

“Planning the technical foundation- [Fashion Shop]”

Technical Requirements:

Frontend:

The main frontend framework for creating a quick, scalable, and SEO-friendly marketplace will be Next.js. It will handle both Server-Side Rendering (SSR) and Static Site Generation (SSG).

Technology Stack:

- React (used by Next.js)
- Tailwind CSS for styling

Backend

Next.js API Routes and Sanity

Next.js will handle the backend logic using API routes, which are built-in serverless functions that execute server-side code for things like authentication, order management, etc. These routes will integrate with Sanity CMS for managing content like products, categories, reviews, and more.

API Routes:

Authentication API: Handles user registration, login, and JWT authentication.

Product API: Fetches products and product details from Sanity CMS.

Order API: Handles cart creation, checkout, and order management.

User API: Manages user profiles and order history.

Payment API: Integrates with a payment gateway (like Stripe, PayPal) for processing payments.

Review API: Allows users to leave and manage product reviews.

Sanity Schema:

Product: Stores product data, including titles, descriptions, images, prices, and stock quantities.

Category: Organizes products into categories like "Shirts", "Pants", "Jackets".

Order: Manages user orders, including product details, total price, shipping info, and status.

User: Stores user profile information, addresses, and past orders.

Review: Stores product reviews and ratings.

Sanity API will be used to retrieve product and category data for the frontend, while Admin users will use Sanity Studio to manage products, categories, and reviews.

Authentication:

For user authentication and management, JWT can be used to securely manage user sessions.

Authentication Flow:

User Registration: A new user signs up using their email and password.

Login: A user logs in, and the server issues a JWT token that is stored in the browser's local storage or cookies.

Payment Gateway Integration:

For processing payments, I will integrate a payment service like Stripe or PayPal. The payment flow involves:

Checkout Page → Payment Processing → Confirmation → Payment API

we will use Supabase or MongoDB for storing user-related data like order history, user profiles, and order statuses.