DBMS : Project Task 1

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

Project Group : 46 , Tutorial Group : 4

Assigned TA(s) : Raghav Verma and Rahul Pardasani

Ride Management System

Introduction

The Ride Management System is a comprehensive database driven solution designed to manage the complete operational workflow of a ride hailing platform. It supports key functionalities including user registration and account management, employee and organizational hierarchy, driver and vehicle management, ride request and lifecycle handling, payment processing, ratings and feedback, and reporting and analytics. The system is built with well defined business rules and technical constraints to ensure data integrity, security, and reliability. Critical operations such as ride assignment, account updates, and payment transactions are implemented using ACID compliant mechanisms to prevent partial failures and maintain consistency. Furthermore, the system is designed to be scalable, maintainable, and secure, enabling efficient handling of large scale transactional and analytical workloads.

Business Requirements

Functional Requirements

1. User Registration and Account Management

Description:
The system shall support user onboarding and account management.

Business Constraints

- Each user shall have exactly one account.
- An account shall not exist independently of a user.

Technical Constraints

- User identifiers must be system generated and unique.
- Account creation must be transactional to prevent partial persistence.

Frequently Used Queries

- Retrieve user profile by user ID
- Fetch account balance for a user
- Validate user authentication credentials

---

2. Employee and Organizational Management

Description:
The system shall manage employees and organizational hierarchy.

Business Constraints

- Every employee must have a defined role.
- Each employee shall report to at most one manager.

Technical Constraints

- Employee roles shall be implemented using role based access control .

Frequently Used Queries

- List employees under a manager
- Retrieve employee details by role
- Count active drivers

---

3. Driver and Vehicle Management

Description:
The system shall manage driver availability and vehicle registration.

Business Constraints

- A driver may be assigned to at most one active ride at a time.
- Each vehicle must be owned by exactly one driver.
- A driver may be associated with zero or more registered vehicles.

Technical Constraints

- Driver availability must be updated atomically during ride assignment.
- Vehicle capacity must be validated against ride requirements.

Frequently Used Queries

- Find available drivers
- Fetch vehicles registered to a driver
- Check driver availability status

---

4. Ride Request and Lifecycle Management

Description:
The system shall manage the complete lifecycle of a ride.

Business Constraints

- A ride must be associated with exactly one user.
- A ride may be associated with at most one driver.
- Each ride must use exactly one vehicle.

Technical Constraints

- Ride status transitions must follow a predefined state model.(REQUESTED , ACCEPTED, IN PROGRESS , COMPLETED) .
- Timestamp fields must be system generated and immutable once set.

Frequently Used Queries

- Retrieve active rides
- Fetch ride history for a user
- Get current ride for a driver

---

5. Payment Processing

Description:
The system shall process ride payments and cancellation fees.

Business Constraints

- Payment shall be generated only after ride completion or eligible cancellation.
- Cancellation fees shall be calculated based on configurable policies.

Technical Constraints

- Payment records must be immutable once created.
- Both successful and failed payments must be persisted.

Frequently Used Queries

- Retrieve payments by ride ID
- Calculate total revenue over a time period
- Identify failed transactions

---

6. Ratings and Feedback

Description:
The system shall allow users to rate drivers.

Business Constraints

- A rating may exist only after a ride is completed or cancelled.
- Each ride may have at most one rating.

Technical Constraints

- Ratings must be existence dependent on rides (weak entity).

- Ratings must not be modifiable after submission.

Frequently Used Queries

- Fetch average rating for a driver
- Retrieve rating for a ride
- Rank drivers by rating

---

7. Reporting and Analytics

Description:
The system shall support operational and analytical reporting.

Business Constraints

- Historical data must not be deleted prematurely.
- Reports must reflect only committed transactions.

Technical Constraints

- Analytical queries must not block transactional operations.

Frequently Used Queries

- Daily ride count
- Revenue per region or time window
- Driver utilization rate

Non-Functional Requirements

1. Performance

Constraints

- Ride assignment queries must return results within acceptable latency.
- Payment processing must not block ride completion.

---

2. Scalability

Constraints

- The system must scale to a very large volume of rides and transactions.

---

3. Reliability and Consistency

Constraints

- Ride, payment, and account updates must be ACID compliant.
- Partial failures must not corrupt the system state.

---

4. Security

Constraints

- Role based access control must be enforced.
- Payment and credential data must be encrypted.

---

5. Data Integrity

Constraints

- Weak entities must not exist without parents.
- Orphan records are not permitted.

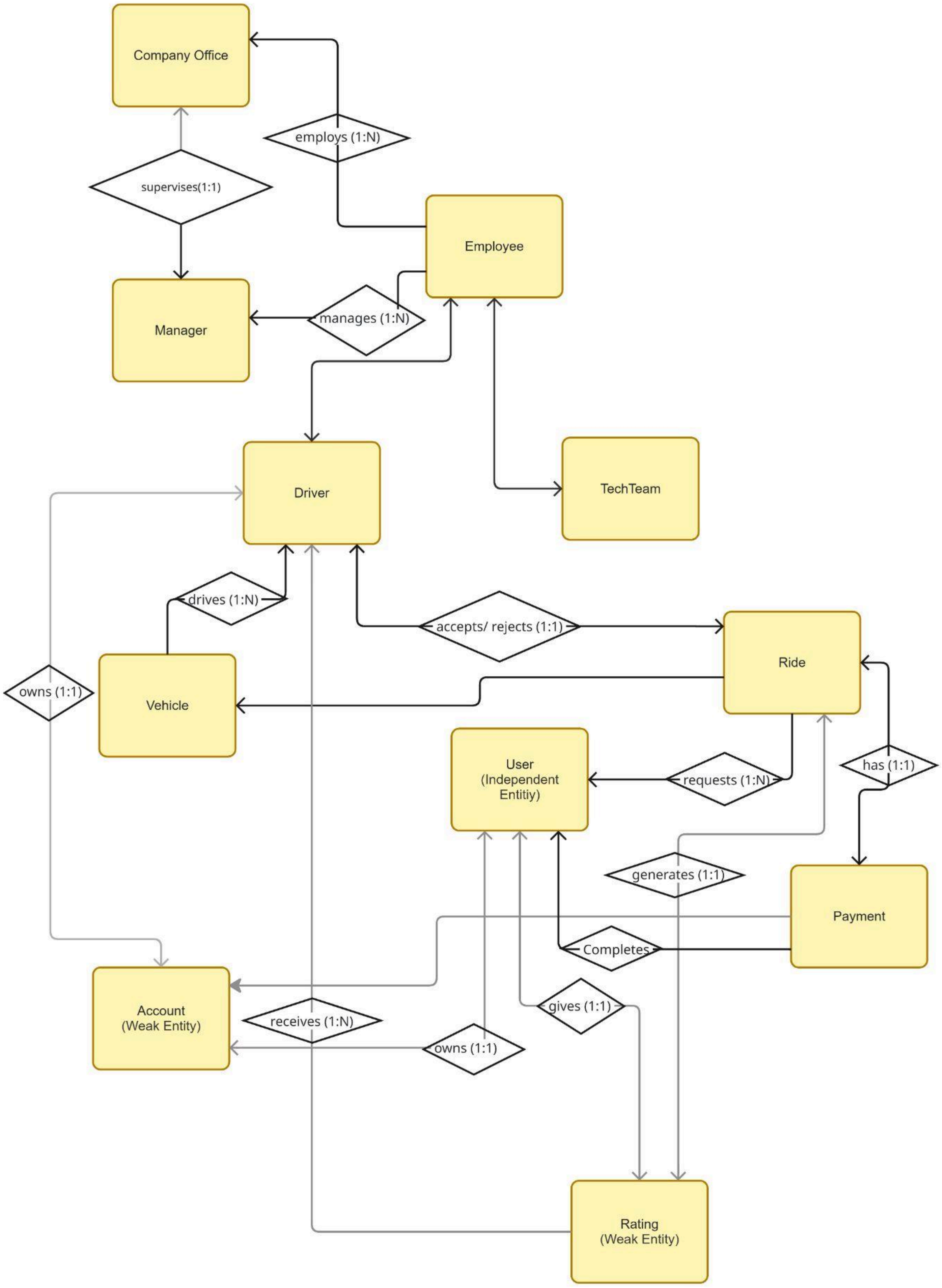
---

6. Maintainability and Configurability

Constraints

- Business rules must be configurable without schema changes.
- Code and schema must support modular updates.

ER DIAGRAM :


Company Office
employs (1:N)
supervises(1:1)
Employee
Manager
manages (1:N)
Driver
TechTeam
drives (1:N)
accepts/ rejects (1:1)
Ride
owns (1:1)
Vehicle
User
(Independent
Entitiy)
requests (1:N)
has (1:1)
generates (1:1)
Payment
Completes
Account
(Weak Entity)
receives (1:N)
owns (1:1)
gives (1:1)
Rating
(Weak Entity)