4. Project

Project Name: Pizzeria

Introduction: Pizzeria is a full-stack real-time pizza delivery application developed using Node.js, Express.js, and MongoDB. The project is hosted on GitHub, Render, and MongoDB Atlas. Designed to provide a seamless pizza ordering experience, it incorporates user-friendly interfaces and robust backend functionality to support real-time updates and efficient order management.

Objective: The primary objective of Pizzeria is to facilitate a hassle-free pizza delivery system where users can order pizzas in real-time, and administrators can manage orders efficiently. This project also focuses on showcasing real-time application development and deployment on modern hosting platforms.

Features:

1. **User Management:**
   - User registration and login.
   - Admin registration and login.
   - Users and admins can update their login details.
2. **Pizza Ordering:**
   - Add pizzas to the cart.
   - Place real-time orders.
   - View order status in real-time.

3. **Admin Dashboard:**
    - Manage pizza orders in real-time.
    - Update order statuses (e.g., preparing, out for delivery, delivered).
    - Update and Add pizzas.
4. **Real-Time Functionality:**
    - WebSocket integration for instant order status updates.
    - Dynamic updates to the user and admin dashboards without refreshing.

Technology Stack:

- **Frontend:** HTML, CSS, JavaScript (EJS Template Engine).
- **Backend:** Node.js, Express.js.
- **Database:** MongoDB (hosted on MongoDB Atlas).
- **Hosting Platforms:** GitHub (source code), Render (backend), MongoDB Atlas (database).

Project Routes:

1. **User Routes:**
    - **/register**: Handles user registration.
    - **/login**: Handles user login.
    - **/logout**: Logs the user out.
    - **/update-profile**: Allows users to update their login details.

2. **Pizza Routes:**
   - **/menu**: Fetches the list of available pizzas.
   - **/cart**: Displays the user's cart.
   - **/add-to-cart/:id**: Adds a selected pizza to the cart.
   - **/remove-from-cart/:id**: Removes a pizza from the cart.
3. **Order Routes:**
   - **/place-order**: Handles order placement.
   - **/orders**: Displays the user's order history.
   - **/order-status/:id**: Fetches the status of a specific order in real-time.
4. **Admin Routes:**
   - **/admin/dashboard**: Displays the admin dashboard.
   - **/admin/orders**: Lists all orders for management.
   - **/admin/update-order/:id**: Updates the status of a specific order.
   - **/admin/dashboard**: Add Pizzas for Users.

Deployment Details:

- **GitHub Repository:** Source code is hosted for version control and collaboration.
- **Render:** Backend is hosted to handle API requests and serve the application.

- **MongoDB Atlas:** Database is hosted for scalable and secure data storage.

Conclusion: Pizzeria demonstrates the potential of real-time web applications in providing exceptional user experiences. By integrating modern technologies and deployment strategies, the project serves as a comprehensive example of a scalable and efficient delivery system. Its real-time features and seamless admin-user interactions make it a valuable solution in the food delivery domain.