INTERNSHIP PROJECT-1 DOCUMENT

ON "RAILWAY RESERVATION SYSTEM (JAVA)"

Submitted by:

Navpreet Singh (INTERN)

Submitted To:-

Kanduri Abhinay (FOUNDER)

RITHIN VERMA (CTO)

INDEX

INTRODUCTION:-

The Railway Reservation System is a console-based Java application designed to handle ticket booking, confirmation, and waitlisting for train routes efficiently. By utilizing file handling for data persistence, the system records and retrieves booking details to simulate real-world reservation operations. The main objective is to streamline ticket allocation based on seat availability and provide a clear user interaction flow, making it suitable for small- to medium-scale rail reservation scenarios.

SOFTWARE REQUIREMENTS:-

Programming Language: Java (JDK 8 or higher)

IDE: VS Code, IntelliJ IDEA, or any Java-compatible editor

Libraries: Uses only Java Standard Library APIs (I/O, Serialization)

Operating System: Platform-independent (Windows, Linux, Mac)

Other Requirements: No external dependencies; runs from the command line

DESIGN:-

The system is organized as follows:

Passenger Model: Represents individual passengers with attributes like name, age, and intended travel route.

ReservationSystem Main Class: Manages core logic including booking confirmation, waitlisting, and user input/output.

Persistence: All bookings are saved in a plain text file (bookings.txt) to maintain booking records across sessions.

Booking Logic: Confirms bookings up to a maximum seat limit; additional passengers are added to a waitlist.

INPUT:-

+------------------------------+

| Railway Reservation System |

+------------------------------+

| 1. Enter Name |

| 2. Enter Age |

| 3. Enter Route |

+-------------------------------+

| (System allocates seat or |

| adds to waitlist) |

+-------------------------------+

User input consists of simple prompts for name, age, and route via the terminal. The program confirms the booking or places the passenger on the waitlist based on seat availability.

IMPLEMENTATION:-

The Passenger class is a Java bean containing fields for name, age, and route with standard getter and setter methods.

The ReservationSystem class keeps two lists: one for confirmed bookings (up to the max seat capacity), and one for the waitlist.

On program start, existing bookings (if any) are loaded from bookings.txt.

Booking a ticket checks if seats are available for confirmation; otherwise, the passenger is waitlisted.

Booking status (confirmed/waitlist) is shown on the terminal after each operation.

Booking data is saved after every transaction for reliability and system recovery.

TESTING:-

The Railway Reservation System underwent several testing approaches:

Manual Input Testing: Booked multiple passengers to verify correct seat allocation and waitlisting.

Persistence Validation: Restarted the program to ensure previously booked data is reloaded from bookings.txt.

Edge Case Checks: Tested behavior when the seat limit is reached and when waitlist is populated.

Robustness: Checked behavior on corrupted or missing booking files (application initializes without crashing).

Menu Flow: Ensured that user prompts are clear and that system feedback is immediate and accurate.

ADVANTAGES:-

Simplicity: User-friendly interface and clear reservation logic, making it well-suited for beginners and small railway agencies.

Portability: Runs on any platform with Java installed, without third-party libraries.

Persistence: Booking records are safely stored and reloaded, enabling real-life applicability and consistency..

Extensibility: Easily extendable to more features like cancellation, notification, or seat charting.

Realistic Simulation: Mimics real-world seat allocation and waitlisting.

CONCLUSION:-

The Railway Reservation System serves as an effective model for understanding file handling, Java collections, user interaction, and core programming concepts. The project emphasizes practical skills like state persistence, modular class design, and robust error handling. It is a strong foundation for developing more complex reservation or inventory management systems in Java.

REFERENCES:-

GeeksForGeeks: Java File Handling and Collections Tutorials

JavaTPoint: Java File I/O and OOP Concepts

Oracle Documentation: Official Java Standard Libraries.