**Full Stack Development with MERN**

# Grocery Web Application Documentation

## 

## 1. Introduction

### Project Title: Grocery webapp

**Team Members:**

* **MOHAMED MARZOOK Back-End Developer**
* **ABU KASIM Front-End Developer**
* **ABDULLAH UI/UX Developer**
* **ABDUL RAHMAN Manual/Automated testing**

## 2. Project Overview

### Purpose

The Grocery Web Application is designed to provide a seamless online shopping experience for customers, enabling them to explore and purchase groceries and everyday essentials through an intuitive digital platform.

### Features

1. User-friendly product browsing and search
2. Secure shopping cart and checkout process
3. Product categorization and filtering
4. User account management
5. Admin dashboard for inventory management
6. Secure payment processing
7. Order tracking and management
8. Real-time inventory updates

## 3. Architecture

### Frontend

1. Built with Angular framework
2. Component-based architecture
3. Responsive design using modern CSS frameworks
4. State management for cart and user sessions
5. RESTful API integration with backend

### Backend

1. Node.js with Express.js framework
2. RESTful API endpoints
3. Middleware for authentication and request processing
4. Error handling and logging
5. Integration with MongoDB database

### Database

MongoDB implementation with following key collections: - Users (Customer and Admin profiles) - Products (Inventory items) - Orders (Purchase transactions) - Cart (Shopping sessions) - Categories (Product classifications)

## 4. Setup Instructions

### Prerequisites

1. Node.js and npm (Latest stable version)
2. MongoDB Community Edition
3. Angular CLI (npm install -g @angular/cli@16.0.0)
4. Git for version control
5. IDE (Visual Studio Code recommended)

### Installation

1. Clone the repository:

* git clone https://github.com/MohdMarzook/E-Commerce-App

1. Install dependencies:

Cd E-Commerce-App-main

Cd server

npm install

cd ..

* cd client/client  
  npm install

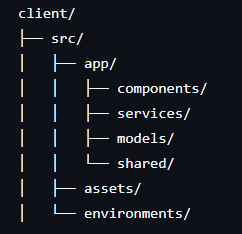
1. Configure environment variables:
   * Create .env file in root directory
   * Add necessary environment variables (MONGO\_URI, PORT)
2. Connecting to Mango DB:
   1. Install MongoDB locally or use a cloud-based MongoDB service.
   2. Or use this DB "mongodb+srv://mmohamedmarzook0701:p7iXscaEX6AzrOuQ@cluster0.afnjy.mongodb.net/?retryWrites=true&w=majority&appName=Cluster0"
3. Admin login

Email : [admin@example.com](mailto:admin@example.com)

Password: admin@070

## 5. Folder Structure

### Client (Angular)



### Server (Node.js)

## 

## 6. Running the Application

### Frontend Server

cd client/client  
npm start  
# Access application at http://localhost:4200

### Backend Server

cd server  
npm run dev  
# Server runs on http://localhost:5100

## 7. API Documentation

### Authentication Endpoints

1. POST /api/auth/register
2. POST /api/auth/login
3. GET /api/auth/profile

### Product Endpoints

1. GET /api/products
2. POST /api/products (Admin)
3. PUT /api/products/:id (Admin)
4. DELETE /api/products/:id (Admin)

### Order Endpoints

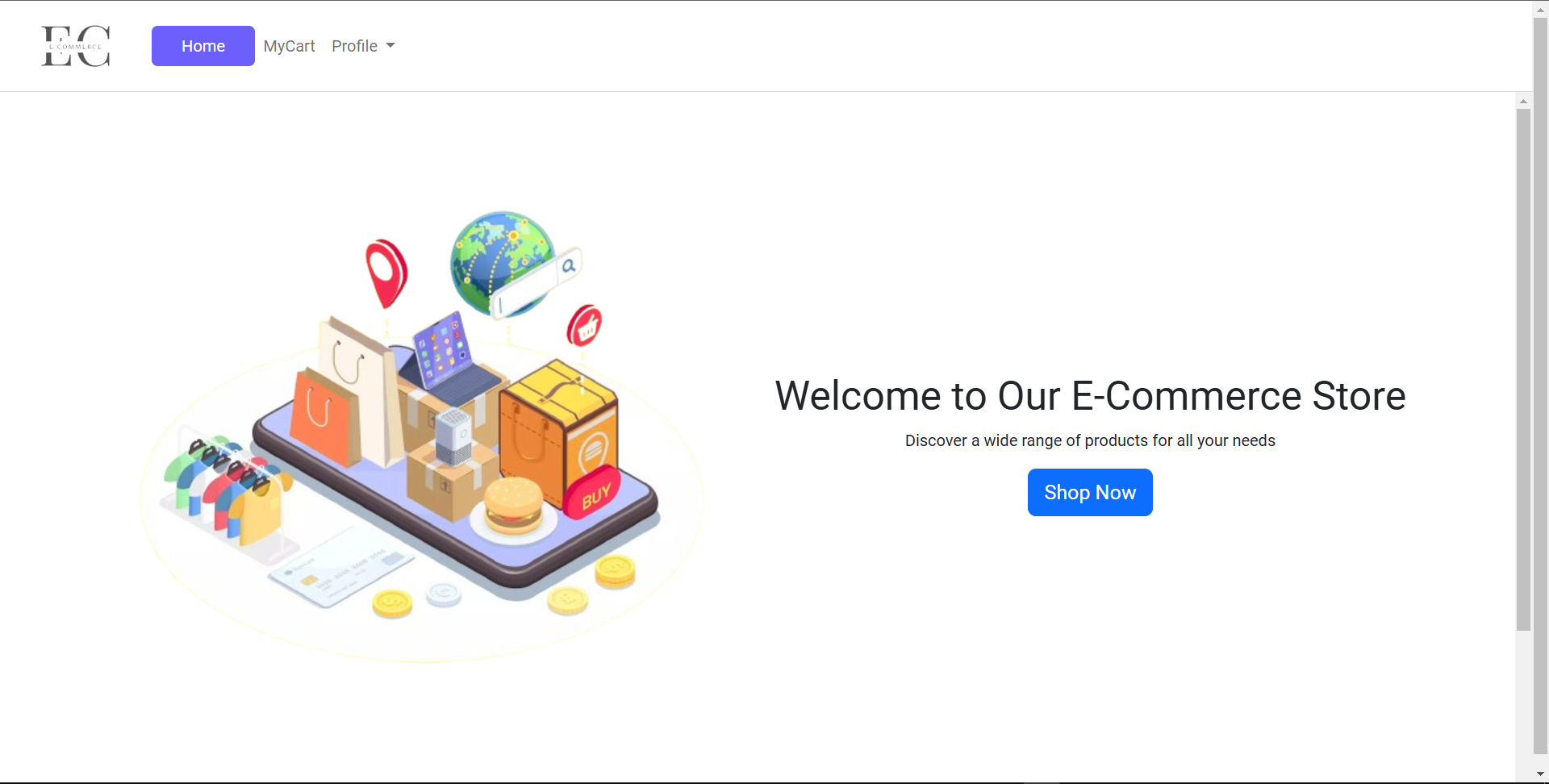
1. GET /api/orders
2. POST /api/orders
3. GET /api/orders/:id

## 8. Authentication

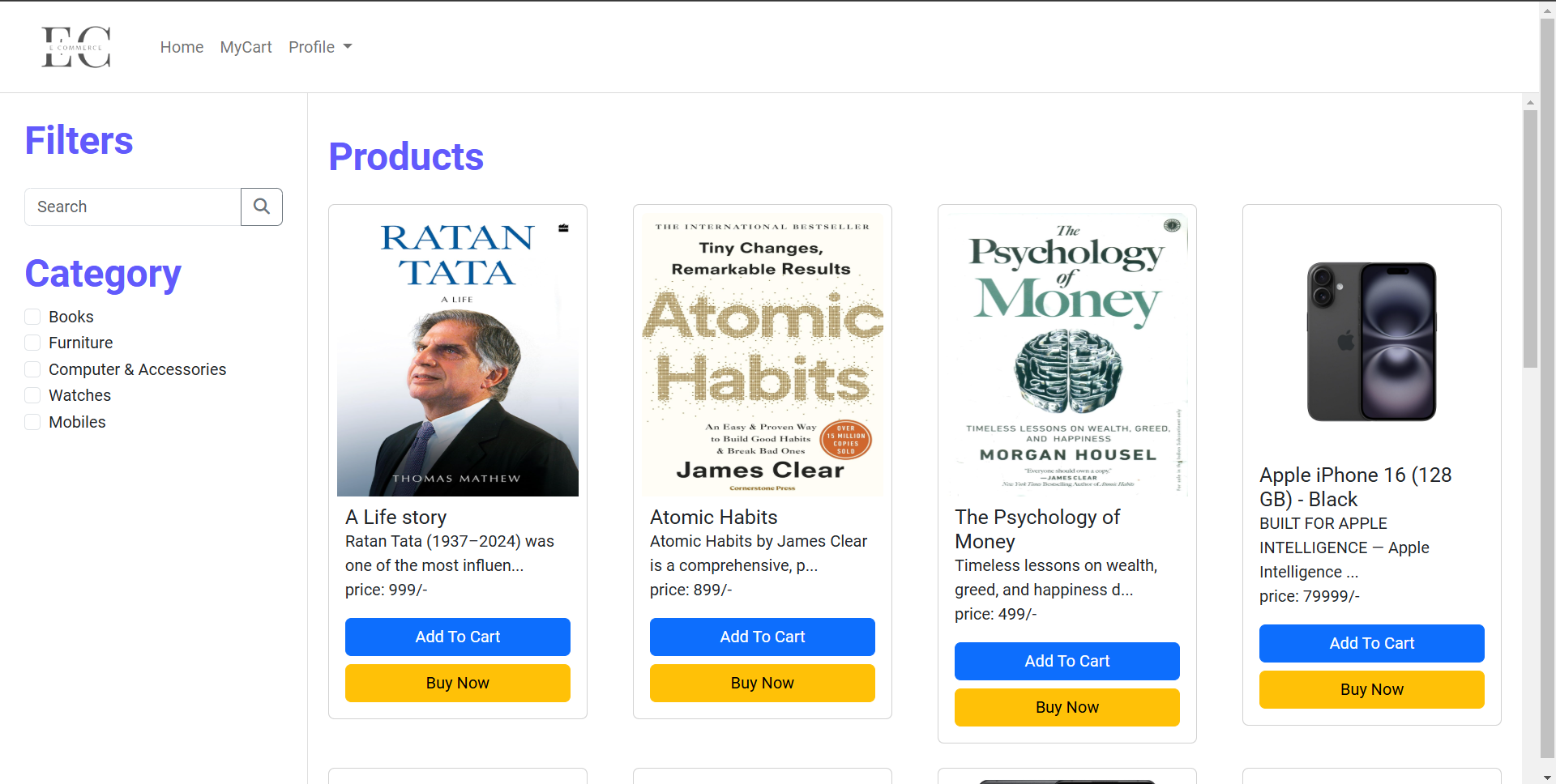
1. JWT-based authentication
2. Role-based access control (Admin/User)
3. Secure password hashing
4. Token expiration and refresh mechanism

## 9. User Interface

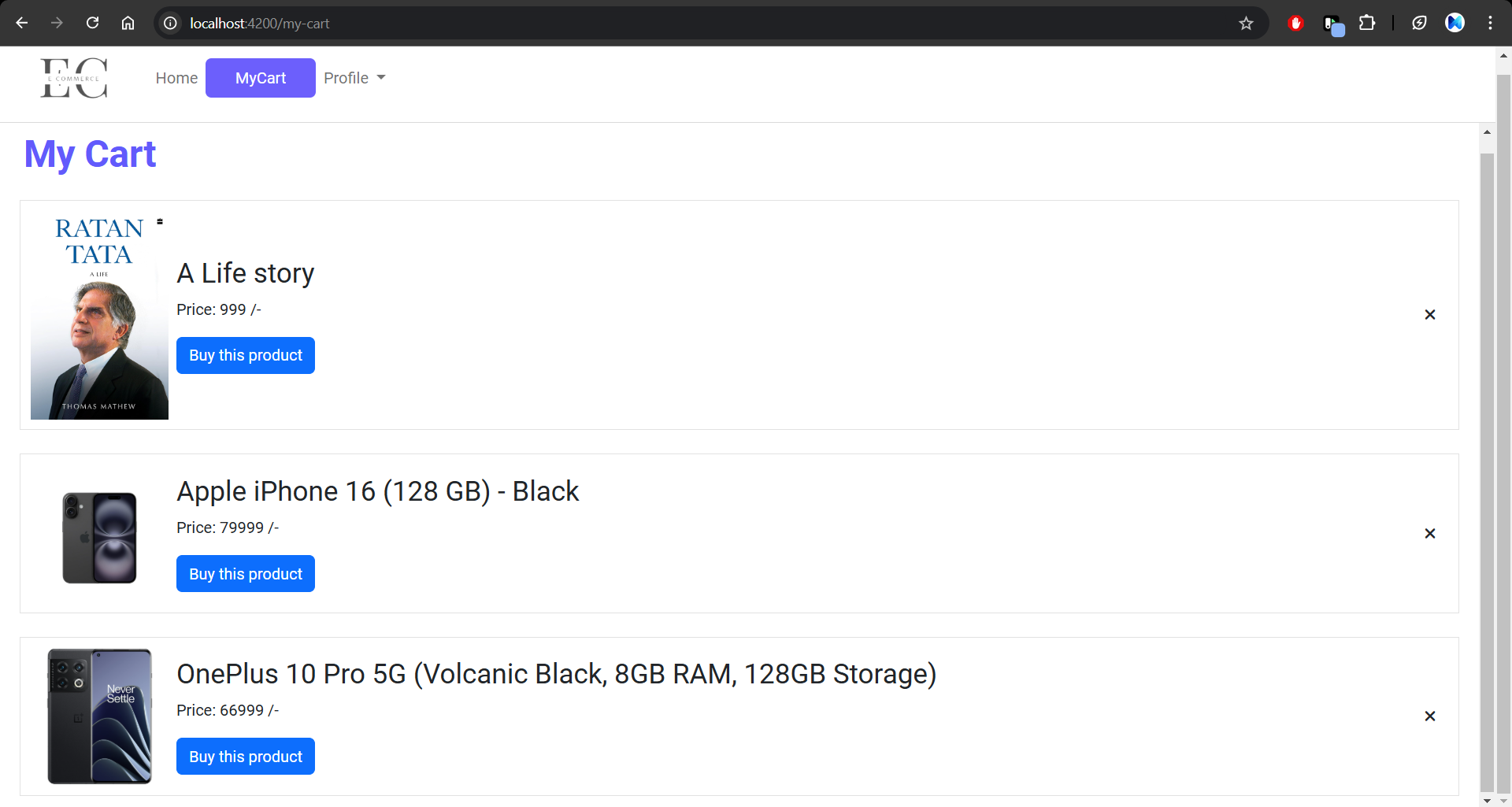
Homepage with product categories



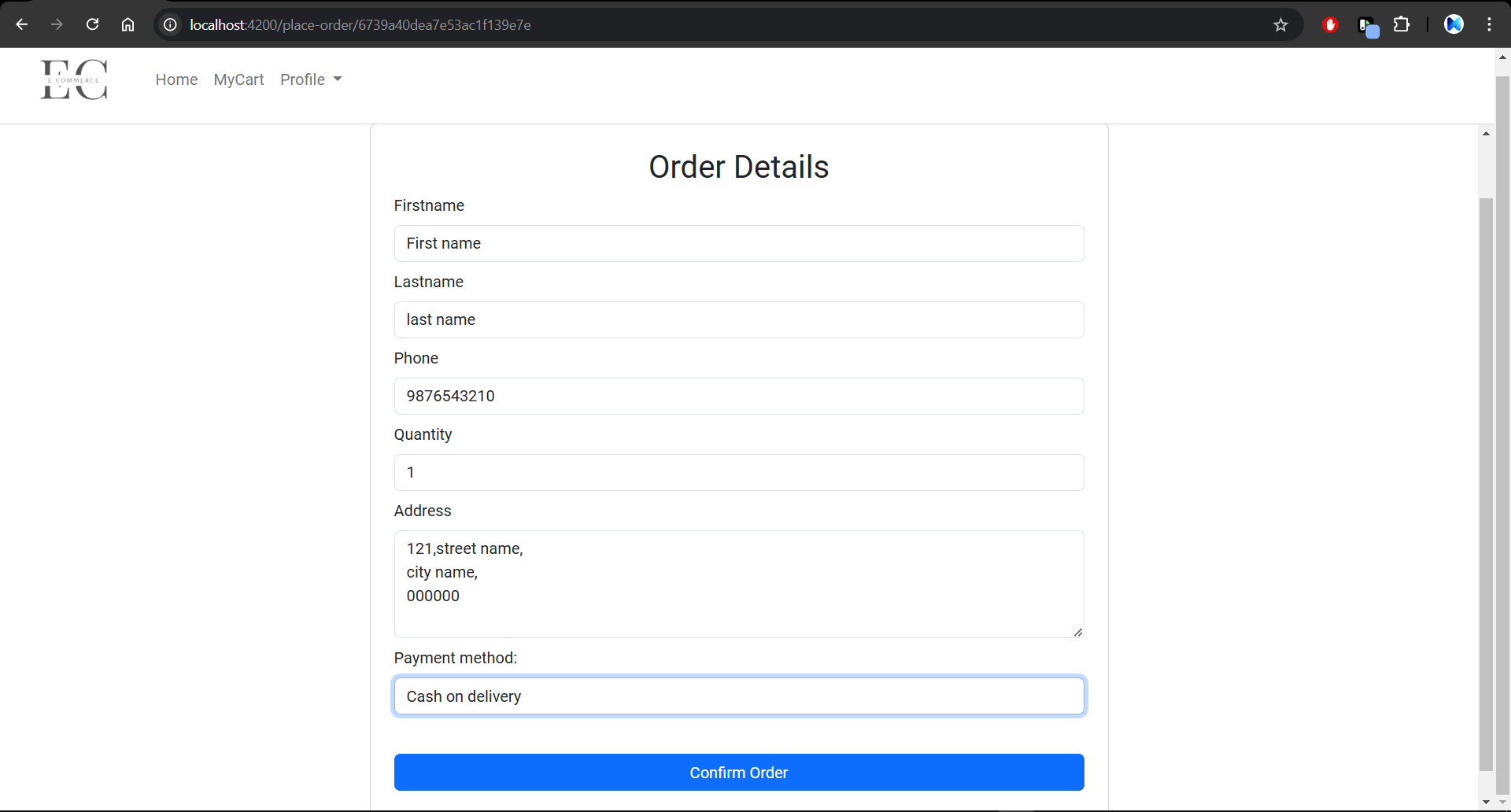
Product listing pages



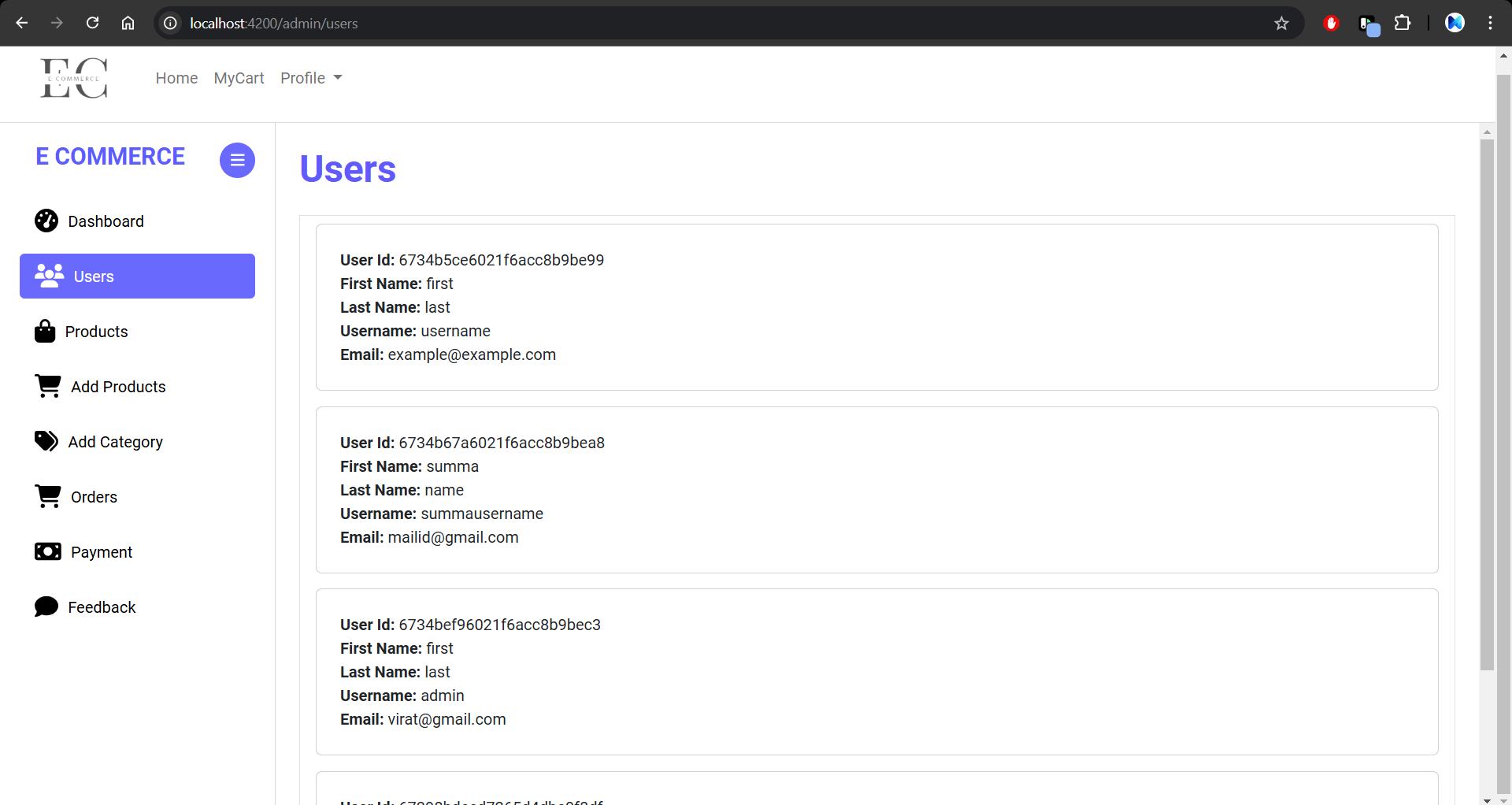
Shopping cart interface



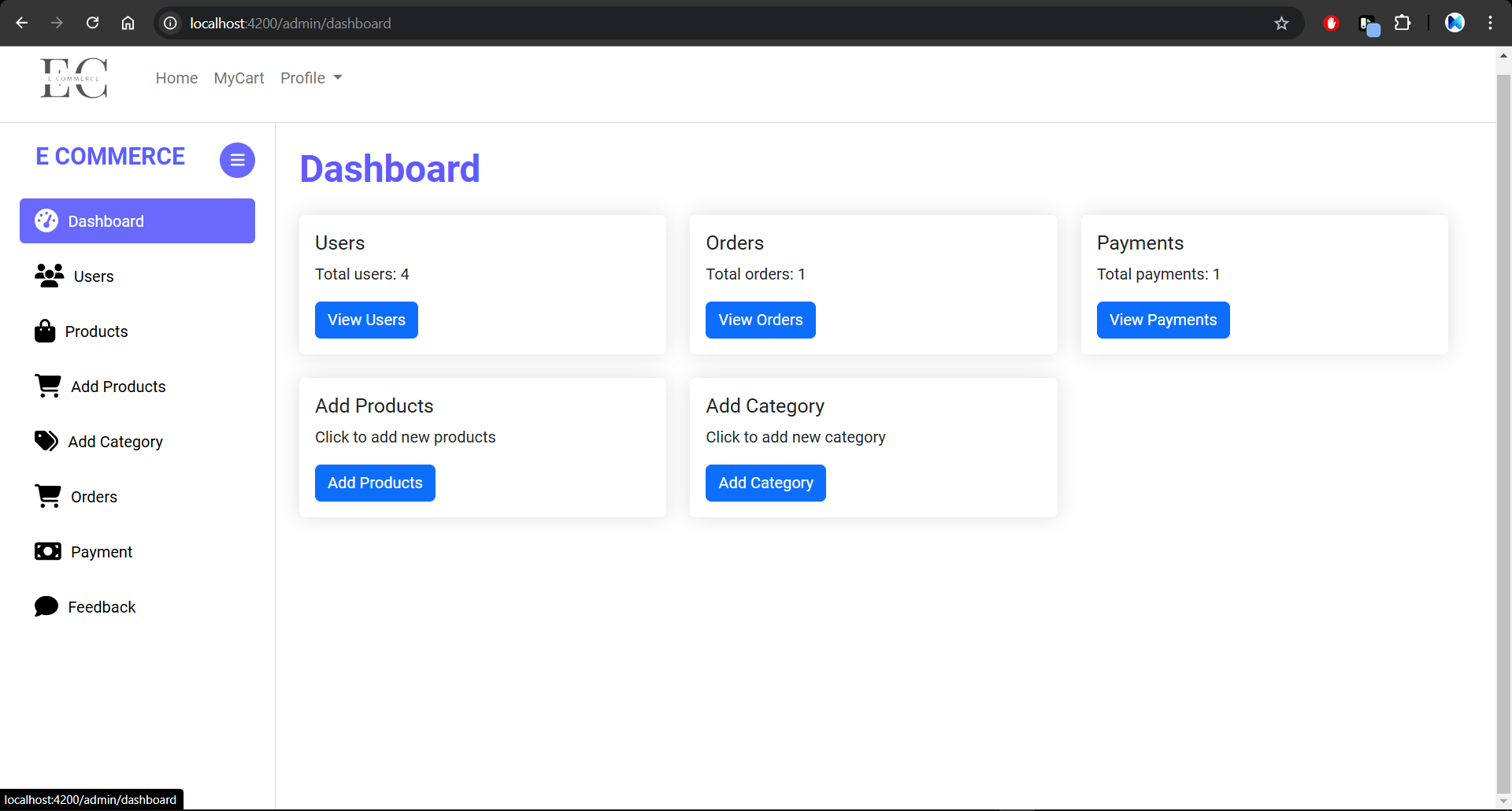
Checkout process



User profile management



Admin dashboard



## 

## 10. Screenshots or Demo

1. Project Repository: https://github.com/MohdMarzook/E-Commerce-App
2. Demo Video: https://github.com/MohdMarzook/E-Commerce-App

## 11. Known Issues

1. Currently implementing responsive design for mobile devices
2. Payment gateway integration in progress
3. Real-time inventory updates need optimization

## 12. Future Enhancements

1. Integration with multiple payment gateways
2. Real-time order tracking
3. Advanced search and filtering options
4. Mobile application development
5. Integration with analytics tools
6. Enhanced reporting features for admin
7. Customer review and rating system
8. Social media integration
9. Loyalty program implementation