

ShopSmart – your digital grocery store experience

Full Stack Development with MERN – Project Documentation

1. Introduction

Project Title: ShopSmart – Your Digital Grocery Store Experience

Team ID : LTVIP2026TMIDS818992

IN TEAM : 5

Team Members:

Team Leader : Shaik Asma

Team member : Shaik Muskan

Team member : Shaik Reshma Reshma

Team member : Thottampuri Krthura

Team member : Telugu Padmini

2. Project Overview

Purpose:



To build a user-friendly online grocery shopping platform where users can browse, search, and purchase groceries, while administrators can manage inventory and orders through a powerful backend.

Key Features:

- User Registration/Login
 - Product Search and Filter
 - Shopping Cart and Wishlist
 - Order Placement and History
 - Admin Dashboard for Product and Order Management
 - Secure authentication and role-based access
-

3. Architecture

Frontend (React):

- Components: Home, ProductList, Cart, Wishlist, Login, Register, Orders, AdminDashboard
- Libraries: React Router, Axios, Bootstrap

Backend (Node.js + Express.js):

- REST APIs for all user and admin operations
- JWT-based authentication and middleware for role protection
- Routes: /auth, /products, /orders, /admin

Database (MongoDB Atlas):

- Collections: Users, Products, Orders
- CRUD operations via Mongoose ORM
- Indexed for performance

4. Setup Instructions

□ **Prerequisites:**

- Node.js(v18+)
- MongoDB Atlas account
- Git

□ **Installation:**

```
git clone https://github.com/yourusername/shopsmart
cd shopsmart
cd server
npm install
cd ../client
npm install
```

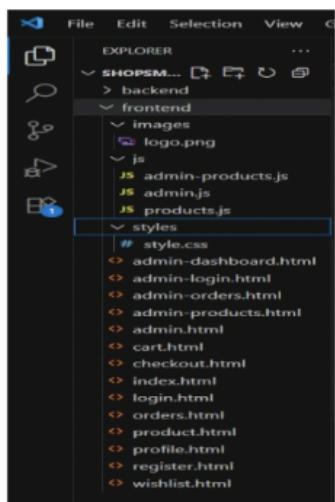
□ **Environment Variables (.env):**

In server/ folder:

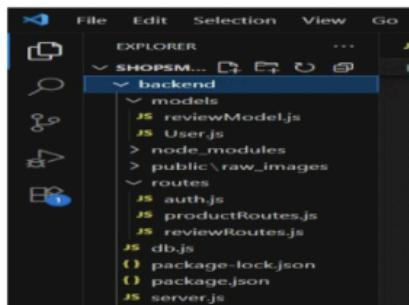
```
ini
MONGO_URI=your_mongodb_atlas_url
JWT_SECRET=your_jwt_secret
```

5. Folder Structure

□ **Client (Frontend):**



□ Server (Node.js Backend):



6. Running the Application

□ Frontend:

```
cd client  
npm start
```

□ Backend:

```
cd server  
npm start  
node  
server.js
```

7. API Documentation

Endpoint	Method	Description
/api/auth/register	POST	User registration
/api/auth/login	POST	User login and token generation
/api/products	GET	List all products
/api/products/:id	PUT/DELETE	Admin: Update/Delete product
/api/orders	POST	Place new order
/api/orders/:userId	GET	Get all orders by user
/api/admin/orders	GET/PUT	Admin: View or update order status

8. Authentication

□ JWT-Based Authentication:

- Users receive a JWTtoken on login
- Token stored in localStorage and passed in headers

- o Role-based middleware (isAdmin, authUsed) used to protect sensitive routes
-

9. User Interface

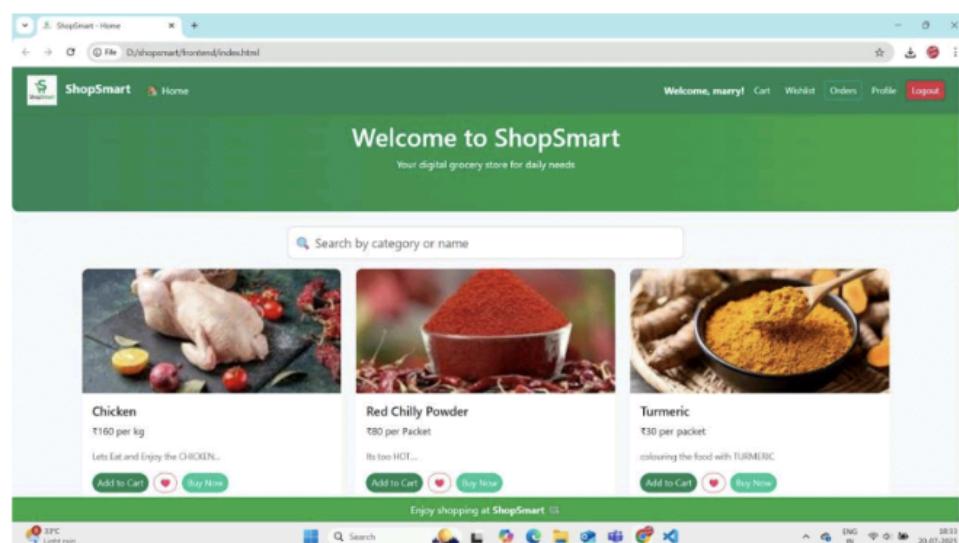
- Pages:
 - o Home Page with product cards
 - o Search bar, product filter
 - o Add to Cart / Wishlist buttons
 - o Order History page
 - o Admin Dashboard: Add/Edit/Delete Products, Manage Orders
 - **UI Technologies:**
 - React + Bootstrap 5
 - Responsive layout for mobile & desktop
-

10. Testing

- **Tools Used:**
 - o Postman (API testing)
 - o Manual UI testing on Chrome, Firefox, Mobile view
 - o Console and network inspection using DevTools
 - **Strategy:**
 - o Unit tests for APIs (optional)
 - o Manual tests for all user/admin flows
-

11. Screenshots & Demo video

- o Home page



o Login/Register

The screenshot shows a web browser window with the URL 'D:/shopsmart/frontend/login.html'. The page title is 'Login to ShopSmart'. It features two input fields: 'Email' and 'Password', both with placeholder text. Below these is a green 'Login' button. At the bottom right of the form area, there is a link 'Don't have an account? [Register](#)'.

o Cart

The screenshot shows a web browser window with the URL 'D:/shopsmart/frontend/cart.html'. The header includes the 'ShopSmart' logo and navigation links: Home, Wishlist, Cart, Orders, and Profile. The main content is titled 'Your Cart' and lists two items: 'Red Chilly Powder (1)' and 'Turmeric (1)'. Each item has a red 'Remove' button. Below the cart list are fields for 'Customer Name' (placeholder: Enter your name) and 'Delivery Address' (placeholder: Enter your full address). A dropdown menu for 'Payment Method' shows 'Cash on Delivery'. The total amount is displayed as 'Total: ₹110.00'. A green 'Place Order' button is at the bottom.

o Orders

The screenshot shows a web browser window with the URL 'D:/shopsmart/frontend/orders.html'. The header includes the 'ShopSmart' logo and navigation links: Home, Wishlist, Cart, Orders, and Profile. The main content is titled 'My Orders' and displays two order entries. The first order is for 'Toor Dal' (Quantity 1) with a total of '₹90'. The second order is for 'Toor Dal' (Quantity 1) with a total of '₹90'. Each order entry includes details: Order Date, Customer Name, Status, Address, Payment Method, and a 'Cancel Order' button. A note at the bottom right of the second order states 'Time left: 0m 40s'.

o Admin Dashboard

The screenshot shows a Windows desktop environment with a browser window titled "Admin - Manage Products". The page displays a table of products with the following data:

Name	Price (₹)	Unit	Image	Category	Description	Actions
Salt	₹20	packet		spices	Add Taste to your FOOD	[Edit] [Delete]
Chicken	₹210	kg		non-veg	Lets eat and enjoy the CHICKEN..	[Edit] [Delete]
Red Chilly Powder	₹80	Packet		spices	Its too HOT...	[Edit] [Delete]
Turmeric	₹30	packet		spices	colouring the food with TURMERIC	[Edit] [Delete]
Basmati rice	₹179	one kg		rice	lets cook some healthy and eat	[Edit] [Delete]
Toor Dal	₹90	one kg		Dals	yummy curry...	[Edit] [Delete]

- **Demo Link:**
https://drive.google.com/file/d/1npDzANr0_yQ9ahc1rsONG8BpFcjdmzxk/view?usp=drive_link
-

12. Known Issues

Payment gateway not integrated
No real-time delivery tracking
Cart items don't persist on logout

13. Future Enhancements

Integrate Razorpay/Stripe for secure payments
Build mobile app with React Native
Add delivery tracking and notification system
Implement product ratings and feedback