Use case no:JP036

Title: Bike rental system

1. Introduction

Transport facility is a matter of headache for those people who do not have any personal transport.

On occasions like Wedding, Vacation, house shifting, and tour and on many other situations they feel the necessity of a bike to sort out the problems. So if it is possible to design or develop a web based application for availing transport whenever and wherever possible, then it will be beneficial for both renter and transport provider. Now a days, by some clicks only, we can get whatever you want at home. We already know about the online shopping, e-banking etc.

Similarly,

The Bike Rental System is the online facility to book bike/scooter online within few clicks only.

Some people can not afford to have a bike, for those people this system becomes very helpful.

This system includes various bike/scooter, as per the customer order and comfort, it place the order

and deliver the bike as per the location within the area. Booking can be done via internet service only.

2.Technologies Used:

- **Front-end development:**
  - HTML
  - CSS
  - JavaScript ○ Bootstrap ○ React.js
- **Back-end development:**
  - JSP ○ Java ○ MySQL ○ Node.js

- **BikeRide caters to two main user categories:**

  - Admin: Manages bike inventory, approves rental requests, and performs CRUD operations on bike details. ○ Customer: Registers, searches for available bikes based on location and dates, rents bikes, and views booking details.

3. Use Case:

- **Scenario:** A user in a city wants to rent a bike for a short commute to work. They don't own a bike and prefer a convenient option compared to public transportation.
- **Steps:**
  1. The user opens the app and creates an account or logs in if they already have one.
  2. They use the app to search for available bikes near their current location and filter by their preferred type (e.g., road bike, e-bike).
  3. The user selects a nearby bike and books it for their desired rental duration.

4. The app provides instructions on how to unlock the bike using their smartphone (e.g., QR code or NFC).
5. The user unlocks the bike, rides to their destination, and locks the bike upon completion of their trip.
6. The app automatically ends the rental and charges the user's payment method based on the chosen duration and applicable rates.

- **Success Criteria:**
  - The user successfully rents a bike, completes their journey, and returns the bike without any issues. ○ The rental process is quick, convenient, and user-friendly through the mobile app.

4. Functional Requirements:

4.1 User (Rider) Section: •

Registration:

- Users create accounts by providing essential information (name, email, phone number). ○ Secure password creation with minimum character length and complexity requirements.
- Account confirmation through email verification.

- **Bike Search:**
  - Search for available bikes based on location, date, and time.
  - Filter results by bike type (e.g., mountain, road, e-bike), features (gears, baskets), and **availability at docking stations** (if applicable).
  - Display search results with detailed information like bike type, picture, location (including docking station details), and price.
- **Booking:**
  - Reserve a chosen bike **at a specific docking station** (if applicable) for a specific duration.
  - System generates a unique booking ID upon successful booking.
  - Option to modify or cancel a booking before a designated timeframe.
- **Unlocking and Locking:**
  - Utilize the mobile app (e.g., QR code, NFC) to unlock and lock rented bikes **at designated docking stations** (if applicable) or designated parking areas (if not).
  - Display basic bike information and features within the app.
- **My Bookings:** ○ View a list of current and past bookings, including:
    - Bike details
    - Rental dates and times
    - Docking station details (if applicable)
    - Booking status (pending, ongoing, completed, cancelled)
    - Total rental cost
- **Payment:** ○ Similar to previous, with integration with secure payment gateways.
- **Reporting Issues:**
  - Report any bike malfunctions or concerns through the app, including the option to upload photos/videos for better assistance.

4.2Admin Section: (Similar to previous, with slight modifications)

- **Login:**
    - Admin login using username and password. •

Bike Management:

- Add new bikes to the system, including details like type, features, **docking station location** (if applicable), pricing, and images. ○ Edit existing bike information (e.g., update location, add/remove images). ○ Track bike availability and location (real-time tracking optional) based on **docking stations** (if applicable) or GPS location (if not). ○ Mark bikes unavailable for maintenance or repairs.

- **Rental Management:** ○ View list of pending customer rental requests. ○ Approve or reject individual rental requests. ○ View details of approved rentals.
    - View Payment History. **4.Non-Functional**

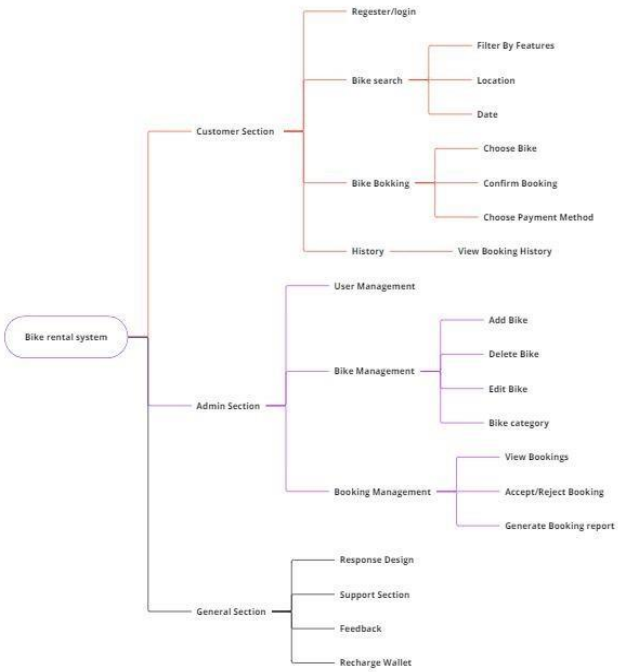
Requirements:

- **Security:**
    - Secure user authentication and authorization. ○ Regular security checks and vulnerability assessments.
    - Performance: ○ Responsive system handling user requests efficiently.
    - Quick display of search results.
- **Scalability:** ○ Accommodate future growth in user base and data volume.
- **Usability:** ○ User-friendly interface with clear navigation and intuitive functionalities.
    - Responsive design for optimal viewing across different devices.

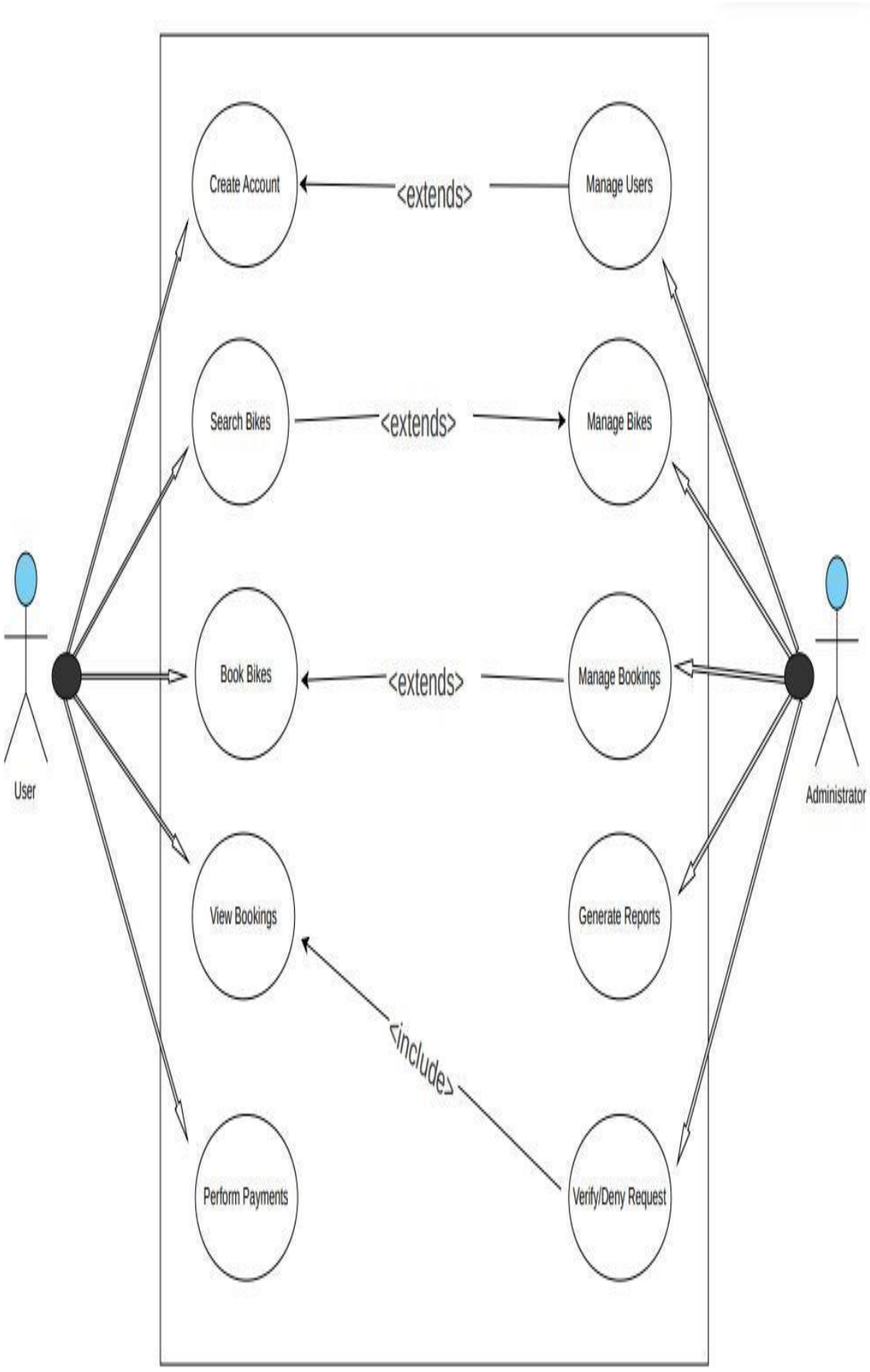
5.Additional Considerations:

- **Customer Service:** ○ Allow customers to report issues or provide feedback regarding service.

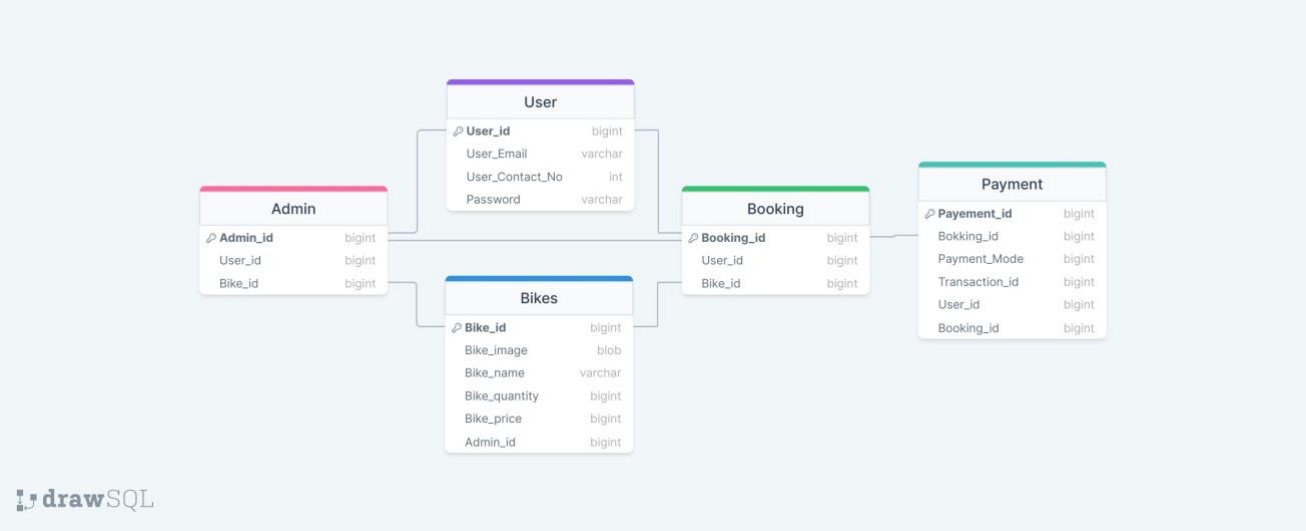
6.Mind Map:


Bike rental system
Customer Section
Regester/login
Bike search
Filter By Features
Location
Date
Bike Bokking
Choose Bike
Confirm Booking
Choose Payment Method
History
View Booking History
Admin Section
User Management
Bike Management
Add Bike
Delete Bike
Edit Bike
Bike category
Booking Management
View Bookings
Accept/Reject Booking
Generate Booking report
General Section
Response Design
Support Section
Feedback
Recharge Wallet

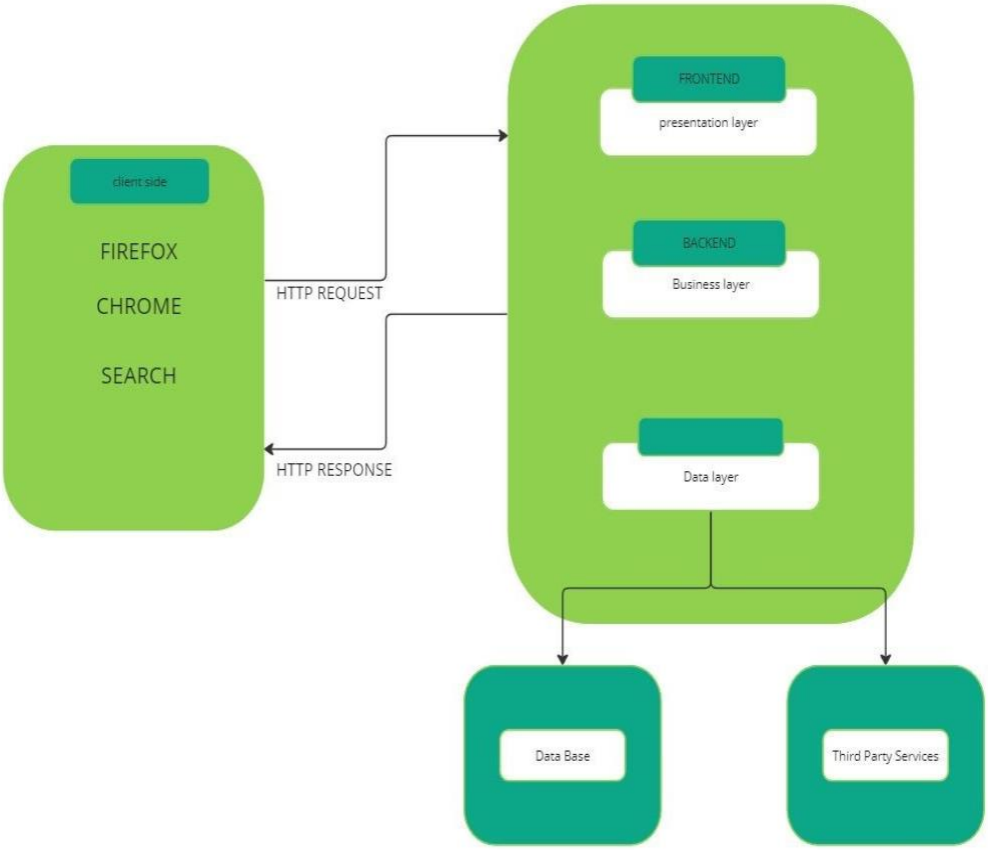
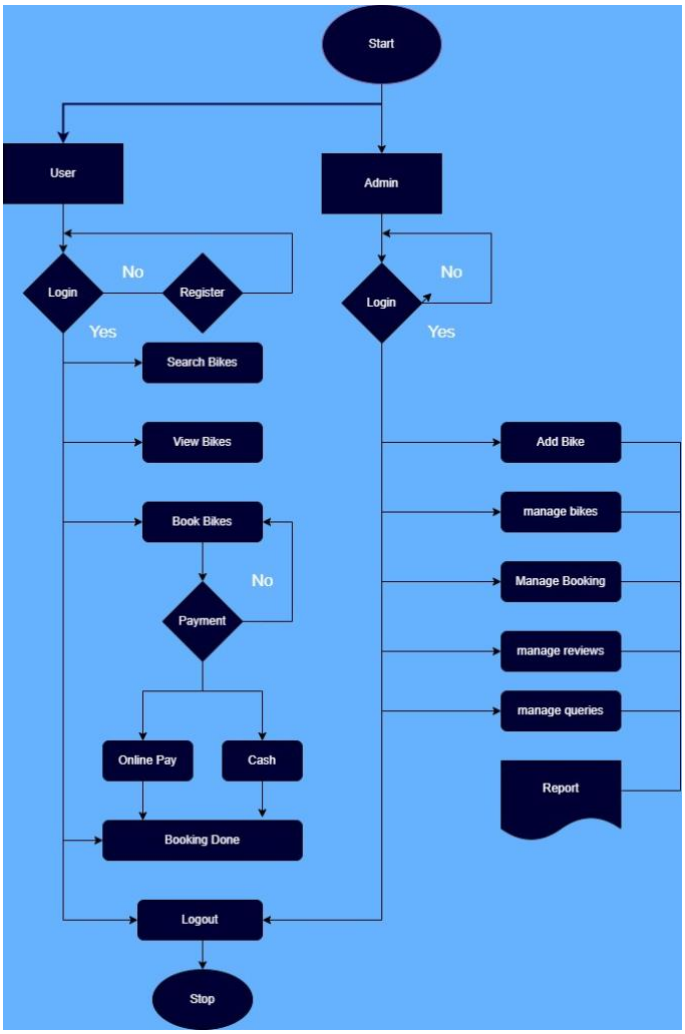
7.Use-case Diagram:



Schema Diagram



Architecture Diagram



Flow Diagram