - Encapsulation, inheritance, and polymorphism					
Functionality - Completeness of features and requirements					
User Interface (if applicable) - Design and usability					
Documentation and Presentation - Clarity and completeness of documentation - Presentation skills					
Total					

ABSTRACT

This project, **Tour Master**, A Travel Agency Management System, exemplifies the potential of Java in building a comprehensive, high-performance application tailored to meet the demands of modern travel agencies. Designed with modular architecture, the system empowers agencies to handle multiple operational aspects seamlessly, ensuring efficient task management across the board.

For travelers, it offers a straightforward registration process, easy navigation through destinations, and a high degree of flexibility in customizing packages. Whether for business trips or leisure vacations, the system supports both domestic and international travel planning, delivering a personalized experience that aligns with users' preferences and budgets.

From the administrative side, the platform provides robust tools for package management, enabling agencies to introduce new destinations, adjust pricing, and offer special deals with ease. Real-time payment processing and secure data handling ensure that transactions are swift and reliable, fostering customer trust.

With an integrated chatbot, users benefit from an enhanced experience, receiving instant support and guidance throughout the booking process. The interactive and modular nature of this system makes it adaptable to future needs, promoting long-term business growth for small to medium-sized travel agencies. Ultimately, this Java-based solution not only automates and optimizes core operations but also transforms the customer journey into a seamless, enjoyable experience.

TABLE OF CONTENTS

CHAPTER NO.	TITLE	PAGE NO.
	ABSTRACT	4
	TABLE OF CONTENTS	5
	LIST OF FIGURES	6
	LIST OF ABBREVIATIONS	7
1	INTRODUCTION	8
	1.1 Project Domain	8
	1.2 Problem Statement	9
	1.3 Project Objective	9
	1.4 Application	10
2	UML DIAGRAMS	11
	2.1 Class diagram	11
	2.2 State diagram	20
3	TECHNOLOGY STACK	21
	3.1 Tools	21
	3.2 Handling Mechanisms	22
4	SYSTEM DESIGN	23
	4.1 Hardware Requirements	23
	4.2 Software Requirements	23
5	IMPLEMENTATION	24
	5.1 Explanation	24
	5.2 Program Code	28
6	SCREENSHOTS	88
7	REFERENCES	93

LIST OF FIGURES

FIGURE NO	TITLE	PAGE NO
2.1.1	Class Diagram	11
2.2.1	State Diagram	20
6.1	Welcome Page	88
6.2	Home Page	88
6.3	User Login	89
6.4	Package Types	89
6.5	Self Chosen Package	90
6.6	Default Package – Type	90
6.7	Default Package - Country	91
6.8	Default Package – Pack Type	91
6.9	Payment Page	92
6.10	Ticket Booking Successful	92

LIST OF ABBREVIATIONS

JAR – Java Archive

AWT – Abstract Window Toolkit

GUI – Graphical User Interface

OOP – Object Oriented Programming

JDK – Java Development Toolkit

IDE – Integrated Development Environment

WB – Windows Builder

JPEG / JPG – Joint Photographic Experts Group

PNG – Portable Network Graphics

src - Source

CHAPTER 1

INTRODUCTION

Background of the Project Domain

The tourism industry has grown exponentially over the past few decades, driving the need for automated travel agency management systems. In a conventional setup, travel agents manually record customer information, bookings, and payments, which can lead to inefficiencies, delays, and human errors. Furthermore, customers today expect online solutions that allow easy trip planning and quick access to various destinations and travel packages. Managing both domestic and international travel logistics in an organized and efficient manner is essential to stay competitive in the market.

This project aims to develop a centralized software solution that can simplify travel planning. Using Java as the core technology, the system ensures smooth operations for administrators and customers alike. It supports role-based access, where the administrator manages destination packages and monitors bookings, while users can explore, register, log in, and book travel packages based on their preferences. Chatbots and custom payment handling systems provide enhanced functionality.

Problem Statement

Manual operations in travel agencies are prone to inconsistencies and inefficiencies. Keeping track of multiple bookings, travel destinations, and payments often involves redundant efforts. Additionally, customers face challenges in exploring travel options, and agents may struggle to customize packages based on individual requirements. This project addresses the need for a digital travel management platform where:

- Users can easily browse domestic and international destinations.
- Admins can efficiently manage packages and customer data.
- Payment systems are integrated to offer seamless booking experiences.

Objectives

The primary objectives of this project are:

- Automate essential travel agency operations, including booking, payment, and package management.
- Provide a modular system where the admin has complete control over tour packages and user data.
- Facilitate user interaction through intuitive navigation and chatbot support.
- Offer secure payment gateways to process bookings without any manual intervention.
- Ensure role-based access to improve system security and performance.

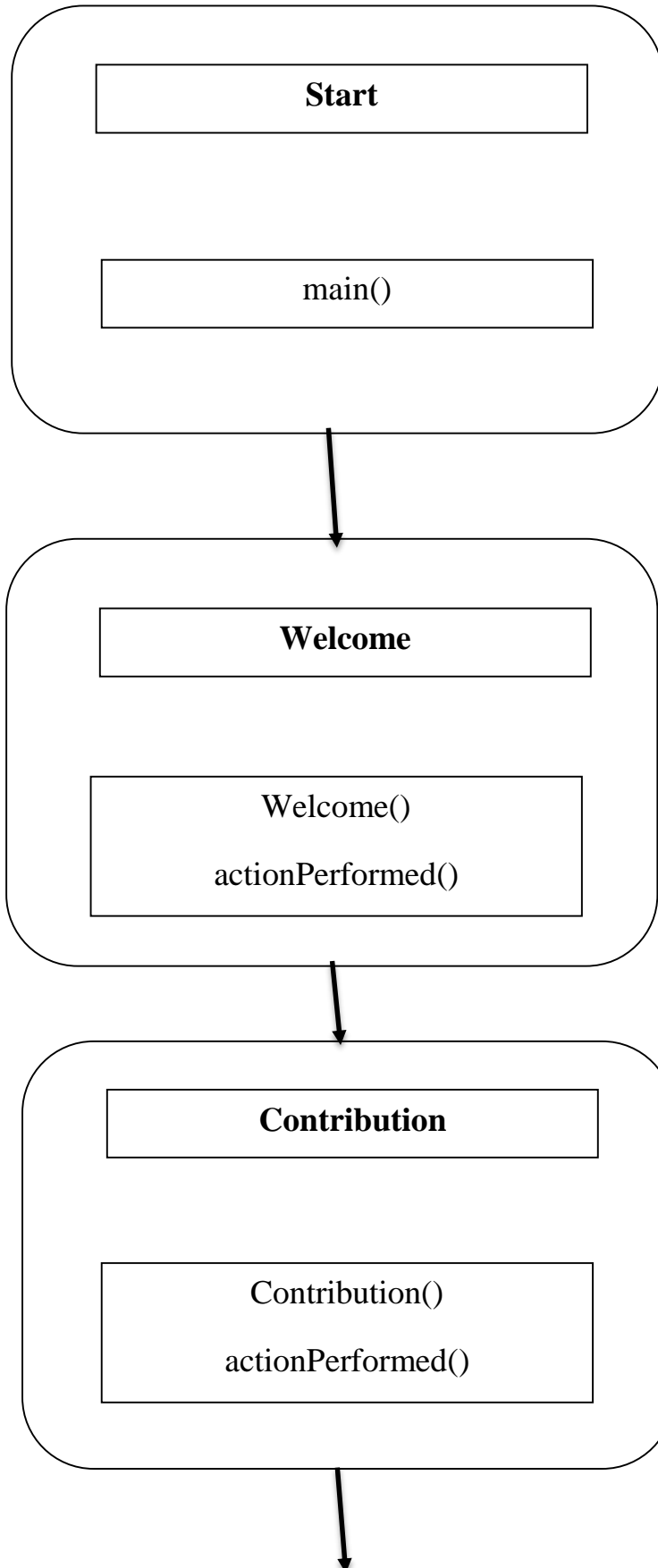
Applications

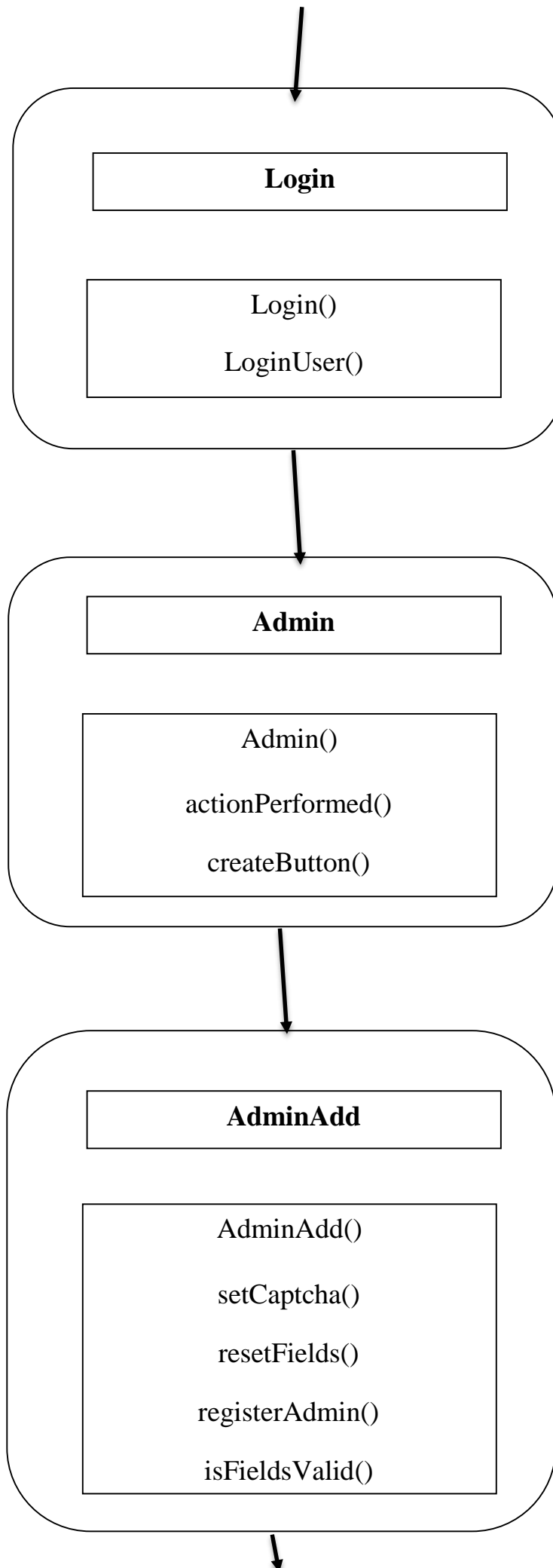
The Travel Agency Management System can be applied in various contexts:

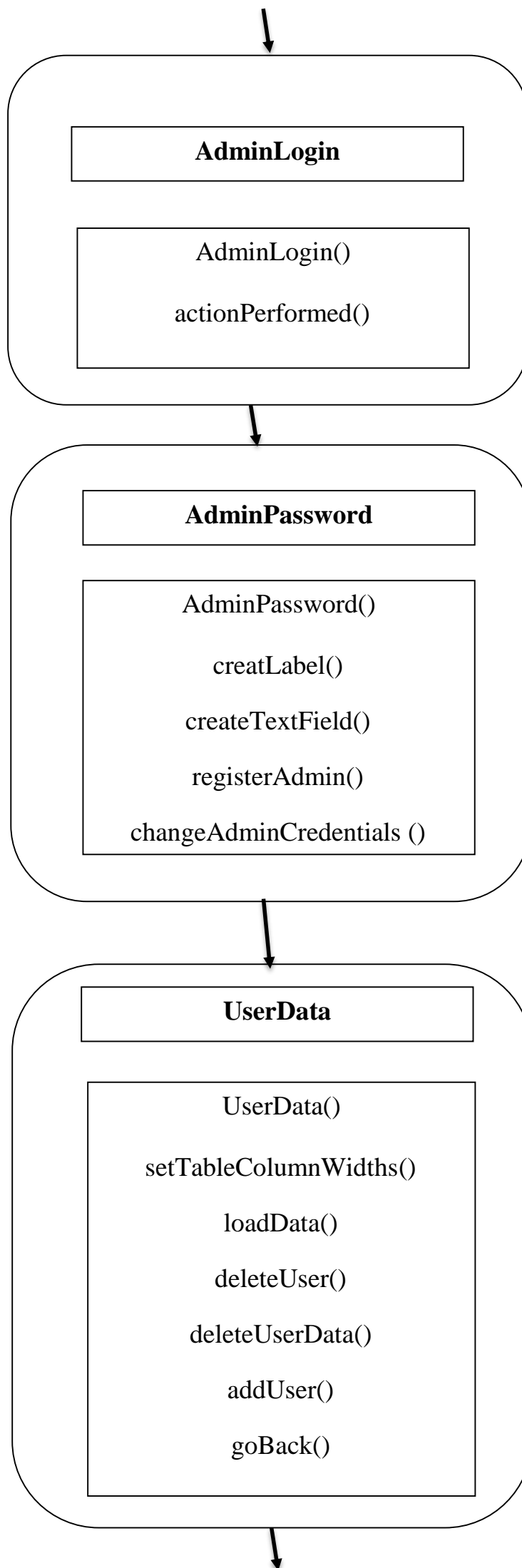
- Small to medium travel agencies looking to digitize their operations and reduce manual effort.
- Educational institutions to showcase practical applications of Java in software development projects.
- Tourism start-ups seeking affordable and efficient solutions to manage bookings and customer interactions.
- Personalized travel portals that need easy customization of packages and seamless payment handling.

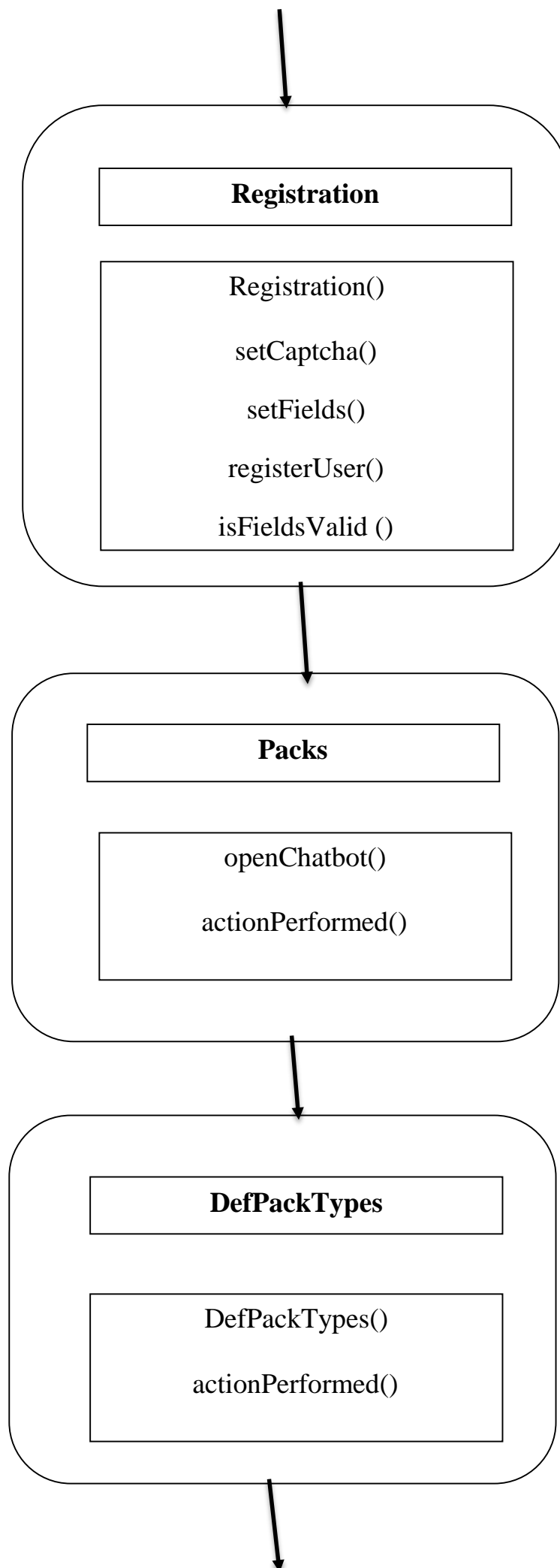
CHAPTER 2

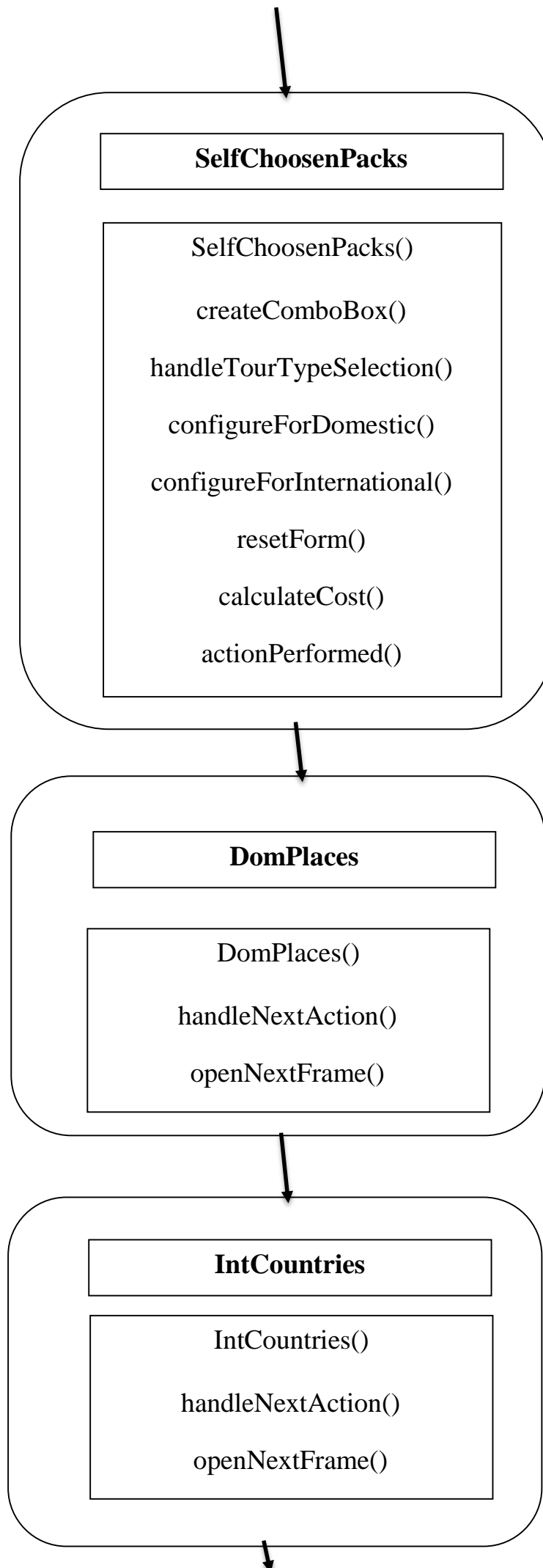
UML DIAGRAMS

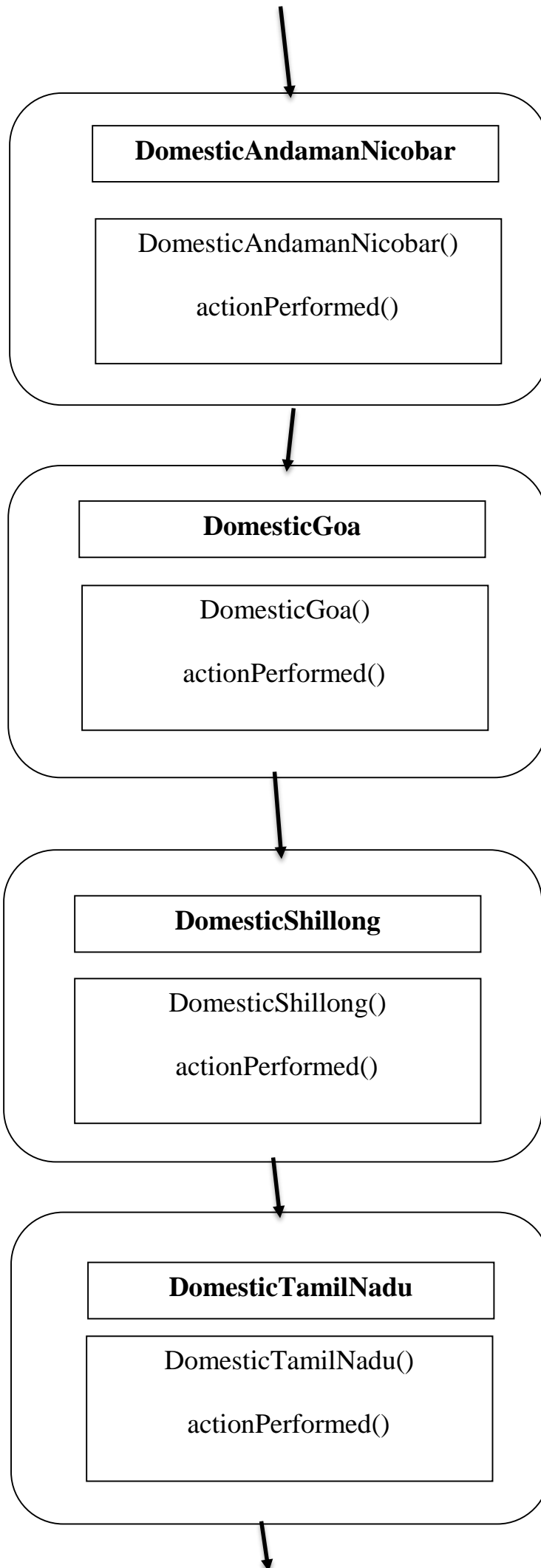


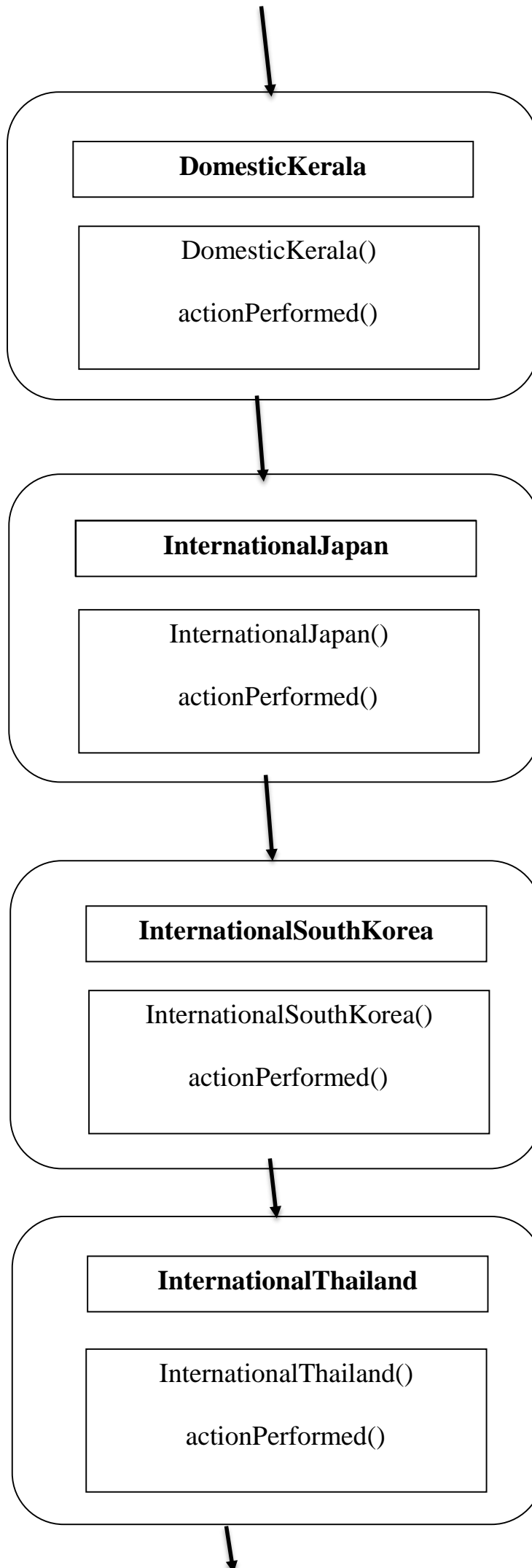


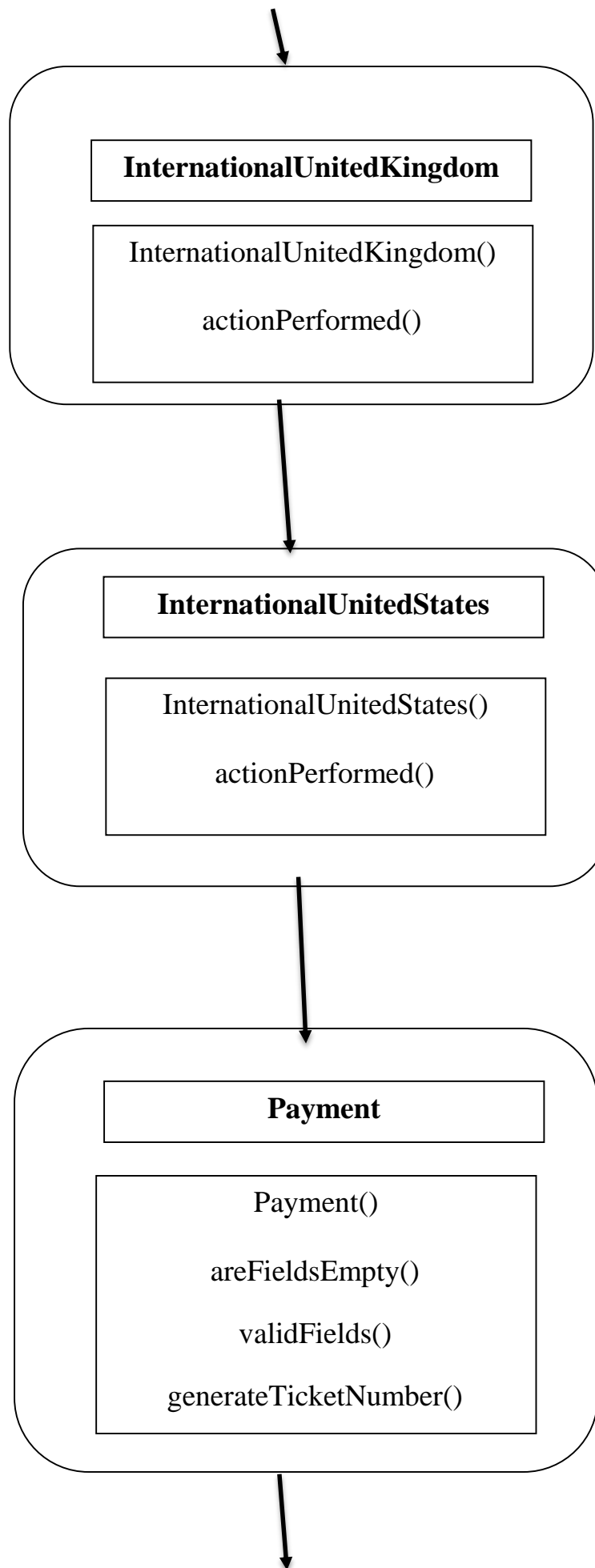


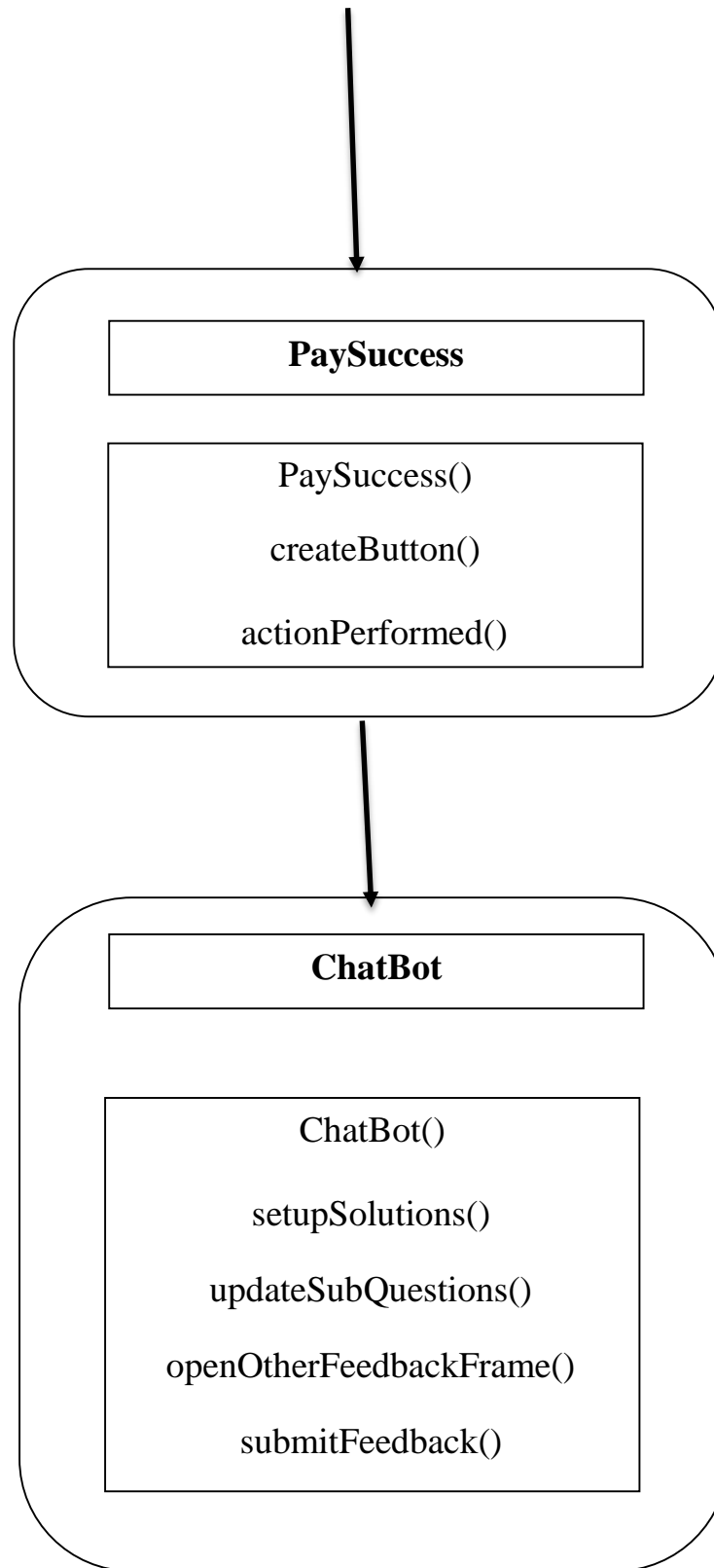




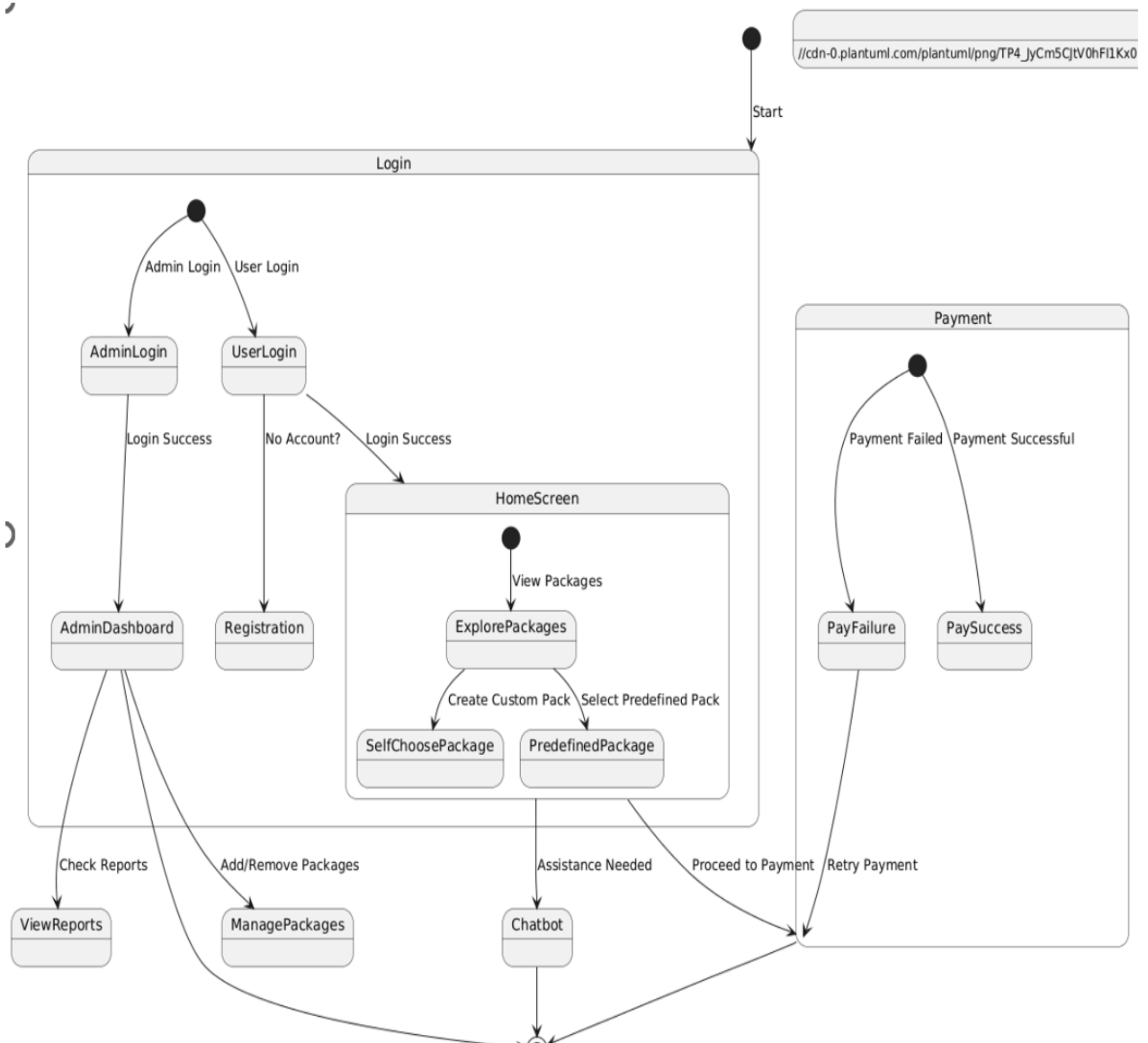








STATE DIAGRAM



CHAPTER 3

TECHNOLOGY STACK

1. Programming Language

Java: The primary language used for application development, leveraging its object-oriented features for modular programming.

2. Integrated Development Environment (IDE)

Eclipse: A powerful IDE for Java development that offers comprehensive tools for coding, debugging, and project management.

3. User Interface Frameworks

Java Swing: Utilized for creating a rich graphical user interface (GUI), providing components like buttons, panels, and dialogs for user interaction.

Java AWT (Abstract Window Toolkit): An optional library for additional UI elements and event handling.

4. Data Storage

Text Files: Simple text files (e.g., `user_data.txt`, `admin_data.txt`) used for storing user and admin information in a human-readable format.

5. File Handling

Java I/O: Leveraging classes from the `java.io` package for effective input/output operations, enabling reading and writing of data to text files.

7. Exception Handling

Java Exception Handling: Utilizing built-in exceptions (e.g., `IOException`, `InputMismatchException`) to gracefully handle errors related to file operations and user input.

8. Version Control (if applicable)

Git: Recommended for version control and collaboration, allowing you to track changes and manage your source code effectively.

CHAPTER 4

SYSTEM DESIGN

Hardware Requirements :

- **Processor:** Dual-core processor (Intel i3 or equivalent)
- **RAM:** 4GB or higher
- **Storage:** 1GB free space for project files and execution
- **Display:** 1024x768 resolution or higher
- **Additional Requirements:** Internet connection (if integrating external APIs or payment gateways)

Software Requirements :

- **Programming Language:** Java
- **Development Environment:** Eclipse or IntelliJ IDEA (or other Java IDEs)
- **User Interface:** Java Swing or AWT
- **Data Storage:** Text files (e.g., admin_data.txt for storing admin/user data)
- **Toolkit :** Spring Boot, Windows Builder.
- **JDK :** Java 8 or higher
- **Java Runtime:** JRE 8 or higher
- **Operating System:** Compatible with Windows, Linux, and macOS

CHAPTER 5

IMPLEMENTATION

The implementation of the Travel Agency Management System is based on Java and follows a modular approach, where each functionality is divided into different classes. The project includes essential features such as user registration, login, package management, payment processing, and chat assistance. Below is a detailed explanation of the implementation, including code structure, key modules, and workflows.

Code Structure

The source code is divided into multiple Java classes, organized under the TravelAgency package. Each class handles a specific feature of the system to ensure better maintainability and scalability. Below is an overview of the key classes:

Home.java: The entry point of the system, providing a welcome screen and navigation options.

AdminLogin.java & Login.java: Classes for handling admin and user login functionalities.

Registration.java: Manages user registration by capturing essential information and storing it in a text file.

Packs.java & SelfChosenPacks.java: Allow users to explore predefined travel packages or create custom packages.

Payment.java & PaySuccess.java: Handle payment workflows, confirming

successful transactions.

ChatBot.java: Provides interactive assistance to users in navigating the system.

Workflow Overview

1. User Registration and Login

- New users register by filling out a form through the Registration.java class.
- Registered users log in through Login.java, while admins authenticate using AdminLogin.java.
- Upon login, users are directed to the home screen, where they can explore travel options.

2. Package Selection

- Users can choose between domestic or international travel packages through dedicated classes like DomesticGoa.java or InternationalJapan.java.
- Alternatively, they can create customized travel packs via the SelfChosenPacks.java class.

3. Admin Management

- The admin logs in via AdminLogin.java and manages user data, packages, and other system settings.
- Admins can add new packages (AdminAdd.java) or modify existing ones.

4. Payment Process

- Once a user selects a package, they proceed to the payment section managed by Payment.java.
- Upon successful payment, the system confirms the transaction through PaySuccess.java.

5. Chatbot Support

- The ChatBot.java class offers guidance to users throughout the booking process, helping them with inquiries about destinations and packages.

Tools and Techniques Used

- **Java Swing:** For building the graphical user interface (GUI).
- **File I/O Operations:** Used for storing user and admin data in text files.
- **Exception Handling:** Ensures the system gracefully handles errors such as invalid input or failed file operations.
- **Modular Design:** Each class handles a specific functionality to maintain clean code architecture.
- **Chatbot Logic:** Provides interactive text-based support to users within the system.

Challenges Faced During Implementation

- **Managing User Data:** Without a database, handling large volumes of text-based data required efficient file I/O operations.
- **Error Handling:** Developing smooth exception handling to manage invalid inputs during registration and login processes.
- **UI Design:** Creating an intuitive interface with Java Swing that is both user-friendly and functional.

PROGRAM :-

Here're some important program of defined classes in our project.

1. Welcome.java

```
package TravelAgency;

import java.awt.*;

import javax.swing.*;

import java.awt.event.*;

public class Welcome extends JFrame implements ActionListener {

    private ImageIcon background;

    private JLabel backgroundLabel, gmail, phno;

    private Font f1;

    private JButton btn1, btn2, btn3;

    Welcome() {

        this.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);

        this.setTitle("Travel Agency");

        this.setSize(1536, 864);

        this.setLocationRelativeTo(null);

        getContentPane().setLayout(null);

        ImageIcon bgImage = new ImageIcon(getClass().getResource("/WelcomeBG.jpg"));

        Image scaledBg = bgImage.getImage().getScaledInstance(this.getWidth(), this.getHeight(),
        Image.SCALE_SMOOTH);

        background = new ImageIcon(scaledBg);
```

```
backgroundLabel = new JLabel(background);

backgroundLabel.setBounds(0, 0, this.getWidth(), this.getHeight());

getContentPane().add(backgroundLabel);


f1 = new Font("Arial Black", Font.BOLD, 25);

Cursor cursor = new Cursor(Cursor.HAND_CURSOR);


int buttonY = this.getHeight() - 150;

int buttonHeight = 50;


btn1 = new JButton("Exit");

btn1.setBounds(100, buttonY, 200, buttonHeight);

btn1.setFont(f1);

btn1.setCursor(cursor);

btn1.setForeground(Color.WHITE);

btn1.setBackground(Color.decode("#C00000"));

backgroundLabel.add(btn1);


gmail = new JLabel("Gmail :tourmaster134@gmail.com");

gmail.setFont(f1);

gmail.setForeground(Color.BLACK);

gmail.setBounds(100, 30, 500, 50);

backgroundLabel.add(gmail);


btn2 = new JButton("Contribution");

btn2.setBounds((this.getWidth() - 245) / 2 - 50, buttonY, 245, buttonHeight);

btn2.setFont(f1);
```

```

    btn2.setCursor(cursor);

    btn2.setForeground(Color.WHITE);

    btn2.setBackground(Color.decode("#2E75B6"));

    backgroundLabel.add(btn2);


    btn3 = new JButton("Next");

    btn3.setBounds(this.getWidth() - 300, buttonY, 200, buttonHeight);

    btn3.setFont(f1);

    btn3.setCursor(cursor);

    btn3.setForeground(Color.WHITE);

    btn3.setBackground(Color.decode("#2E75B6"));

    backgroundLabel.add(btn3);


    phno = new JLabel("Contact no : 9876543210");

    phno.setFont(f1);

    phno.setForeground(Color.BLACK);

    phno.setBounds(this.getWidth() - 500, 30, 400, 50);

    backgroundLabel.add(phno);


    btn1.addActionListener(this);

    btn2.addActionListener(this);

    btn3.addActionListener(this);
}


public void actionPerformed(ActionEvent e) {

    if (e.getSource() == btn2) {

        try {

```

```

        new Contribution().setVisible(true);
    } catch (Exception ex) {
        System.err.println("Error opening Contribution frame: " + ex.getMessage());
    }
} else if (e.getSource() == btn3) {
    setVisible(false);
    try {
        new Home().setVisible(true);
    } catch (Exception ex) {
        System.err.println("Error opening Home frame: " + ex.getMessage());
    }
} else if (e.getSource() == btn1) {
    System.exit(0);
}
}

public static void main(String[] args) {
    new Welcome().setVisible(true);
}
}

```

2. Home.java

```

package TravelAgency;

import java.awt.*;
import javax.swing.*;
import java.awt.event.*;
import java.net.URL;

public class Home extends JFrame {

```

```

private JLabel titleLabel1, titleLabel2;

private Font titleFont, buttonFont;

private JButton loginButton, registerButton, exitButton, adminLoginButton;

private JPanel backgroundPanel;

Home() {

    setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);

    setTitle("Travel Agency");

    setSize(1536, 864);

    setLocationRelativeTo(null);

    setResizable(false);

    backgroundPanel = new JPanel() {

        @Override

        protected void paintComponent(Graphics g) {

            super.paintComponent(g);

            URL backgroundImageUrl = getClass().getResource("/User.jpg");

            if (backgroundImageUrl != null) {

                ImageIcon backgroundImage = new ImageIcon(backgroundImageUrl);

                g.drawImage(backgroundImage.getImage(), 0, 0, getWidth(), getHeight(), this);

            }

        }

    };

    backgroundPanel.setLayout(null);

    getContentPane().add(backgroundPanel);

```



```
titleFont = new Font("Segoe UI Black", Font.BOLD, 48);

buttonFont = new Font("Segoe UI Black", Font.PLAIN, 25);


titleLabel1 = new JLabel("Travel Anywhere", SwingConstants.CENTER);

titleLabel1.setBounds(0, 170, 1536, 65);

titleLabel1.setFont(titleFont);

backgroundPanel.add(titleLabel1);


titleLabel2 = new JLabel("In the World!", SwingConstants.CENTER);

titleLabel2.setBounds(0, 300, 1536, 65);

titleLabel2.setFont(titleFont);

backgroundPanel.add(titleLabel2);


Cursor cursor = new Cursor(Cursor.HAND_CURSOR);


loginButton = createButton("Login", 400, 450, cursor);

registerButton = createButton("Register", 1000, 450, cursor);

exitButton = createButton("Exit", 400, 600, cursor, Color.decode("#C00000"));

adminLoginButton = createButton("Admin Login", 1000, 600, cursor);


backgroundPanel.add(loginButton);

backgroundPanel.add(registerButton);

backgroundPanel.add(exitButton);

backgroundPanel.add(adminLoginButton);


loginButton.addActionListener(e -> openFrame(new Login()));

registerButton.addActionListener(e -> openFrame(new Registration()));
```

```

exitButton.addActionListener(e -> System.exit(0));

adminLoginButton.addActionListener(e -> {

    openFrame(new AdminLogin());

    JOptionPane.showMessageDialog(null, "By default, Admin Name and Password is : 'ugesh
041105'", "Information!", JOptionPane.INFORMATION_MESSAGE);

});

}

private JButton createButton(String text, int x, int y, Cursor cursor) {

    return createButton(text, x, y, cursor, Color.decode("#2E75B6"));

}

private JButton createButton(String text, int x, int y, Cursor cursor, Color bgColor) {

    JButton button = new JButton(text);

    button.setBounds(x, y, 195, 50);

    button.setFont(buttonFont);

    button.setCursor(cursor);

    button.setForeground(Color.WHITE);

    button.setBackground(bgColor);

    return button;

}

private void openFrame(JFrame frame) {

    setVisible(false);

    frame.setVisible(true);

}

```

```

public static void main(String[] args) {

    SwingUtilities.invokeLater(() -> new Home().setVisible(true));

}

}

```

3. Login.java

```

package TravelAgency;

import java.awt.*;

import javax.swing.*;

import java.awt.event.*;

import java.io.*;

public class Login extends JFrame {

    private JLabel titleLabel, backgroundLabel, userLabel, passLabel;

    private JTextField usernameField;

    private JPasswordField passwordField;

    private JButton exitButton, backButton, loginButton;

    private Font titleFont, buttonFont, labelFont, fieldFont;

    public Login() {

        this.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);

        this.setTitle("Travel Agency - Login");

        this.setSize(1536, 864);

        this.setLocationRelativeTo(null);

        this.setResizable(false);

        getContentPane().setLayout(null);

        titleLabel = new Font("Segoe UI Black", Font.BOLD, 60);

        buttonFont = new Font("Segoe UI Black", Font.PLAIN, 25);

```

```
labelFont = new Font("Segoe UI", Font.BOLD, 33);

fieldFont = new Font("Segoe UI", Font.PLAIN, 22);


ImageIcon bgImage = new ImageIcon(getClass().getResource("/User.jpg"));

Image scaledBg = bgImage.getImage().getScaledInstance(this.getWidth(), this.getHeight(),
Image.SCALE_SMOOTH);

ImageIcon background = new ImageIcon(scaledBg);

backgroundLabel = new JLabel(background);

backgroundLabel.setBounds(0, 0, this.getWidth(), this.getHeight());

getContentPane().add(backgroundLabel);


titleLabel = new JLabel("User Login");

titleLabel.setFont(titleFont);

titleLabel.setBounds(549, 160, 500, 90);

backgroundLabel.add(titleLabel);


userLabel = new JLabel("User Name");

userLabel.setFont(labelFont);

userLabel.setBounds(450, 302, 246, 50);

backgroundLabel.add(userLabel);


usernameField = new JTextField();

usernameField.setFont(fieldFont);

usernameField.setBounds(761, 314, 200, 35);

backgroundLabel.add(usernameField);


passLabel = new JLabel("Password");
```

```

passLabel.setFont(labelFont);

passLabel.setBounds(450, 444, 293, 50);

backgroundLabel.add(passLabel);


passwordField = new JPasswordField();

passwordField.setFont(fieldFont);

passwordField.setBounds(761, 456, 200, 35);

backgroundLabel.add(passwordField);


exitButton = createButton("Exit", 30, 650, "#C00000", e -> System.exit(0));

backgroundLabel.add(exitButton);


backButton = createButton("Back", 630, 650, "#2E75B6", e -> {

    setVisible(false);

    new Home().setVisible(true);

});

backgroundLabel.add(backButton);


loginButton = createButton("Login", 1150, 650, "#2E75B6", e -> loginUser());

backgroundLabel.add(loginButton);

}


private JButton createButton(String text, int x, int y, String colorHex, ActionListener action) {

    JButton button = new JButton(text);

    button.setFont(buttonFont);

    button.setForeground(Color.WHITE);

    button.setBackground(Color.decode(colorHex));

```

```

button.setBounds(x, y, 300, 70);

button.setCursor(new Cursor(Cursor.HAND_CURSOR));

button.addActionListener(action);

return button;
}

```

```

private void loginUser() {

    String usernameInput = usernameField.getText().toLowerCase();

    String passwordInput = new String(passwordField.getPassword());

    if (usernameInput.isEmpty() || passwordInput.isEmpty()) {

        JOptionPane.showMessageDialog(this, "Please fill all of the fields.", "Warning",
JOptionPane.WARNING_MESSAGE);

        return;

    }
}

```

```

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

    String line;

    boolean isAuthenticated = false;

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

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

            String fileUsername = line.substring(11).trim();

            String passwordLine = reader.readLine();

            passwordLine = reader.readLine();

            if (passwordLine != null && passwordLine.startsWith("Password: ")) {

```

```

        String filePassword = passwordLine.substring(10).trim();

        if (fileUsername.equalsIgnoreCase(usernameInput) &&
filePassword.equals(passwordInput)) {

            isAuthenticated = true;

            break;

        }

    }

}

}

if (isAuthenticated) {

    JOptionPane.showMessageDialog(this, "Login Successful.", "Travel Agency",
JOptionPane.INFORMATION_MESSAGE);

    setVisible(false);

    new Packs().setVisible(true);

} else {

    JOptionPane.showMessageDialog(this, "Invalid Username or Password!", "Error",
JOptionPane.ERROR_MESSAGE);

    }

} catch (IOException e) {

    JOptionPane.showMessageDialog(this, "Error accessing user data.", "Error",
JOptionPane.ERROR_MESSAGE);

    }

}

public static void main(String[] args) {

    SwingUtilities.invokeLater(() -> new Login().setVisible(true));

}

```

```
}
```

4. Registration.java

```
package TravelAgency;
```

```
import java.awt.*;
```

```
import javax.swing.*;
```

```
import java.awt.event.*;
```

```
import java.io.*;
```

```
import java.time.LocalDateTime;
```

```
import java.time.format.DateTimeFormatter;
```

```
import java.util.Random;
```

```
public class Registration extends JFrame {
```

```
    private JLabel titleLabel, userNameLabel, emailLabel, passwordLabel, securityQLabel,  
    answerLabel, captchaLabel;
```

```
    private JTextField userNameField, emailField, answerField, captchaField;
```

```
    private JPasswordField passwordField;
```

```
    private JComboBox<String> securityQComboBox;
```

```
    private JButton exitButton, backButton, resetButton, registerButton;
```

```
    private int captchaResult;
```

```
    public Registration() {
```

```
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
```

```
        setTitle("Travel Agency - User Registration");
```

```
        setSize(1536, 864);
```

```
        setLocationRelativeTo(null);
```

```
        setResizable(false);
```



```
setLayout(null);
```

```
ImageIcon bgImage = new ImageIcon(getClass().getResource("/Admin.jpg"));
```

```
JLabel backgroundLabel = new JLabel(new  
ImageIcon(bgImage.getImage().getScaledInstance(this.getWidth(), this.getHeight(),  
Image.SCALE_SMOOTH)));
```

```
backgroundLabel.setBounds(0, 0, 1536, 864);
```

```
add(backgroundLabel);
```

```
titleLabel = new JLabel("Enter Your Information");
```

```
titleLabel.setFont(new Font("Segoe UI Black", Font.BOLD, 45));
```

```
titleLabel.setForeground(Color.WHITE);
```

```
titleLabel.setBounds(500, 50, 600, 50);
```

```
backgroundLabel.add(titleLabel);
```

```
userNameLabel = createLabel("User Name", 400, 150);
```

```
backgroundLabel.add(userNameLabel);
```

```
userNameField = createTextField(700, 150);
```

```
backgroundLabel.add(userNameField);
```

```
emailLabel = createLabel("Email", 400, 250);
```

```
backgroundLabel.add(emailLabel);
```

```
emailField = createTextField(700, 250);
```

```
backgroundLabel.add(emailField);
```

```
passwordLabel = createLabel("Password", 400, 350);
```

```
backgroundLabel.add(passwordLabel);
```

```
passwordField = new JPasswordField();
```

```

passwordField.setFont(new Font("Segoe UI", Font.PLAIN, 22));

passwordField.setBounds(700, 350, 400, 40);

backgroundLabel.add(passwordField);


securityQLabel = createLabel("Security Question", 400, 450);

backgroundLabel.add(securityQLabel);

securityQComboBox = new JComboBox<>(new String[] { "Choose a Security Question...",
"Your dream job?", "Your favorite song?", "First pet's name?", "Your favorite hobby?" });

securityQComboBox.setFont(new Font("Segoe UI", Font.PLAIN, 22));

securityQComboBox.setBounds(700, 450, 400, 40);

backgroundLabel.add(securityQComboBox);


answerLabel = createLabel("Answer", 400, 550);

backgroundLabel.add(answerLabel);

answerField = createTextField(700, 550);

backgroundLabel.add(answerField);


captchaLabel = createLabel("", 400, 650);

captchaLabel.setForeground(Color.RED);

setCaptcha();

backgroundLabel.add(captchaLabel);

captchaField = createTextField(700, 650);

backgroundLabel.add(captchaField);


exitButton = createButton("Exit", 340, 750, e -> System.exit(0));

backgroundLabel.add(exitButton);

backButton = createButton("Back", 540, 750, e -> new Home().setVisible(true));

```

```
backgroundLabel.add(backButton);

resetButton = createButton("Reset", 740, 750, e -> resetFields());

backgroundLabel.add(resetButton);

registerButton = createButton("Register", 940, 750, e -> registerUser());

backgroundLabel.add(registerButton);
}

private JLabel createLabel(String text, int x, int y) {
    JLabel label = new JLabel(text);

    label.setFont(new Font("Segoe UI", Font.BOLD, 30));

    label.setForeground(Color.WHITE);

    label.setBounds(x, y, 300, 40);

    return label;
}

private JTextField createTextField(int x, int y) {
    JTextField textField = new JTextField();

    textField.setFont(new Font("Segoe UI", Font.PLAIN, 22));

    textField.setBounds(x, y, 400, 40);

    return textField;
}

private JButton createButton(String text, int x, int y, ActionListener action) {
    JButton button = new JButton(text);

    button.setFont(new Font("Segoe UI Black", Font.PLAIN, 30));

    button.setForeground(Color.WHITE);

    button.setBackground(Color.decode("#2E75B6"));
```

```

        button.setBounds(x, y, 180, 50);

        button.addActionListener(action);

        return button;
    }

```

```

private void setCaptcha() {

    Random rand = new Random();

    int a = rand.nextInt(10), b = rand.nextInt(10);

    captchaResult = a + b;

    captchaLabel.setText("Captcha: " + a + " + " + b + " = ?");

}

```

```

private void resetFields() {

    userNameField.setText("");

    emailField.setText("");

    passwordField.setText("");

    answerField.setText("");

    captchaField.setText("");

    securityQComboBox.setSelectedIndex(0);

    setCaptcha();

}

```

```

private void registerUser() {

    try {

        if (isFieldsValid()) {

            try (PrintWriter pw = new PrintWriter(new BufferedWriter(new FileWriter("user_data.txt",
true)))) {

```

```

        String formattedDate =
LocalDateTime.now().format(DateTimeFormatter.ofPattern("HH:mm a, dd/MM/yyyy"));

        pw.printf("User Name: %s%nEmail: %s%nPassword: %s%nSecurity Question:
%s%nAnswer: %s%nDate & Time:
%s%n===== %n",

                userNameField.getText(), emailField.getText(), new
String(passwordField.getPassword()),

                securityQComboBox.getSelectedItem(), answerField.getText(), formattedDate);

        JOptionPane.showMessageDialog(this, "Registration Successful!", "Success",
JOptionPane.INFORMATION_MESSAGE);

        resetFields();

    } catch (IOException e) {

        JOptionPane.showMessageDialog(this, "Error saving data.", "Error",
JOptionPane.ERROR_MESSAGE);

    }

}

} catch (RegistrationException e) {

    JOptionPane.showMessageDialog(this, e.getMessage(), "Validation Error",
JOptionPane.WARNING_MESSAGE);

}

}

private boolean isFieldsValid() throws RegistrationException {

    if (userNameField.getText().isEmpty() || emailField.getText().isEmpty() ||
passwordField.getPassword().length == 0

        || securityQComboBox.getSelectedIndex() == 0 || answerField.getText().isEmpty() ||
captchaField.getText().isEmpty()) {

        throw new RegistrationException("Please fill all fields.");

    }

    if (!emailField.getText().matches("^([A-Za-z0-9+_.-]+@[A-Za-z0-9.-]+)$")) {

        throw new RegistrationException("Invalid email format.");

```

```

    }

    if (userNameField.getText().length() < 3) {
        throw new RegistrationException("Username must be at least 3 characters long.");
    }

    if (!captchaField.getText().equals(String.valueOf(captchaResult))) {
        setCaptcha();
        throw new RegistrationException("Incorrect captcha.");
    }

    return true;
}

public static void main(String[] args) {
    SwingUtilities.invokeLater(() -> new Registration().setVisible(true));
}
}

class RegistrationException extends Exception {
    public RegistrationException(String message) {
        super(message);
    }
}

5. AdminPassword.java

package TravelAgency;

import java.awt.*;

```

```

import javax.swing.*;

import java.awt.event.*;

import java.io.*;

import java.net.URL;


public class AdminPassword extends JFrame {


    private JLabel titleLabel1, titleLabel2, nameLabel, passwordLabel, backgroundLabel;

    private Font titleFont, fieldFont, buttonFont;

    private JTextField nameField;

    private JPasswordField passwordField;

    private JButton closeButton, changeButton;

    private Cursor handCursor;


    AdminPassword() {

        setTitle("Travel Agency - Change Admin Credentials");

        setSize(1536, 864);

        setLocationRelativeTo(null);

        setResizable(false);

        setDefaultCloseOperation(JFrame.DISPOSE_ON_CLOSE);

        getContentPane().setLayout(null);


        URL bgImageUrl = getClass().getResource("/Admin.jpg");

        if (bgImageUrl != null) {

            ImageIcon bgImage = new ImageIcon(bgImageUrl);

            Image bg = bgImage.getImage();

            Image scaledBg = bg.getScaledInstance(this.getWidth(), this.getHeight(),

```

```
Image.SCALE_SMOOTH);
```

```
    backgroundLabel = new JLabel(new ImageIcon(scaledBg));
```

```
    backgroundLabel.setBounds(0, 0, 1536, 864);
```

```
    getContentPane().add(backgroundLabel);
```

```
    backgroundLabel.setLayout(null);
```

```
}
```

```
titleFont = new Font("Segoe UI Black", Font.BOLD, 48);
```

```
fieldFont = new Font("Segoe UI", Font.PLAIN, 25);
```

```
buttonFont = new Font("Segoe UI Black", Font.PLAIN, 25);
```

```
titleLabel1 = createLabel("Change Admin Name", 0, 50, 1536, 60, titleFont, Color.WHITE,
backgroundLabel);
```

```
titleLabel2 = createLabel("And Password", 0, 130, 1536, 60, titleFont, Color.WHITE,
backgroundLabel);
```

```
nameLabel = createLabel("Name:", 450, 280, 200, 50, new Font("Segoe UI", Font.BOLD, 28),
Color.WHITE, backgroundLabel);
```

```
nameField = createTextField(660, 280, 350, 50, fieldFont, backgroundLabel);
```

```
passwordLabel = createLabel("Password:", 450, 500, 200, 50, new Font("Segoe UI",
Font.BOLD, 28), Color.WHITE, backgroundLabel);
```

```
passwordField = createPasswordField(660, 500, 350, 50, fieldFont, backgroundLabel);
```

```
handCursor = new Cursor(Cursor.HAND_CURSOR);
```

```
closeButton = createButton("Close", 400, 700, 220, 60, "#C00000", handCursor, buttonFont,
backgroundLabel);
```

```
changeButton = createButton("Change", 850, 700, 220, 60, "#2E75B6", handCursor, buttonFont,
backgroundLabel);
```



```

closeButton.addActionListener(ae -> setVisible(false));

changeButton.addActionListener(ae -> changeAdminCredentials());

}

```

```

private JLabel createLabel(String text, int x, int y, int width, int height, Font font, Color color,
Container container) {

```

```

    JLabel label = new JLabel(text, SwingConstants.CENTER);

    label.setBounds(x, y, width, height);

    label.setFont(font);

    label.setForeground(color);

    container.add(label);

    return label;

}

```

```

private JTextField createTextField(int x, int y, int width, int height, Font font, Container container)
{

```

```

    JTextField field = new JTextField();

    field.setBounds(x, y, width, height);

    field.setFont(font);

    container.add(field);

    return field;

}

```

```

private JPasswordField createPasswordField(int x, int y, int width, int height, Font font, Container
container) {

```

```

    JPasswordField field = new JPasswordField();

    field.setBounds(x, y, width, height);

```

```
field.setFont(font);

field.setEchoChar('*');

container.add(field);

return field;

}
```

```
private JButton createButton(String text, int x, int y, int width, int height, String colorHex, Cursor
cursor, Font font, Container container) {

    JButton button = new JButton(text);

    button.setBounds(x, y, width, height);

    button.setFont(font);

    button.setForeground(Color.WHITE);

    button.setBackground(Color.decode(colorHex));

    button.setCursor(cursor);

    container.add(button);

    return button;

}
```

```
private void changeAdminCredentials() {

    String adminName = nameField.getText().toLowerCase();

    String adminPassword = new String(passwordField.getPassword());

    if (adminName.isEmpty() || adminPassword.isEmpty()) {

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

        return;

    }

}
```

```

String adminFile = "./admin_data.txt";

File oldFile = new File(adminFile);

oldFile.delete();

try (PrintWriter pw = new PrintWriter(new BufferedWriter(new FileWriter(adminFile, true)))) {

    pw.println("=====");

    pw.println("=== ### Travel Agency Admin Data ### ===");

    pw.println("=====");

    pw.println("Name : " + adminName);

    pw.println("Password : " + adminPassword);

    pw.println("=====");

    JOptionPane.showMessageDialog(null, "Admin Name and Password have been changed.",
    "Success!", JOptionPane.INFORMATION_MESSAGE);

    setVisible(false);

} catch (IOException ex) {

    ex.printStackTrace();

}

}

public static void main(String[] args) {

    AdminPassword frame = new AdminPassword();

    frame.setVisible(true);

}

}

```

6. DefPackTypes.java

```

package TravelAgency;

import javax.swing.*;

import java.awt.*;

import java.awt.event.*;

public class DefPackTypes extends JFrame {

    private JButton exitButton, backButton, nextButton;

    private JRadioButton international, domestic;

    private ButtonGroup radioButtonGroup;

    private int defPack = 0;

    DefPackTypes() {

        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);

        setTitle("Travel Agency - Choose Tour Plan");

        setSize(1536, 864);

        setLocationRelativeTo(null);

        setResizable(false);

        setLayout(null);

        ImageIcon bgImage = new ImageIcon(getClass().getResource("/International.jpg"));

        Image img = bgImage.getImage();

        Image scaledImg = img.getScaledInstance(this.getWidth(), this.getHeight(),
        Image.SCALE_SMOOTH);

        JLabel backgroundLabel = new JLabel(new ImageIcon(scaledImg));

        backgroundLabel.setBounds(0, 0, this.getWidth(), this.getHeight());

        add(backgroundLabel);
    }

```

```
JLabel label1 = new JLabel("Choose Tour Plan");

label1.setBounds(510, 63, 691, 100);

label1.setFont(new Font("Segoe UI Black", Font.PLAIN, 60));

backgroundLabel.add(label1);


international = new JRadioButton("International");

international.setBounds(230, 169, 423, 139);

international.setFont(new Font("Segoe UI Italic", Font.PLAIN, 50));

international.setOpaque(false);

backgroundLabel.add(international);


domestic = new JRadioButton("Domestic");

domestic.setBounds(1000, 169, 423, 139);

domestic.setFont(new Font("Segoe UI Italic", Font.PLAIN, 50));

domestic.setOpaque(false);

backgroundLabel.add(domestic);


radioButtonGroup = new ButtonGroup();

radioButtonGroup.add(international);

radioButtonGroup.add(domestic);


exitButton = new JButton("Exit");

exitButton.setBounds(200, 728, 215, 50);

exitButton.setFont(new Font("Segoe UI Black", Font.PLAIN, 35));

exitButton.setForeground(Color.WHITE);

exitButton.setBackground(Color.decode("#C00000"));
```

```
backgroundLabel.add(exitButton);
```

```
backButton = new JButton("Back");
```

```
backButton.setBounds(700, 728, 215, 50);
```

```
backButton.setFont(new Font("Segoe UI Black", Font.PLAIN, 35));
```

```
backButton.setForeground(Color.WHITE);
```

```
backButton.setBackground(Color.decode("#2E75B6"));
```

```
backgroundLabel.add(backButton);
```

```
nextButton = new JButton("Next");
```

```
nextButton.setBounds(1200, 730, 215, 50);
```

```
nextButton.setFont(new Font("Segoe UI Black", Font.PLAIN, 35));
```

```
nextButton.setForeground(Color.WHITE);
```

```
nextButton.setBackground(Color.decode("#2E75B6"));
```

```
backgroundLabel.add(nextButton);
```

```
Handler handler = new Handler();
```

```
international.addActionListener(handler);
```

```
domestic.addActionListener(handler);
```

```
exitButton.addActionListener(handler);
```

```
backButton.addActionListener(handler);
```

```
nextButton.addActionListener(handler);
```

```
}
```

```
class Handler implements ActionListener {
```

```
    @Override
```

```
    public void actionPerformed(ActionEvent e) {
```

```

if (e.getSource() == backButton) {

    new Packs().setVisible(true);

    setVisible(false);

} else if (e.getSource() == nextButton) {

    if (defPack == 1) {

        new IntCountries().setVisible(true);

        setVisible(false);

    } else if (defPack == 2) {

        new DomPlaces().setVisible(true);

        setVisible(false);

    } else {

        JOptionPane.showMessageDialog(null, "Please select a package type!", "Error",
JOptionPane.ERROR_MESSAGE);

    }

} else if (e.getSource() == exitButton) {

    System.exit(0);

} else if (e.getSource() == international) {

    defPack = 1;

} else if (e.getSource() == domestic) {

    defPack = 2;

}

}

}

```

```

public static void main(String[] args) {

    SwingUtilities.invokeLater(() -> {

        DefPackTypes frame = new DefPackTypes();

```

```

        frame.setVisible(true);

    });

}

}

```

7. DomesticShillong.java

```

package TravelAgency;

import javax.swing.*.*;
import java.awt.*.*;

public class DomesticShillong extends JFrame {

    private ImageIcon background;

    private JLabel label1, label2, label3, label3_1, label3_2, backgroundLabel;

    private Font f1, f2, f3, f4, f5;

    private JButton exitButton, backButton, payButton;

    private Cursor cursor;

    private JRadioButton pack1, pack2, pack3;

    private ButtonGroup radioButtonGroup;

    private int selected = 0;

    DomesticShillong() {

        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);

        setTitle("Travel Agency - Shillong");

        setSize(1536, 864);

        setLocationRelativeTo(null);

        setResizable(false);

        setLayout(null);
    }
}

```



```
background = new ImageIcon(getClass().getResource("/Shillong.jpg"));
backgroundLabel = new JLabel(background);
backgroundLabel.setBounds(0, 0, getWidth(), getHeight());
add(backgroundLabel);
```

```
f1 = new Font("Segoe UI Black", Font.PLAIN, 50);
f2 = new Font("Segoe UI", Font.BOLD, 30);
f3 = new Font("Segoe UI", Font.BOLD, 42);
f4 = new Font("Segoe UI", Font.BOLD, 40);
f5 = new Font("Segoe UI Black", Font.PLAIN, 28);
cursor = new Cursor(Cursor.HAND_CURSOR);
```

```
label1 = new JLabel("3 Packs Available for Shillong Tour!");
label1.setBounds(100, 34, 1324, 70);
label1.setFont(f1);
label1.setHorizontalAlignment(JLabel.CENTER);
backgroundLabel.add(label1);
```

```
label2 = new JLabel("Select One:");
label2.setBounds(671, 97, 500, 50);
label2.setFont(f2);
backgroundLabel.add(label2);
```

```
pack1 = new JRadioButton("Pack 1");
pack1.setBounds(115, 200, 168, 50);
pack1.setFont(f3);
pack1.setBackground(Color.decode("#F2F2F2"));
```

```

pack1.setCursor(cursor);

backgroundLabel.add(pack1);

label3 = new JLabel("<html>* First Class<br>* 5 Star Hotel<br>* By Flight<br>* 6  
Persons<br>* 7 Days<br>* Cost: ₹90,000</html>");

label3.setForeground(Color.WHITE);

label3.setBounds(100, 260, 412, 324);

label3.setFont(f4);

backgroundLabel.add(label3);


pack2 = new JRadioButton("Pack 2");

pack2.setBounds(674, 200, 178, 50);

pack2.setFont(f3);

pack2.setBackground(Color.decode("#F2F2F2"));

pack2.setCursor(cursor);

backgroundLabel.add(pack2);

label3_1 = new JLabel("<html>* Second Class<br>* 4 Star Hotel<br>* By Train<br>* 4  
Persons<br>* 5 Days<br>* Cost: ₹60,000</html>");

label3_1.setForeground(Color.WHITE);

label3_1.setBounds(640, 260, 412, 324);

label3_1.setFont(f4);

backgroundLabel.add(label3_1);


pack3 = new JRadioButton("Pack 3");

pack3.setBounds(1145, 200, 168, 50);

pack3.setFont(f3);

pack3.setBackground(Color.decode("#F2F2F2"));

pack3.setCursor(cursor);

backgroundLabel.add(pack3);

```

```
label3_2 = new JLabel("<html>* Third Class<br>* 3 Star Hotel<br>* By Bus<br>* 2  
Persons<br>* 3 Days<br>* Cost: ₹30,000</html>");
```

```
label3_2.setForeground(Color.WHITE);
```

```
label3_2.setBounds(1110, 260, 412, 324);
```

```
label3_2.setFont(f4);
```

```
backgroundLabel.add(label3_2);
```

```
radioButtonGroup = new ButtonGroup();
```

```
radioButtonGroup.add(pack1);
```

```
radioButtonGroup.add(pack2);
```

```
radioButtonGroup.add(pack3);
```

```
exitButton = new JButton("Exit");
```

```
exitButton.setBounds(30, 650, 300, 70);
```

```
exitButton.setFont(f5);
```

```
exitButton.setCursor(cursor);
```

```
exitButton.setForeground(Color.WHITE);
```

```
exitButton.setBackground(Color.decode("#C00000"));
```

```
backgroundLabel.add(exitButton);
```

```
backButton = new JButton("Back");
```

```
backButton.setBounds(1150, 650, 300, 70);
```

```
backButton.setFont(f5);
```

```
backButton.setCursor(cursor);
```

```
backButton.setForeground(Color.WHITE);
```

```
backButton.setBackground(Color.decode("#2E75B6"));
```

```
backgroundLabel.add(backButton);
```

```

payButton = new JButton("Pay");

payButton.setBounds(630, 650, 300, 70);

payButton.setFont(f5);

payButton.setCursor(cursor);

payButton.setForeground(Color.WHITE);

payButton.setBackground(Color.decode("#2E75B6"));

backgroundLabel.add(payButton);


exitButton.addActionListener(e -> System.exit(0));


backButton.addActionListener(e -> {

    setVisible(false);

    new DomPlaces().setVisible(true);

});


payButton.addActionListener(e -> {

    selected = pack1.isSelected() ? 1 : pack2.isSelected() ? 2 : pack3.isSelected() ? 3 : 0;

    if (selected != 0) {

        setVisible(false);

        new Payment().setVisible(true);

    } else {

        JOptionPane.showMessageDialog(null, "Please select a package before proceeding!",
"Warning!", JOptionPane.WARNING_MESSAGE);

    }

});

}

```

```

public static void main(String[] args) {

    SwingUtilities.invokeLater(() -> {

        DomesticShillong frame = new DomesticShillong();

        frame.setVisible(true);

    });

}
}

```

8. InternationalSouthKorea.java

```

package TravelAgency;

import javax.swing.*;
import java.awt.*;
import java.awt.event.*;

public class InternationalSouthKorea extends JFrame {

    private ImageIcon background;

    private JLabel label1, backgroundLabel, label3, label3_1, label3_2;

    private JButton exitButton, backButton, payButton;

    private JRadioButton pack1, pack2, pack3;

    private ButtonGroup radioButtonGroup;

    private int selected = 0;

    InternationalSouthKorea() {

        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);

        setTitle("Travel Agency - South Korea");

        setSize(1536, 864);

        setLocationRelativeTo(null);
    }
}

```

```
setResizable(false);
```

```
setLayout(null);
```

```
background = new ImageIcon(new ImageIcon(getClass().getResource("/South  
Korea.jpg")).getImage().getScaledInstance(getWidth(), getHeight(), Image.SCALE_SMOOTH));
```

```
backgroundLabel = new JLabel(background);
```

```
backgroundLabel.setBounds(0, 0, getWidth(), getHeight());
```

```
add(backgroundLabel);
```

```
label1 = new JLabel("3 Packs Available for South Korea Tour!");
```

```
label1.setBounds(83, 25, 1324, 70);
```

```
label1.setFont(new Font("Segoe UI Black", Font.PLAIN, 50));
```

```
label1.setHorizontalAlignment(JLabel.CENTER);
```

```
backgroundLabel.add(label1);
```

```
JLabel label2 = new JLabel("Select One:");
```

```
label2.setBounds(678, 94, 500, 50);
```

```
label2.setFont(new Font("Segoe UI", Font.BOLD, 30));
```

```
backgroundLabel.add(label2);
```

```
pack1 = new JRadioButton("Pack 1");
```

```
pack1.setBounds(115, 200, 168, 50);
```

```
pack1.setFont(new Font("Segoe UI", Font.BOLD, 42));
```

```
backgroundLabel.add(pack1);
```

```
label3 = new JLabel("<html>* First Class<br>* 5 Star Hotel<br>* By Airplane<br>* 6  
Persons<br>* 10 Days<br>* Cost: ₩6,000,000</html>");
```

```
label3.setForeground(Color.WHITE);
```

```
label3.setBounds(100, 260, 412, 324);
```

```

label3.setFont(new Font("Segoe UI", Font.BOLD, 40));

backgroundLabel.add(label3);


pack2 = new JRadioButton("Pack 2");

pack2.setBounds(674, 200, 178, 50);

pack2.setFont(new Font("Segoe UI", Font.BOLD, 42));

backgroundLabel.add(pack2);

label3_1 = new JLabel("<html>* Second Class<br>* 4 Star Hotel<br>* By Airplane<br>* 4  
Persons<br>* 7 Days<br>* Cost: ₩4,500,000</html>");

label3_1.setForeground(Color.WHITE);

label3_1.setBounds(640, 260, 412, 324);

label3_1.setFont(new Font("Segoe UI", Font.BOLD, 40));

backgroundLabel.add(label3_1);


pack3 = new JRadioButton("Pack 3");

pack3.setBounds(1145, 200, 168, 50);

pack3.setFont(new Font("Segoe UI", Font.BOLD, 42));

backgroundLabel.add(pack3);

label3_2 = new JLabel("<html>* Third Class<br>* 3 Star Hotel<br>* By Airplane<br>* 3  
Persons<br>* 5 Days<br>* Cost: ₩3,000,000</html>");

label3_2.setForeground(Color.WHITE);

label3_2.setBounds(1110, 260, 412, 324);

label3_2.setFont(new Font("Segoe UI", Font.BOLD, 40));

backgroundLabel.add(label3_2);


radioButtonGroup = new ButtonGroup();

radioButtonGroup.add(pack1);

radioButtonGroup.add(pack2);

```

```
radioButtonGroup.add(pack3);
```

```
exitButton = new JButton("Exit");
```

```
exitButton.setBounds(30, 650, 300, 70);
```

```
exitButton.setFont(new Font("Segoe UI Black", Font.PLAIN, 28));
```

```
exitButton.setForeground(Color.WHITE);
```

```
exitButton.setBackground(Color.decode("#C00000"));
```

```
backgroundLabel.add(exitButton);
```

```
backButton = new JButton("Back");
```

```
backButton.setBounds(1150, 650, 300, 70);
```

```
backButton.setFont(new Font("Segoe UI Black", Font.PLAIN, 28));
```

```
backButton.setForeground(Color.WHITE);
```

```
backButton.setBackground(Color.decode("#2E75B6"));
```

```
backgroundLabel.add(backButton);
```

```
payButton = new JButton("Pay");
```

```
payButton.setBounds(630, 650, 300, 70);
```

```
payButton.setFont(new Font("Segoe UI Black", Font.PLAIN, 28));
```

```
payButton.setForeground(Color.WHITE);
```

```
payButton.setBackground(Color.decode("#2E75B6"));
```

```
backgroundLabel.add(payButton);
```

```
exitButton.addActionListener(e -> System.exit(0));
```

```
backButton.addActionListener(e -> {
```

```
    setVisible(false);
```



```

        new IntCountries().setVisible(true);
    });

    payButton.addActionListener(e -> {
        if (pack1.isSelected()) selected = 1;
        else if (pack2.isSelected()) selected = 2;
        else if (pack3.isSelected()) selected = 3;

        if (selected != 0) {
            setVisible(false);
            new Payment().setVisible(true);
        } else {
            JOptionPane.showMessageDialog(null, "Please select a package before proceeding!",
"Warning!", JOptionPane.WARNING_MESSAGE);
        }
    });
}

```

```

public static void main(String[] args) {
    SwingUtilities.invokeLater(() -> new InternationalSouthKorea().setVisible(true));
}
}

```

9.SelfChosenPacks.java

```

package TravelAgency;

import java.awt.*;

import javax.swing.*;

import java.awt.event.*;

```

```

public class SelfChosenPacks extends JFrame {

    private JLabel label0, label1, totalCostLabel, backgroundLabel;

    private Font f1, f2, f3, f4, f5;

    private JComboBox<String> cb1, cb2, cb3, cb4, cb5, cb6, cb7;

    private JButton btn1, btn2, btn3, btn4;

    private Cursor cursor;

    private double totalCost = 0;

    SelfChosenPacks() {

        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);

        setTitle("Travel Agency - Registration by admin");

        setSize(1536, 864);

        setLocationRelativeTo(null);

        setResizable(false);

        setLayout(null);

        ImageIcon bgImage = new ImageIcon(getClass().getResource("/Admin.jpg"));

        backgroundLabel = new JLabel(new
        ImageIcon(bgImage.getImage().getScaledInstance(this.getWidth(), this.getHeight(),
        Image.SCALE_SMOOTH)));

        backgroundLabel.setBounds(0, 0, 1536, 864);

        add(backgroundLabel);

        f1 = new Font("Segoe UI Black", Font.PLAIN, 35);

        f2 = new Font("Segoe UI Black", Font.PLAIN, 25);

        f3 = new Font("Segoe UI Semibold", Font.PLAIN, 35);

```

```
f4 = new Font("Segoe UI", Font.PLAIN, 25);  
f5 = new Font("Segoe UI", Font.PLAIN, 19);  
  
cursor = new Cursor(Cursor.HAND_CURSOR);  
  
label1 = createLabel("Select Tour Details", 482, 24, f1);  
backgroundLabel.add(label1);  
  
JLabel tourTypeLabel = createLabel("Tour Type", 430, 75, f4);  
backgroundLabel.add(tourTypeLabel);  
  
String[] cb1Str = { "Choose tour type...", "Domestic", "International" };  
cb1 = createComboBox(cb1Str, 580, 84);  
backgroundLabel.add(cb1);  
  
label0 = createLabel("Destination", 430, 140, f4);  
backgroundLabel.add(label0);  
  
String[] cb2Str = { "Choose tour type first!" };  
cb2 = createComboBox(cb2Str, 580, 150);  
cb2.setEnabled(false);  
backgroundLabel.add(cb2);  
  
JLabel personLabel = createLabel("Person", 430, 200, f4);  
backgroundLabel.add(personLabel);  
  
String[] cb3Str = { "Total person number...", "1", "2", "3", "4", "5" };
```

```
cb3 = createComboBox(cb3Str, 580, 210);
```

```
backgroundLabel.add(cb3);
```

```
JLabel daysLabel = createLabel("Total Days", 430, 300, f4);
```

```
backgroundLabel.add(daysLabel);
```

```
String[] cb4Str = { "Number of days...", "1", "2", "3", "4", "5", "6", "7" };
```

```
cb4 = createComboBox(cb4Str, 580, 310);
```

```
backgroundLabel.add(cb4);
```

```
JLabel hotelLabel = createLabel("Hotel Type", 430, 400, f4);
```

```
backgroundLabel.add(hotelLabel);
```

```
String[] cb5Str = { "Select tour type first!" };
```

```
cb5 = createComboBox(cb5Str, 580, 410);
```

```
cb5.setEnabled(false);
```

```
backgroundLabel.add(cb5);
```

```
JLabel travelByLabel = createLabel("Travel By", 430, 500, f4);
```

```
backgroundLabel.add(travelByLabel);
```

```
String[] cb6Str = { "Select tour type first!" };
```

```
cb6 = createComboBox(cb6Str, 580, 510);
```

```
cb6.setEnabled(false);
```

```
backgroundLabel.add(cb6);
```

```
JLabel vehicleLabel = createLabel("Vehicle Type", 430, 600, f4);
```

```
backgroundLabel.add(vehicleLabel);
```

```
String[] cb7Str = { "Select vehicle first!" };
```

```
cb7 = createComboBox(cb7Str, 580, 610);
```

```
cb7.setEnabled(false);
```

```
backgroundLabel.add(cb7);
```

```
totalCostLabel = createLabel("Total Cost: $0", 450, 650, f3);
```

```
backgroundLabel.add(totalCostLabel);
```

```
btn1 = createButton("Exit", 58, 740, "#C00000", e -> System.exit(0));
```

```
backgroundLabel.add(btn1);
```

```
btn2 = createButton("Back", 387, 740, "#2E75B6", e -> {
```

```
    Packs frame = new Packs();
```

```
    frame.setVisible(true);
```

```
    setVisible(false);
```

```
});
```

```
backgroundLabel.add(btn2);
```

```
btn3 = createButton("Next", 1226, 740, "#2E75B6", e -> {
```

```
    totalCost = calculateCost();
```

```
    totalCostLabel.setText("Total Cost: $" + totalCost);
```

```
    JOptionPane.showMessageDialog(this, "Total Cost is: $" + totalCost, "Total Cost",
JOptionPane.INFORMATION_MESSAGE);
```

```
    if (totalCost != 0) {
```

```

        setVisible(false);

        Payment frame = new Payment();

        frame.setVisible(true);

    } else {

        JOptionPane.showMessageDialog(this, "Please select a package before proceeding!",
"Warning!", JOptionPane.WARNING_MESSAGE);

    }

});

backgroundLabel.add(btn3);


btn4 = createButton("Reset", 784, 740, "#2E75B6", e -> resetForm());

backgroundLabel.add(btn4);


cb1.addActionListener(ae -> handleTourTypeSelection());


cb2.addActionListener(ae -> {

    if (cb1.getSelectedIndex() == 1 || cb1.getSelectedIndex() == 2) {

        cb5.setEnabled(true);

        cb6.setEnabled(true);

    }

});


cb6.addActionListener(ae -> cb7.setEnabled(cb6.getSelectedIndex() != 0));

}


private JLabel createLabel(String text, int x, int y, Font font) {

    JLabel label = new JLabel(text);

```

```

    label.setBounds(x, y, 500, 50);

    label.setFont(font);

    label.setForeground(Color.WHITE);

    return label;
}

private JComboBox<String> createComboBox(String[] items, int x, int y) {

    JComboBox<String> comboBox = new JComboBox<>(items);

    comboBox.setBounds(x, y, 250, 30);

    comboBox.setFont(f5);

    comboBox.setBackground(Color.WHITE);

    return comboBox;
}

private JButton createButton(String text, int x, int y, String colorHex, ActionListener action) {

    JButton button = new JButton(text);

    button.setBounds(x, y, 183, 50);

    button.setFont(f2);

    button.setCursor(cursor);

    button.setForeground(Color.WHITE);

    button.setBackground(Color.decode(colorHex));

    button.addActionListener(action);

    return button;
}

private void handleTourTypeSelection() {

    int s = cb1.getSelectedIndex();

```

```

cb2.removeAllItems();

if (s == 1) {

    label0.setText("State");

    cb2.addItem("Select State...");

    cb2.addItem("Tamil Nadu");

    cb2.addItem("Goa");

    cb2.addItem("Shillong");

    cb2.addItem("Andaman & Nicobar");

    cb2.addItem("Kerala");

    cb2.setEnabled(true);

    configureForDomestic();

} else if (s == 2) {

    label0.setText("Country");

    cb2.addItem("Select Country...");

    cb2.addItem("United States");

    cb2.addItem("United Kingdom");

    cb2.addItem("Thailand");

    cb2.addItem("Japan");

    cb2.addItem("South Korea");

    cb2.setEnabled(true);

    configureForInternational();

}

}

```

```

private void configureForDomestic() {

    cb5.removeAllItems();

    cb5.addItem("Hotel Type...");

```



```

        cb5.addItem("Luxury");
        cb5.addItem("Budget");
        cb5.addItem("Standard");
        cb5.setEnabled(true);

        cb6.removeAllItems();
        cb6.addItem("Travel By...");
        cb6.addItem("Train");
        cb6.addItem("Bus");
        cb6.addItem("Flight");
        cb6.setEnabled(true);

        cb7.removeAllItems();
        cb7.addItem("Vehicle Type...");
        cb7.addItem("Car");
        cb7.addItem("Bike");
        cb7.setEnabled(true);
    }

    private void configureForInternational() {
        cb5.removeAllItems();
        cb5.addItem("Hotel Type...");
        cb5.addItem("Luxury");
        cb5.addItem("Budget");
        cb5.addItem("Standard");
        cb5.setEnabled(true);
    }

```

```
        cb6.removeAllItems();

        cb6.addItem("Travel By...");

        cb6.addItem("Flight");

        cb6.addItem("Cruise");

        cb6.addItem("Bus");

        cb6.setEnabled(true);


        cb7.removeAllItems();

        cb7.addItem("Vehicle Type...");

        cb7.addItem("Luxury Car");

        cb7.addItem("Economy Car");

        cb7.setEnabled(true);
    }


    private void resetForm() {

        cb1.setSelectedIndex(0);

        cb2.setSelectedIndex(0);

        cb3.setSelectedIndex(0);

        cb4.setSelectedIndex(0);

        cb5.setSelectedIndex(0);

        cb6.setSelectedIndex(0);

        cb7.setSelectedIndex(0);

        totalCost = 0;

        totalCostLabel.setText("Total Cost: $" + totalCost);
    }


    private double calculateCost() {
```

```

int tourType = cb1.getSelectedIndex();

int destination = cb2.getSelectedIndex();

int person = cb3.getSelectedIndex();

int days = cb4.getSelectedIndex();

int hotel = cb5.getSelectedIndex();

int travelBy = cb6.getSelectedIndex();

int vehicle = cb7.getSelectedIndex();


    if (tourType > 0 && destination > 0 && person > 0 && days > 0 && hotel > 0 && travelBy > 0
    && vehicle > 0) {

        totalCost = (person * days * 5000) + (hotel * 10000) + (travelBy * 8000) + (vehicle * 2000);

    } else {

        totalCost = 0;

    }

    return totalCost;

}


public static void main(String[] args) {

    SwingUtilities.invokeLater(() -> new SelfChoosenPacks().setVisible(true));

}

}

10.IntCountries.java

package TravelAgency;

import javax.swing.*;

import java.awt.*;


public class IntCountries extends JFrame {

```

```

private JLabel label1, backgroundLabel;

private Font f1, f3;

private JButton btnExit, btnBack, btnNext;

private Cursor cursor;

private JRadioButton country1, country2, country3, country4, country5;

private ButtonGroup buttonGroup;

private int country = 0;

IntCountries() {

    setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);

    setTitle("Travel Agency");

    setSize(1536, 864);

    setLocationRelativeTo(null);

    setResizable(false);

    setLayout(null);

    backgroundLabel = new JLabel(new ImageIcon(getScaledBackground()));

    backgroundLabel.setBounds(0, 0, getWidth(), getHeight());

    add(backgroundLabel);

    f1 = new Font("Segoe UI Black", Font.PLAIN, 45);

    f3 = new Font("Segoe UI Black", Font.PLAIN, 25);

    cursor = new Cursor(Cursor.HAND_CURSOR);

    label1 = new JLabel("Which Country Would You Like To Visit?", SwingConstants.CENTER);

    label1.setForeground(Color.WHITE);

    label1.setBounds(150, 29, 1166, 50);

```

```

label1.setFont(f1);

backgroundLabel.add(label1);


country1 = createRadioButton("United States", 100, 1);
country2 = createRadioButton("Thailand", 200, 2);
country3 = createRadioButton("United Kingdom", 300, 3);
country4 = createRadioButton("Japan", 400, 4);
country5 = createRadioButton("South Korea", 500, 5);


buttonGroup = new ButtonGroup();
buttonGroup.add(country1);
buttonGroup.add(country2);
buttonGroup.add(country3);
buttonGroup.add(country4);
buttonGroup.add(country5);


btnExit = createButton("Exit", 80, 700, Color.decode("#C00000"));
btnBack = createButton("Back", 650, 700, Color.decode("#2E75B6"));
btnNext = createButton("Next", 1250, 700, Color.decode("#2E75B6"));


btnExit.addActionListener(e -> System.exit(0));
btnBack.addActionListener(e -> openFrame(new DefPackTypes()));
btnNext.addActionListener(e -> handleNextAction());


backgroundLabel.add(country1);
backgroundLabel.add(country2);
backgroundLabel.add(country3);

```

```

backgroundLabel.add(country4);

backgroundLabel.add(country5);

backgroundLabel.add(btnExit);

backgroundLabel.add(btnBack);

backgroundLabel.add(btnNext);

}

```

```

private Image getScaledBackground() {

    ImageIcon bgImage = new ImageIcon(getClass().getResource("/DefPacks.jpg"));

    return bgImage.getImage().getScaledInstance(getWidth(), getHeight(),
Image.SCALE_SMOOTH);

}

```

```

private JRadioButton createRadioButton(String text, int yPos, int countryIndex) {

    JRadioButton radioButton = new JRadioButton(text);

    radioButton.setForeground(Color.WHITE);

    radioButton.setBounds(460, yPos, 400, 50);

    radioButton.setFont(new Font("Segoe UI", Font.BOLD, 45));

    radioButton.setOpaque(false);

    radioButton.setCursor(cursor);

    radioButton.addActionListener(e -> country = countryIndex);

    return radioButton;

}

```

```

private JButton createButton(String text, int xPos, int yPos, Color bgColor) {

    JButton button = new JButton(text);

    button.setBounds(xPos, yPos, 215, 50);

}

```

```

        button.setFont(f3);

        button.setCursor(cursor);

        button.setForeground(Color.WHITE);

        button.setBackground(bgColor);

        return button;
    }

```

```

private void handleNextAction() {

    switch (country) {

        case 1: openFrame(new InternationalUnitedStates()); break;

        case 2: openFrame(new InternationalThailand()); break;

        case 3: openFrame(new InternationalUnitedKingdom()); break;

        case 4: openFrame(new InternationalJapan()); break;

        case 5: openFrame(new InternationalSouthKorea()); break;

        default: JOptionPane.showMessageDialog(null, "Please select a country.", "Warning!!!",
JOptionPane.WARNING_MESSAGE);

    }

}

```

```

private void openFrame(JFrame frame) {

    setVisible(false);

    frame.setVisible(true);

}

```

```

public static void main(String[] args) {

    SwingUtilities.invokeLater(() -> new IntCountries().setVisible(true));

}

```

```
}
```

```
11.Payment.java
```

```
package TravelAgency;
```

```
import java.awt.*;
```

```
import javax.swing.*;
```

```
import java.awt.event.*;
```

```
import java.util.Date;
```

```
public class Payment extends JFrame {
```

```
    private ImageIcon icon, background;
```

```
    private JLabel titleLabel, nameOnCardLabel, cardNumberLabel, cvvCodeLabel, validOnLabel;
```

```
    private JLabel acceptLabel, cardDetailsLabel, imgLabel, backgroundLabel, travelDateLabel;
```

```
    private JTextField nameField, cardNumberField, validOnField;
```

```
    private JPasswordField cvvField;
```

```
    private JSpinner travelDateSpinner;
```

```
    private JButton exitButton, backButton, confirmButton;
```

```
    private Cursor handCursor;
```

```
    public Payment() {
```

```
        this.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
```

```
        this.setTitle("Travel Agency");
```

```
        this.setSize(1536, 864);
```

```
        this.setLocationRelativeTo(null);
```

```
        this.setResizable(false);
```

```
        getContentPane().setLayout(null);
```



```

ImageIcon bgImage = new ImageIcon(getClass().getResource("/Register.jpg"));

Image scaledBg = bgImage.getImage().getScaledInstance(this.getWidth(), this.getHeight(),
Image.SCALE_SMOOTH);

background = new ImageIcon(scaledBg);

backgroundLabel = new JLabel(background);

backgroundLabel.setBounds(0, 0, this.getWidth(), this.getHeight());

getContentPane().add(backgroundLabel);


ImageIcon visaImage = new ImageIcon(getClass().getResource("/VisaCard.png"));

imgLabel = new JLabel(visaImage);

imgLabel.setBounds(630, 120, visaImage.getIconWidth(), visaImage.getIconHeight());

backgroundLabel.add(imgLabel);


ImageIcon masterCardImage = new ImageIcon(getClass().getResource("/MasterCard.png"));

imgLabel = new JLabel(masterCardImage);

imgLabel.setBounds(720, 120, masterCardImage.getIconWidth(),
masterCardImage.getIconHeight());

backgroundLabel.add(imgLabel);


Font titleFont = new Font("Segoe UI Black", Font.PLAIN, 30);

Font labelFont = new Font("Segoe UI Semibold", Font.PLAIN, 25);

Font fieldFont = new Font("Segoe UI", Font.PLAIN, 20);

Font smallFont = new Font("Segoe UI", Font.PLAIN, 15);

titleLabel = new JLabel("Complete Your Payment");

titleLabel.setBounds(500, 20, 500, 50);

titleLabel.setFont(titleFont);

```

```
backgroundLabel.add(titleLabel);
```

```
handCursor = new Cursor(Cursor.HAND_CURSOR);
```

```
exitButton = createButton("Exit", 150, 700, handCursor, Color.WHITE,  
Color.decode("#C00000"), titleFont);
```

```
backgroundLabel.add(exitButton);
```

```
backButton = createButton("Back", 700, 700, handCursor, Color.WHITE,  
Color.decode("#2E75B6"), titleFont);
```

```
backgroundLabel.add(backButton);
```

```
confirmButton = createButton("Confirm Payment", 1150, 700, handCursor, Color.WHITE,  
Color.decode("#2E75B6"), titleFont);
```

```
confirmButton.setBounds(1150, 700, 350, 50);
```

```
backgroundLabel.add(confirmButton);
```

```
travelDateLabel = new JLabel("Travel Date");
```

```
travelDateLabel.setBounds(409, 480, 500, 50);
```

```
travelDateLabel.setFont(fieldFont);
```

```
backgroundLabel.add(travelDateLabel);
```

```
travelDateSpinner = new JSpinner(new SpinnerDateModel());
```

```
JSpinner.DateEditor dateEditor = new JSpinner.DateEditor(travelDateSpinner, "yyyy-MM-dd");
```

```
travelDateSpinner.setEditor(dateEditor);
```

```
travelDateSpinner.setBounds(600, 493, 120, 30);
```

```
travelDateSpinner.setFont(smallFont);
```

```
backgroundLabel.add(travelDateSpinner);
```

```
validOnLabel = new JLabel("Valid On");  
validOnLabel.setBounds(387, 536, 500, 50);  
validOnLabel.setFont(fieldFont);  
backgroundLabel.add(validOnLabel);  
  
validOnField = createTextField(541, 549, 90, 30, smallFont);  
backgroundLabel.add(validOnField);  
  
cvvCodeLabel = new JLabel("CVV Code");  
cvvCodeLabel.setBounds(672, 536, 500, 50);  
cvvCodeLabel.setFont(fieldFont);  
backgroundLabel.add(cvvCodeLabel);  
  
cvvField = new JPasswordField();  
cvvField.setBounds(776, 550, 80, 30);  
cvvField.setFont(smallFont);  
cvvField.setHorizontalAlignment(JPasswordField.CENTER);  
cvvField.setEchoChar('*');  
backgroundLabel.add(cvvField);  
  
cardNumberLabel = new JLabel("Card Number");  
cardNumberLabel.setBounds(409, 418, 500, 50);  
cardNumberLabel.setFont(fieldFont);  
backgroundLabel.add(cardNumberLabel);  
  
cardNumberField = createTextField(600, 431, 235, 30, smallFont);
```

```
backgroundLabel.add(cardNumberField);
```

```
nameOnCardLabel = new JLabel("Name On Card");
```

```
nameOnCardLabel.setBounds(409, 358, 500, 50);
```

```
nameOnCardLabel.setFont(fieldFont);
```

```
backgroundLabel.add(nameOnCardLabel);
```

```
nameField = createTextField(600, 371, 235, 30, smallFont);
```

```
backgroundLabel.add(nameField);
```

```
cardDetailsLabel = new JLabel("Card Details");
```

```
cardDetailsLabel.setBounds(409, 235, 500, 50);
```

```
cardDetailsLabel.setFont(labelFont);
```

```
backgroundLabel.add(cardDetailsLabel);
```

```
acceptLabel = new JLabel("We Accept Only");
```

```
acceptLabel.setBounds(387, 116, 500, 50);
```

```
acceptLabel.setFont(labelFont);
```

```
backgroundLabel.add(acceptLabel);
```

```
exitButton.addActionListener(e -> System.exit(0));
```

```
confirmButton.addActionListener(e -> {
```

```
    try {
```

```
        validateFields();
```

```
        Date travelDate = (Date) travelDateSpinner.getValue();
```

```
        String ticketNumber = generateTicketNumber(); // Generate or retrieve a ticket number
```

```
        PaySuccess successFrame = new PaySuccess(travelDate.toString(), ticketNumber);
```

```

        successFrame.setVisible(true);

        this.dispose();

    } catch (InvalidYearException | InvalidCVVException | InvalidCardNumberException ex) {

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

    } catch (Exception ex) {

        JOptionPane.showMessageDialog(null, "An unexpected error occurred.", "Error",
JOptionPane.ERROR_MESSAGE);

    }

});

}

```

```

private JButton createButton(String text, int x, int y, Cursor cursor, Color fgColor, Color bgColor,
Font font) {

```

```

    JButton button = new JButton(text);

    button.setBounds(x, y, 190, 50);

    button.setFont(font);

    button.setCursor(cursor);

    button.setForeground(fgColor);

    button.setBackground(bgColor);

    return button;

}

```

```

private JTextField createTextField(int x, int y, int width, int height, Font font) {

```

```

    JTextField textField = new JTextField();

    textField.setBounds(x, y, width, height);

    textField.setFont(font);

    textField.setHorizontalAlignment(JTextField.CENTER);

```

```

    return textField;
}

```

```

private boolean areFieldsEmpty(JTextField... fields) {
    for (JTextField field : fields) {
        if (field.getText().isEmpty()) {
            return true; }
    }
    return false;
}

```

```

private void validateFields() throws InvalidYearException, InvalidCVVException,
InvalidCardNumberException {
    if (areFieldsEmpty(nameField, cardNumberField, validOnField, cvvField)) {
        throw new IllegalArgumentException("Please fill all of the fields.");
    }

    String validOnText = validOnField.getText();
    if (validOnText.length() != 4 || !validOnText.matches("\\d{4}")) {
        throw new InvalidYearException("Valid On must be a valid year (4 digits).");
    }

    String cvvText = new String(cvvField.getPassword());
    if (cvvText.length() != 3 || !cvvText.matches("\\d{3}")) {
        throw new InvalidCVVException("CVV must be a 3-digit number.");
    }

    String cardNumberText = cardNumberField.getText();
    if (cardNumberText.length() != 16 || !cardNumberText.matches("\\d{16}")) {
        throw new InvalidCardNumberException("Card number must be a 16-digit number.");
    }
}

```

```

    }
}

private String generateTicketNumber() {

    return "TK" + System.currentTimeMillis();

}

class InvalidYearException extends Exception {

    public InvalidYearException(String message) {

        super(message);

    }

}

class InvalidCVVException extends Exception {

    public InvalidCVVException(String message) {

        super(message); }

}

class InvalidCardNumberException extends Exception {

    public InvalidCardNumberException(String message) {

        super(message);}

}

public static void main(String[] args) {

    Payment paymentFrame = new Payment();

    paymentFrame.setVisible(true);

}

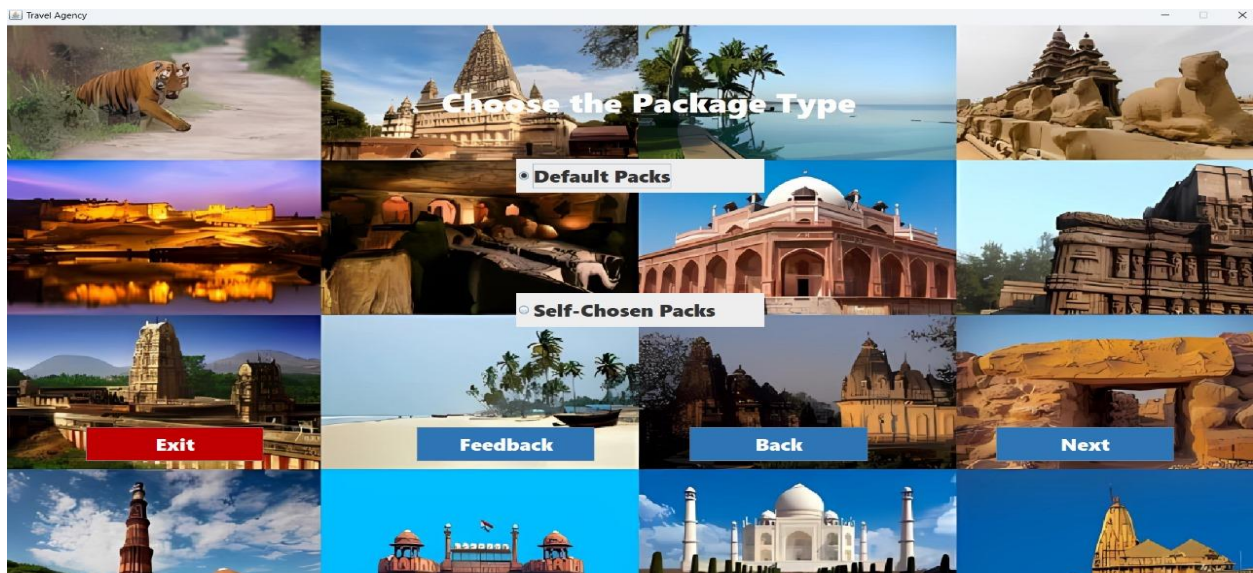
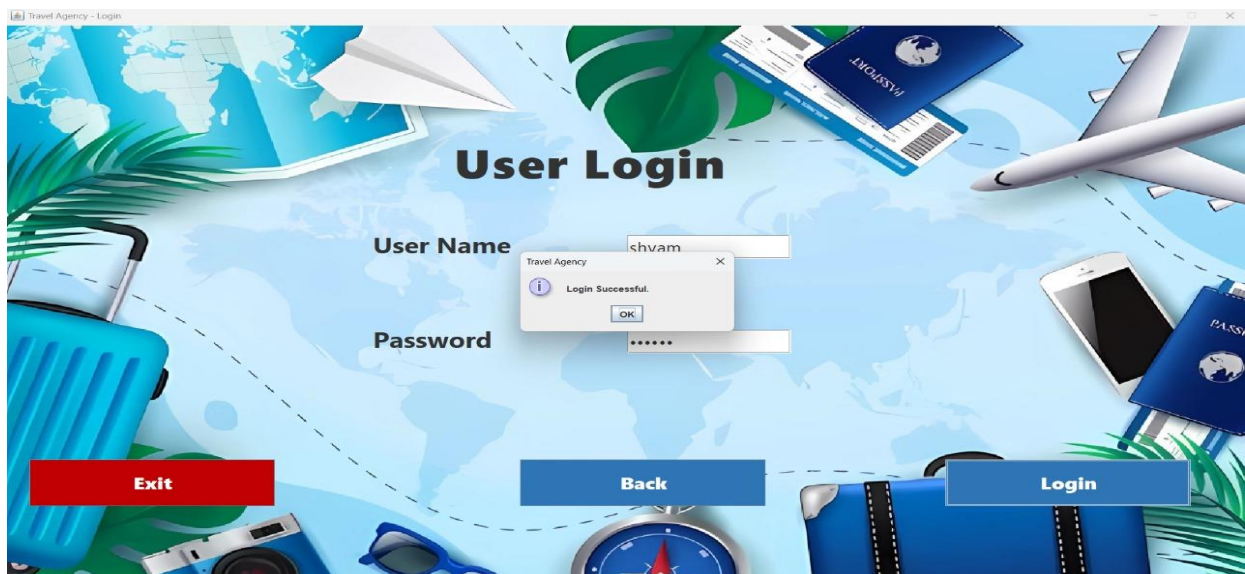
}

```

CHAPTER 6

SCREENSHOTS





Travel Agency - Registration by admin

Select Tour Details

Tour Type

State

Person

Total Days

Hotel Type

Travel By

Vehicle Type

Total Cost: \$1150.0

Exit **Back** **Reset** **Next**



Total Cost
Total Cost is: \$1150.0
OK





Travel Agency

Complete Your Payment

We Accept Only  

Card Details

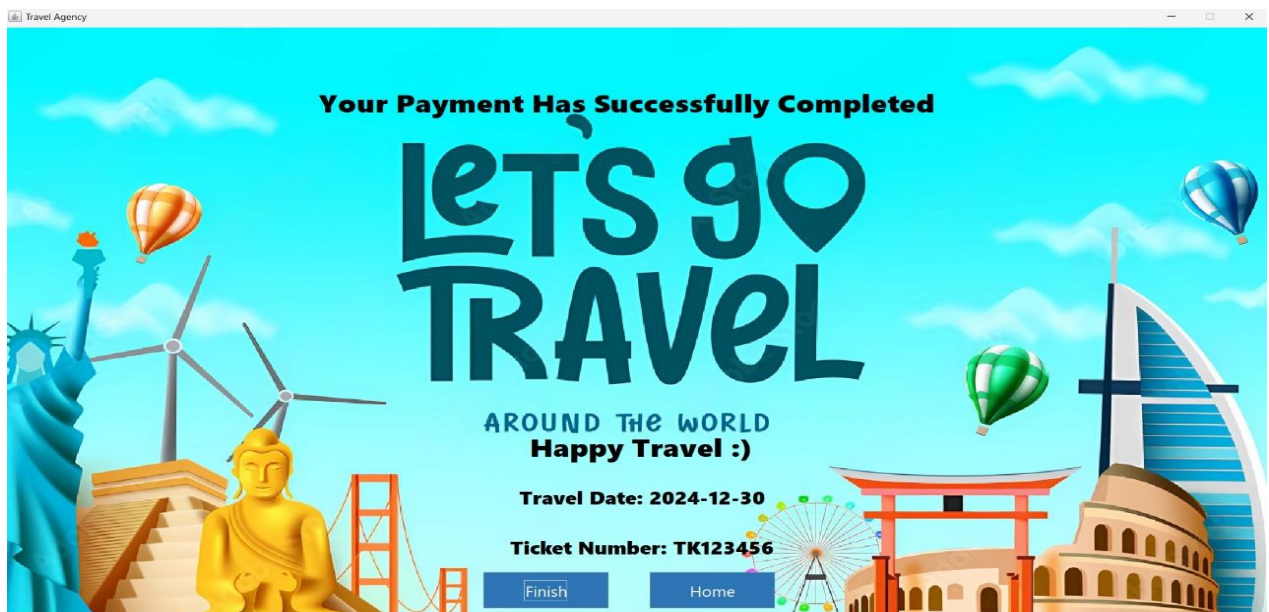
Name On Card

Card Number

Travel Date

Valid On CVV Code

[Exit](#) [Back](#) [Confirm Payment](#)



REFERENCES

1. Anita Seth and B.L. Juneja, 'Java One Step Ahead' (2017) – Oxford University Press
2. 'The Complete Reference Java' , 8th Edition, Herbert Schildt, McGraw Hill Publications, India.
3. 'Core Java Volume I Fundamentals', 12th Edition, Cay S. Horstmann by Oracle Press.
4. 'Core Java Volume II Advanced Features', 12th Edition, Cay S. Horstmann by Oracle Press.