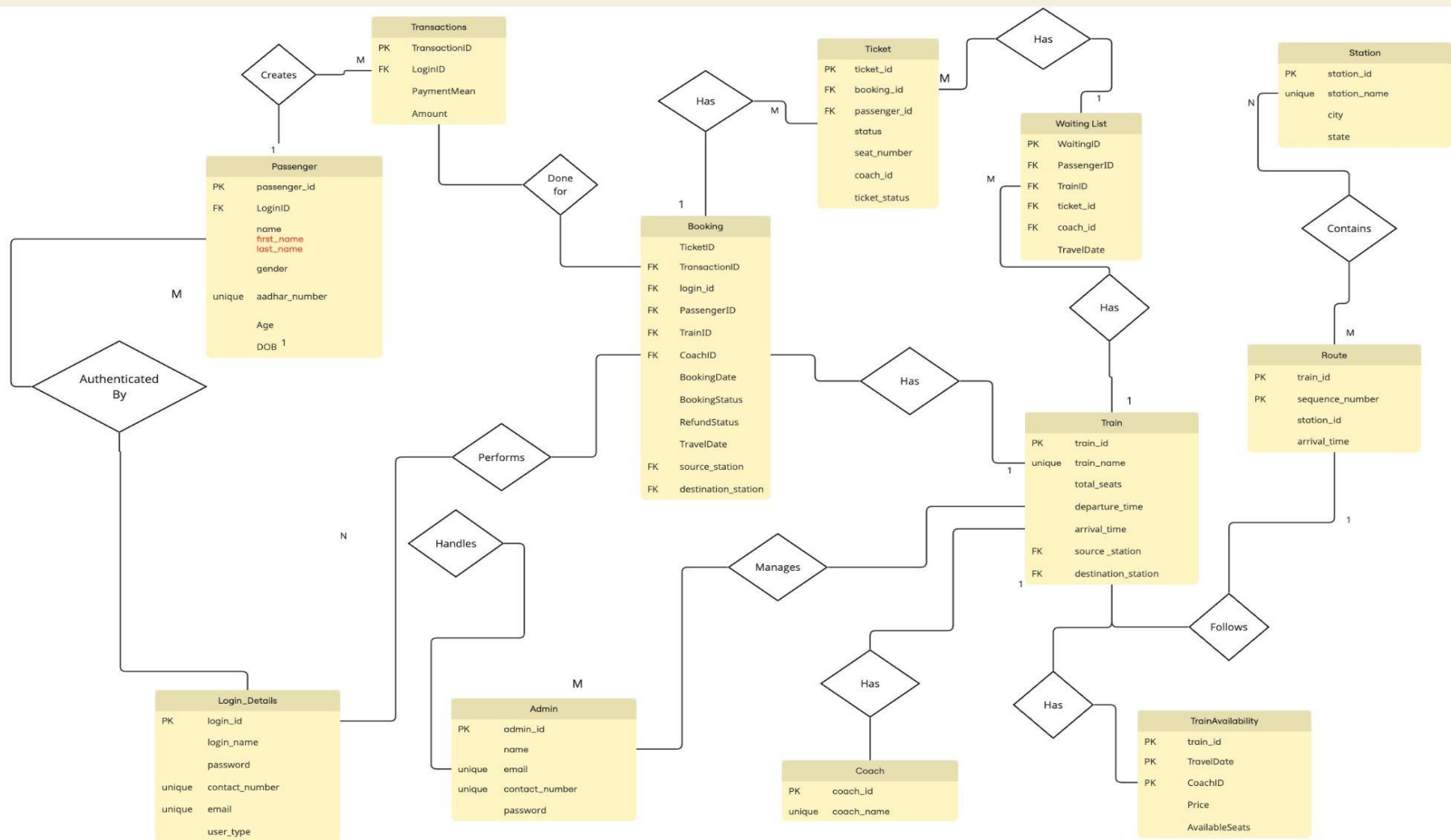


GoRail – A DBMS-Based Train Booking System

Introduction:

GoRail is a database-backed train ticket booking system inspired by IRCTC. It enables users to search for trains, book tickets, manage profiles, and track their bookings. The project demonstrates real-world DBMS application design with entity relationships, backend integration, and user-friendly interfaces.





: Primary Key
 : Foreign Key
 : Attributes

: passenger_id
 : login_id
 : train_id
 : booking_id
 : ticket_id
 : coach_id
 : station_id

Project Landing Page

The **landing page** of GoRail provides a clean and intuitive interface where users can log in or sign up to access the platform. It highlights the purpose of the system — allowing passengers to search, book, and manage train tickets with ease.

- Role Based Access: Landing Page prompts user to choose between Passenger or Admin roles, directing them to appropriate interface.
- The user data is begin fetched from the backend connection to DBMS, which validates the login.
- Acts as a gateway to role specific work

User Functionalities

1. **User Profile:** View and edit personal information like name, email and contact details.
2. **View Past Transactions**
3. **Train Search and Bookings:** Search for trains based on specific dates, source and destination.
4. **Book ticket and View ticket :** Select train and class, enter passenger details
5. **Manage Bookings, Cancel ticket.**

Admin Functionalities

1. **Manage Passengers** : View list of registered users
2. **Manage Bookings** : View all passenger bookings
3. **Transactions** : Process payments and funds
4. **Manage Trains** : Edit train details. Search trains based on city, stations, arrival time. Manage train finances.
5. **Manage Routes** : View train routes

Tools and Technologies

Database

- **MySQL Workbench** : Used for designing and implementing the relational database schema, running SQL queries and managing data visually.

Frontend

- **HTML** : For Creating the structures of pages
- **CSS** : For styling and layout design, ensuring a clean UI
- **JavaScript** : For adding interactivity and dynamic behavior to web pages.

Backend

- **Node.js** : JavaScript runtime for building the backend server and handling user requests.
- **Express.js** : Node.js web framework used to build APIs and handle HTTP requests and routing efficiently.

Project Requirements Changes

- Removed Complaints table from ER-diagram
- Replaced transactions instead of payments
- Mapped PassengerID with LoginID

THANK YOU

Team Members

- **Aakash Chityala:** Took on the most *mission-critical* task — making the final PPT after the project was already completed. A true legend of *last-minute presence*. Also offered top-tier *emotional support* during the grind (when available).
- **Harshit Anand:** Implemented the *View Bookings*, *Booking Details*, and *Ticket Cancellation* features.
- **Abhas Gupta:** Built and managed the complete *Admin Dashboard*.
- **Divyanshu Yadav:** Developed the *Edit Profile* and *Book Ticket* functionalities, and also managed the project's *GitHub repository*.