**Grocery Webapp with MERN**

**Project Documentation**

1. **Introduction:**

**Project Title:** Grocery Web App

**Team Members:**

* **Bhavadharani. V–** Project Manager
* **Jayasree. K–** Frontend Developer
* **Basheera Fathimah K. F –** Backend Developer
* **Kirthana. G –** Database Administrator
* **Keerthika. B –** Testing

**2. Project Overview**

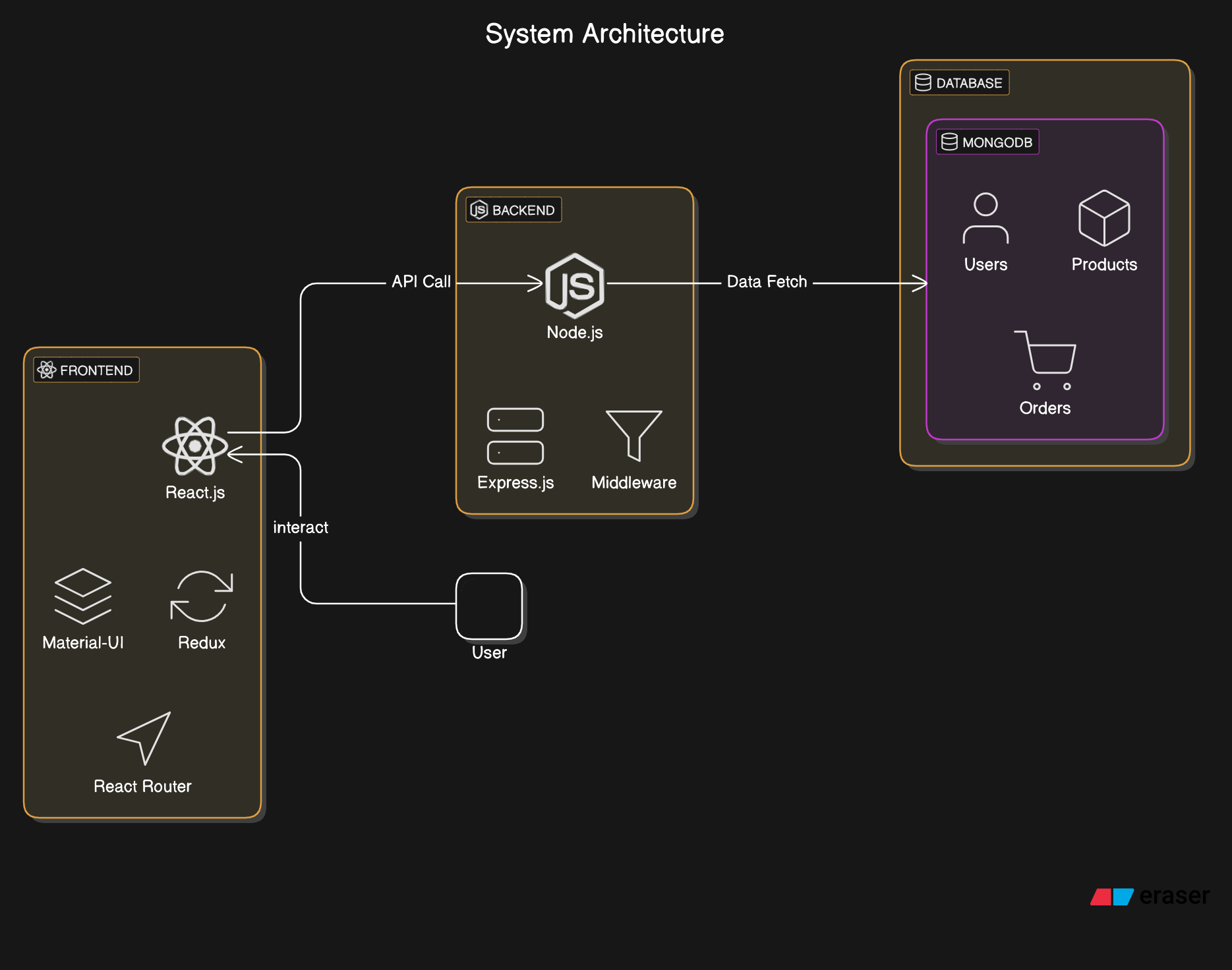
**Purpose:**

The purpose of this Grocery Web App is to provide users with a seamless platform to browse, purchase, and manage grocery items online. The app aims to simplify grocery shopping with features like smart recommendations, real-time inventory updates, and a user-friendly interface, ensuring convenience and efficiency for customers.

**Features:**

* **User-Friendly Dashboard:** A clean and intuitive dashboard for customers to navigate.
* **Product Search and Filtering:** Advanced search with filters for categories, brands, and price ranges.
* **Real-Time Inventory Updates:** Availability of products displayed dynamically.
* **Cart and Checkout:** Smooth cart management with a secure checkout process.
* **Order Tracking:** Updates on order status from placement to delivery**.**
* **Promotions and Discounts:** Personalized deals and discount codes.
* **Authentication:** Secure login/signup with token-based authentication.
* **Admin Panel:** Manage inventory, orders, and user accounts efficiently**.**

**3. Architecture**

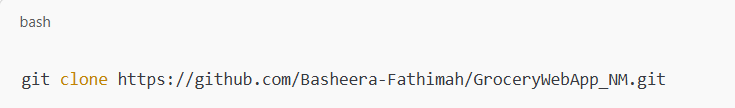
****

**4. Setup Instructions**

**Prerequisites:**

* Node.js (v14 or above)
* MongoDB (v4.4 or above)
* npm
* A code editor like VS Code

**Installation:**

1. **Clone the repository: **
2. **Navigate to the project directory:**
3. **Install dependencies for both frontend and backend:**

****

****

1. **Set up environment variables:**
   * **Create a .env file in the server folder with the following keys:**

Make file

MONGO\_URI=<MongoDB Connection String>

JWT\_SECRET=<Your JWT Secret>

1. **Start MongoDB locally or connect to a hosted instance.**

**5. Folder Structure**

**Frontend:**

**/src:**

* **components/:** Reusable React components.
* **pages/:** Pages like Home, Product List, Cart, and Checkout.
* **redux/:** State management files for actions, reducers, and stores.
* **styles/:** CSS or SCSS files for styling.

**Backend:**

* **/routes/:** Defines API endpoints (e.g., /products, /users, /orders).
* **/models/:** Mongoose schemas for Users, Products, and Orders.
* **/middleware/:** Authentication, error handling, and validation logic.
* **/controllers/:** Logic for handling requests (CRUD operations).

**6. Running the Application**

* **Frontend:  
  Navigate to the client directory and start the development server:**

**Bash**

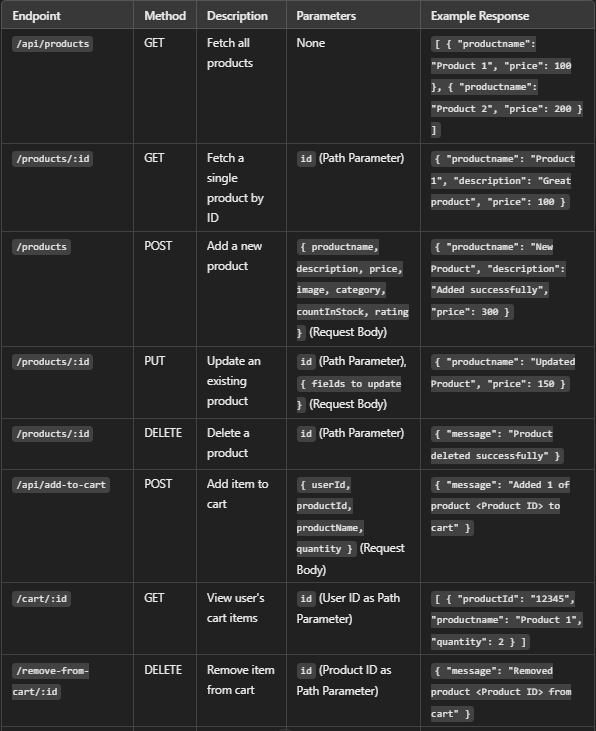
**npm start**

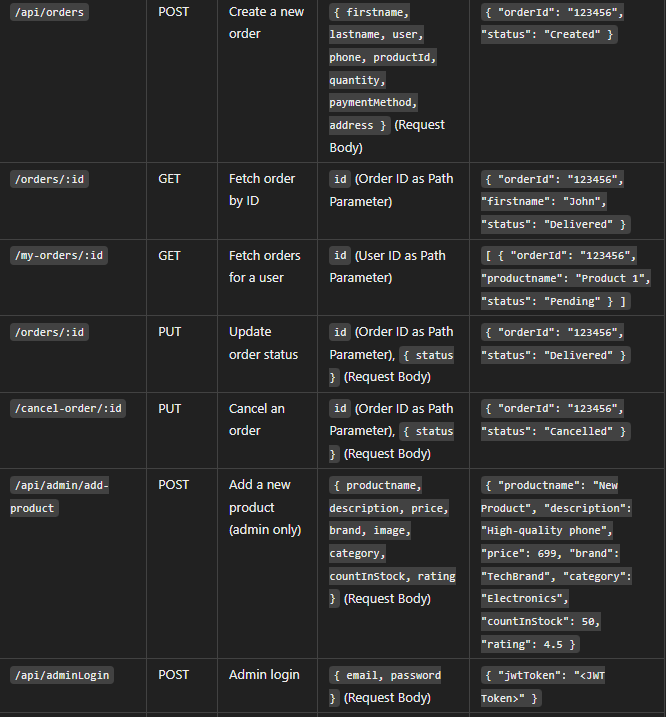
* **Backend:  
  Navigate to the server directory and start the Node.js server:**

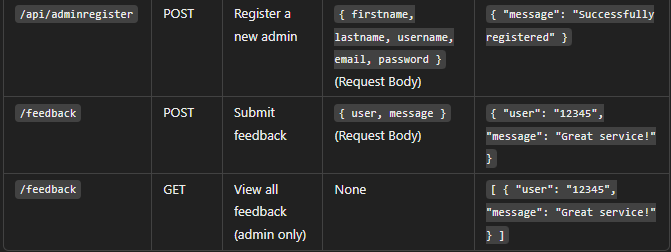
**bash**

**npm start**

1. **API Documentation**

****

****

****

**8. Authentication**

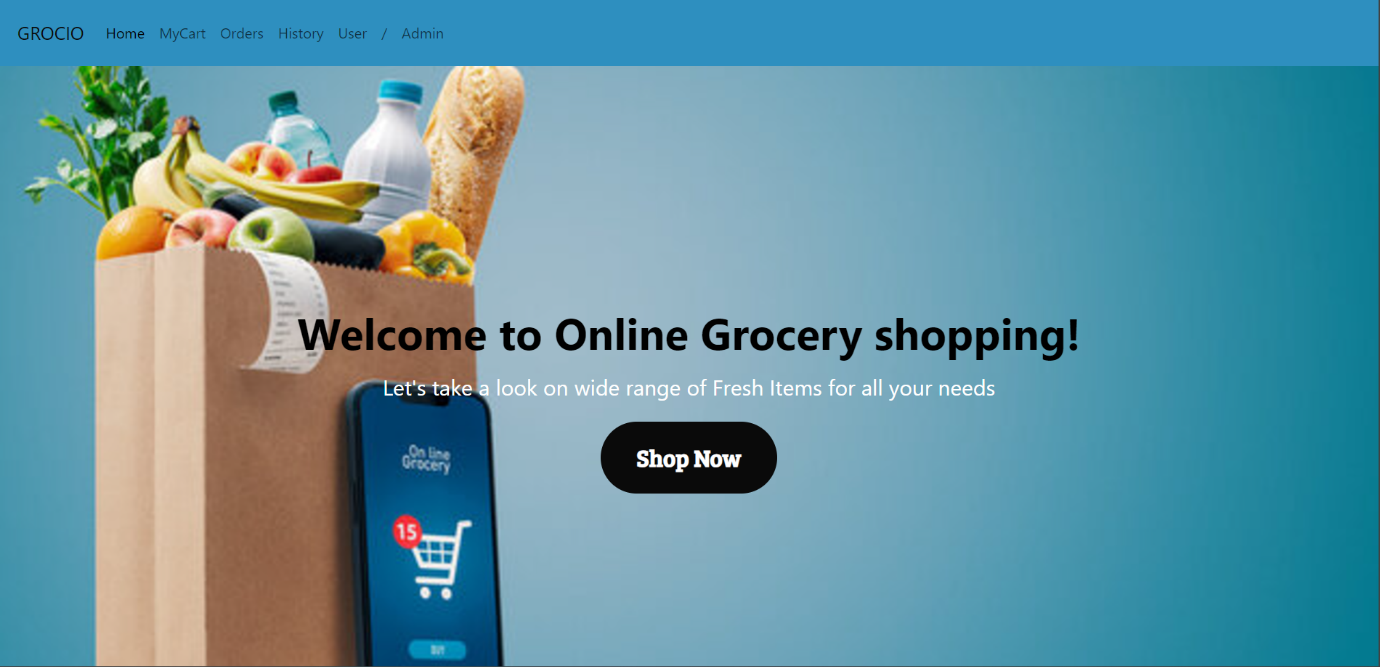
**The project uses JWT-based authentication:**

* Users authenticate using a unique token issued during login.
* Middleware ensures secure routes for cart and order operations**.**

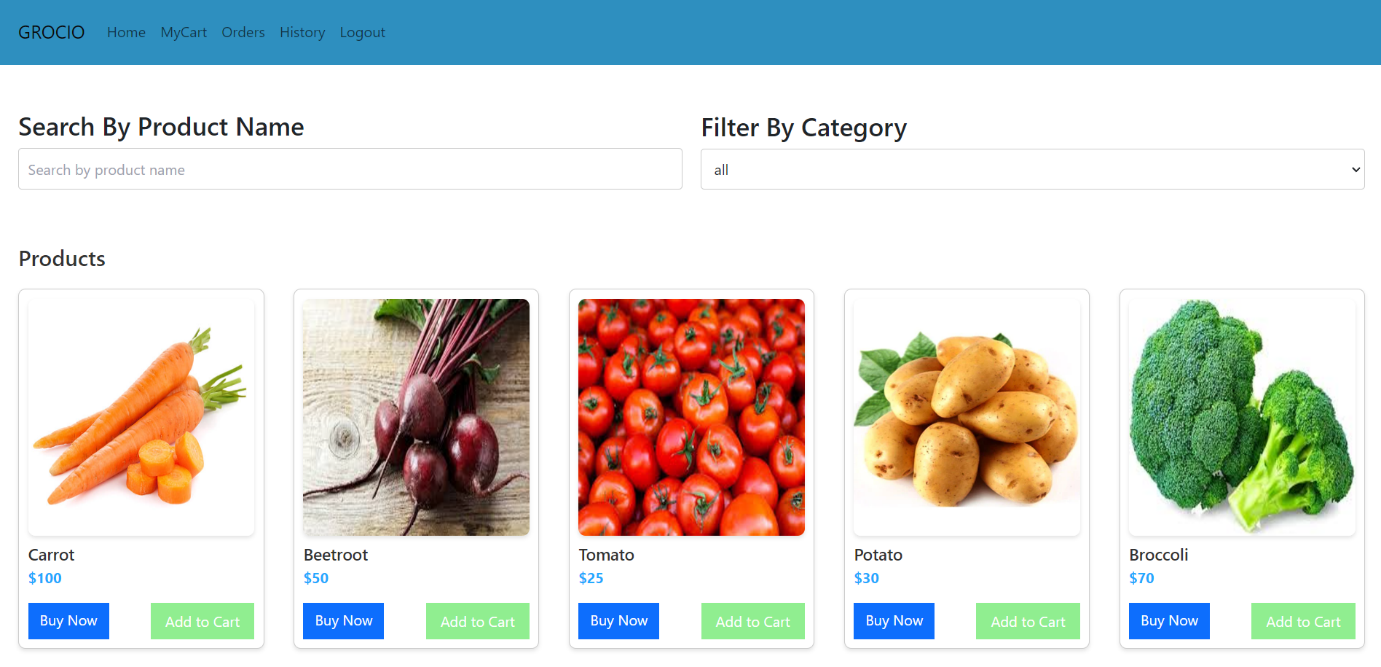
**9. User Interface**

**Screenshots showcasing:**

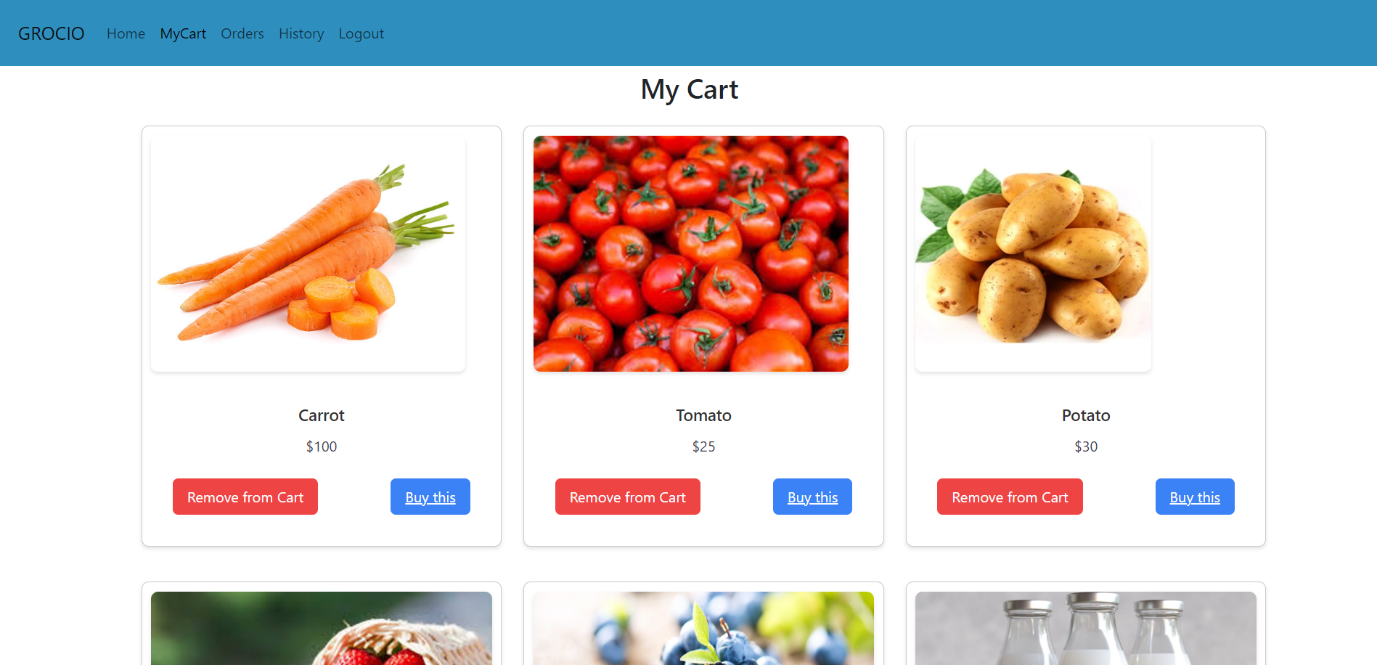
1. **Home Page:** Displays featured categories and promotions.



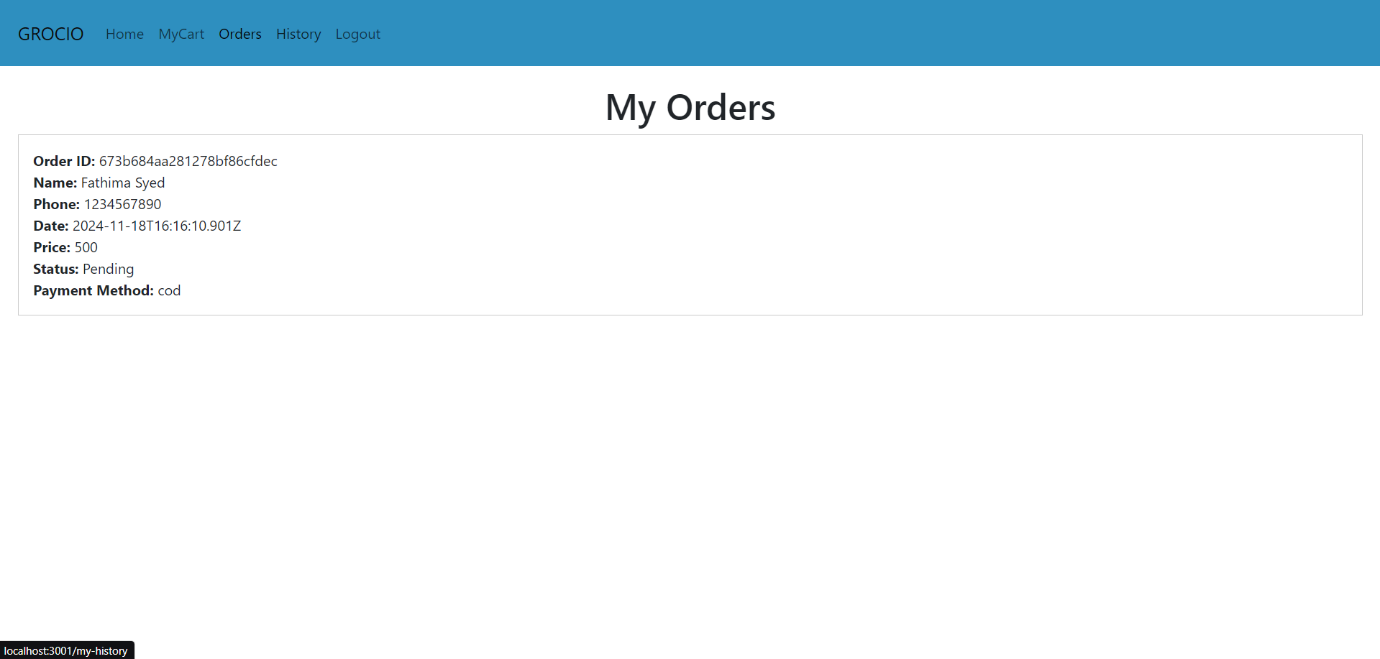
1. **Product Page:** Lists products with sorting options.



1. **Cart and Checkout:** Interactive cart summary and payment options.



1. **Order Tracking:** Displays order progress**.**

****

**10. Testing:**

* **Unit Testing:** Jest for frontend and backend.
* **API Testing:** Postman collections for endpoint testing.
* **Integration Testing:** Cypress for end-to-end tests.

**11. Screenshots or Demo**

**Provide:**

* **Link to the demo:** https://drive.google.com/file/d/13VfRMx1IxogRuAcHf1DuU5sKWWhxwIUb/view?usp=sharing

**12. Known Issues**

* Limited payment gateway integrations.
* Currently lacks multi-language support.

**13. Future Enhancements**

* Add AI-based personalized product recommendations.
* Enable voice-based search functionality.
* Integrate multiple payment gateways like Stripe and Razorpay.
* Mobile app development for better accessibility.