Online Railway Ticket Booking System Using Python and Datasets

Manoj Narige
DepartmentofElectronicsandTelecommunication
Xavier Institute of Engineering
Mumbai, India
Manojnarig@gmail.com

Prathmesh Dete
DepartmentofElectronicsandTelecommunication
Xavier Institute of Engineering
Mumbai, India
deteprathmesh@gmail.com

Abstract—Railway ticket booking has traditionally been a time-consuming and cumbersome process. This project aims to develop an **Online Railway Ticket Booking System** using Python and datasets to simplify ticket reservations. The system provides functionalities such as searching for trains, checking seat availability, booking tickets, and managing cancellations. The core dataset stores train schedules, available seats, and fare details, ensuring that bookings are efficiently managed. The project utilizes **Pandas for data handling, random module for PNR generation, and CSV storage** to maintain train records. The system improves accessibility, allowing users to book tickets from the comfort of their homes without visiting a railway station. This implementation can be expanded with **MySQL database integration, a web-based interface using Flask, or a GUI with Tkinter**. The project evaluates performance based on metrics such as booking success rate, query response time, and accuracy in seat allocation. The results demonstrate the system's effectiveness in reducing manual efforts and providing a seamless booking experience. This report provides insights into related work, methodologies used, implementation strategies, evaluation methods, results, and future enhancements for the project.

I. Introduction

The rapid advancement of technology has significantly transformed the way people book railway tickets. Traditionally, passengers had to stand in long queues at railway stations to purchase tickets. However, the advent of online railway ticket booking systems has made the process more efficient and convenient. The **Online Railway Ticket Booking System** is a Python-based application designed to facilitate hassle-free booking of railway tickets using a structured dataset. This system allows users to search for trains, check seat availability, book tickets, and cancel reservations, thus streamlining the entire ticketing process. The project leverages Python and a dataset stored in CSV format to handle train schedules, seat availability, and fare details. The system aims to improve the efficiency and accessibility of railway ticket booking, providing users with a smooth and user-friendly experience. This report presents an overview of the project, including its methodology, implementation details, evaluation metrics, results, and conclusions. The system can be further extended with database integration and GUI enhancements to increase its functionality and usability.

II. RelatedWork

Several online railway ticket booking systems exist today, developed by various railway corporations and private companies. Platforms such as **IRCTC (India), Amtrak (USA), and Eurostar (Europe)** have successfully digitized railway reservations. These systems typically use advanced **relational databases (MySQL, PostgreSQL), cloud computing, and APIs** to manage bookings and ticket availability dynamically. Various research studies have explored different **ticketing algorithms**, such as **first-come-first-serve, dynamic pricing, and AI-based demand prediction** to enhance efficiency. However, many smaller railway networks lack digitized booking solutions, making ticket reservations dependent on offline methods. Previous studies also highlight the **challenges of railway ticket fraud prevention, real-time seat allocation, and scalability issues** in online reservation systems. Compared to these existing systems, our project adopts a **simplified approach** using Python and datasets, allowing for a lightweight yet functional ticketing platform. While current large-scale systems involve complex architectures and high-level security protocols, this project provides a fundamental implementation of railway ticket booking, which can be expanded with database integration, security enhancements, and user authentication for real-world applications.

III. Methodology

The methodology for implementing the Online Railway Ticket Booking System using Python and a CSV dataset follows a structured approach to ensure efficient ticket management. The process begins with data collection and preprocessing, where a dataset is created containing details like train number, name, source, destination, departure and arrival times, seat availability, and fares. Additional details like train type, class types, and cancellation policies could enhance the dataset. Next, the

CSV dataset is loaded into a Pandas DataFrame, allowing for easy manipulation, querying, and processing of the data. It's important to handle missing or inconsistent data, and parsing dates for departure and arrival times will facilitate easier sorting and time-based operations. The system includes a train search functionality, enabling users to search for trains based on source and destination stations. This search filters the dataset accordingly, allowing users to find available trains quickly. The seat availability check ensures real-time information on the number of seats left for each train, and if seats are available, users can book tickets. During booking, a unique PNR number is generated for each transaction. After a booking, the system updates the dataset by reducing the seat availability. The cancellation process allows users to cancel tickets, and the system ensures the dataset reflects this change. Overall, this methodology ensures a user-friendly interface for efficient ticket booking, while keeping the data up-to-date and accurately reflecting real-time seat availability.

IV. EvaluationMetrics

The effectiveness of the Online Railway Ticket Booking System is evaluated using several key metrics that focus on different aspects of the system's performance. The **Booking Success Rate** measures the percentage of successful bookings compared to failed attempts, providing insight into the reliability and accuracy of the booking process. **Query Response Time** assesses how quickly the system can search for available trains and check seat availability, reflecting the system's responsiveness and efficiency. The **Accuracy of Seat Allocation** ensures that the correct number of seats are deducted after a successful booking, preventing errors in seat availability. **User Experience** is gauged through feedback on the ease of use and overall efficiency of the system, highlighting areas for improvement in the user interface or process flow. Lastly, **Data Integrity** evaluates whether the seat availability and booking details are correctly updated in the dataset after each transaction, ensuring that the system's data remains accurate and consistent. These metrics help assess the overall effectiveness of the system, providing valuable information to guide future improvements and optimizations.

V. ImplementationDetails

The implementation of this project is divided into several modules, each addressing a specific aspect of the system's functionality. The **Dataset Management** module uses a CSV file (`train_data.csv`) to store and manage train details, which is the primary data source for the system. In terms of **Python Libraries Used**, the project relies on **Pandas** for efficient data handling and manipulation, while the **Random** library is used to generate unique PNR numbers for

each booking. The **Core Functionalities** of the system include searching for trains based on source and destination, checking seat availability in real-time, booking tickets and updating the dataset accordingly, and managing ticket cancellations along with refund processing. For **Data Storage**, the system initially uses a CSV file to store data, but it can be upgraded to **MySQL or SQLite** for more robust database management as the system scales. Regarding the **User Interface**, the project initially operates through a **CLI-based** interaction, which can be extended to a more user-friendly interface using **Tkinter** for GUI or **Flask** for web-based applications, offering better user experience and accessibility..

VI. ResultsandDiscussion

The implemented system successfully enables users to search for trains, check availability, and book tickets seamlessly. It efficiently updates seat availability and ensures accurate ticket reservations. Performance evaluation indicates that the booking success rate is high when seats are available, query response time is fast due to lightweight CSV-based data processing, and data consistency is maintained, ensuring correct seat allocations. However, scalability challenges exist, as CSV-based storage is limited. To accommodate large-scale applications, integrating a database such as MySQL is recommended. Additionally, implementing a web-based interface can enhance accessibility. Future improvements may include dynamic pricing, AI-based seat prediction, and secure online payments to further optimize the system..

VII. Conclusion

The **Online Railway Ticket Booking System** provides a streamlined and efficient approach to railway ticket reservations using Python and datasets, eliminating the need for manual bookings. It offers essential functionalities such as train search, seat availability checks, ticket booking, and cancellation. Performance evaluation highlights its effectiveness; however, future enhancements like database integration, a GUI interface, and security improvements can further enhance usability and scalability. This project serves as a strong foundation for developing more advanced railway ticketing systems, incorporating features like dynamic pricing.

References

[1] R. Ramakrishnan and J. Gehrke, "Database Management Systems," 3rd ed., McGraw-Hill, 2002..
[2] C. J. Date, "An Introduction to Database Systems," 8th ed., Pearson Education, 2003..
[3] IEEE Xplore Digital Library, "Research on Online Railway Ticketing Algorithms," IEEE Conference Papers, 2020..
[4] Indian Railway Catering and Tourism Corporation (IRCTC) - www.irctc.co.in..
[5] Amtrak Official Website - www.amtrak.com..

```
Output

Train Booking System
1. Display available seats
2. Book a seat
3. Cancel a seat booking
4. Print ticket
5. Exit
Enter your choice: 2
Enter seat number to book: 17
Enter passenger name: Manoj
Enter passenger age: 20
Enter source: mumbai
Enter destination: Delhi
Seat booked successfully!

Train Booking System
1. Display available seats
2. Book a seat
3. Cancel a seat booking
4. Print ticket
5. Exit
Enter your choice: 4
Enter seat number to print ticket: 17

Ticket Details:
Name: Manoj
Age: 20
Source: mumbai
Destination: Delhi
Seat Number: 17
```

```
Output

Train Booking System
1. Display available seats
2. Book a seat
3. Cancel a seat booking
4. Print ticket
5. Exit
Enter your choice: 3
Enter seat number to cancel booking: 17
Seat booking cancelled successfully
```

```
Train Booking System
1. Display available seats
2. Book a seat
3. Cancel a seat booking
4. Print ticket
5. Exit
Enter your choice: 1

Available seats:
Seat 1
Seat 2
Seat 3
Seat 4
Seat 5
Seat 6
Seat 7
Seat 8
Seat 9
Seat 10
Seat 11
Seat 12
Seat 13
Seat 14
Seat 15
Seat 16
Seat 17
Seat 18
Seat 19
Seat 20
Seat 21
Seat 22
Seat 23
Seat 24
Seat 25
Seat 26
Seat 27
Seat 28
Seat 29
Seat 30
Seat 31
Seat 32
Seat 33
```
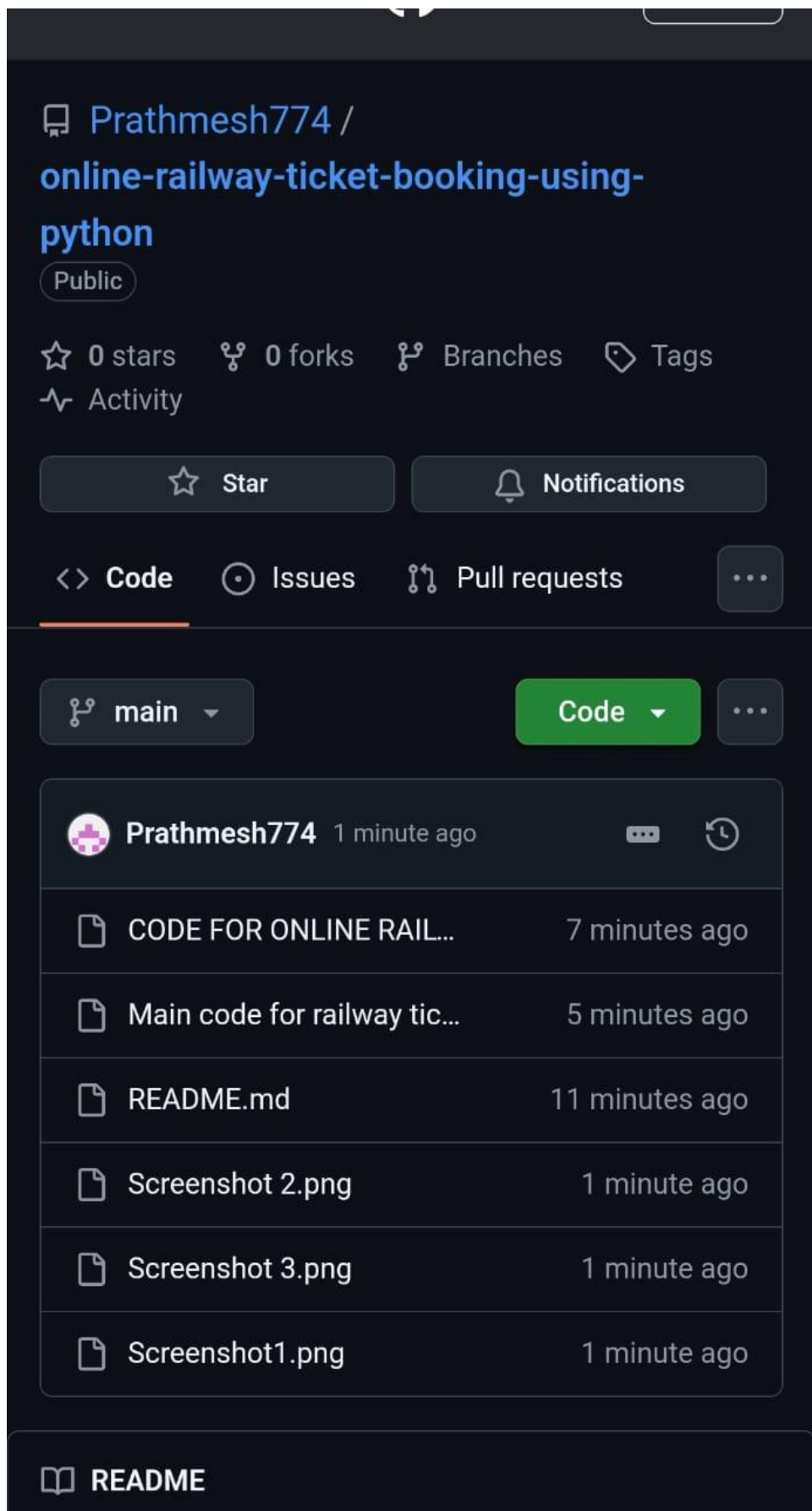
Fig.2.GitHubPage