

ShopSmart—your digital grocery store experience

Full Stack Development with MERN—Project Documentation

1. Introduction

- **Project Title:**Shop Smart – Your Digital Grocery Store Experience
 - **Team ID:**LTVIP2026TMIDS47270
 - **Team Size:**5
 - **Team Members:**
 - **Team Leader:**Kollam lohitha
 - **Team member:**Konde poojitha
 - **Team member:**Yuvaraju Konedhala
 - **Team member:**Middepogu yuva raju
 - **Team member:**M Mallavika
-

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, Admin Dashboard
 - 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(MongoDBAtlas):**
 - 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/Lohithareddy071/shopsmart
cd shopsmart
cd server
npm install
cd../client
npm install
```

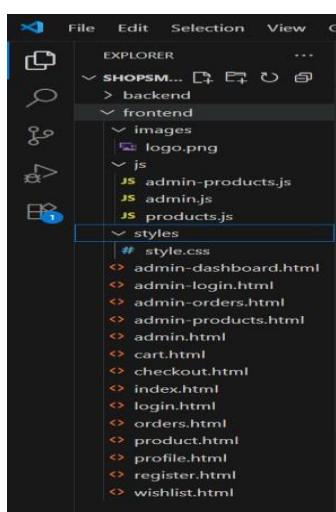
- **Environment Variables(.env):**

In server/folder:

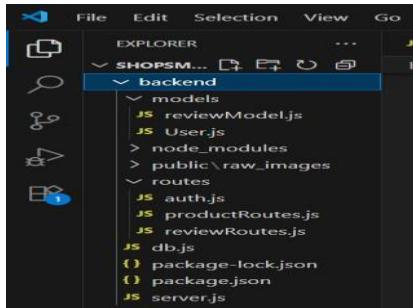
```
ini
MONGO_URI=your_mongodb_atlas_url
JWT_SECRET=your_jwt_secret
```

5. Folder Structure

- **Client (Frontend):**



- **Server(Node.jsBackend):**



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 JWT token on login
- Token stored in local Storage and passed in headers

- Role-based middleware(`isAdmin`,`authUser`) used to protect sensitive routes
-

9. User Interface

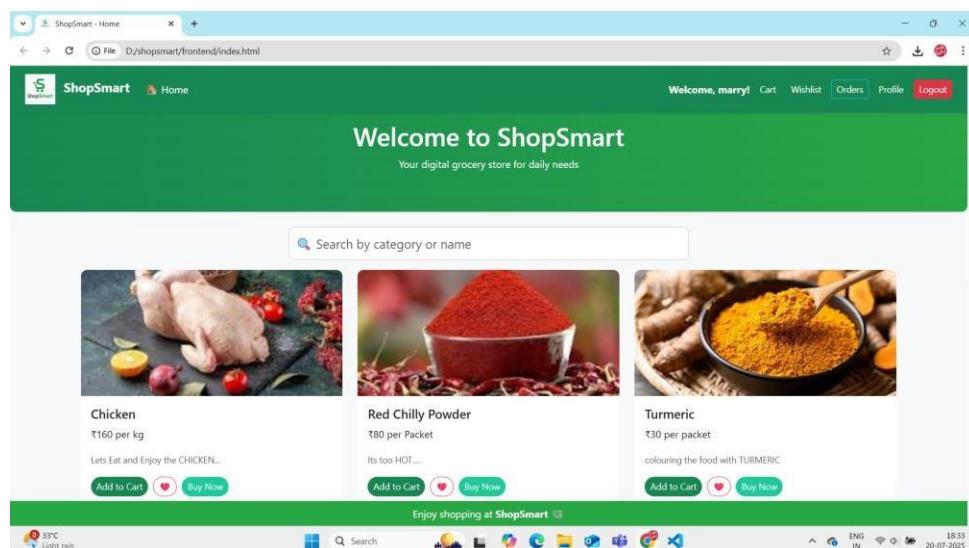
- Pages:
 - Home Page with productcards
 - Search bar,productfilter
 - Add to Cart/ Wishlist buttons
 - Order Historypage
 - Admin Dashboard: Add/Edit/Delete Products, Manage Orders
 - **UI Technologies:**
React+Bootstrap5
Responsive layout for mobile & desktop
-

10. Testing

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

11. Screenshots & Demovideo

- Homepage



- 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 of the form, 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 'ShopSmart' is visible with navigation links: Home, Wishlist, Cart, Orders, and Profile. The main content area is titled 'Your Cart' and lists two items: 'Red Chilly Powder (1)' and 'Turmeric (1)'. Each item has a '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 listed 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 'ShopSmart' is visible with navigation links: Home, Wishlist, Cart, Orders, and Profile. The main content area is titled 'My Orders' and displays two order entries. The first order is for 'Toor Dal x 1' with a total of '₹90'. It includes details: 'Order Date: 8/3/2025, 6:42:42 PM', 'Customer Name: shaik mehtaj', 'Status: Pending', 'Address: tirupati', and 'Payment Method: Cash on Delivery'. A 'Cancel Order' button is present. The second order is for 'Toor Dal x 1' with a total of '₹90'. It includes details: 'Order Date: 7/20/2025, 4:21:41 PM', 'Customer Name: marry', 'Status: Processing', 'Address: tirupati', and 'Payment Method: Cash on Delivery'. A 'Time left: 0m 40s' message is shown next to the second order's cancel button.

- o Admin Dashboard

The screenshot shows a web-based admin dashboard titled "Admin: Manage Products". At the top, there's a form for adding or editing a product, with fields for Product Name, Price, Unit, Image URL, and Category. Below the form is a table listing six products:

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

- **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 ReactNative
- Add delivery tracking and notification system
- Implement product ratings and feedback