Grocery Webapp – Detailed Project Report

1.Introduction:

Project Title: Grocery Webapp : Simplifying Grocery Shopping

Team Members:

- Sudarvizhi K - Project Head , Database Engineer
- Monisha C - Frontend Developer
- U D Jasritha - Backend Developer
- B M Monica - Quality Assurance Tester

2.Project Overview:

Purpose:

To create an intuitive web application that allows users to browse, search, and purchase groceries online, making grocery shopping efficient and user-friendly.

Features:

- User authentication and profile management.
- Real-time product search and filtering.
- Cart management with seamless checkout.
- Admin dashboard for managing inventory.

3. Architecture:

Frontend:

The frontend is built using React with component-based architecture, styled using TailwindCSS for modern and responsive design.

Backend:

Node.js and Express.js power the backend, managing API endpoints and handling requests.

Database:

MongoDB serves as the database, with collections for users, products, orders, and categories.

4. Setup Instructions

Prerequisites:

- Node.js (v16+)
- MongoDB (local or cloud instance)
- Git

5.Installation:

A. Clone the repository:

git clone git https://github.com/Bharath136/Grocery-Web-App-MERN.git

B. Navigate to directories and install dependencies:

```
cd client
npm install
cd ../server
npm install
```

C. Set up environment variables:

- Create .env file in the server directory.
- Add variables like DB_URI, JWT_SECRET, and PORT.

5. Folder Structure

Client:

- src/components/ - React components.
- src/pages/ - Page-level components.
- src/services/ - API interaction logic.

Server:

- routes/ - API routes.
- controllers/ - Business logic.
- models/ - Mongoose schemas.

6.Running the Application:

Frontend:

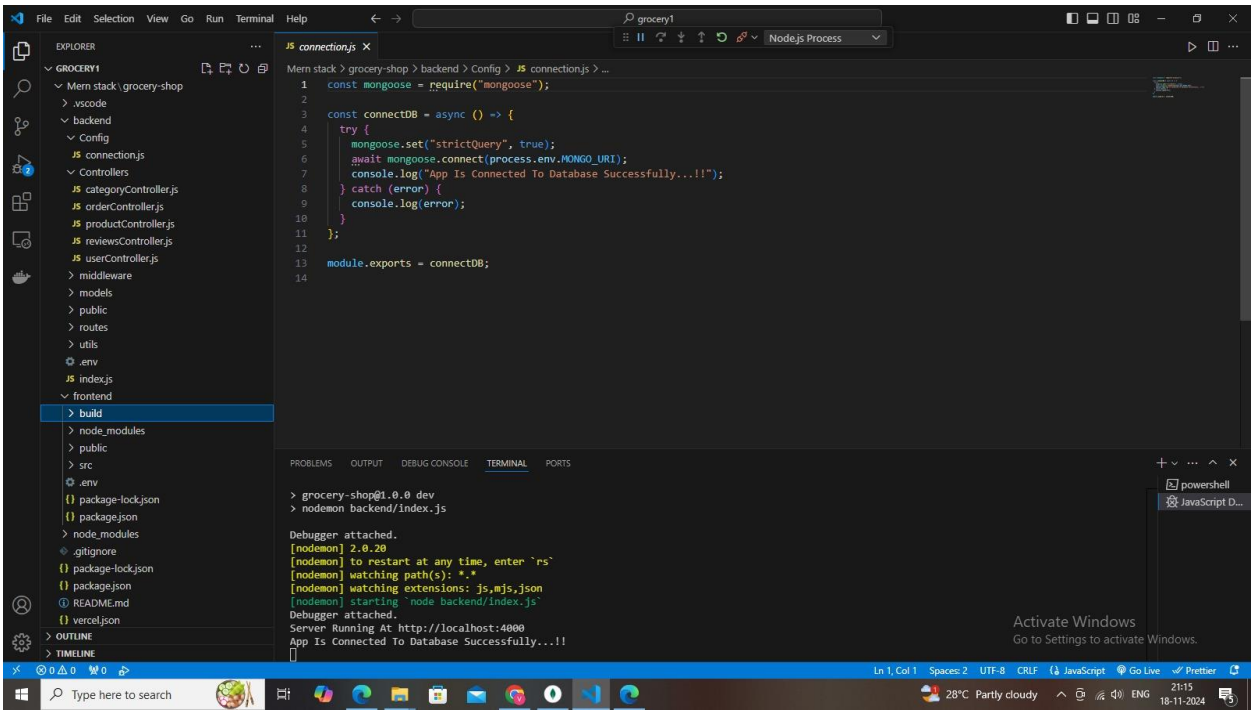
cd client

npm start

Backend:

cd server

npm start



7.API Documentation:

- GET /api/products: Fetch all products.
- POST /api/auth/login: User login.
- POST /api/orders: Place an order.

Example:

Request:

POST /api/auth/login

```
{
  "email": "sample@gmail.com",
  "password": "Sample@123"
}
```

Response:

json

```
{
  "token": "jwt-token",
  "user": { "name": "John Doe" }
}
```

8.Authentication:

JWT-based authentication is implemented.

- Tokens are issued at login and verified for secured endpoints.
- Middleware ensures only authorized users access restricted resources.

9.User Interface:

The user interface includes:

- A home page showcasing featured products.
- A responsive product search and filter page.
- A secure checkout flow.

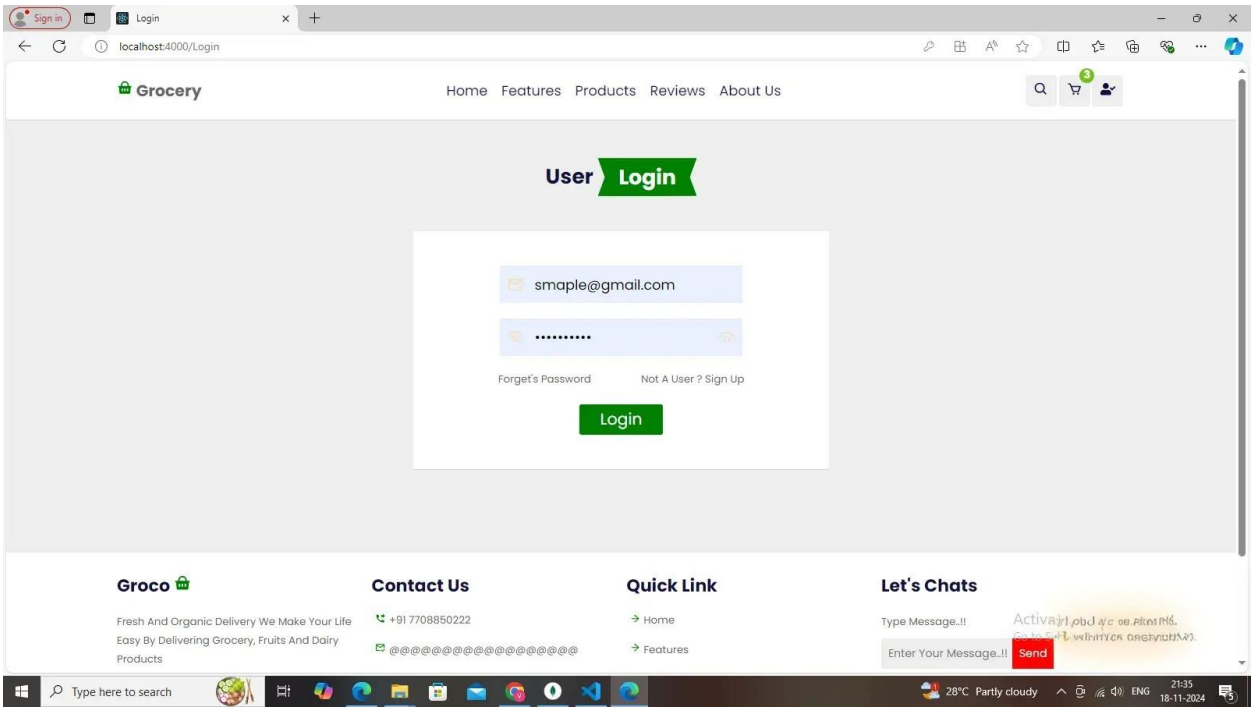
10.Testing:

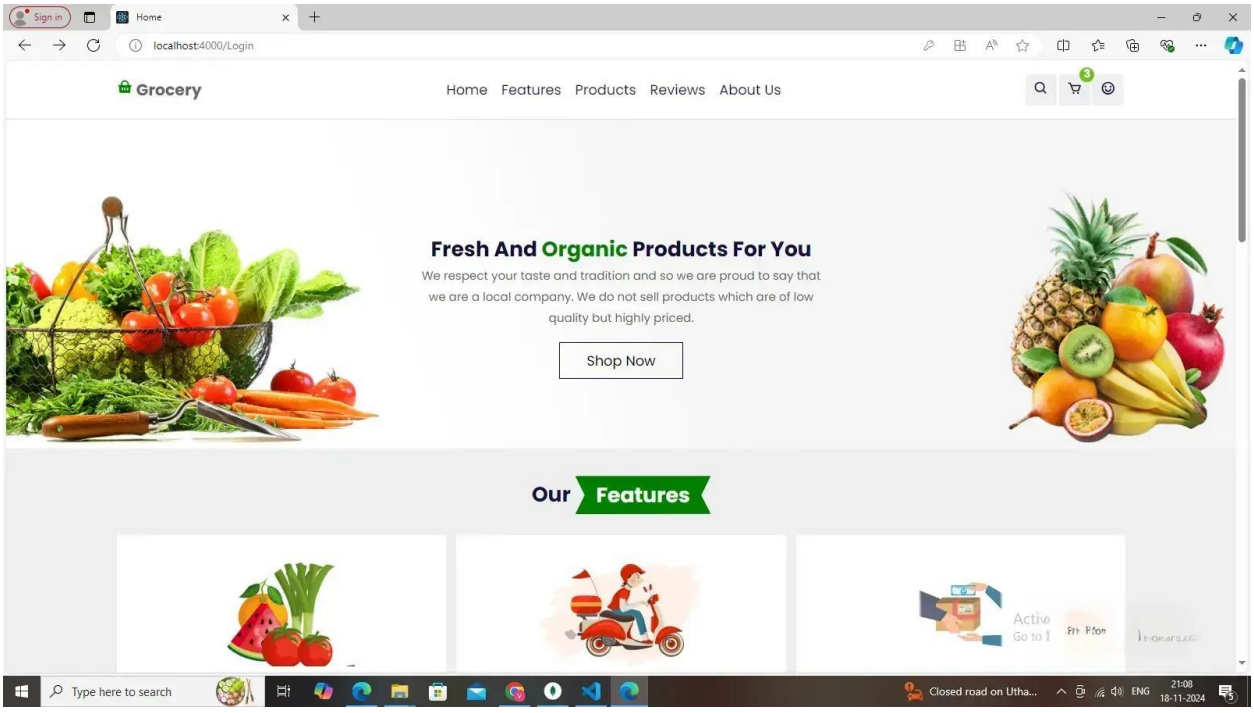
Tools Used: Jest and Cypress.

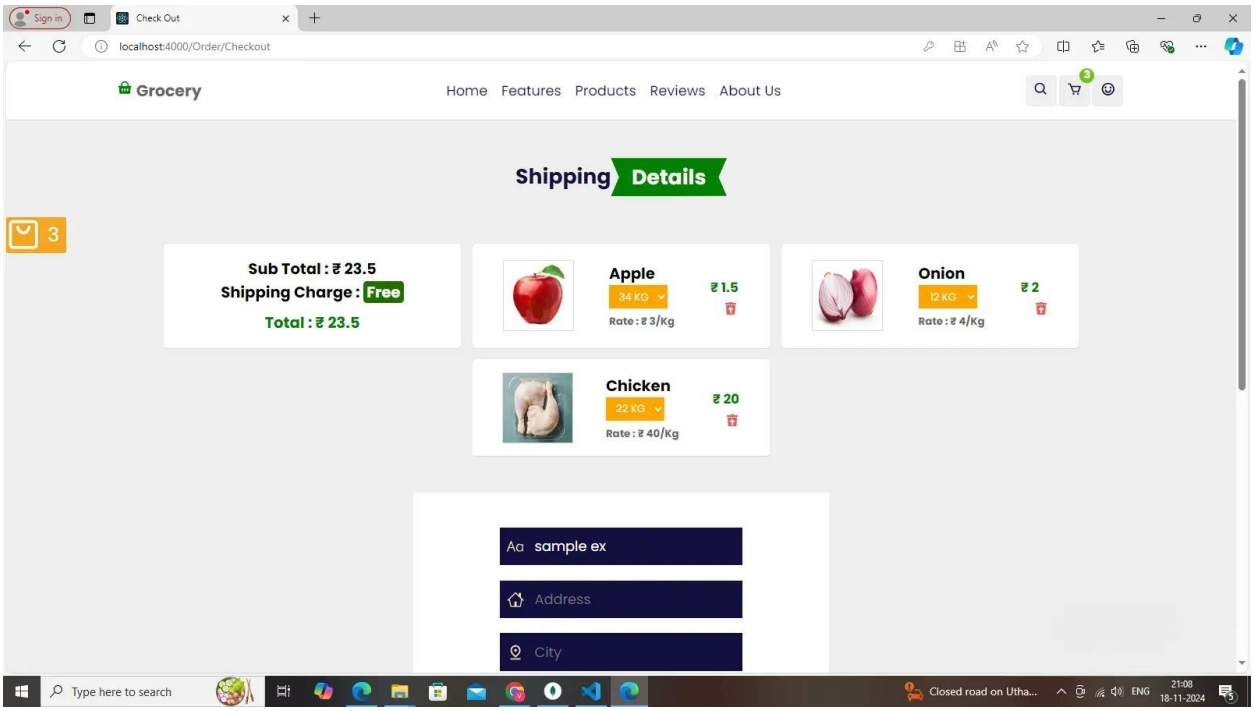
- Unit tests for critical functions.
- End-to-end tests for checkout and user registration flows.

11.Screenshots or Demo:

Screenshots:


Sign in
Login
localhost:4000/Login
Grocery
Home Features Products Reviews About Us
User Login
smaple@gmail.com
Forget's Password
Not A User ? Sign Up
Login
Groco
Fresh And Organic Delivery We Make Your Life Easy By Delivering Grocery, Fruits And Dairy Products
Contact Us
+91 7708850222
Quick Link
Home
Features
Let's Chats
Type Message..!!
Enter Your Message..!!
Send
Type here to search
28°C Partly cloudy
ENG
21:35
18-11-2024


Sign in
Home
localhost:4000/Login
Grocery
Home Features Products Reviews About Us
Fresh And Organic Products For You
We respect your taste and tradition and so we are proud to say that we are a local company. We do not sell products which are of low quality but highly priced.
Shop Now
Our Features
Type here to search
Closed road on Utha...
ENG
21:08
18-11-2024



Demo Video:[Live Demo Link]

https://drive.google.com/file/d/1MGNz0rbUqQ23NS8kvXULlxFj35t4JRpq/view?usp=sharing

12.Known Issues:

- Minor UI glitches on older browsers.
- Payment gateway sandbox mode occasionally times out.

13.Future Enhancements:

- Implement AI-driven product recommendations.
- Add support for multiple languages.
- Introduce subscription-based discounts.