**National University of Computer & Emerging Sciences**

**Karachi Campus**



**Project Report**

**Vehicle Rental System**

**Section: BSE-4A**

**Group Members:**

**23k-3012 Umer Ahmed Shaikh**

**23k-3014 Abdul Jibran**

**23k-0529 Omer Khalid**

**1. Introduction**

The Vehicle Rental Management System is a full-stack web application designed to streamline the process of renting vehicles online. It caters to two main user roles: Admin (responsible for managing the platform) and Customer (responsible for renting vehicles). The system ensures smooth operations of vehicle rentals through a centralized interface, robust database design, and intuitive user experience.

**2. Objectives**

* To provide an online platform for renting vehicles.
* To simplify vehicle and maintenance management for administrators.
* To maintain accurate rental records and track payments.
* To support a scalable and secure database for business logic.
* To implement role-based access control.

**3. Features**

**3.1 Customer Features**

* User Registration & Login with secure password hashing.
* Browse & Search Vehicles by model, type, brand, or availability.
* Rental Booking with date selection and fee calculation.
* Review System to rate and comment on completed rentals.
* View Rental & Payment History.

**3.2 Admin Features**

* Vehicle Management: Add, update, or remove vehicles.
* Maintenance Management: Track and manage vehicle maintenance.
* Rental Monitoring: Track ongoing, completed, and cancelled bookings.
* Business Statistics : Get up to date business stats, helpful in making business decisions.
* Admin Logs: Record admin activities (add, delete, update).

**4. System Design**

**4.1 Front-End**

* Developed using React.js and Tailwind CSS.

**4.2 Back-End**

* Implemented using Node.js and Express.js.
* Handles business logic for user roles, rental processing, and data validation.

**4.3 Database**

* PostgreSQL with a normalized relational schema.
* Major tables:
  + users, customer, admin
  + vehicle, rental, payment, review, maintenance\_record, admin\_logs
* Uses ENUM types for controlled value inputs (e.g., rental\_status, payment\_status, user\_type).

**5. Data Integrity & Security**

* Passwords are securely stored using bcrypt hashing via pgcrypto.
* Relational constraints ensure data consistency and prevent duplicates (e.g., unique email, license number).
* Foreign keys enforce relationships between tables.
* ENUM types prevent invalid statuses and improve query reliability.

**6. Challenges & Solutions**

* Challenge: Maintaining accurate rental and payment flow.
  + Solution: Enforced constraints and logical validation at both database and application layers.
* Challenge: Admin activity tracking.
  + Solution: Introduced a logging mechanism via the admin\_logs table with timestamped actions.

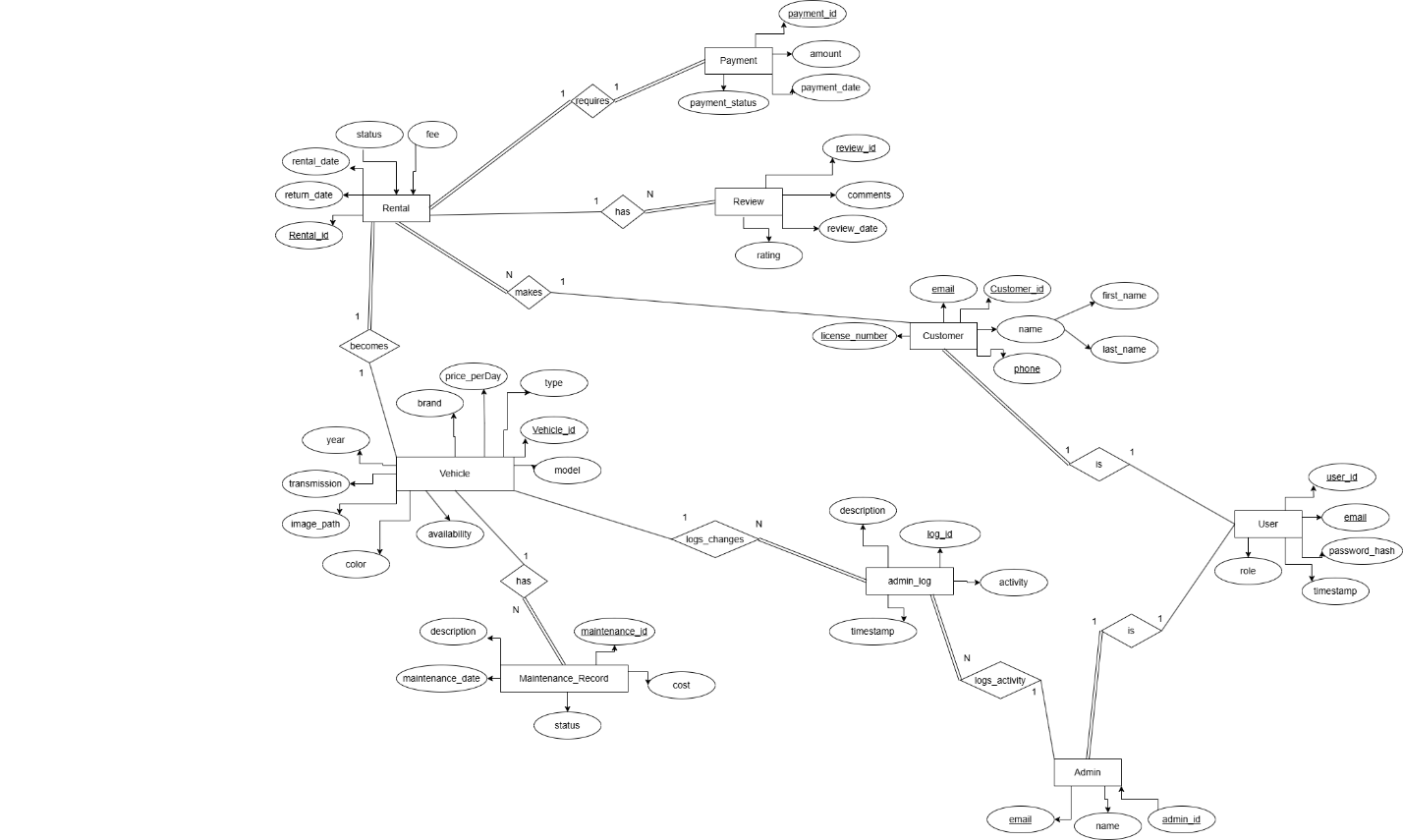
**7. Conclusion**

This project successfully demonstrates the design and implementation of a comprehensive vehicle rental platform. It supports real-world functionalities like vehicle management, rental scheduling, payment processing, and administrative oversight. The use of a normalized database, secure authentication, and clean separation of concerns ensures both robustness and scalability.

**8. Future Enhancements**

* Integration with real payment gateways (e.g., Stripe).
* SMS/email notifications for booking status and payment confirmations.

**9. ER diagram**

****

**10. Class diagramA diagram of a computer flowchart

AI-generated content may be incorrect.**