**A Project Report on**

**Railway Management System**



**BS Computer Science**

## **Database System**

**(LAB)**

**Semester:**

4th

**Project Report Submitted By:**

**Group Participants**

**Muhammad Shakeel Khan**

(222202011)

**Nimra Sajid**

(222202028)

**Submitted To:**

Ma’am Rida Bajwa

**Dated:** Jan 11th,2024

**Introduction:**

**Railway Management System**

Railway Management System, a comprehensive solution designed to revolutionize railway operations and passenger experiences. Anchored by a robust Entity-Relationship Diagram (ERD), our system seamlessly integrates key entities such as Login, Passenger, Admin, Train, Schedule, Class, and Payment. With a focus on efficiency, security, and modernization, our Railway Management System aims to streamline scheduling, enhance class offerings, and provide a secure, user-friendly platform for passengers and administrators alike.

**Features:**

1. **User Authentication:**

Secure login system for passengers and administrators.

Role-based access control ensuring appropriate privileges.

1. **Train Inquiry:**

Comprehensive train listing with details like ticket number, name, description, departure, and arrival times.

Real-time availability status of seats.

1. **Ticket Booking and Cancellation:**

Effortless online ticket booking for passengers.

Convenient ticket cancellation process.

1. **Personalized Passenger Profiles:**

User profiles for passengers with personal details.

Ability to update and manage personal information.

1. **Admin Scheduling and Management:**

Administrator control over train scheduling.

Efficient management of train routes and classes.

1. **Class-based Seating:**

Differentiation of seat classes (e.g., Business, Economy) with specific details for each class.

1. **Station Management:**

Systematic management of train stations.

Display of arrival and departure times for each station.

1. **Payment Options:**

Multiple payment methods, including card payments and EasyPaisa transactions.

1. **Ticket Details:**

Display of detailed ticket information, including source, destination, date, and time.

1. **Real-time Updates:**

Instant updates on train schedules, seat availability, and booking status.

Notifications for successful bookings and cancellations.

**Problem statement**

The traditional railway management system faces various challenges that hinder the efficiency and user experience for both passengers and administrators. These challenges include:

1. **Manual Booking and Inquiry:**

Traditional ticket booking and train inquiry methods are time-consuming and often require passengers to visit railway stations in person.

1. **Limited Accessibility:**

Lack of an online platform restricts passengers from easily accessing train schedules, checking seat availability, and making bookings from the comfort of their homes.

1. **Inefficient Administration:**

Administrators struggle with manual scheduling, managing seat classes, and overseeing station details, leading to potential errors and delays.

1. **Data Discrepancies:**

Manual data entry and management may lead to inconsistencies, inaccuracies, and difficulties in maintaining a centralized and updated record of train schedules, seat availability, and passenger information.

1. **Limited User Profiles:**

A lack of personalized passenger profiles prevents users from efficiently managing their information and preferences within the system.

1. **Outdated Information:**

The absence of real-time updates leaves passengers unaware of sudden changes in schedules, seat availability, or other critical information.

**----------------------------------------------**

**Database Implementation**

**Relational Database Management System (RDBMS):** Utilizing a strong RDBMS i.e., Microsoft SQL Server for creating tables and managing data.

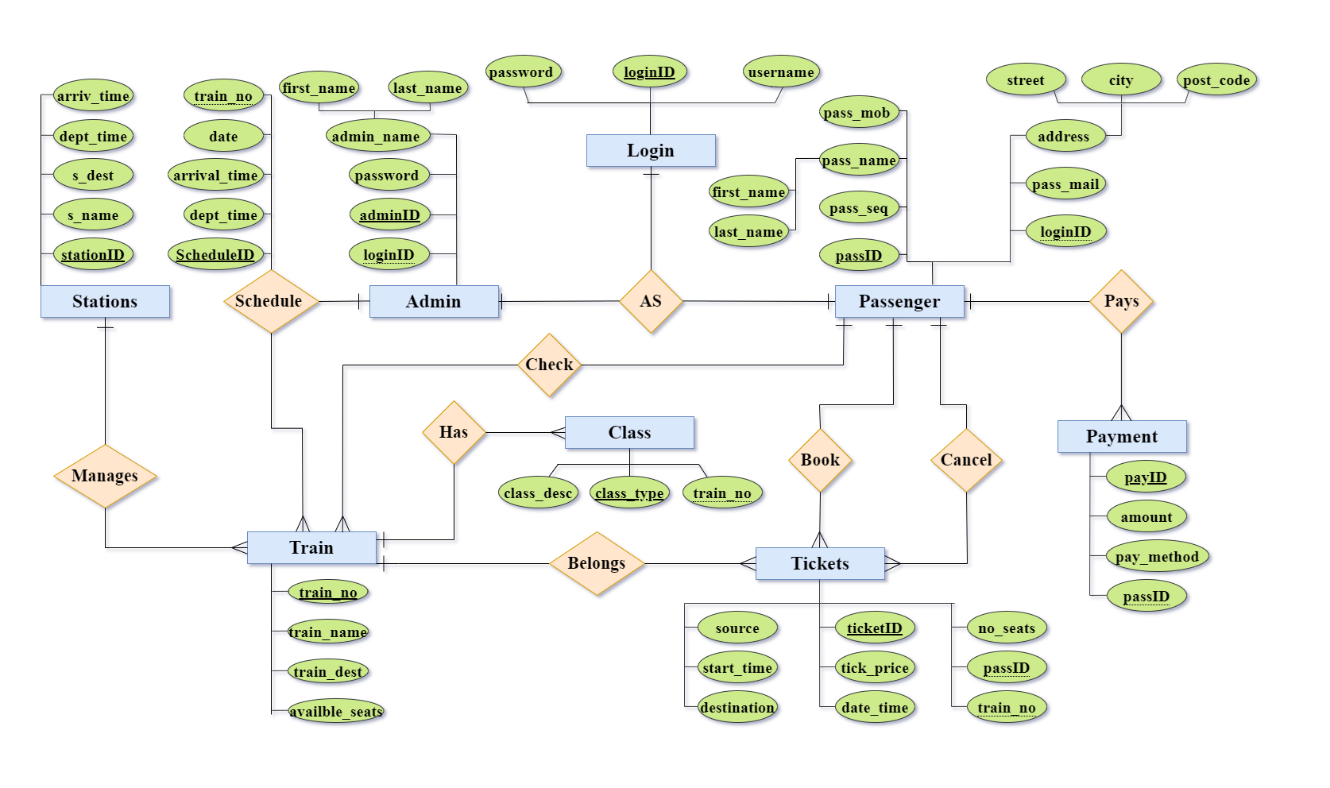
**Tables and Relationships:** Creating well-defined tables to represent entities and establishing relationships between them to model the system accurately.

**Primary and Foreign Keys:** Implementing primary keys for unique identification and foreign keys to establish relationships between tables, ensuring referential integrity.

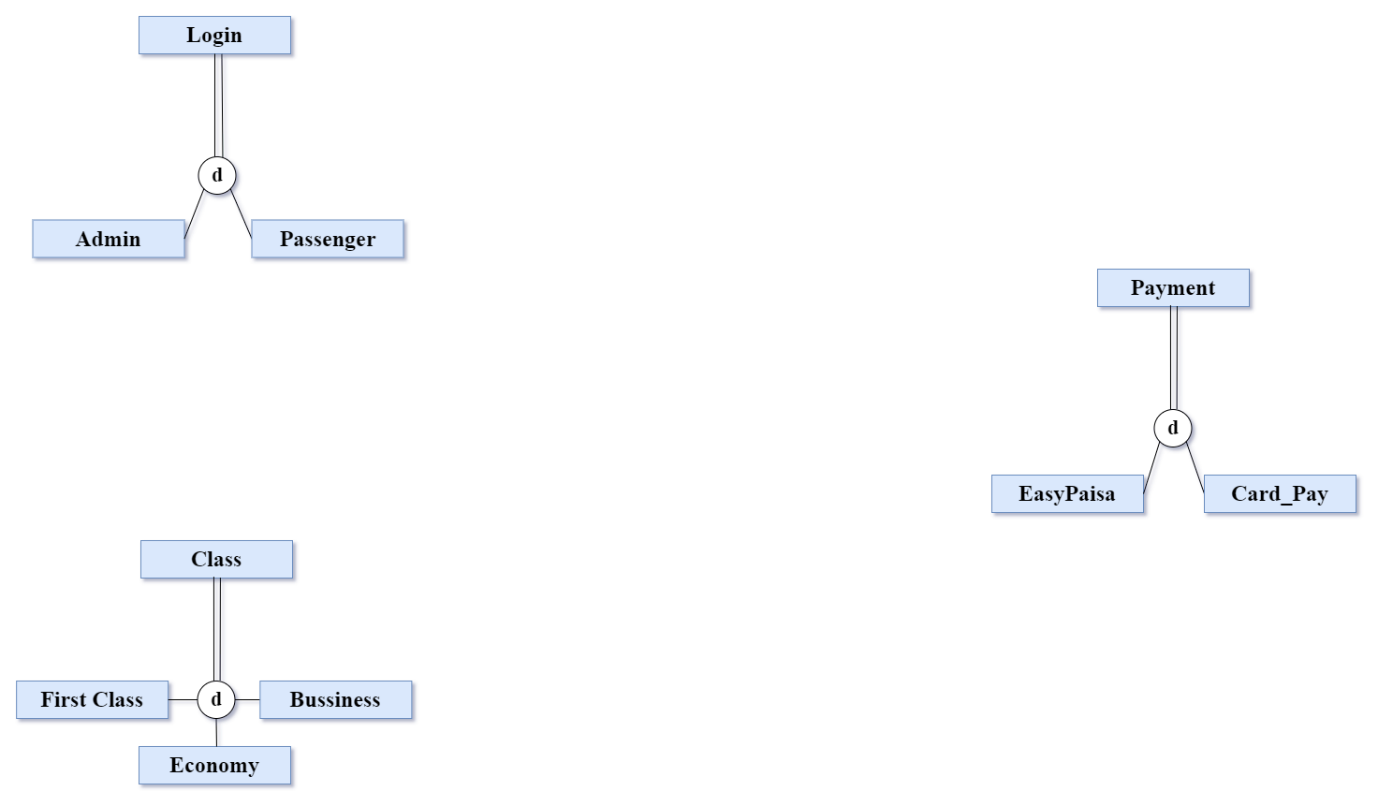
**Data Types:** Selecting appropriate data types for each attribute to ensure efficient storage and retrieval.

**----------------------------------------------**

**ERD**

**----------------------------------------------**

**EERD Concepts**



**Class:**

Class entity has subtypes representing different classes on trains. Allows passengers to choose their preferred class during booking.

**Payment:**

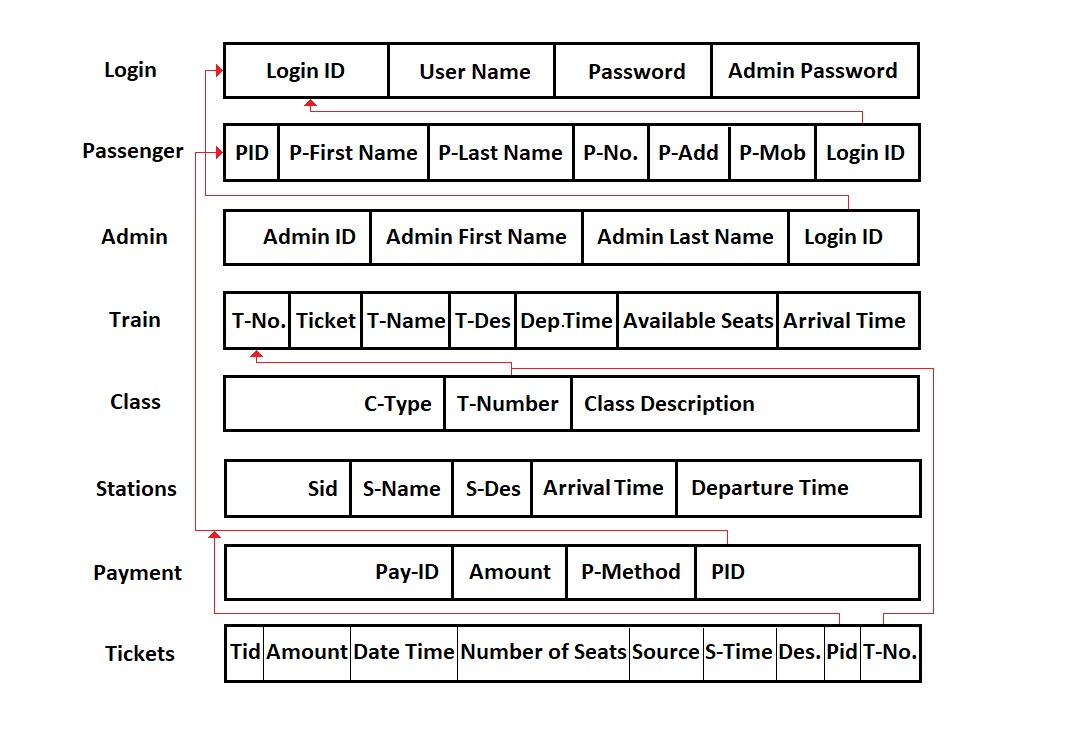
Payment entity has subtypes for different payment methods, including EasyPaisa and Card pay. Captures diverse payment options available to passengers.

**Login:**

The Login entity has subtypes, catering to both administrators and passengers. Enables a clear distinction between administrator and passenger logins.

**----------------------------------------------**

**Relational Schema:**



**Example Queries:**

**Query to fetch schedule details with source and destination**

SELECT s.schedule\_id, s.departure\_time, s.arrival\_time, s.date, t.source, t.destination FROM Schedule s JOIN Train t ON s.tno = t.tno;

**Retrieve all trains and their available seats**

SELECT Train.tno, Train.tticket, Train.tname, Train.tdes, Train.departure\_time, Train.arrival\_time, Train.available\_seats FROM Train;

**Retrieve the payment details for a specific passenger**

SELECT Payment.payid, Payment.amount, Payment.pmethod .FROM Payment

WHERE Payment.pid = {passenger\_id};

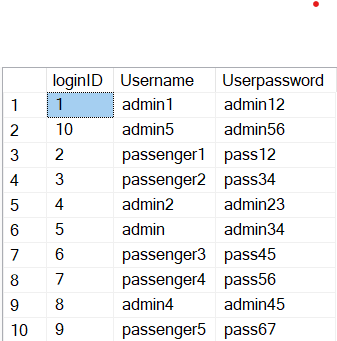
**Retrieve the classes available for a specific train**

SELECT Class.ctype, Class.class\_description FROM Class WHERE Class.tno = {train\_number};

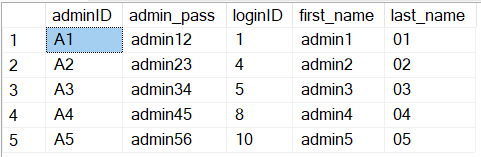
**----------------------------------------------**

**Relations With Data**

**Login Table**

****

**Admin Table**



**Passenger Table**

----------------------------------------------