Marketplace Technical Foundation - My Coffee Shop

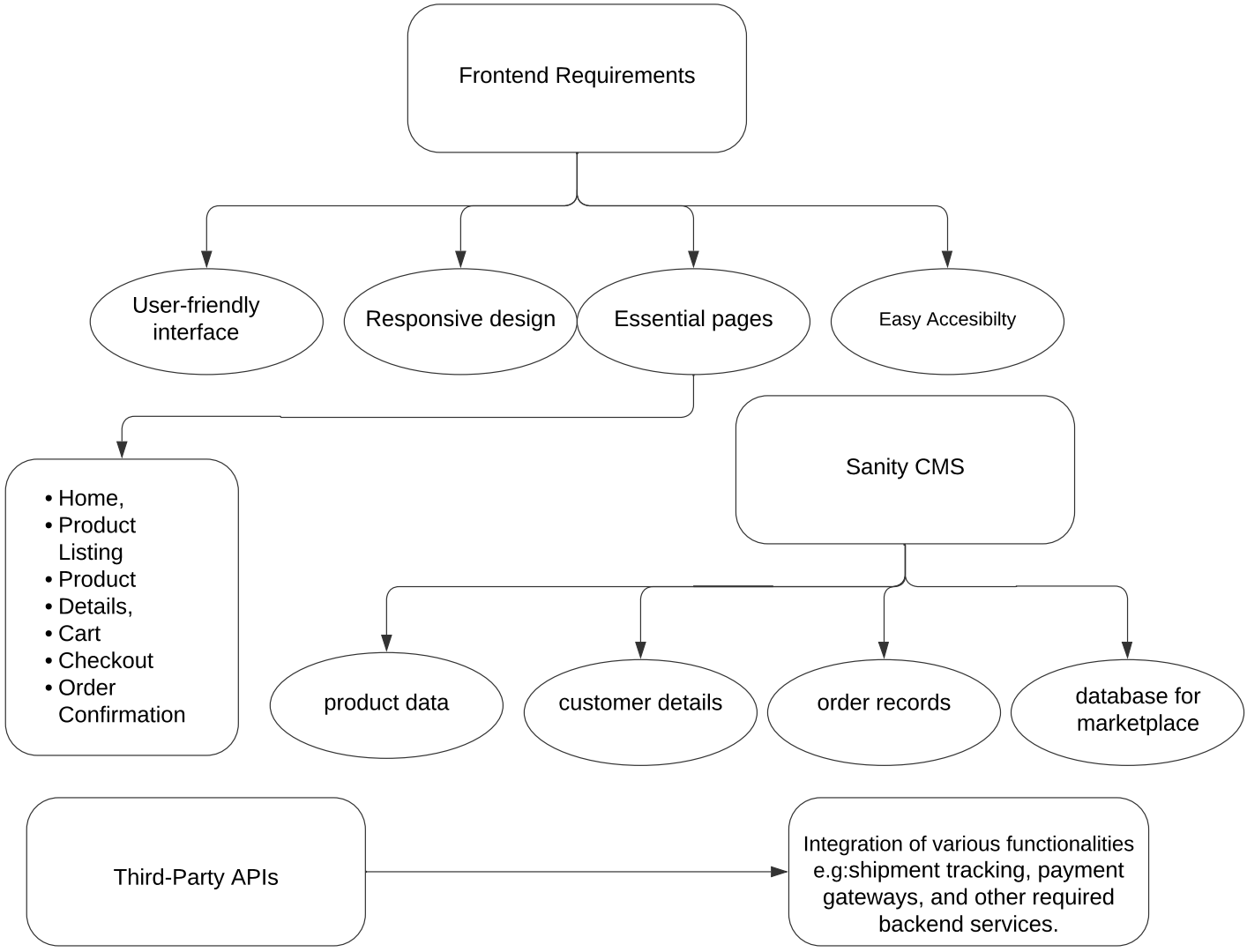
Overview

This document serves as a **blueprint for a Q-Commerce Marketplace**, detailing the essential components and workflows that define its architecture. It provides a comprehensive plan tailored to the unique requirements of a fast-paced, on-demand commerce ecosystem. The document is structured to help visualize the interactions between various system components and to guide effective planning and integration.
Key Highlights:

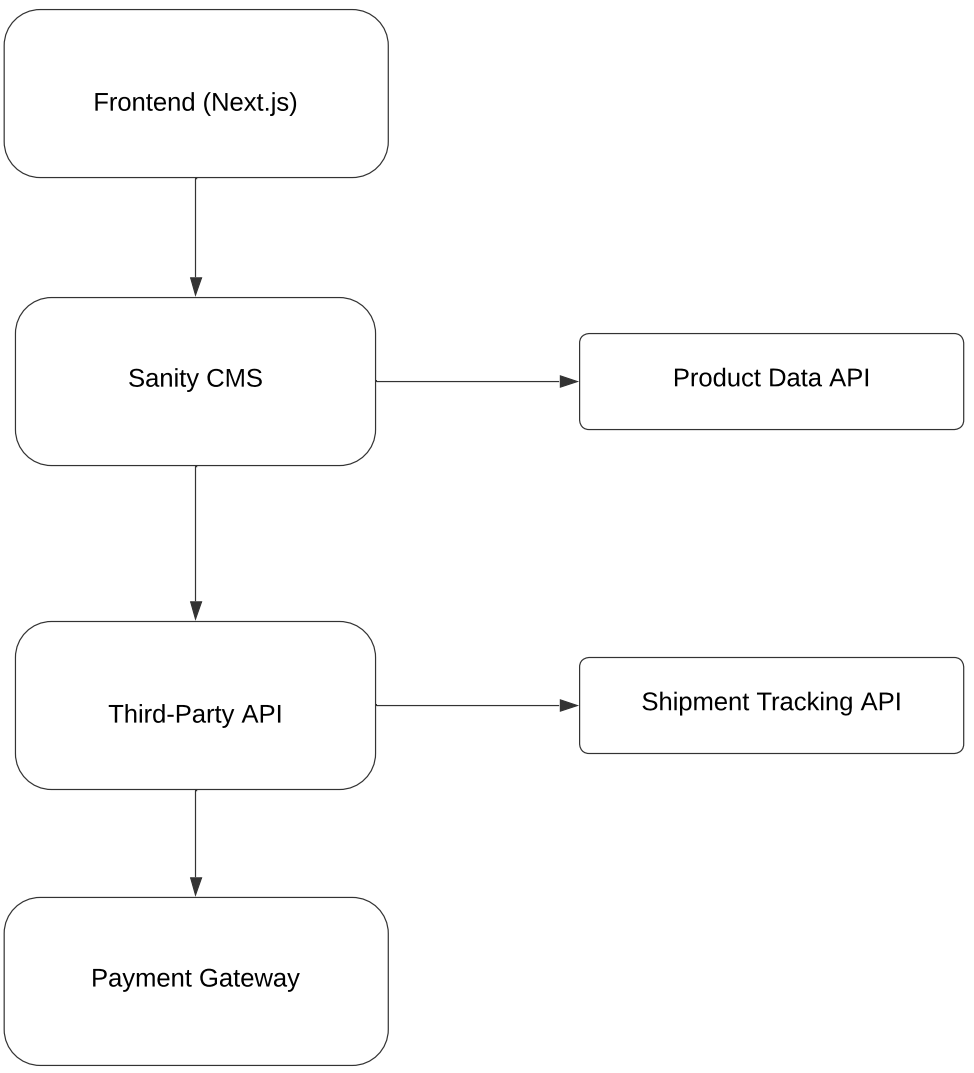
1. **System Architecture and Frontend-Sanity CMS Integration**
   - A clear diagram illustrating how the **frontend interacts with Sanity CMS** for dynamic content management.
   - The architecture outlines workflows customized for the q-commerce model, enabling seamless content updates and user experiences.
   - **High-Level System Design and Workflow Description**
   - Includes a high-level diagram showcasing the interaction between system components, such as:
     - Frontend
     - Sanity CMS for content management
     - Third-party APIs for logistics management
     - Payment gateways for secure transaction processing.
   - These visualizations help identify dependencies and streamline integration planning.
   - **API Endpoints and Schema Examples**
   - Provides examples of API endpoints critical for operations, including:
     - Product catalog retrieval
     - Order placement and tracking
     - Inventory updates.
   - Includes a sample schema to demonstrate data structure and facilitate backend development.

This blueprint will act as a foundation for building an efficient, scalable, and user-focused q-commerce marketplace, ensuring all key aspects of system design, interaction, and integration are addressed comprehensively.

Technical Requirements



High-level System Architecture Diagram



Workflow

1. User Registration

- **Step 1:** The user signs up by entering their details on the frontend (Next.js).
- **Step 2:** The entered data is securely stored in the **Sanity CMS** under the "User Data Storage" module.
- **Step 3:** A confirmation message (email or SMS) is sent to the user, completing the registration process.

2. Product Browsing

- **Step 1:** The user navigates the product categories or searches for specific items via the frontend.
- **Step 2:** The **Sanity CMS Product Data API** fetches the relevant product information based on the user's input.
- **Step 3:** The fetched product data is displayed dynamically on the frontend for the user to browse.

3. Order Placement

- **Step 1:** The user selects products and adds them to the cart on the frontend.
- **Step 2:** At checkout, the order details, including product selections, user information, and payment preferences, are captured.
- **Step 3:** The order details are stored in the **Sanity CMS Order Management** module.
- **Step 4:** The **Payment Gateway** processes the payment, confirming the transaction, and updates the order status in Sanity.

4. Shipment Tracking

- **Step 1:** Once an order is placed, the shipment status is updated in real-time using a **Third-Party Shipment Tracking API**.
- **Step 2:** The API retrieves the current status of the delivery and sends the information back to the frontend.
- **Step 3:** The shipment updates are displayed to the user in an easy-to-understand format, ensuring transparency in the delivery process.