DBMS : **Project Task 1**

Names & Roll. No : Mayank Rana (2024341), Rishi Raina (2024465) and Mukul (2024359)

Project Group : 50

Tutorial Group : 46

Assigned TA(s) : Raghav Verma and Rahul Pardasani

Ride Management System – Business Requirements

A ride-hailing company similar to Uber or Ola plans to build a centralized database-driven application to manage its on-demand transportation services. The objective of the system is to efficiently connect users who request rides with available drivers, manage ride execution, handle payments and support administrative and operational decision-making.

The company operates through a central Company Office that manages its workforce and oversees day-to-day operations. The company employs multiple Employees who are categorized into Drivers and Technical Team members based on their roles. Managers, appointed by the company office, supervise employees and monitor platform operations. Employee records maintain information related to roles, employment details and reporting structure within the organization.

Users are independent entities in the system and represent customers who use the platform to request rides. A user registers on the platform by providing personal and contact details and is associated with an Account that stores authentication credentials and wallet balance. The account is modelled as a weak entity since it cannot exist without a corresponding user. Users may request multiple rides over time.

Drivers are employees of the company who provide transportation services to users. Each driver is associated with one or more Vehicles that are registered in the system with information such as vehicle type and capacity. A driver may accept or reject ride requests based on availability. At any given time, a driver can be assigned to at most one active ride. Drivers are also associated with accounts that are used for handling payments and balances.

When a user requests a ride, the system creates a Ride record that captures details such as pickup location, drop location, ride status and timestamps. The ride request is processed by the system and an available driver may accept or reject it. Once accepted, the ride is associated with both the requesting user and the assigned driver. Each ride is completed or cancelled using a specific vehicle.

The system supports payment processing at different stages of a ride. A Payment record may be generated when a ride is completed or when a ride is cancelled after a certain point, in which case a percentage of the estimated fare may be charged as a cancellation fee. Payment records store information related to fare amount, payment method, transaction status and timestamp. Both successful and failed transactions are recorded for auditing and analysis. Analytical queries may be performed on payment data to evaluate revenue, demand patterns and system performance.

After a ride is completed or cancelled, users can provide feedback in the form of Ratings for drivers. Ratings are set as weak entities and are associated with rides. The system stores rating information and uses it to assess driver performance and improve service quality.

Entity – Relationship Diagram