

Build & Learn E-Commerce

Project Overview

We are developing an advanced e-commerce application to practise and enhance our skills across multiple technologies. Our project is divided into four parts, each handled by a different team member, focusing on various components such as the mobile app, web interface, admin panel, backend, and UI/UX design. We are using Flutter for the mobile app and UI/UX design, Laravel Filament for the admin panel, and Laravel for API development and backend functionality, and Next Js for frontend Web development. The application will include comprehensive features to provide a rich user experience and robust backend capabilities, ensuring a seamless and efficient e-commerce platform.

Project Breakdown and Requirements

- **1. UI/UX Design in Figma**
- **2. App Development**
- **3. Admin Panel**
- **4. API Development**
- **5. Web Frontend Development**

UI/UX Designer

Key Features:

- User Research & Wireframing:
 - Research user needs and behaviours.
 - Create wireframes for key screens:
 - Home
 - Product Listing
 - Product Details
 - Cart
 - Checkout
 - User Profile
 - Order History
- High-Fidelity Design:
 - Design high-fidelity mockups for each screen, ensuring modern aesthetics.
 - Include detailed interactions and animations.
 - Create designs for both mobile and tablet views.
- Prototyping & User Testing:
 - Develop interactive prototypes to simulate user flows.
 - Conduct user testing sessions and iterate based on feedback.
- Design System:
 - Create a reusable design system with components, color palettes, typography, and iconography.

Flutter Developer

Key Features:

- User Interface:
 - Home Screen: Display product categories and featured products.
 - Product Listing: Show products with filtering and sorting options.
 - Searching.
 - Infinity scrolling.
 - Related products.
 - Category Product list.
 - Product Details: Show product information, images, reviews, and related products.
 - Cart: Allow users to add, view, and update cart items.
 - Checkout: Implement a streamlined checkout process.
 - User Profile: Manage user information, order history, and settings.
 - Authentication: User registration, login, logout, and password reset.
- Navigation & State Management:
 - Implement navigation using Flutter Navigator 2.0 or GoRouter.
 - Use state management solutions like Provider, Riverpod, or Bloc.
- API Integration:
 - Integrate with Laravel backend APIs for data fetching and user actions.

Filament Admin Developer

Key Features:

- Admin Dashboard:
 - Display key metrics and charts, including total sales, number of users, and most sold products.
- CRUD Operations:
 - Manage products, categories, users, and orders with CRUD operations.
 - Ensure proper validation and error handling.
- Order Management:
 - View order details, update order status, and manage returns.
- User Management:
 - Manage user roles and permissions.
 - Handle customer inquiries and support tickets.
- Reports:
 - Generate and export sales and inventory reports.

Backend Developer

Key Features:

- Authentication:
 - Implement user registration, login, logout, and password reset using JWT or Sanctum.
- Product Management:
 - Endpoints for fetching products, categories, and product details.
- Cart Management:
 - Endpoints for adding items to the cart, viewing the cart, updating the cart, and checking out.
- Order Management:
 - Endpoints for placing orders, viewing order history, and tracking order status.
- User Profile Management:
 - Endpoints for updating user information and changing passwords.
- Database Design:
 - Design schemas for products, categories, users, orders, and reviews.
- Security & Testing:
 - Secure all endpoints and perform validation.
 - Write unit and integration tests.
 - Document API endpoints using tools like Swagger or Postman.

Web Developer

Key Features:

- User Interface:
 - Home Screen: Display product categories and featured products.
 - Product Listing: Show products with filtering and sorting options.
 - Product Details: Show product information, images, reviews, and related products.
 - Searching.
 - Infinity scrolling.
 - Related products.
 - Category Product list.
 - Cart: Allow users to add, view, and update cart items.
 - Checkout: Implement a streamlined checkout process.
 - User Profile: Manage user information, order history, and settings.
 - Authentication: User registration, login, logout, and password reset.
- State Management:
 - Use a state management solution like Redux (for React) or Vuex/Pinia (for Vue.js).
- API Integration:
 - Integrate with Laravel backend APIs for data fetching and user actions.
- Responsive Design:
 - Ensure the web application is fully responsive and works well on both desktop and mobile devices.

Detailed Requirements for Each Part

UI/UX Design (1)

- Home Screen:
 - Showcase categories and featured products.
 - Include a search bar and promotional banners.
- Product Listing:
 - Display products with options to filter by category, price, and rating.
 - Include sorting options (e.g., price, popularity).
- Product Details:
 - Show detailed product information, including images, descriptions, specifications, and customer reviews.
 - Include a section for related products.
- Cart & Checkout:
 - Cart: Display items with options to update quantities or remove items.
 - Checkout: Collect shipping information, payment details, and review the order.
- User Profile:
 - Display user information and order history.
 - Provide options to update profile details and change the password.

Flutter App Development (2)

- Home Screen:
 - Implement a grid view for categories and a carousel for featured products.
 - Add a search functionality.
- Product Listing:
 - Implement a list view with filtering and sorting options.
 - Include lazy loading for better performance.
- Product Details:
 - Create a detailed view with image carousel, product details, and reviews.
 - Implement a "Related Products" section.
- Cart & Checkout:
 - Implement a cart view with item update and removal options.
 - Develop a multi-step checkout process.
- User Profile:
 - Create a profile view with editable fields.
 - Display order history with detailed order information.
- Authentication:
 - Implement registration, login, logout, and password reset screens.

Filament Admin Panel (3)

- Dashboard:
 - Create widgets for displaying sales, users, and product statistics.
- Product Management:
 - Implement CRUD operations for managing products and categories.
 - Include bulk actions and export options.
- Order Management:
 - Develop views for viewing and updating order statuses.
 - Implement return and refund management.
- User Management:
 - Create views for managing user roles and permissions.
 - Develop a support ticket system.
- Reports:
 - Generate sales and inventory reports.
 - Provide export options for reports.

Laravel API Development (4)

- Authentication:
 - Implement endpoints for user registration, login, logout, and password reset.
 - Use JWT or Sanctum for authentication.
- Product Management:
 - Develop endpoints for fetching products, categories, and product details.
 - Implement filtering and sorting capabilities.
- Cart Management:
 - Create endpoints for adding items to the cart, viewing the cart, updating cart items, and checking out.
- Order Management:
 - Develop endpoints for placing orders, viewing order history, and tracking order status.
- User Profile Management:
 - Implement endpoints for updating user information and changing passwords.
- Database Design:
 - Design schemas for products, categories, users, orders, and reviews.
- Security & Testing:
 - Secure endpoints and perform validation.
 - Write unit and integration tests.
 - Document API endpoints using Swagger or Postman.

Web Frontend Development (5)

- **Home Screen:**
 - Implement a grid view for categories and a carousel for featured products.
 - Add a search functionality.
- **Product Listing:**
 - Implement a list view with filtering and sorting options.
 - Include lazy loading for better performance.
- **Product Details:**
 - Create a detailed view with image carousel, product details, and reviews.
 - Implement a "Related Products" section.
- **Cart & Checkout:**
 - Implement a cart view with item update and removal options.
 - Develop a multi-step checkout process.
- **User Profile:**
 - Create a profile view with editable fields.
 - Display order history with detailed order information.
- **Authentication:**
 - Implement registration, login, logout, and password reset screens.
- **Responsive Design:**
 - Ensure the web application is fully responsive and works well on both desktop and mobile devices.

The End for now