P2: Database Design (Initial ERD)

BLUE BIKE DATABASE MANAGEMENT SYSTEM

ABSTRACT

How can a bike-share system's network dynamics generate information to improve its inventory? In this work, we simulate station demand and route flow in relation to time and weather. These models are used to make both long-term and short-term choices on rider incentives. Such incentives are intended to increase inventory, which in turn increases service satisfaction and the company's profit. As a result, we recommend employing a visualization to aid decision making about these incentives throughout the day.

MISSION STATEMNET

The typical station-based bike-sharing system will be the subject of this project. Blue Bikes is a bicycle sharing program in Boston, Massachusetts. The bike-sharing scheme began on July 28, 2011. This program was designed for anyone who needed it on a short-term basis for a fee. Individuals can borrow a bike from a dock station and return it to another dock station after using it.

Every bike-sharing station has a varying demand depending on the time of day. When some of the stations are full of bikes, riders are unable to park their bikes. During busy hours, however, several of the stations have no available bikes. The operator utilizes force-balancing, which implies a bike moving truck, to maintain its balance. The balance is necessary for the optimal use of a station's assets. In the near run, an uneven bike inventory reduces potential users' chances of riding a bike. In the worst-case scenario, people abandon their expectations of the bike-sharing system eventually, and no one wants to use it.

OBJECTIVE

The scope of a blue bike database management system would depend on the specific requirements and goals of the blue bike sharing service. However, some common elements of the system's scope could include:

• Storing and maintaining bike availability information, such as bike location, status (e.g., available, in use, in repair), and any associated facts.

• Keeping track of client data, such as contact information, Rental, and payment information.

• Recording and maintaining rental transactions, including rental start and finish times, bikes involved, and costs charged.

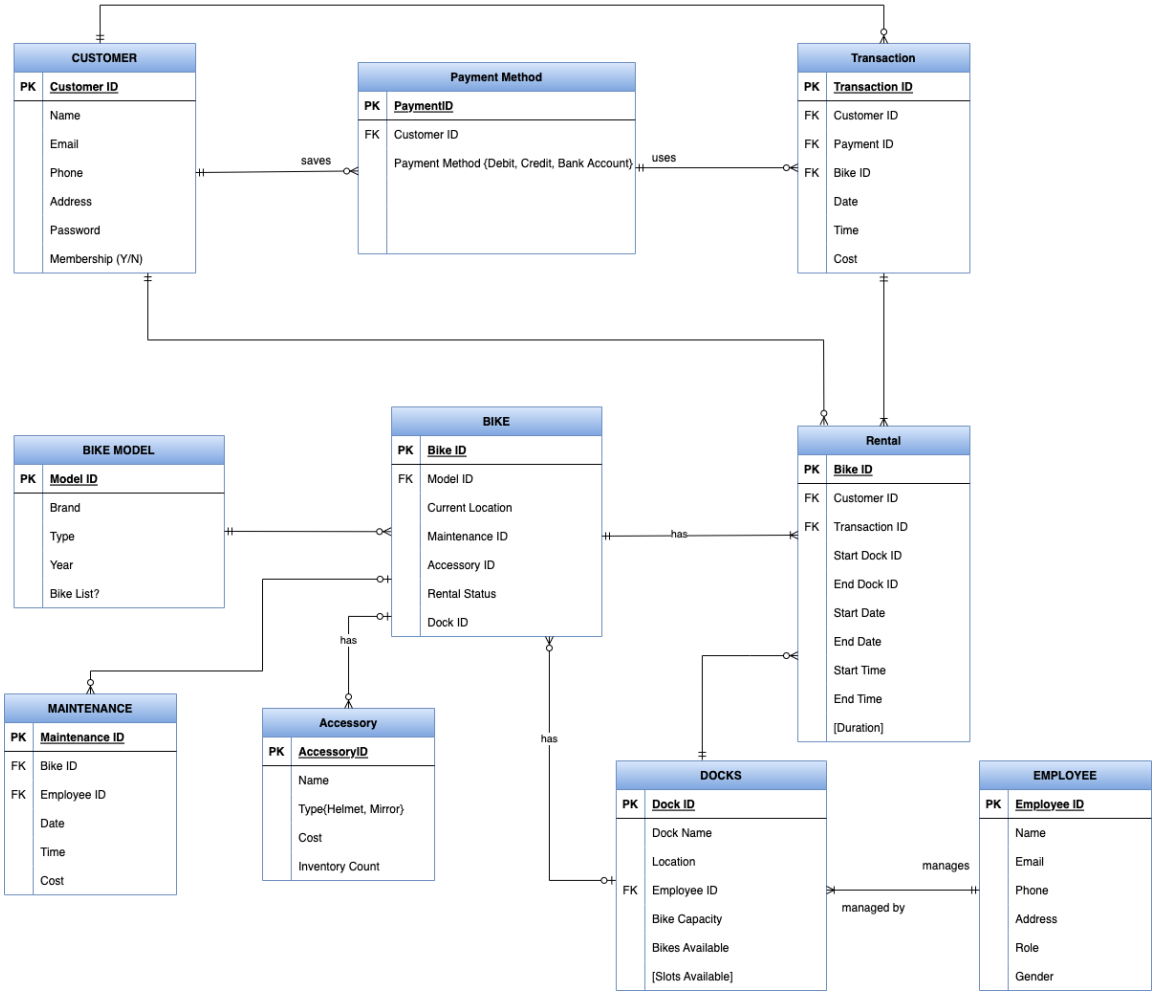
• Keeping track of maintenance data, such as when bikes were serviced, what was done, and any associated expenditures.

• By analyzing bike usage trends, identifying locations where bikes are often used or in high demand, and monitoring the financial success of the bike sharing service, real-time information and insights are provided to assist wise business decisions and enhance operational efficiency.

The specific scope of the blue bike database management system would depend on the requirements and goals of the blue bike sharing service.

ER Diagram

This ERD includes the basic entities and relationships of a Blue Bike management database system.



ENTITIES

1. **Customer**: The Customer entity represents the people who rent the bikes. It contains personal details of the customer. It has attributes such as Name, Email ID, Phone, Address, Password, Membership Status. It is associated with Payment Method, Transactions, and Rental.
2. **Payment Method**: This entity represents various payment methods used for bike rentals by a customer. It has attributes such as the payment method, amount, and Transactions. It is associated with other entities such as Customer and Transactions.
3. **Bike:** This entity represents a bike that can be rented. It has attributes such as the model, serial number, current location, and condition, and it is associated with other entities such as Bike Model, Maintenance, Accessory, Rental, and Docks.
4. **Bike Model:** This entity represents the different bike models, which have attributes such as the brand, type, and year. It is associated with only one entity, i.e., Bike, as a Bike will have a Bike Model.
5. **Docks**: This entity represents a location where bikes can be rented from and returned to. It has attributes such as the name, location, capacity, number of bikes available, and dock supervisor. It is associated with other entities such as Bike and Employee, as it will have multiple bikes and an employee responsible for managing the docks.
6. **Employee:** This entity represents an employee responsible for managing the docks and maintaining bikes**.** It has attributes such as the name, email, phone, address, role, and gender. It is associated with other entities such as Docks and Maintenance as an Employee is responsible for managing Docks and completing maintenance on bikes.
7. **Maintenance:** This entity represents the maintenance history of a bike, which includes the bike id, employee id who did the maintenance, date, time, and cost. It is only associated with Bike as an Employee is responsible for performing maintenance on bikes, and a Bike can have multiple maintenance histories.
8. **Accessory:** This entity represents an accessory put on a bike, such as a helmet, basket, or lock**.** It has attributes such as the name, type, cost, and inventory count. It is associated with the Bike entity, as an accessory can be compatible with multiple bikes.
9. **Rental:** This entity represents the rental transactions of a bike, which includes the bike ID, customer ID, start and end date and time, start and end dock, and duration as a derived attribute. It is associated with other entities such as Bike, Customer, and Transactions.
10. **Transaction:** This entity represents the Transactions of the bike rental, which includes the payment date and time, amount, customer, and bike for the transaction. It is associated with other entities such as Customer, Payment Method, and Rental.

RELATIONSHIPS

- A Customer can have multiple Payment Method (1:M)
- A Customer can have multiple Transaction (1:M)
- A Customer can have multiple Rental (1:M)
- A Payment Method can have multiple Transaction (1:M)
- A Payment Method will be associated with only one Customer (1:1)
- A Transaction will be associated with only one Customer (1:1)
- A Transaction will have only one Payment Method (1:1)
- A Transaction will be associated with multiple Rental (1:M)
- A Bike Model can have multiple Bikes (1:M)
- A Bike can be of only one Bike Model (1:1)
- A Bike can have multiple Maintenance records (1:M)
- A Bike can have multiple Accessories (1:M)
- A Bike can be stationed at only one Dock (1:1)
- A Bike can have multiple Rental (1:M)
- A Rental will be associated with only one Transaction (1:1)
- A Rental will be associated with only one Customer (1:1)
- A Rental will be associated with only one Bike (1:1)
- A Rental will have a Dock (1:1)
- A Maintenance record will be of only one Bike (1:1)
- An Accessory will be associated with only Bike (1:1)
- A Dock can have multiple Bikes (1:M)
- A Dock will be managed by only one Employee (1:1)
- A Dock can have multiple Rentals (1:M)
- An Employee will manage multiple Docks (1:M)