Project Name: The ultimate bike rental experience.

Introduction:

The ultimate bike rental experience is your go-to place for bike rentals! It's designed for both admins managing the bikes and users looking to rent one easily.

Project description:

The ultimate bike rental experience is a comprehensive bike rental platform that connects users with a wide range of bikes available for rent. Admins can efficiently manage inventory, handle user accounts, and oversee rentals, ensuring a smooth experience for everyone. Users can easily browse, book, and enjoy their biking adventures, making it simple to explore the great outdoors

Key Features and Functionalities

- **Home Page**:
  - View available bikes for rent.
  - Apply coupons for discounts.
  - Spin a wheel for additional discount offers.
- **All Bikes Page**:
  - View a list of all available bikes.
  - Navigate to the bike detail page for more information.
  - Book a bike by selecting the start time and proceeding with payment.
- **User Dashboard**:
  - View and update user profile details.
- **My Rentals**:
  - View rentals under "Paid" and "Unpaid" tabs.
  - Pay for unpaid rentals directly from the list.
- **About Page**:
  - Learn about the platform's mission and vision.
  - View contact information and location details.

Admin Features

- **Bike Management**:
  - Create, update, and delete bike listings.
  - Manage bike returns and calculate rental costs.
- **User Management**:
  - Promote users to admin.
  - Delete user accounts.

These features make the platform user-friendly and efficient for both regular users and admins

Technology Stack

Frontend:

- **React**: For building user interfaces.
- **Redux Toolkit**: For state management.
- **RTK Query**: For handling data fetching and caching.
- **React Hook Form**: For managing form state and validation.
- **Ant Design (Antd)**: For pre-built UI components.
- **Tailwind CSS**: For utility-first styling.
- **Daisy UI**: For additional UI components based on Tailwind.

Backend:

- **Express**: For building the server and API endpoints.
- **MongoDB**: As the database for storing bike, user, and rental information.
- **Mongoose**: For object data modeling (ODM) with MongoDB.
- **MongoDB Aggregation**: For advanced data processing and querying.

Other:

- **TypeScript**: For type safety and better code maintainability across both frontend and backend.

Installation Step

Installation Steps for Bike Rental Project

1. **Clone the Repository**: Download the project by cloning it using Git. Navigate to the project directory.
2. **Setup Backend**:
   - Navigate to the backend folder.
   - Create a .env file to store environment variables like the MongoDB connection string and JWT secret.
   - Install necessary backend dependencies like Express, Mongoose, jsonwebtoken, etc.
   - Run the backend server using nodemon in development mode.
3. **Setup Frontend**:
   - Navigate to the frontend folder.
   - Create a .env file to store the API URL for connecting to the backend.
   - Install frontend dependencies like React, Redux Toolkit, RTK Query, Ant Design, Tailwind CSS, etc.

- Run the frontend server to start the development environment.

4. **Run the Project**:
    - Start the backend and frontend servers using the respective npm run start:dev commands.
    - The backend will run on http://localhost:5000, and the frontend will run on http://localhost:5173.

Configuration

```
PORT=5000
DATABASE_URL=mongodb+srv://bike_rental:T4dcezZ96MGo2U3h@cluster0.vthybgs.mongodb.
net/?retryWrites=true&w=majority&appName=Cluster0
CREATE_TOKEN_SECRATE=dsfhgklsdfjjjjjjjjjjjjjjjjjjjjjjjj345432kfhjk
EXPIRES_TOKEN=10d
NODE_ENV=development
SALT=10
```