COMSATS University, Islamabad Pakistan

LAB SEMESTER PROJECTS – SPRING 2025

Project Report
For
TRAIN MANGEMENT SYSTEM

(Pakistan Railway System)

Submitted By:

Muhammad Ali **FA23-BSE-062**

Hassan Muhammad **FA23-BSE-104**

Supervisor

Mr. Saif Ullah Ijaz

Submission Date: (18th June 2025)
Course: CSC241 Object Oriented Programming
BS-R

Bachelor of Science in Software Engineering (2023-2027)
Computer Science Department

Contents

FA23-BSE-062
FA23-BSE-104

1. Abstract

The Pakistan Railway System is a desktop-based GUI application developed in Java using Object-Oriented Programming principles. It enables users to search trains, book tickets, view and cancel bookings, and navigate through various system options in an interactive graphical interface. The application simulates a real-world ticket reservation experience while storing booking information in local text files. Designed for academic purposes, it highlights the use of classes, file handling, encapsulation, and GUI building using Java Swing.

2. Introduction

Background

Railway systems around the world depend on digital management tools to streamline booking and ticketing operations. This project mimics a basic version of such a system with a strong focus on OOP concepts, enabling students to understand how programming logic maps to real-world applications.

Problem Statement

Console-based or manual ticketing systems are inefficient, prone to human error, and lack user-friendliness. This project addresses this by offering an intuitive, modular, GUI-driven ticket booking system using Java Swing.

Objectives

- Develop a GUI-based train reservation system
- Apply OOP principles: encapsulation, modularity, and event-driven programming
- Enable users to search trains, select dates, book, and cancel tickets
- Use file handling to save booking records persistently

3. System Overview

Main Features

- Train search using source/destination dropdowns
- Calendar for selecting travel dates
- Ticket booking and data storage in text file
- Ticket viewing and cancellation functionality
- Navigation between screens with a consistent UI style

FA23-BSE-062
FA23-BSE-104

4. System Design and Architecture

Major classes and their purposes:

- Train: Represents a train with details like ID, source, destination, and time.
- Passenger: Contains details about the user booking the ticket.
- Ticket: Encapsulates train and passenger details and travel date.
- PakistanRailwaySystem: Main class that manages all GUI screens and events.

OOP Concepts Applied:

- **Encapsulation**: All classes hide data and provide access via getters/setters.
- **Inheritance**: GUI forms follow reusable structures.
- **Polymorphism**: Action listeners handle events dynamically.
- **Modularity**: GUI logic is separated into different panels and methods.

5. OOP Principles in Action

Encapsulation

```
public class Train {
    private String trainID;
    private String source;
    private String destination;
    private String time;

    public Train(String trainID, String source, String destination, String time) {
        this.trainID = trainID;
        this.source = source;
        this.destination = destination;
        this.time = time;
    }

    public String getTrainID() { return trainID; }
    public String getSource() { return source; }
    public String getDestination() { return destination; }
    public String getTime() { return time; }
}
```

Explanation:

The Train class uses encapsulation to hide internal data and provide controlled access through getters.

FA23-BSE-062
FA23-BSE-104

Event Handling

```
bookButton.addActionListener(new ActionListener() {
    public void actionPerformed(ActionEvent e) {
        String name = nameField.getText();
        String cnic = cnicField.getText();
        String trainID = selectedTrain.getTrainID();
        String date = selectedDate;

        try (FileWriter writer = new FileWriter("bookings.txt", true)) {
            writer.write(name + "," + cnic + "," + trainID + "," + date + "\n");
        } catch (IOException ex) {
            ex.printStackTrace();
        }
    }
});
```

Explanation:
When the "Book Ticket" button is clicked, the listener captures user input, constructs a booking record, and appends it to bookings.txt.

Polymorphism

```
JButton exitButton = new JButton("Exit");
exitButton.addActionListener(e -> System.exit(0));
```

Explanation:
The ActionListener attached to buttons allows for polymorphic behavior depending on the context of the button's use (exit, back, etc.).

6. Technologies Used

- **Java (JDK 21)** – Core logic and OOP design
- **Java Swing** – GUI development
- **Text Files** – For booking persistence
- **VS Code** – Development environment

7. GUI Flow Overview

Although actual screenshots aren't shown here, below is the logical GUI flow:

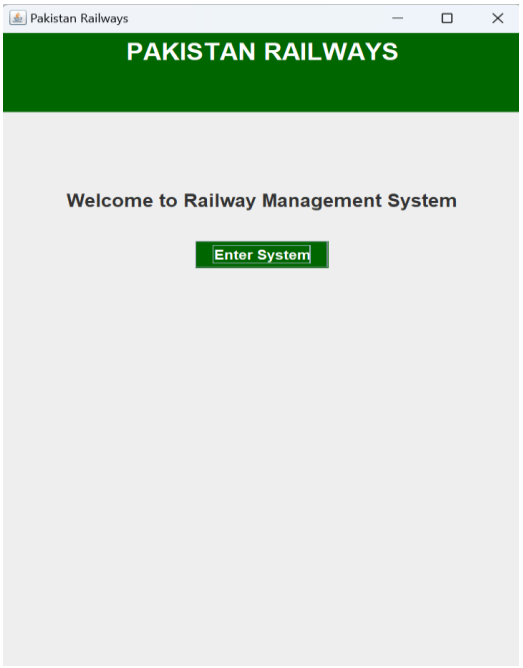
1. **Main Menu** – Options: Book Ticket, View Bookings
2. **Train Search** – Dropdowns to select source/destination
3. **Calendar View** – Select travel date
4. **Passenger Info Entry** – Input name, Gender

FA23-BSE-062
FA23-BSE-104

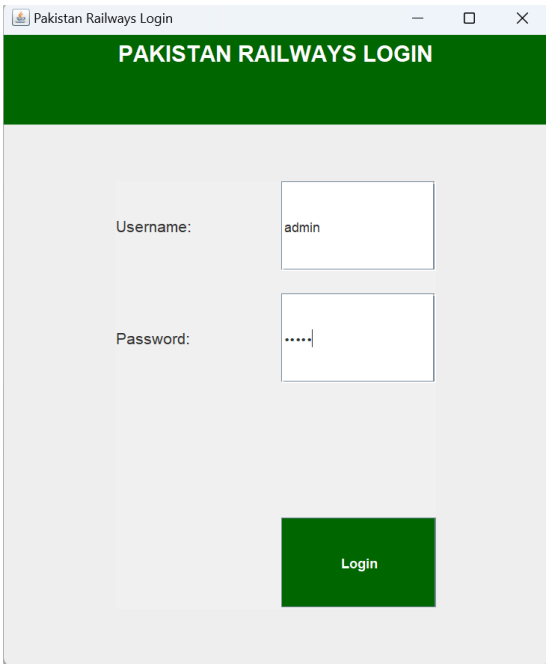
5. **Confirmation Page** – Shows thank-you message
6. **Cancel Ticket** – Remove existing bookings

8. UI Overview

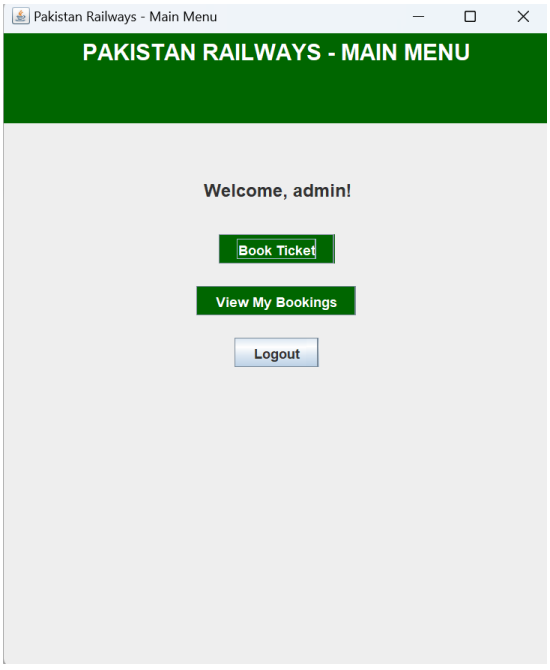
Title Screen



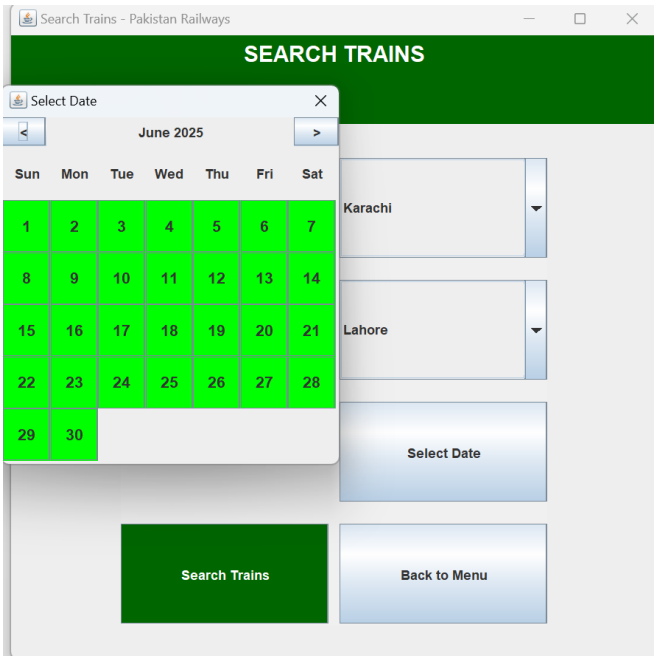
Login page



Main menu

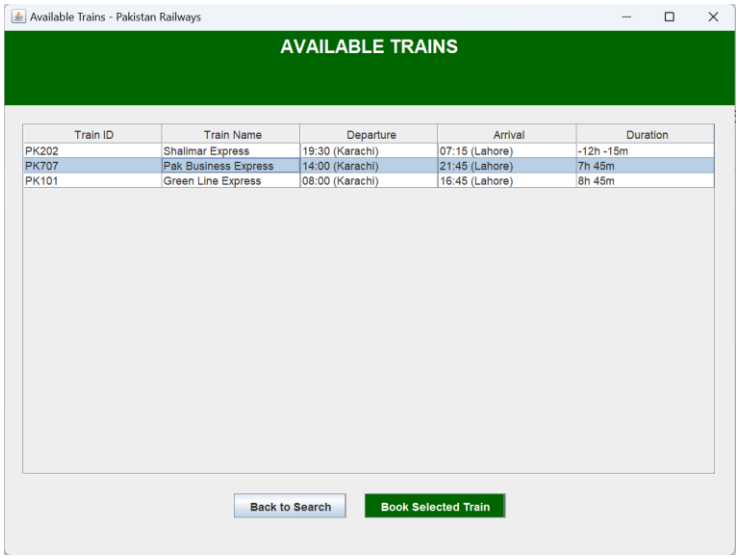


Search Trains

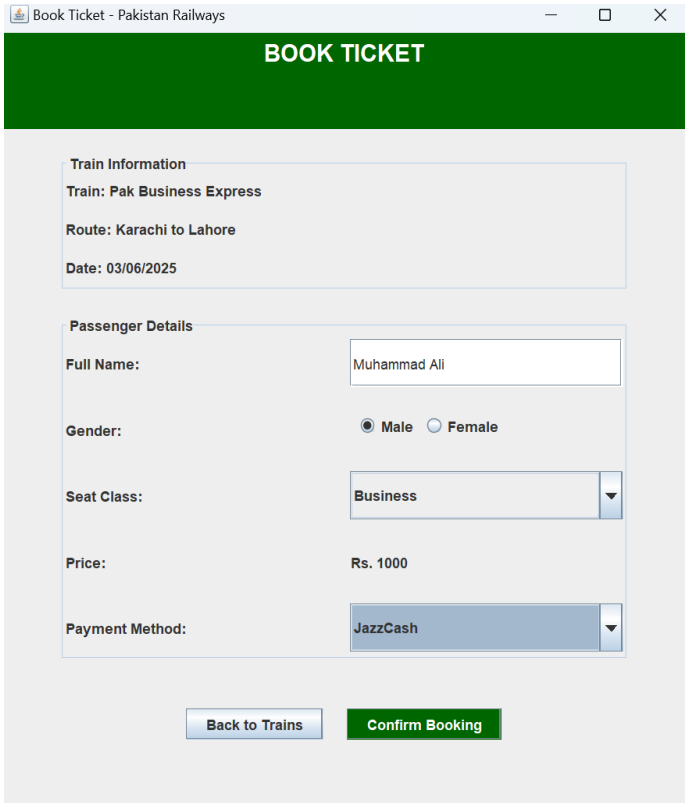


FA23-BSE-062
FA23-BSE-104

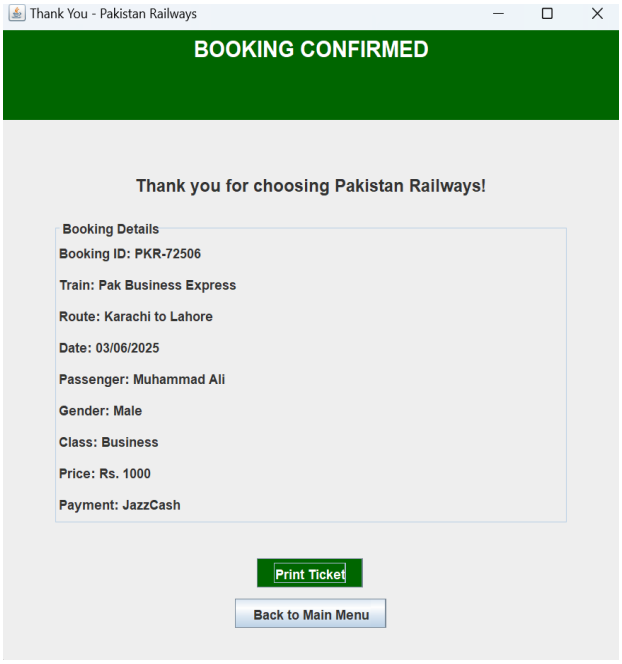
Available Trains



Book Ticket

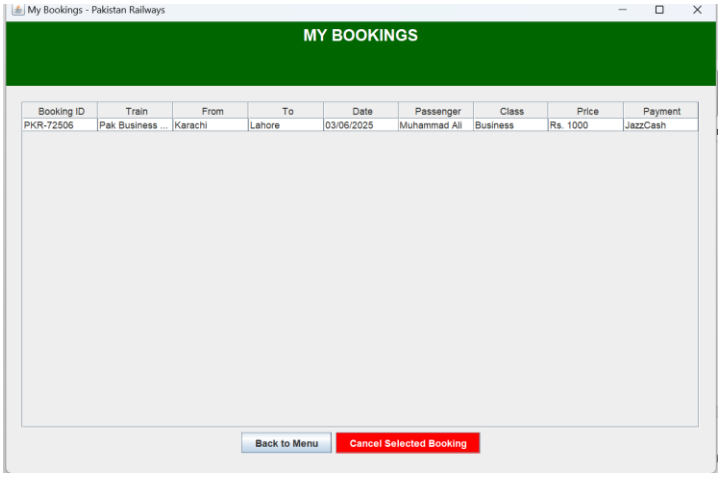


Thank You Page

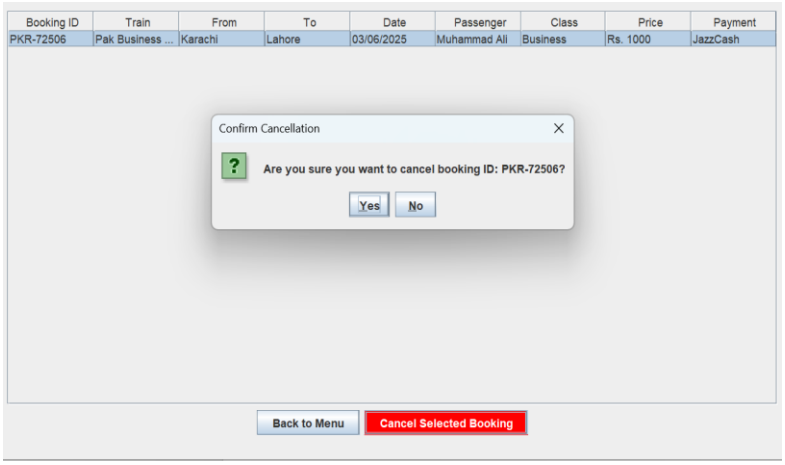


FA23-BSE-062
FA23-BSE-104

My Bookings



Cancel Booking



9. Data Handling

File Handling

```
try (FileWriter writer = new FileWriter("bookings.txt", true)) {
    writer.write(name + "," + cnic + "," + trainID + "," + date + "\n");
}
```

Explanation:

Bookings are stored as comma-separated lines in a .txt file. This simulates a basic database while keeping the project file-based for simplicity.

10. Testing and Validation

The application was tested for:

- Field validation (empty inputs)
- Navigation between screens
- Duplicate booking entries
- Files append accuracy
- Booking cancellation flow

Basic error handling and input checks are implemented.

FA23-BSE-062
FA23-BSE-104

11. Conclusion

The Pakistan Railway System demonstrates the practical use of OOP principles and GUI development in Java. It effectively fulfills the need for a beginner-friendly, file-based ticket booking simulation. Through modular class design and structured event handling, the project delivers a clean and educational implementation of a real-world use case.

FA23-BSE-062
FA23-BSE-104