

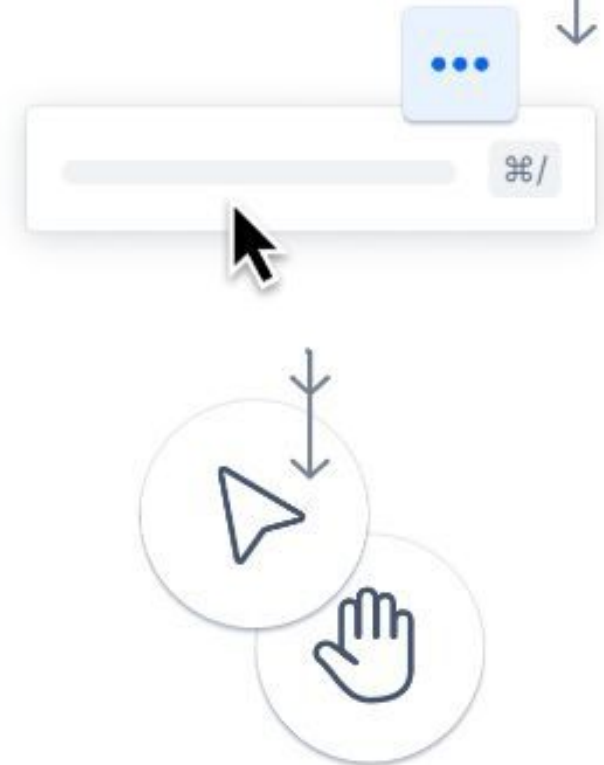
Marketplace Technical Foundation - E-commerce Website

System Architecture Overview

The architecture for this e-commerce marketplace integrates various components for a seamless user experience and robust backend management. The system includes a frontend (Next.js), Sanity CMS for content management, third-party services for tracking and payment processing, and an API layer to handle data flow.

Key Components

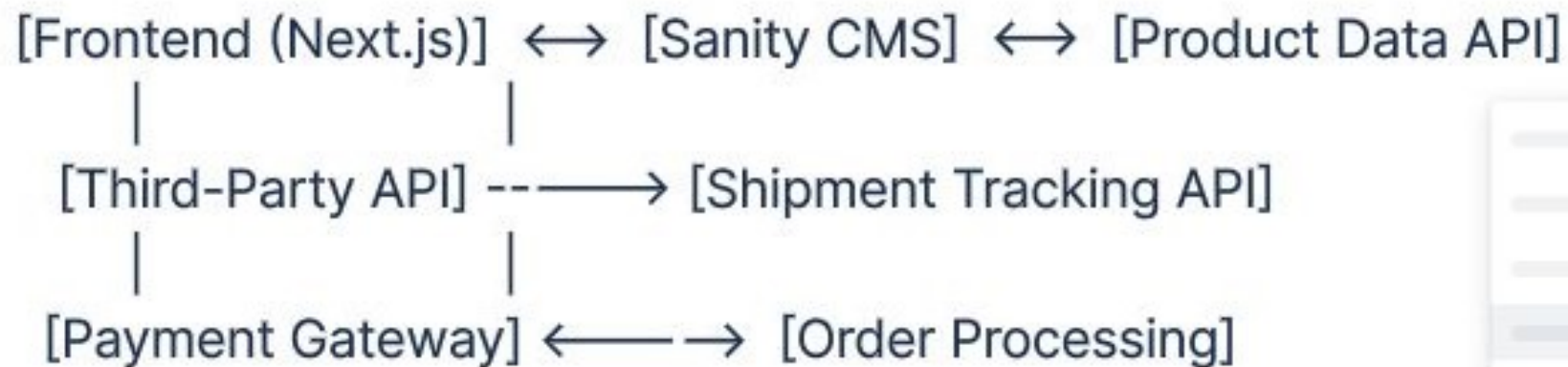
- **Frontend (Next.js):** Dynamic user interface for browsing, searching, and purchasing products.
- **Sanity CMS:** Manages product catalog, customer details, and order data.
- **Third-Party APIs:** External services for shipment tracking, real-time delivery updates, and payment gateway integration.
- **Payment Gateway:** Secure processing of customer payments

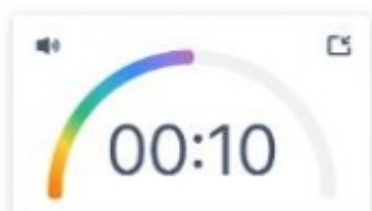


Key Components

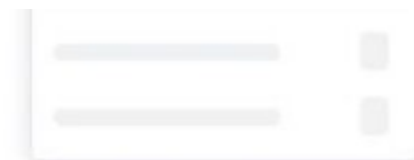
- **Frontend (Next.js):** Dynamic user interface for browsing, searching, and purchasing products.
- **Sanity CMS:** Manages product catalog, customer details, and order data.
- **Third-Party APIs:** External services for shipment tracking, real-time delivery updates, and payment gateway integration.
- **Payment Gateway:** Secure processing of customer payments

System Architecture Diagram





Detailed Workflows



User Registration and Authentication

- **Step 1:** User creates an account by providing details such as name, email, and password.
- **Step 2:** Data is stored securely in Sanity CMS under the users schema.
- **Step 3:** Confirmation email sent to the user for account verification.

2. Product Browsing

- **Step 1:** User navigates through various categories of products.
- **Step 2:** Product data is dynamically fetched from Sanity CMS using a GET API endpoint (/products).
- **Step 3:** Products are displayed on the frontend (Next.js) with detailed descriptions, prices, and images.

3. Shopping Cart and Order Placement

- **Step 1:** User adds items to the shopping cart.
- **Step 2:** User proceeds to checkout, entering shipping and payment details.
- **Step 3:** Order details are sent to Sanity CMS for storage and tracking.
- **Step 4:** Payment gateway processes the payment securely and returns the status.

4. Shipment Tracking

- **Step 1:** After order placement, the user is redirected to the order confirmation page with shipment tracking information.
- **Step 2:** Real-time shipment updates are fetched from the Third-Party Shipment API and displayed on the frontend.

Schema Relationship

Explanation:

1. **Product to Order:** The **Product** schema is referenced in the **Order** schema via the orderDetails field. Each order can contain multiple products.
2. **User to Order:** The **User** schema is linked to the **Order** schema through the customerInfo field. This establishes that an order is placed by a specific user.
3. **Order's Data:** The **Order** schema tracks the total amount, payment status, and order status. It also aggregates data about the **User** and **Product(s)** involved in the transaction.

Schemas Diagram

```
+-----+
| Product |
+-----+
| id      |
| name    |
| price   |
| description |
| stock   |
| image   |
+-----+
```

v

```
+-----+
| Order   |
+-----+
| orderId |
| totalAmount |
| paymentStatus |
| orderStatus |
| customerInfo |
| orderDetails |
+-----+
```

v

```
+-----+
| User    |
+-----+
| id      |
| name    |
| email   |
| address |
| password |
+-----+
```