A blue and red logo

Description generated with high confidence

University of Dhaka

Department of Computer Science & Engineering

2nd Year 1st Semester

Session: 2022-23

**CSE – 2112: Object Oriented Programming Lab**

**Project title: Travel and Tourism Management System**

Submitted to

**Dr. Chowdhury Farhan Ahmed**

Professor, Dept. of CSE

**Redwan Ahmed Rizvee,**

Lecturer, Dept. of CSE

Submitted by

Md. Sarif - 14

Md. Mahmudur Rahman Moin – 30

**Table of Contents**

1. **Introduction**
   1. Introduction
   2. Motivation
   3. Objectives
   4. Features
   5. Technology Used
2. **Design and Implementation**
   1. UX Flow
   2. UML Diagram
   3. UI
   4. System Design
   5. System Flow Diagram
   6. Detailed Class Description
   7. Discussion
   8. Use of OOP
3. **Conclusion**
   1. Challenges and Solutions
   2. Future Enhancement
   3. Reference
   4. Contribution

***Project Title: Travel and Tourism Management System***

* 1. **Introduction**

The **Travel and Tourism Management System** is an Object-Oriented Programming (OOP) project designed to automate and manage travel – related services such as tour bookings, hotel reservations, transportations and customer management.

This system aims to enhance the experience of travelers by providing an efficient and user-friendly platform for planning and managing trips.

* 1. **Motivation**

Traveling is an essential part of life, whether for leisure, business, or exploration. However, planning a trip can be a time-consuming and complex process involving multiple components such as booking hotels, transportation, and sightseeing. Traditional methods often lead to inefficiencies, mismanagement, and inconvenience for travelers.

This project is motivated by the need for a streamlined, automated system that simplifies the booking process, ensures better organization, and enhances customer satisfaction. By integrating various travel services into a single platform, this system aims to provide an efficient, accessible, and secure solution for travelers and travel agencies alike.

Furthermore, the system can encourage huge community of local people to connect with the digital world in an easier way.

* 1. **Objectives**
* To automate the process of booking and managing tours.
* To provide an easy-to-use interface for Customers.
* To improve operational efficiency in the travel industry.
* To maintain an organized database of customers, hotels, transport, and bookings.
  1. **Features**
  + Spectacular interfaces including colors and images
  + Splash window at intro
  + Phone number and Email in Sign up menu
  + Password recovery system using phone number and email
  + Loading screen after logging in
  + Dashboard menu to access all the necessary functions
  + Log out button in dashboard
  + Add and update personal information
  + Deleting account
  + Checking available hotels, packages and tour destination
  + Booking hotels and packages
  + Utility application: Calculator
  + JDBS for storing and using multiple user and hotel information
  1. **Technology Used**

Programming Language: Java

GUI Design tool: Java Swing

Database: MySQL JDBS

IDE: Apache NetBeans IDE 24

OS: Windows 10 / 11

1. **Design and Implementation**

**2.1 UX Flow**

The UX flow of the Travel and Tourism Management System is designed to provide an intuitive and seamless user experience. Whether users are booking trips, managing their reservations, or accessing administrative features, the system ensures smooth navigation and efficient functionality.

**At the beginning of project:**

The Splash class displays a splash screen for 5 seconds before launching the login window, creating a smooth introduction to the **Travel and Tourism Management System**.

**User Account Signup and Login**

The Signup class allows new users to register by storing their details (name, username, password, phone number, email) in the database and redirecting them to the login page.

The Login class handles user authentication by verifying credentials and granting access to the system.

**Recovery Password**

First searches by the username. If username matches from sql database, it prints full name. Phone number and email, which is used to sign up, must be remembered to regain password.

**Dashboard: Control Panel Engagement**

Once logged in, users are redirected to the Dashboard. Dashboard is the main navigation window with buttons for different functionalities. which contains following components:

**a. Add Personal Details:** This section allows user to fill their personal data.

**b. Update** **Personal Details:** This section allows user to update their personal data.

**c. View** **Personal Details:** Enables users to access and review their saved personal information.

**d. Delete** **Personal Details:** Allows users to remove their account or data from the system.

**e. Check Package:** Checking available tour package.

**f. Book Package:** Allows users to select a travel package, input the number of persons, check the price, and book the package by saving the details in the system.

**g. View Package:** Displays a user's booked travel package details.

**h. View Hotel:** displays a GUI window with tabs for different hotels, each containing an image and name, allowing users to view information about various hotels.

**i. Book Hotel:** allows users to book hotel rooms by selecting a hotel, entering personal details, and confirming the booking through a graphical user interface.

**j. View Booked Hotel:** Displays a user's booked hotel details.

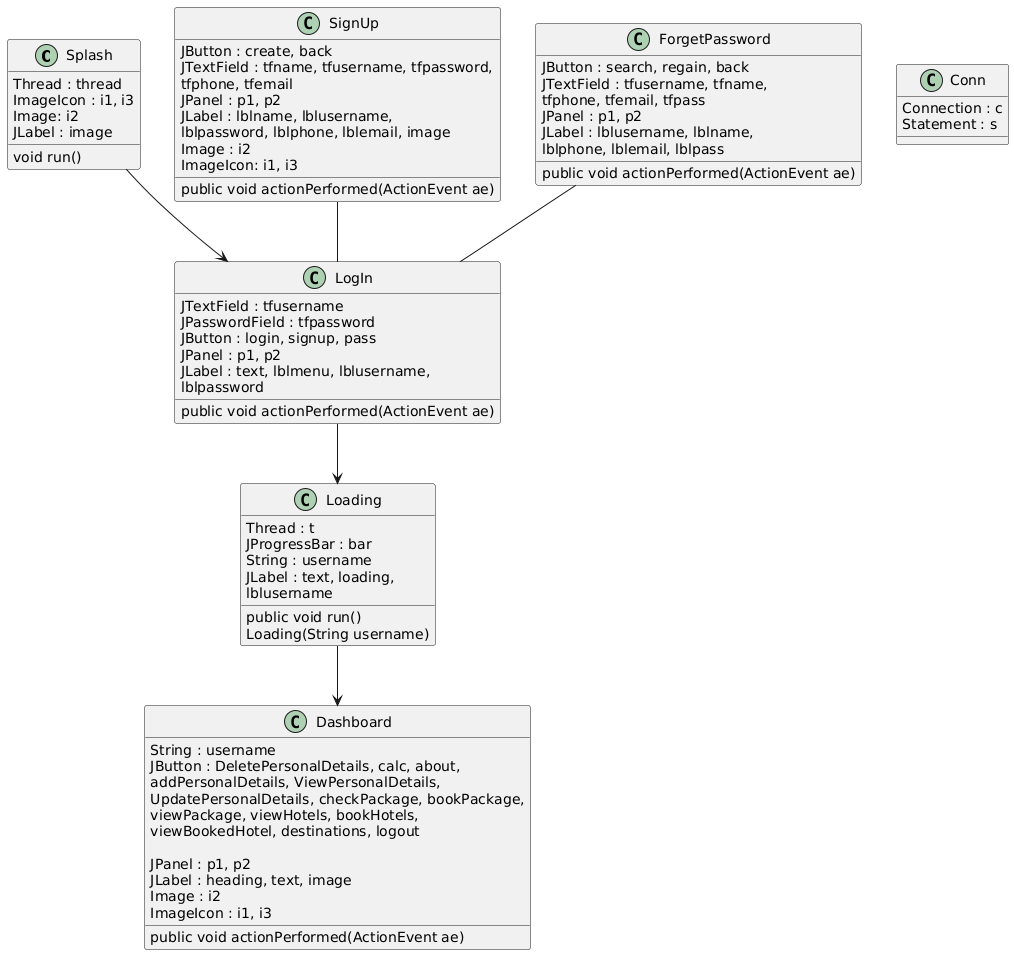
**l. Destinations:** allowing users to view offering destinations by clicking different tab.

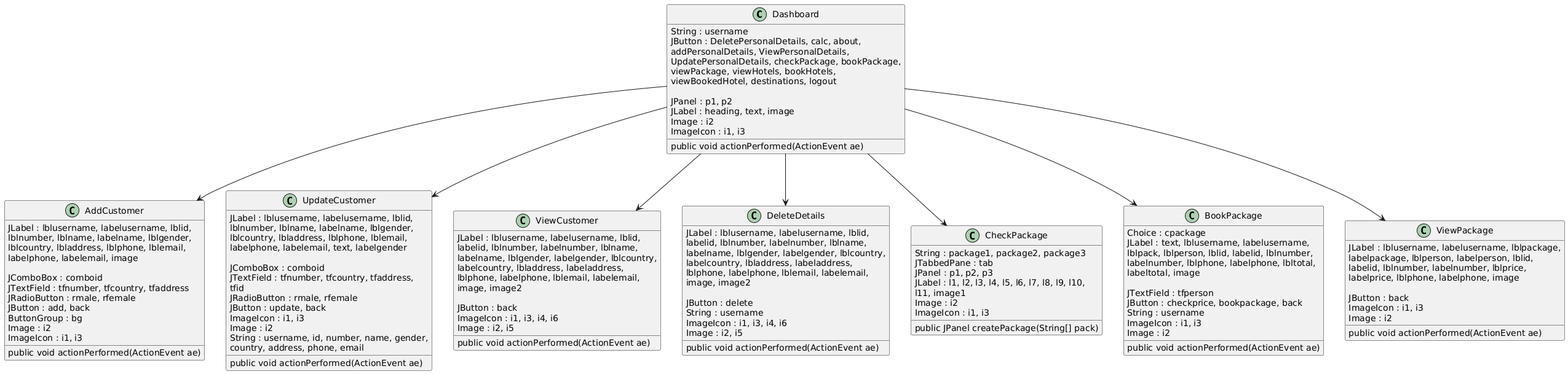
**m. Calculator:** It’s a utility application.

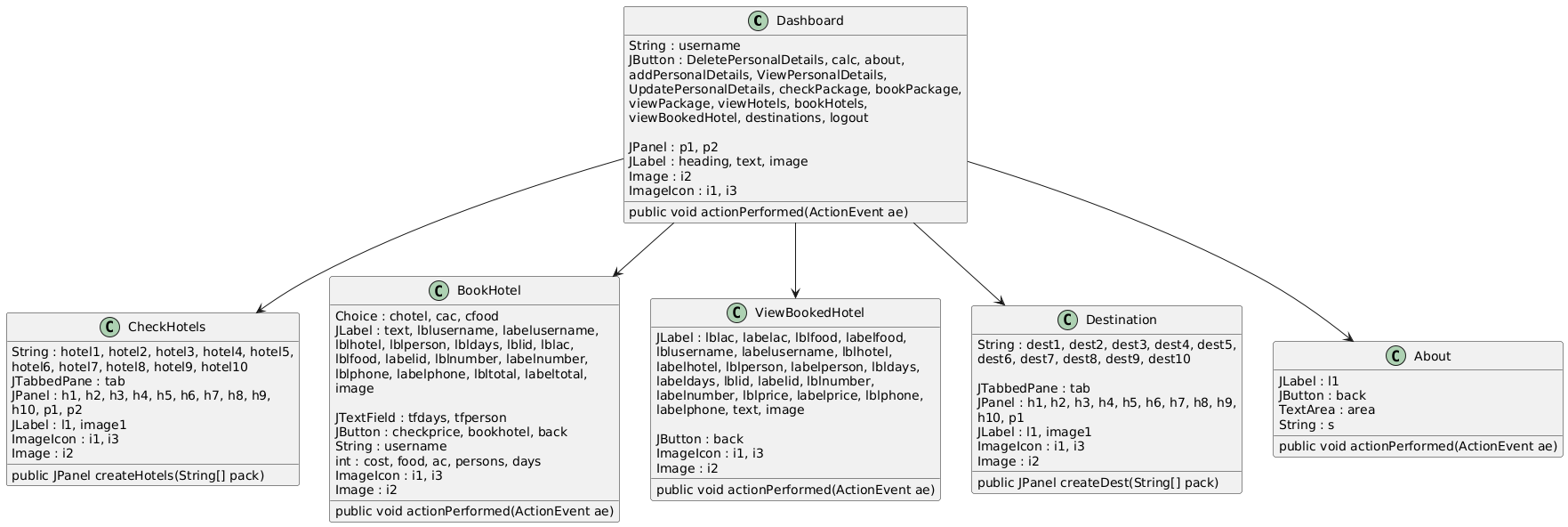
**n. About:** provides information about a travel and tourism management system.

**o. Logout:** handles user session termination and redirects the user to the login screen or exits the application.

**2.2 UML Diagram**

****

****

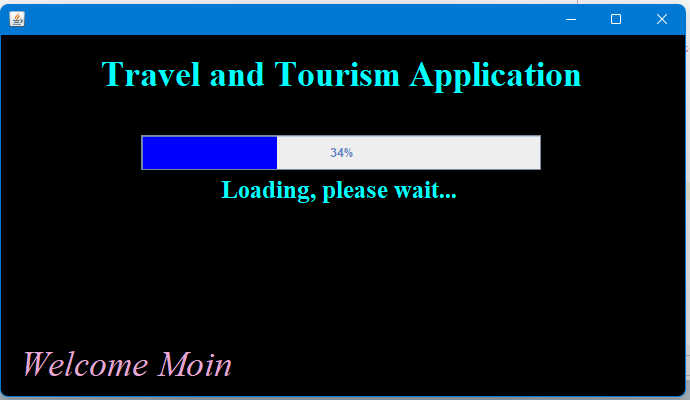
****

* 1. **UI**

**A screenshot of a computer screen

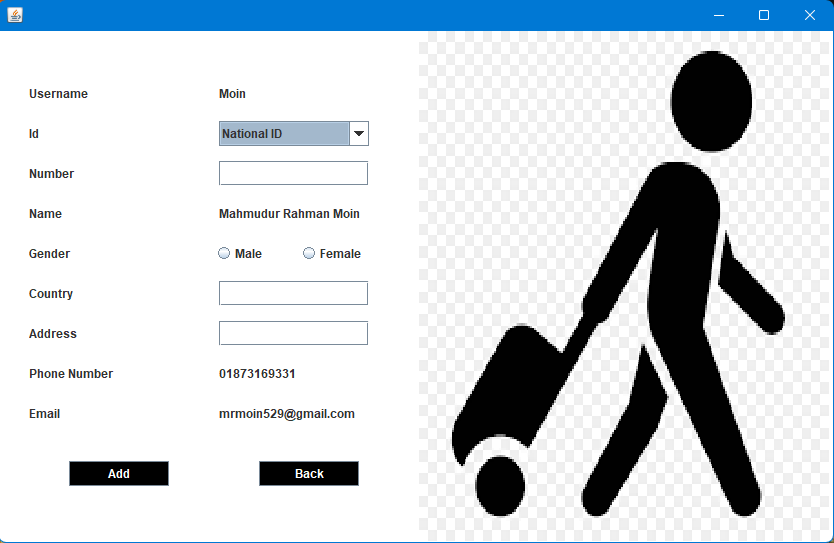
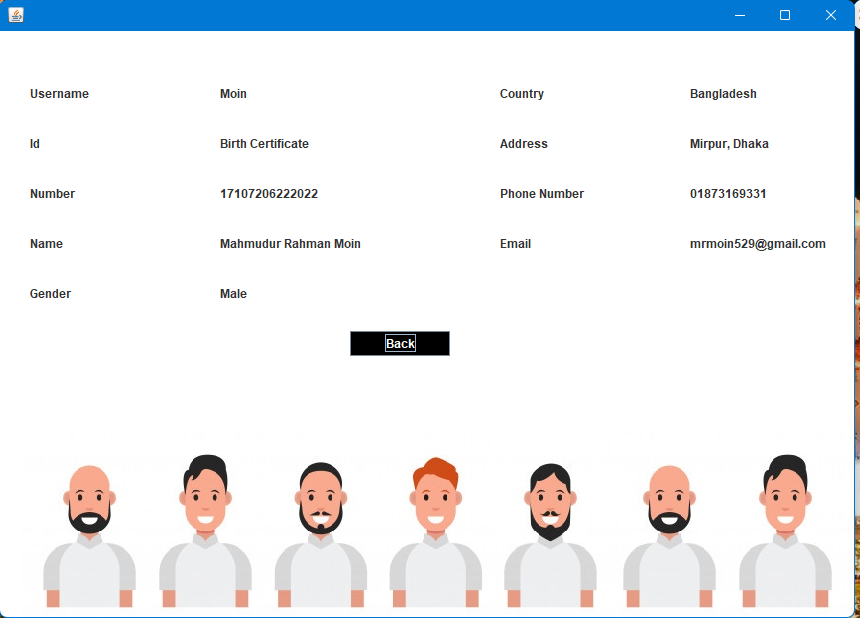
Description automatically generated A computer screen shot of a person

Description automatically generatedA screenshot of a computer

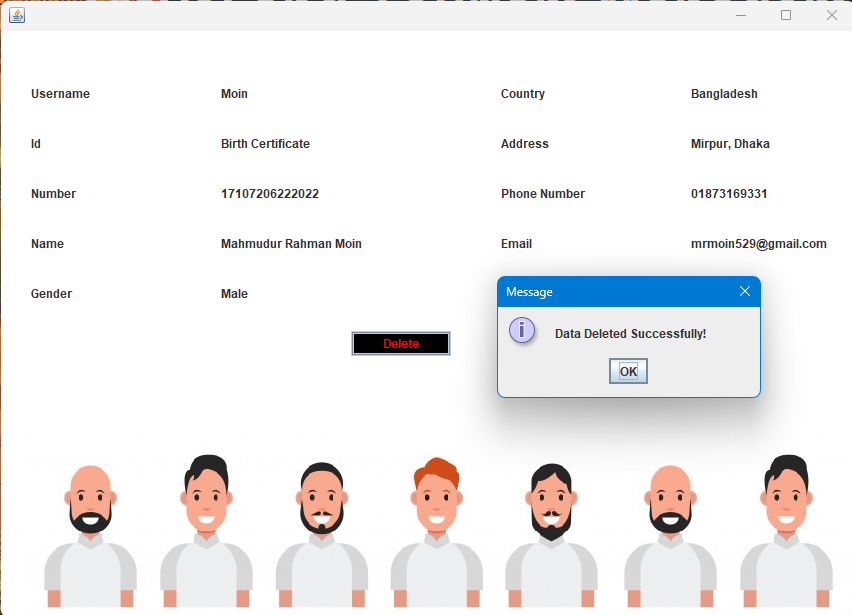
Description automatically generated **

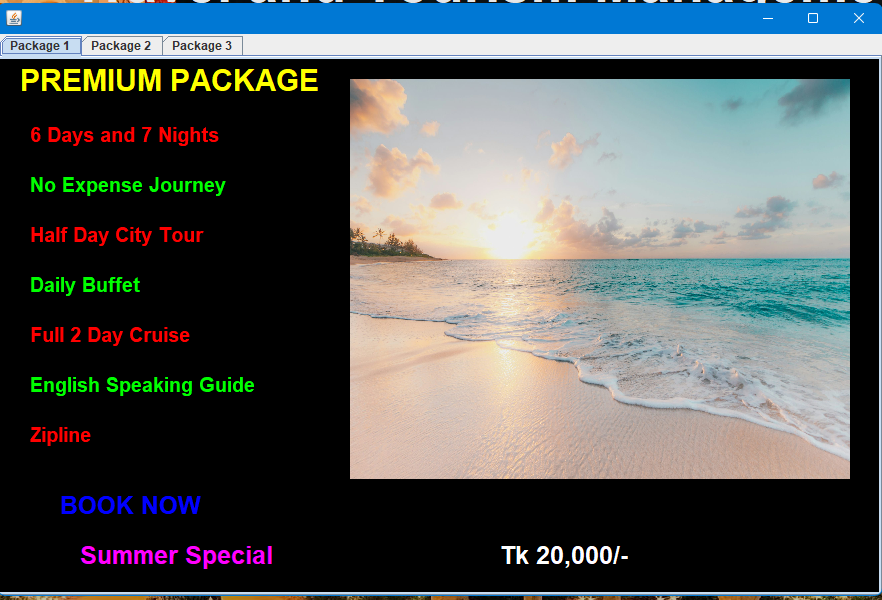
**A screenshot of a computer screen

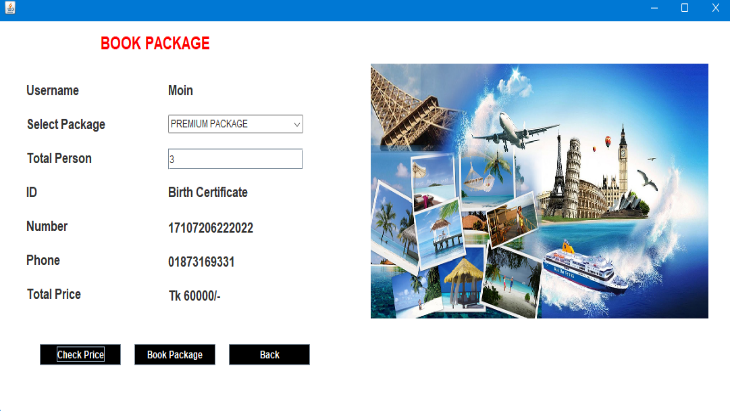
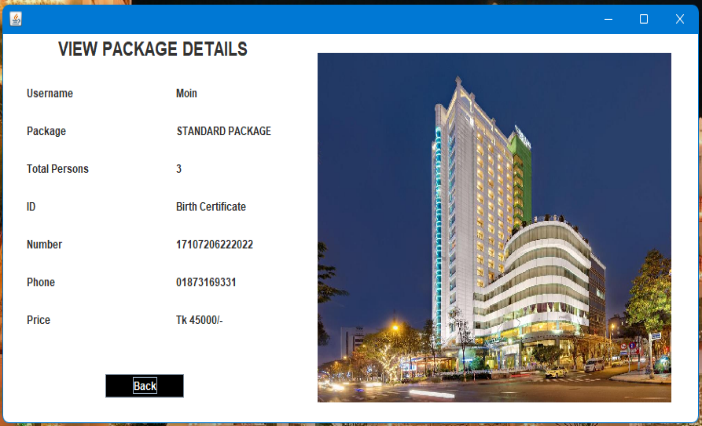
Description automatically generated**

** **

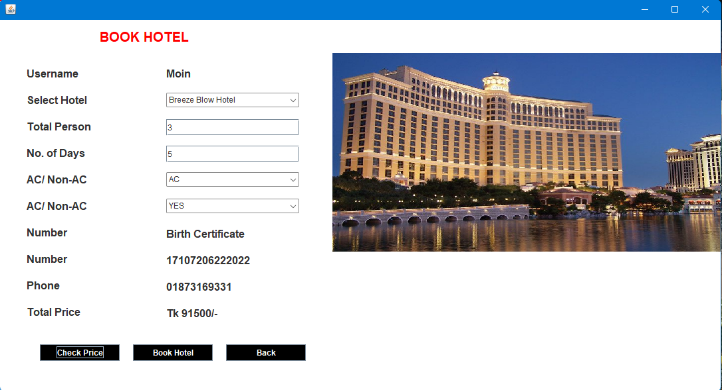
**A screenshot of a computer

Description automatically generated **

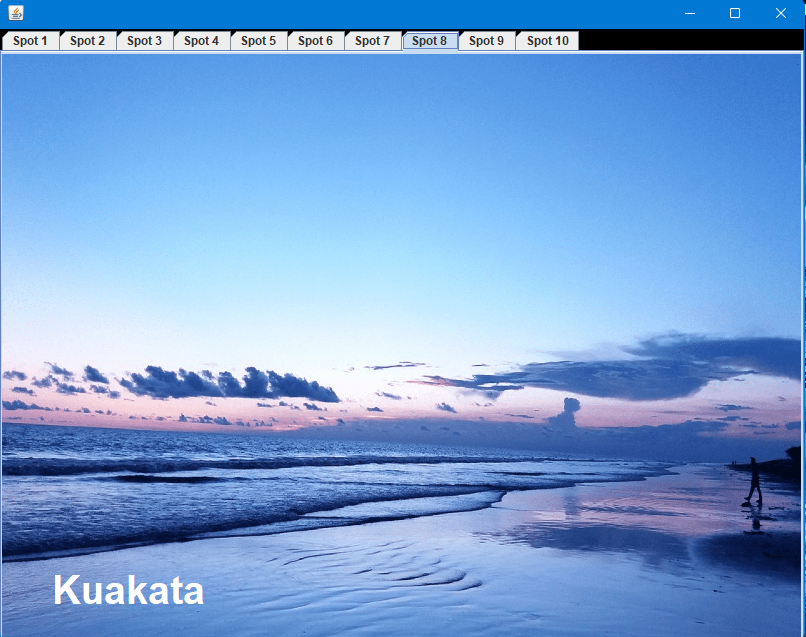
****

** **

**A pool with palm trees and a building

Description automatically generated  A screenshot of a hotel

Description automatically generated**

****

**2.4 System Design**

The **‘Travel and Tourism Management System’** is a desktop-based application. To build this, we have used three tools such as Javax(swing) for front-end, Java for Backend and SQL for database. Discuss these tools below:

1. **Presentation Layer (Frontend – Java Swing)**

Provides the graphical user interface (GUI) for users to interact with the system.

Components: Dashboard, AddCustomer, ViewCustomer etc.

1. **Business Logic Layer (Service Layer – Java Classes)**

Handles core functionalities like customer management, booking management, and payments.

Components: BookPackage, BookHotel, No. of Customer etc.

1. **Data Layer (Database - MySQL)**

Stores and retrieves user and booking details.

Components: account, customer etc.

* 1. **System Flow Diagram**

Users logs in 🡪 Dashboard opens 🡪 selects an option (e.g. ViewPackage) 🡪 system fetches available option 🡪 Check all available packages, hotels and destinations 🡪 **User selects and confirms** a package/hotel 🡪 System stores booking details in the database 🡪 **User makes payment 🡪 User can view booked packages/hotels**

**2.6 Detailed Class Descriptions**

**Splash Class**

Attributes: Thread, ImageIcon, Image

Methods: run(), Splash()

**LogIn Class**

Attributes: login, signup, pass, tfusername, tfpassword, text, lblmenu, lblusername, lblpassword

Methods: actionPerformed(ActionEvent ae)

**SignUp Class**

Attributes: create, back, tfname, tfusername, tfpassword, tfphone, tfemail, lblname, lblusername, lblpassword, lblphone, lblemail, image, ImageIcon

Methods: actionPerformed(ActionEvent ae)

**ForgetPassword Class**

Attributes: search, regain, back, tfname, tfusername, tfpass, tfphone, tfemail, lblname, lblusername, lblpassword, lblphone, lblemail, image, ImageIcon

Methods: actionPerformed(ActionEvent ae)

**Loading Class**

Attributes: Thread t, JProgressBar bar, String username, Label text, loading, lblusername

Methods: run(), Loading(String username)

**Conn Class**

Attributes: statement, connection

**Dashboard Class**

Attributes: String username, Button DeletePersonalDetails, calc, about, addPersonalDetails, ViewPersonalDetails, UpdatePersonalDetails, checkPackage, bookPackage, viewPackage, viewHotels, bookHotels, viewBookedHotel, destinations, logout, Label heading, text, image, ImageIcon

Methods: actionPerformed(ActionEvent ae)

**AddCustomer Class**

Attributes: Label lblusername, labelusername, lblid, lblnumber, lblname, labelname, lblgender, lblcountry, lbladdress, lblphone, lblemail, labelphone, labelemail, image, ComboBox comboid, RadioButton rmale, rfemale, Button add, back, ImageIcon

Methods: actionPerformed(ActionEvent ae)

**UpdateCustomer Class**

Attributes: Label lblusername, labelusername, lblid, lblnumber, lblname, labelname, lblgender, lblcountry, lbladdress, lblphone, lblemail, labelphone, labelemail, image, text, ComboBox comboid, RadioButton rmale, rfemale, Button update, back, ImageIcon, TextField tfnumber, tfcountry, tfaddress, tfid, String username, id, number, name, gender, country, address, phone, email

Methods: actionPerformed(ActionEvent ae)

**ViewCustomer Class**

Attributes: Label lblusername, labelusername, lblid, labelid, lblnumber, labelnumber, lblname, labelname, lblgender, labelgender, lblcountry, labelcountry, lbladdress, labeladdress, lblphone, labelphone, lblemail, labelemail, image, image2, Button back, ImageIcon

Methods: actionPerformed(ActionEvent ae)

**DeleteDetails Class**

Attributes: Label lblusername, labelusername, lblid, labelid, lblnumber, labelnumber, lblname, labelname, lblgender, labelgender, lblcountry, labelcountry, lbladdress, labeladdress, lblphone, labelphone, lblemail, labelemail, image, image2, Button delete, ImageIcon, String username

Methods: actionPerformed(ActionEvent ae)

**CheckPackage Class**

Attributes: Label l1, l2, l3, l4, l5, l6, l7, l8, l9, l10, l11, image1, TabbedPane tab, ImageIcon, String package1, package2, package3

Methods: createPackage(String[] pack)

**BookPackage Class**

Attributes: Label text, lblusername, labelusername, lblpack, lblperson, lblid, labelid, lblnumber, labelnumber, lblphone, labelphone, lbltotal, labeltotal, image, Button checkprice, bookpackage, back, ImageIcon, TextField tfperson, Choice cpackage

Methods: actionPerformed(ActionEvent ae)

**ViewPackage Class**

Attributes: Label lblusername, labelusername, lblpackage, labelpackage, lblperson, labelperson, lblid, labelid, lblnumber, labelnumber, lblprice, labelprice, lblphone, labelphone, image, Button back, ImageIcon

Methods: actionPerformed(ActionEvent ae)

**CheckHotels Class**

Attributes: Label l1, image1, ImageIcon, TabbedPane tab, String hotel1, hotel2, hotel3, hotel4, hotel5, hotel6, hotel7, hotel8, hotel9, hotel10

Methods: createHotels(String[] pack)

**BookHotel Class**

Attributes: Label text, lblusername, labelusername, lblhotel, lblperson, lbldays, lblid, lblac, lblfood, labelid, lblnumber, labelnumber, lblphone, labelphone, lbltotal, labeltotal, image, Choice chotel, cac, cfood, TextField tfdays, tfperson, Button checkprice, bookhotel, back, ImageIcon, String username, int cost, food, ac, persons, days

Methods: actionPerformed(ActionEvent ae)

**ViewBookedHotel Class**

Attributes: Label lblac, labelac, lblfood, labelfood, lblusername, labelusername, lblhotel, labelhotel, lblperson, labelperson, lbldays, labeldays, lblid, labelid, lblnumber, labelnumber, lblprice, labelprice, lblphone, labelphone, text, image, Button back, ImageIcon

Methods: actionPerformed(ActionEvent ae)

**Destination Class**

Attributes: Label l1, image1, ImageIcon, String dest1, dest2, dest3, dest4, dest5, dest6, dest7, dest8, dest9, dest10, TabbedPane tab

Methods: createDest(String[] pack)

**About Class**

Attributes: Label l1, Button back, TextArea area, String s

Methods: actionPerformed(ActionEvent ae)

**2.7 Discussion**

The main aim of this project is to help the tourists to manage their trip. It makes all operation of the tour company easy and accurate. The standalone platform makes tourism management easy by handling requests and providing servers for the customers located at different parts of the various cities. Different modules have been incorporated in this project to handle different parts and sector of the tour management field. The process of the system we can consider here, can maintain the databases of the system. We can insert to the databases and retrieve all the information.

**2.8 Use of OOP**

This project implements our main OOP principles:

**Encapsulation:**

Users’ personal information and booking details are securely handled in different classes.

**Inheritance:**

LogIn class extends our Dashboard class for role-specific behaviour.

**Polymorphism:**

actionPerformed method is customized in different classes for button behaviour. Also run method is overridden for multithread operations.

**Abstraction:**

In Java Swing ActionListener is an interface which is used for button actions. This ensures flexible and modular system design.

1. **Conclusion**

**3.1 Challenges and Solutions**

**Updating and Maintaining the System:**

Implemented modular coding to make future updates easier.

**Packaging and Deployment:**

Packaging the project as an executable JAR file and using MySQL dump for easy database setup.

**Slow Data Retrieval:**

Optimized queries using MySQL database.

**Dynamic Pricing and Discounts:**

Implementing pricing algorithms that adjust fares dynamically

**Managing Booking Conflicts:**

Implementing a unique booking ID system by using database transactions.

**3.2 Future Enhancement**

* Adding a payment method for paying booked prices
* More enhanced Graphical interfaces with less effort
* Spots from abroad will be available soon
* After introduction of abroad tourism flight price will be added to the packages
* More developed security system in securing personal information of the users
* Easier ways to password recovery and log in

**3.3 Reference**

* <https://www.w3schools.com>
* <https://www.javatpoint.com>
* <https://www.codecademy.com>
* <https://www.stackoverflow.com>

**3.4 Contribution**

Md. Mahmudur Rahman Moin – 30

* Database integration using MySQL
* Development of Dashboard UI
* Managing user’s personal details
* Account deletion system
* Package details using multiple tab
* Package booking system
* Destination tabs
* Utility app : Calculator

Md. Sharif – 14

* Splash intro
* Login and Signup mechanism
* Password recovery system
* Loading screen system
* Checking hotel details using multiple tab
* Hotel booking system
* **Video Link :** <https://youtu.be/9hC3Vglsp28>
* **GitHub Source Code:** [**https://github.com/Moin529/CSE-2112-OOP-Project/tree/main**](https://github.com/Moin529/CSE-2112-OOP-Project/tree/main)