TRAIN TICKET RESERVATION SYSTEM

PROJECT PROPOSAL

Our Team,

1) AS2021403 - D.P.G.A.H. Kulathilaka
2) AS2021367 - T.G.D.S. Chathuranga
3) AS2021573 - M.A.S.E.R. Wijerathna
4) AS2021457 - R.M.D.D. Malinda
5) AS2021517 - P.T.S. Karunarathna
6) AS2021535 - H.R.S.U. Ranasinghe
7) AS2021637 - M.A. Sathsarani
8) AS2021359 - G.P.G. Pasqual

INTRODUCTION

In today's fast-paced world, efficient and user-friendly train travel booking systems are crucial for both passengers and train operators. Traditional booking methods can be time-consuming, inconvenient, and lack real-time information. This proposal outlines the design and implementation of a modern train ticket booking system leveraging a robust database management system (DBMS) to address these challenges and create a seamless travel experience.

Problem Statements

1. **Inefficient booking process**: Long queues at counters, and outdated technology lead to frustration and delays for passengers.
2. **Lack of real-time information**: Inaccurate or delayed schedule updates, limited seat availability information, and minimal train progress updates hinder informed travel decisions.
3. **Inflexible booking options**: Difficulty managing bookings, limited cancellation policies, and limited booking channels create inconvenience and inflexibility.
4. **Poor communication**: Delayed or unavailable information about emergencies, schedule changes, and disruptions significantly impact passenger journeys.

Objectives of the system

1. **Streamline booking process**: Implement online and mobile booking options, enable self-service ticket management, and provide real-time seat availability.
2. **Enhance data management**: Create a centralized database for train schedules, fares, user information, and booking details, ensuring accuracy and efficiency.
3. **Improve user experience**: Offer user-friendly interfaces, multiple payment options, personalized booking histories, and real-time train progress updates.
4. **Strengthen communication**: Integrate an emergency alert system to keep passengers informed about delays, cancellations, and track disruptions.

Functional Requirements

Core Functionalities:

- Booking Management:
    - Allow customers to search for train schedules and fares based on their travel preferences.
    - Facilitate online booking of tickets with secure payment options.
    - Enable cancellation of bookings up to 3 days before the departure date.
    - Store and manage booking details, including passenger information, payment records, and ticket status.

- User Management:

    - Provide options for customer registration and account management.
    - Allow guest bookings using a mobile number and identity card number.
    - Store and manage user profiles, including personal information, booking history, and preferences.

- Train Information Management:
    - Maintain a comprehensive database of train schedules, routes, fares, and seat availability.
    - Enable staff to update train information and schedules as needed.

- Station Information Management:
    - Maintain a database of train stations, including names, locations, and facilities.
    - Display station information to customers during booking and journey planning.

- Emergency Alert Management:
    - Integrate with train tracking systems to receive real-time updates on delays, cancellations, and track disruptions.
    - Send timely notifications to passengers about emergencies via push notifications, SMS, and in-app messaging.

System Entities

1. **Customers**: Users who book train tickets through the system.
2. **Staff**: Users who manage the system and provide customer support.
3. **Trains**: The trains that are available for booking.
4. **Stations**: The stations where the trains stop.
5. **Bookings**: The bookings made by customers.
6. **Emergency Details**: The details of any emergencies that occur during train rides.

Implementation Details

The system will be implemented using a combination of programming languages, including Java, and SQL. The front-end of the system will be developed using HTML, CSS, and JavaScript. The back-end of the system will be developed using Java. The database will be developed using SQL.