FULL STACK DEVELOPMENT WITH MERN

Shopez- MERN Full Stack E-commerce

INTRODUCTION:

- Project Title: Shopez
- Team Members: Individual project

PROJECT OVERVIEW

Purpose:

The Project, Shopez, aims to provide a robust and scalable solution for online retailers. It'll feature user authentication, product catalog, payment gateway, and order management. Shopez will be a comprehensive ecommerce solution with a responsive UI and secure backend

Features:

1. **Product Catalog:**
   Shopez offers an extensive range of products with detailed descriptions, customer reviews, pricing, and available discounts.

2. **Shop Now Button:**
   Every product includes a "Shop Now" button to initiate the order process quickly.

3. **Order Management:**
   After clicking "Shop Now," users are redirected to an order details page to enter their shipping address, payment information, and specific product requirements.

4. **Secure & Efficient Checkout:**
   The platform ensures safe handling of personal data and provides a fast, reliable checkout experience.

5. **Order Confirmation & Review:**
   Users receive confirmation after placing orders and can view complete order details, including shipping and payment information.

ARCHITECTURE

- **Frontend (React JS)**
  The frontend is built using React and includes UI components for user authentication, product listings, cart, user profile, and the admin dashboard. It uses React components, Context API for state management, and React Router for navigation.

- **Backend (Node.js + Express.js)**
  The backend provides RESTful API endpoints for users, products, cart, and orders. It handles admin authentication, dashboard operations, and all business logic using Express.js.

- **Database (MongoDB)**
  MongoDB stores collections such as Users, Products, Cart, and Orders. Mongoose is used for defining schemas and managing database interactions.

SETUP INSTRUCTIONS

Prerequisites

- Node.js
- MongoDB (Local or MongoDB Atlas)
- Git

Installation

1. Clone the repository

   ```
   git clone [your_repo_link]
   ```

2. Install backend dependencies

   ```
   cd server
   npm install
   ```

3. Install frontend dependencies

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

4. Create .env file in the /server folder

   ```
   PORT=4000
   MONGO_URI=your_connection_string
   JWT_SECRET=your_secret_key
   ```

FOLDER STRUCTURE

Client (React Frontend)

```
client/
├── public/          # Static assets
├── src/             # Main source code
│   ├── components/      # Reusable UI components
│   ├── context/         # Global state management
│   ├── image/           # Images and icons
│   ├── pages/           # Application pages/screens
│   ├── styles/          # CSS files (App.css, etc.)
│   ├── App.js           # Root component
│   ├── index.js         # Entry point
│   ├── index.css        # Global styles
│   ├── App.test.js      # App tests
│   ├── reportWebVitals.js  # Performance metrics
│   └── setupTests.js    # Test configuration
│
├── .gitignore
├── package-lock.json
└── README.md
```

Server (node.js + express.js)

```
  server/
|  package.json
|  package-lock.json
|  index.js
|  server.js
|  .env
|
 ├─ controllers/
 ├─ models/
 ├─ routes/
 ├─ middleware/
└─ config/
```

RUNNING THE APPLICATION

Frontend

- cd client
- npm start

Backend

- cd server
- npm start

API DOCUMENTATION

User APIs

- **POST /api/auth/register** – Register a new user
- **POST /api/auth/login** – User login
- **GET /api/auth/profile** – Get logged-in user profile *(protected)*

Product APIs

- **GET /api/products** – Get all products
- **GET /api/products/:id** – Get a single product by ID
- **POST /api/products** – Add a product *(admin only)*
- **PUT /api/products/:id** – Update a product *(admin only)*
- **DELETE /api/products/:id** – Delete a product *(admin only)*

Cart APIs

- **POST /api/cart/add** – Add item to cart
- **GET /api/cart/:userId** – Get user cart
- **DELETE /api/cart/:itemId** – Remove item from cart

Order APIs

- **POST /api/orders** – Place a new order
- **GET /api/orders/:userId** – Get orders of a user
- **GET /api/orders/details/:orderId** – Get order details

Admin APIs

- **POST /api/admin/login** – Admin login
- **GET /api/admin/users** – Get all users
- **GET /api/admin/orders** – Get all orders

AUTHENTICATION

- **JWT Generation:**

  When a user logs in, the backend verifies the credentials and generates a JWT token.

- **Token Storage:**

  The JWT token is stored on the frontend using **localStorage** or **sessionStorage**.

- **Protected Routes:**

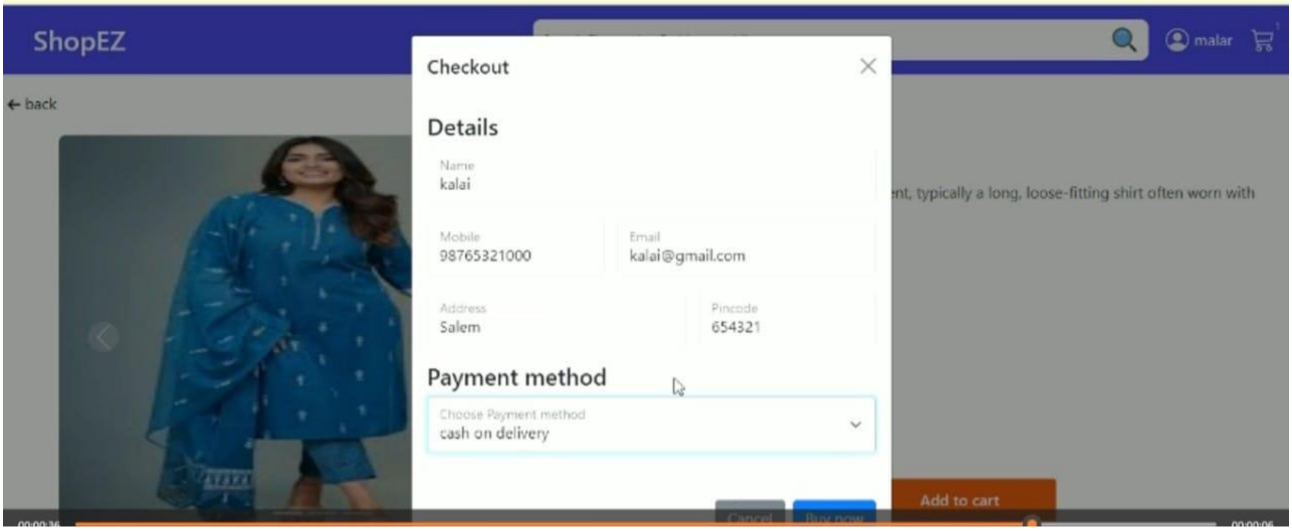
  For every protected API request, the token is included in the Authorization header as:

  ```
  Authorization: Bearer <token>
  ```

- **Authorization:**

  The backend validates the token before granting access to restricted routes such as user profile, cart, orders, and admin operations.

USER INTERFACE



Checkout

Details

Name
kalai

Mobile
98765321000

Email
kalai@gmail.com

Address
Salem

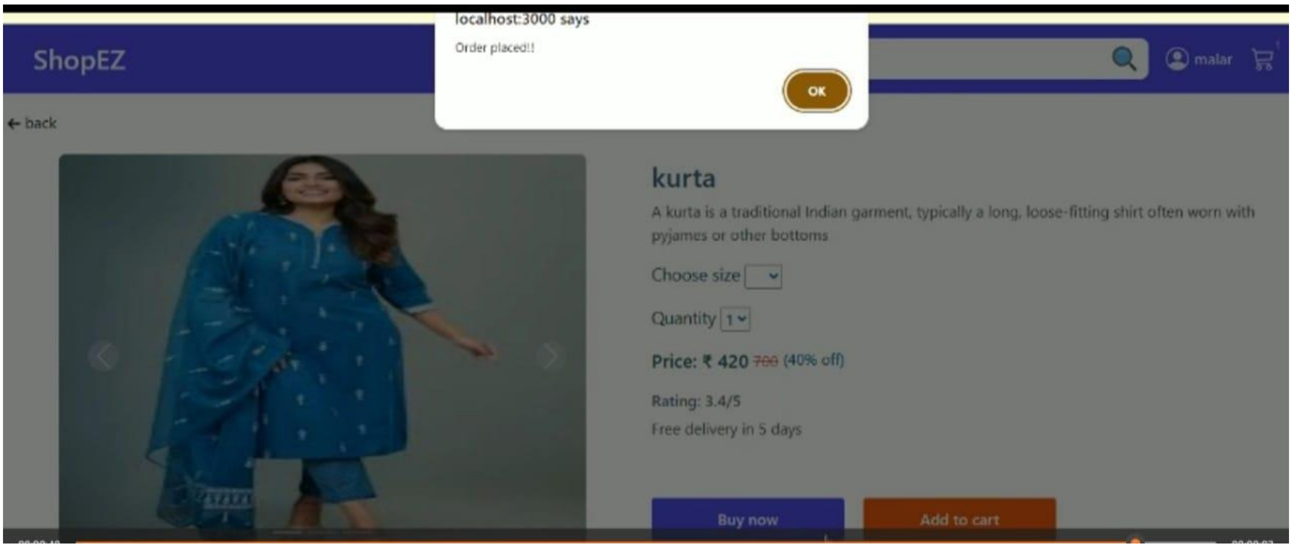
Pincode
654321

Payment method

Choose Payment method
cash on delivery

...nt, typically a long, loose-fitting shirt often worn with

Order Details:


localhost:3000 says

Order placed!!

kurta

A kurta is a traditional Indian garment, typically a long, loose-fitting shirt often worn with pyjames or other bottoms

Choose size

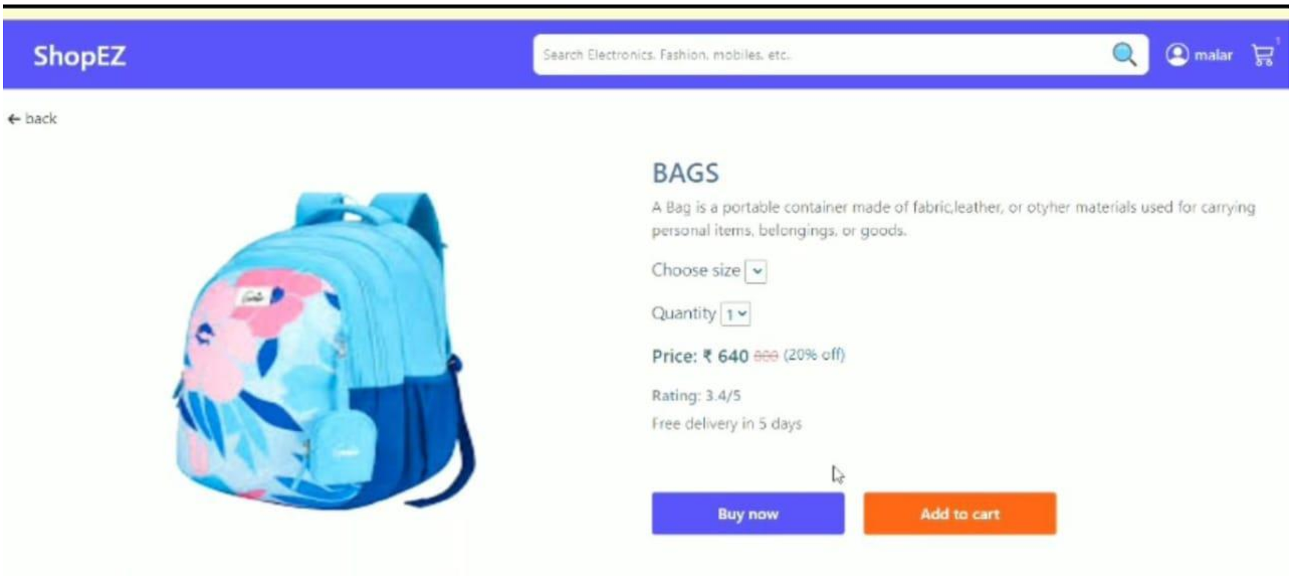
Quantity 1

Price: ₹ 420 ~~700~~ (40% off)

Rating: 3.4/5

Free delivery in 5 days

Order Details Page



BAGS

A Bag is a portable container made of fabric,leather, or otyher materials used for carrying personal items, belongings, or goods.

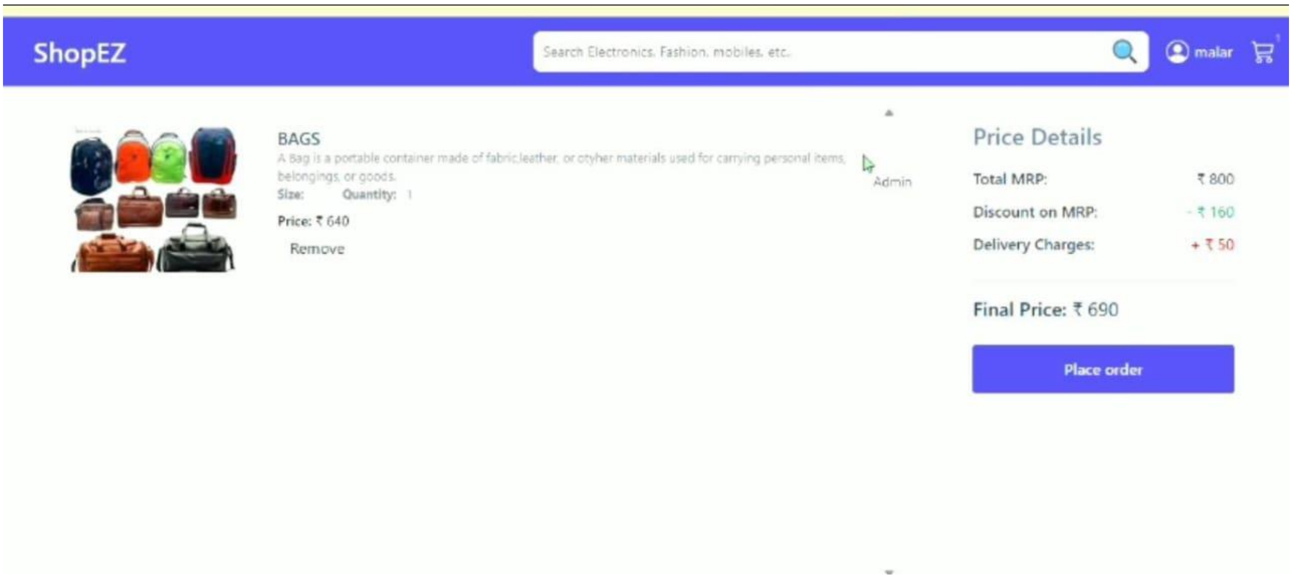
Choose size

Quantity 1

Price: ₹ 640 ~~800~~ (20% off)

Rating: 3.4/5

Free delivery in 5 days



BAGS

A Bag is a portable container made of fabric,leather, or otyher materials used for carrying personal items, belongings, or goods.

Size: Quantity: 1

Price: ₹ 640

Price Details

Total MRP:	₹ 800
Discount on MRP:	- ₹ 160
Delivery Charges:	+ ₹ 50

Final Price: ₹ 690

TESTING

Testing Strategy

- **UI Testing:**
  Each page (Home, Products, Cart, Profile, Admin Dashboard) was manually tested to verify layout, navigation, and user interactions.

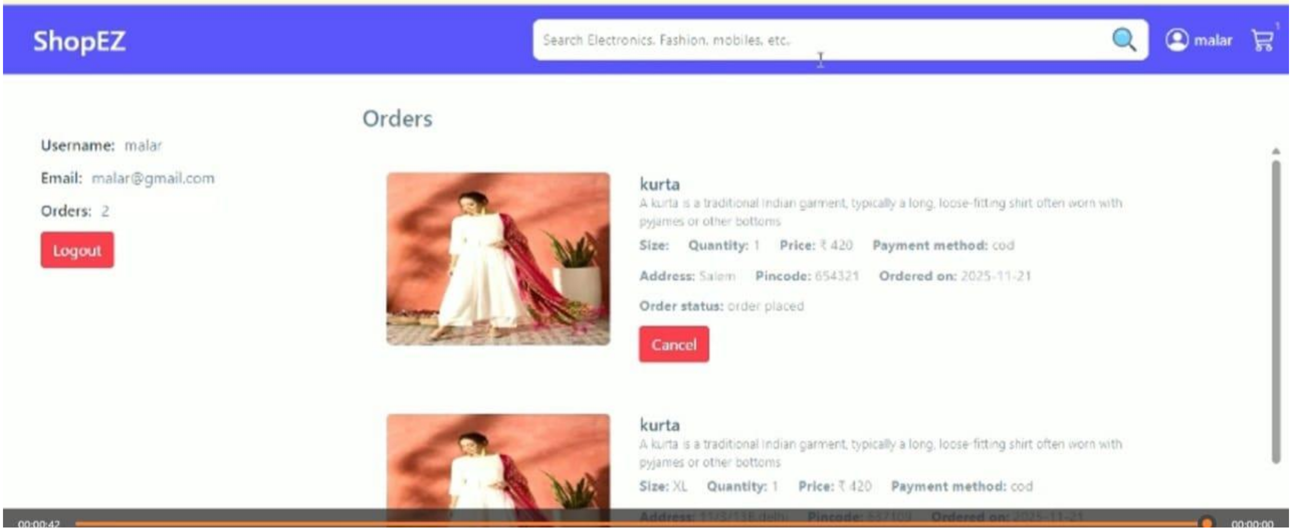
- **API Testing:**
  All backend API endpoints were tested using **Postman** to validate request methods, parameters, responses, and error handling.

Tools Used

- **Manual UI Testing** (Browser-based testing)
- **Postman** for API endpoint testing
- **Console Logs & Debugging Tools** for resolving errors

SCREENSHOTS OR DEMO

Home Page



Orders

Username: malar

Email: malar@gmail.com

Orders: 2

kurta

A kurta is a traditional indian garment, typically a long, loose-fitting shirt often worn with pyjames or other bottoms

Size: Quantity: 1 Price: ₹ 420 Payment method: cod

Address: Salem Pincode: 654321 Ordered on: 2025-11-21

Order status: order placed

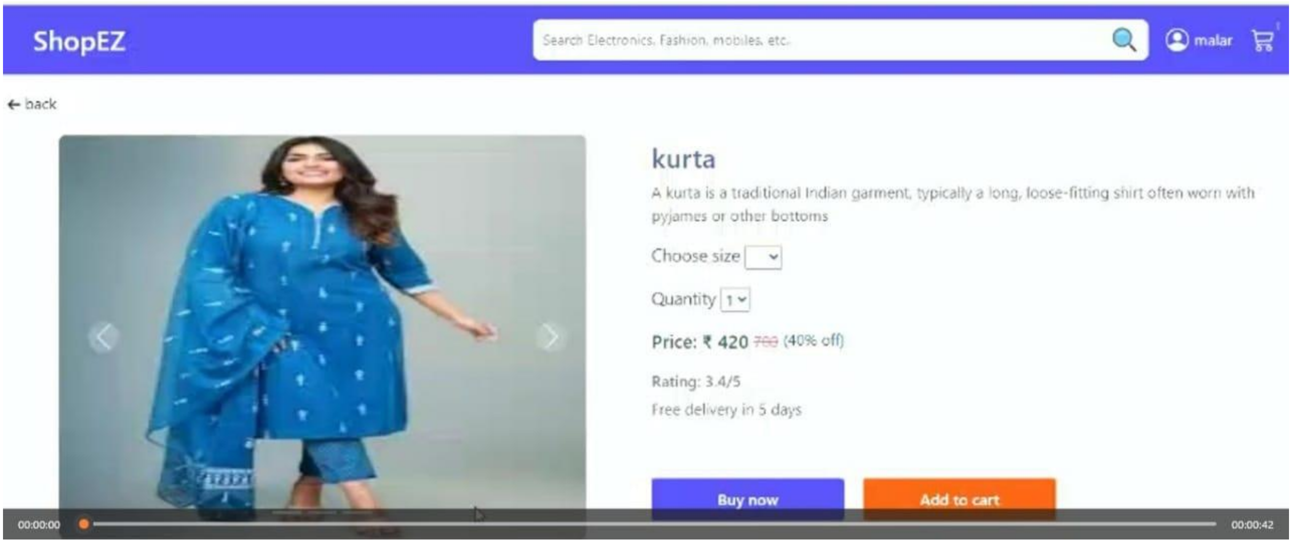
kurta

A kurta is a traditional indian garment, typically a long, loose-fitting shirt often worn with pyjames or other bottoms

Size: XL Quantity: 1 Price: ₹ 420 Payment method: cod

Address: 11/3/138 delhi Pincode: 557109 Ordered on: 2025-11-21

Order Confirmation Page



kurta

A kurta is a traditional Indian garment, typically a long, loose-fitting shirt often worn with pyjames or other bottoms

Choose size

Quantity 1

Price: ₹ 420 ~~700~~ (40% off)

Rating: 3.4/5

Free delivery in 5 days

KNOWN ISSUES

- Delivery- Related Problem
- Return and Refund Difficulties.
- Product Quality and Authenticity Concerns.
- Billing and Payment Disputes.
- Platform Usability Transparency Issues .

FUTURE ENHANCEMENTS

- Add advanced search and filters for better user experience.
- Implement payment gateway integration for online payments.
- Omni channel Retailing.
- Improve mobile responsiveness and overall UI design.
- Voice Commerce.
- Add wish list, product reviews, and ratings.
- AI-Powered Personalization.