

# Full Stack Development (MERN) - SHOPEZ Application Documentation

## 1. Introduction

**Project Title:** SHOPEZ – E-Commerce Web Application (MERN Stack)

**Team Members:** Suganthi K (Full Stack Developer)

---

## 2. Project Overview

### Purpose

The SHOPEZ project is an e-commerce application built using MERN stack. It provides product browsing, user authentication, cart management, order placement, admin management, and banner display functionalities.

### Features

- User authentication (login/register)
  - Product listing and product details
  - Add to cart and checkout
  - Order management
  - Admin dashboard for product & user control
  - Banner management
  - API-based architecture with secure routes
- 

## 3. Architecture

### Frontend (React + Vite)

- React component-based UI structure
- Context API for global state
- API integration with Axios
- Pages + Components separation
- Assets & images folder for UI resources

### Backend (Node.js + Express.js)

- REST API architecture
- Multiple controllers handling separate business logic
- Route-level separation

- JWT authentication middleware

## Database (MongoDB + Mongoose)

- Centralized Schema.js for all collections
  - Collections include Users, Products, Cart, Orders, Admin, Banner, etc.
- 

## 4. Setup Instructions

### Prerequisites

- Node.js
- MongoDB
- VS Code

### Backend Setup

```
cd Backend  
npm install
```

Create .env file:

```
MONGO_URL=your_mongo_url  
JWT_SECRET=your_secret  
PORT=5000
```

Run server:

```
npm start
```

### Frontend Setup

```
cd Frontend  
npm install  
npm run dev
```

---

## 5. Folder Structure

### Backend Folder Structure (SHOPEZ)

```
Backend/  
  └── config/  
      └── db.js  
  └── controllers/  
      ├── adminController.js  
      ├── authController.js  
      ├── bannerController.js  
      └── cartController.js
```

```
    └── orderController.js  
    └── productController.js  
    └── userController.js  
  └── middleware/  
    └── authMiddleware.js  
  └── models/  
    └── Schema.js  
  └── routes/  
    ├── adminRoutes.js  
    ├── bannerRoutes.js  
    ├── cartRoutes.js  
    ├── orderRoutes.js  
    ├── productRoutes.js  
    └── userRoutes.js  
  └── .env  
└── index.js
```

---

## Frontend Folder Structure (SHOPEZ)

```
Frontend/  
  └── public/  
    ├── favicon.ico  
    ├── index.html  
    ├── logo192.png  
    ├── logo512.png  
    ├── manifest.json  
    ├── robots.txt  
    └── vite.svg  
  └── src/  
    ├── assets/  
    ├── components/  
    │   ├── FlashSale.jsx  
    │   ├── Footer.jsx  
    │   ├── Login.jsx  
    │   ├── Navbar.jsx  
    │   ├── Products.jsx  
    │   └── Register.jsx  
    ├── context/  
    │   └── GeneralContext.jsx  
    ├── images/  
    │   ├── home-banner-1.png  
    │   └── home-banner-2.png  
    ├── pages/  
    ├── App.jsx  
    ├── App.css  
    ├── index.css  
    └── main.jsx
```

```
└── package.json
└── vite.config.js
└── node_modules/
```

---

## 6. Running the Application

### Backend

```
cd Backend
npm start
```

### Frontend

```
cd Frontend
npm run dev
```

---

## 7. API Documentation

### User Routes

- POST /user/register
- POST /user/login
- GET /user/profile

### Product Routes

- GET /product/all
- GET /product/:id

### Cart Routes

- POST /cart/add
- GET /cart
- DELETE /cart/remove/:id

### Order Routes

- POST /order/create
- GET /order/user

### Admin Routes

- GET /admin/users
- GET /admin/products

### Banner Routes

- GET /banner
- POST /banner

---

## 8. Authentication

- JWT-based authentication
  - Private routes protected by authMiddleware.js
  - Token generated during login
- 

## 9. User Interface

- React components display all UI pages
  - Flash sale banner
  - Product grid view
  - Login/Register forms
  - Navbar for navigation
- 

## 10. Testing

- API tested using Postman
  - Frontend tested manually through browser
- 

## 11. Screenshots or Demo

---

## 12. Known Issues

- None reported currently
- 

## 13. Future Enhancements

- Add payment gateway
- Add admin analytics dashboard
- Integrate product search & filtering
- Add wishlist functionality