

HackathonDay-2

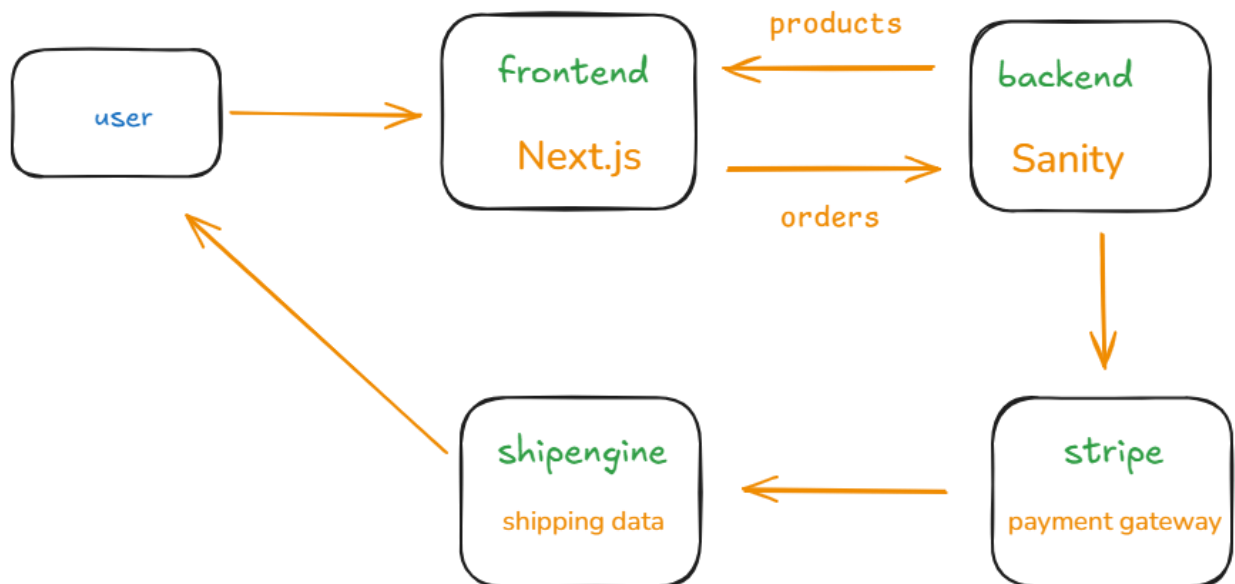
Marketplace Technical Foundation – E-Commerce Website

1. System Architecture

Overview

This document outlines the technical foundation for the shoe marketplace e-commerce website, detailing frontend and backend interactions, API endpoints, and database schemas.

Architecture Diagram



Data Flow

1. **User Interaction:** Users browse products and add them to the cart.
2. **Product Retrieval:** Data is fetched from Sanity CMS.

3. **Checkout & Payment Processing:** Order details are stored in Sanity, and payment is processed via Stripe.
 4. **Order Confirmation:** Once payment is confirmed, order details are updated.
 5. **Shipping & Tracking:** Order details are sent to ShipEngine for shipping management.
 6. **Order Status Updates:** Users can track their order using ShipEngine APIs.
-

2. Technical Requirements

Frontend (Next.js, TypeScript, Tailwind CSS)

- Home Page: Display featured and trending shoes.
- Product Listing Page: List all shoes with filtering options.
- Product Detail Page: Show detailed shoe information.
- Cart Page: Enable users to add/remove multiple products before checkout.
- Checkout Page: Collect user information and process payments via Stripe.
- Order Confirmation Page: Display order details and tracking information.

Backend (Sanity CMS)

- **Products:** Store product name, price, description, stock, images, and category.
- **Orders:** Save customer info, ordered products, total price, and order status.
- **Customers:** Manage users with names, emails, and order history.

Third-Party APIs

- **Stripe:** Handle payment processing.
 - **ShipEngine:** Manage shipping and order tracking.
-

3. API Requirements

Endpoints & Methods

Product APIs

- GET /api/products → Fetch all shoes.
- GET /api/products/:id → Fetch details of a single shoe.

Order APIs

- POST /api/orders → Create a new order.

- GET /api/orders/:id/status → Fetch order status.

Payment APIs

- POST /api/checkout → Process payment via Stripe.

Shipping APIs

- GET /api/shipping/:id → Retrieve shipping details from ShipEngine.
-