**Full Stack Development with MERN**

**Project Documentation**

**1. Introduction**

* **Project Title: Shopez eCommerce application**
* **Team Members: Sivaramakrishnan, Lokeshwaran, Nirmal, Rishab Kumar**

**2. Project Overview**

* **Purpose:  
  This project aims to provide users with an intuitive and secure online shopping experience. It allows vendors to list products and customers to browse, search, and purchase items seamlessly.**
* **Key Features:**
  + **User authentication with role-based access (Admin, Vendor, Customer).**
  + **Product catalog with categories and search functionality.**
  + **Shopping cart and checkout system.**
  + **Payment gateway integration.**
  + **Admin dashboard for managing users, orders, and products.**
  + **Responsive and mobile-friendly design.**

**3. Architecture**

* **Frontend:**
  + **Developed using React.**
  + **Folder structure includes:**
    - **/components: Houses reusable UI components like Navbar, Footer, ProductCard, etc.**
    - **/pages: Contains page-level components like Home, Product Details, Cart, and Login.**
    - **/redux or /context: Manages application state for authentication, cart, and product listings.**
* **Backend:**
  + **Built with Node.js and Express.js.**
  + **Includes middleware for request validation, error handling, and authentication.**
  + **Organized with /routes for endpoints and /controllers for business logic.**
* **Database:**
  + **MongoDB stores user data, product details, and order history.**
  + **Schema design includes collections for:**
    - **Users: Stores user profiles, hashed passwords, and roles.**
    - **Products: Details such as name, price, category, stock, and images.**
    - **Orders: Tracks user orders, payment status, and shipping details.**

**4. Setup Instructions**

* **Prerequisites:**
  + **Node.js (v16 or higher).**
  + **MongoDB (local or cloud-based).**
  + **IDE (e.g., VS Code).**
  + **NPM or Yarn.**
* **Installation:**
* **Clone the repository :**

**git clone** [**https://github.com/Saiganesh-2705/shopEZ-Ecommerce.git**](https://github.com/Saiganesh-2705/shopEZ-Ecommerce.git)

* **Navigate to the project directory.**
* **Install dependencies for both frontend and backend :**

**cd client && npm install**

**cd ../server && npm install**

* **Configure environment variables in .env files for both client and server (e.g., API keys, database URLs).**

**5. Folder Structure**

* **Client (Frontend):**
  + **/src/components: Contains reusable UI elements (e.g., Button, Modal).**
  + **/src/pages: Page-specific components like LoginPage, ProductPage.**
  + **/src/redux: Redux store configuration and slices for cart and user states.**
* **Server (Backend):**
  + **/routes: API routes for authentication, products, and orders.**
  + **/models: MongoDB schemas for users, products, and orders.**
  + **/controllers: Implements logic for CRUD operations.**
  + **/middleware: Middleware for authentication and error handling.**

**6. Running the Application**

* **Frontend:**
  + **Start the development server:**

**cd client**

**npm start**

* **Backend:**
  + **Start the backend server:**

**cd server**

**npm start**

**7. API Documentation**

**Example:**

* **Get All Products**
  + **Method: GET**
  + **URL: /api/products**
  + **Description: Fetches a list of all available products.**
  + **Response:**

**[**

**{**

**"id": "123",**

**"name": "Product A",**

**"price": 50.0,**

**"category": "Electronics"**

**}**

**]**

* **Create Order**
  + **Method: POST**
  + **URL: /api/orders**
  + **Body:**

**{**

**"userId": "user123",**

**"products": [**

**{ "productId": "prod123", "quantity": 2 }**

**]**

**"totalPrice": 100.0**

**}**

**8. Authentication**

* **Method:**
  + **JWT-based authentication for session management.**
  + **Passwords are hashed using bcrypt.**
* **Roles:**
  + **Admins: Manage users, products, and orders.**
  + **Vendors: Add and manage their product listings.**
  + **Customers: Browse and purchase products.**

**9. User Interface**

* **Features Highlighted:**
  + **Navigation bar with search and category filters.**
  + **Dynamic product display grid.**
  + **Checkout and payment flow.**
  + **Admin panel for managing site content.**

**10. Testing**

* **Tools:**
  + **Unit Tests: Jest for frontend components.**
  + **Integration Tests: Postman for API testing.**
  + **End-to-End Tests: Cypress for testing workflows like login and checkout.**

**11. Screenshot**

