

FASHION STORE

Project Documentation

Under the supervision of the **eng**ineer : **Ashraf Sadek   
  
Developed by :**



**1. Project Overview**

**Project Title:** Fashion Store  
**Objective:** Build a fashion e-commerce platform for browsing, adding items to a cart, and completing purchases.

**2. Key Features**

* Product display
* Shopping cart functionality (stored in database)
* Checkout and payment integration

**3. Database Models and Schemas**

**3.1 User Schema**

A black screen with white text

Description automatically generated

**Purpose:** Stores user information for authentication and profile management.

**3.2 Product Schema**

A screen shot of a computer program

Description automatically generated

**Purpose:** Contains details about the products available for purchase.

**3.3 Cart Schema**

A computer screen with white and pink text

Description automatically generated

**Purpose:** Manages the shopping cart for each user, linking products with quantities.

**4. Backend Technologies**

* **Technologies:** Node.js, Express.js
* **Functionality:**
  + API for handling requests (e.g., fetching products, managing the cart, processing orders).
  + Communicates with MongoDB to store and retrieve data.

**5. API Endpoints**

**5.1 Users API**

* **POST** http://localhost:5000/api/users/register
  + **Description:** Register a new user.
  + **Request Body:**

A screenshot of a computer

Description automatically generated

* **POST** <http://localhost:5000/api/users/login>
  + **Description:** Log in a user.
  + **Request Body:**

A screenshot of a computer screen

Description automatically generated

**5.2 Products API**

* **GET** http://localhost:5000/api/products/
  + **Description:** List all products.
* **POST** http://localhost:5000/api/products/
  + **Description:** Add a new product.
  + **Request Body:**

A screenshot of a computer

Description automatically generated

**5.3 Cart API**

* **GET** http://localhost:5000/api/cart/:user\_id
  + **Description:** View the user's cart.
* **DELETE** http://localhost:5000/api/cart/:user\_id/:product\_id
  + **Description:** Remove a product from the user's cart.
* **POST** http://localhost:5000/api/cart/
  + **Description:** Add a product to the cart.
  + **Request Body:**

A screenshot of a computer

Description automatically generated

**6. Frontend Technologies**

* **Technologies:** React, useContext (for user state), HTML, CSS, JavaScript
* **Functionality:**
  + User interface for displaying products, managing the cart, and handling user interactions.

**7. Key Functionalities**

* **Product Listing and Detail Pages:** Displays products with details such as name, price, and description.
* **Shopping Cart:** Add items to the cart, which is stored in MongoDB for persistence.
* **Checkout:** Order summary, shipping details, and payment processing.

**8. User State Management**

* **useContext:** Managing user login status and ensuring the cart is linked to the user session.
* **Why useContext:**
  + Simplifies user state management across components compared to Redux.

**9. Frontend Pages**

**9.1 Home Page**

* **Description:** Displays a list of products with options to view details and add items to the cart.

**9.2 About Page**

* **Description:** Provides information about the company, its mission, and values.

**9.3 Contact Us Page**

* **Description:** Allows users to reach out for inquiries or support.

**9.4 Login Page**

* **Description:** Enables users to log in to their accounts.

**9.5 Register Page**

* **Description:** Allows new users to create an account.

**9.6 Cart Page**

* **Description:** Displays the items added to the user's cart.

**10. Future Improvements**

**10.1 Checkout System**

* **Description:** Implement a streamlined checkout process to enhance user experience.

**10.2 Payment System**

* **Description:** Integrate a secure payment gateway.

**10.3 Secure Sessions**

* **Description:** Enhance user security during sessions.

**10.4 Admin Portal**

* **Description:** Create an interface for admins to manage products and orders.

**10.5 UI Improvements Based on Customer Feedback**

* **Description:** Refine the user interface to meet user needs better.

**11. Conclusion**

* **Final Thoughts:** This e-commerce web app provides a user-friendly platform for purchasing fashion products with a modern tech stack.