System Architecture Document

1. Introduction

Purpose: To provide a high-level overview of the architecture and design of the eCommerce system, detailing how components interact to deliver seamless functionality.

Scope: Focuses on the interaction between the frontend, CMS, APIs, and external services (payment and shipment) to enable smooth workflows.

2. Architecture Diagram

Components:

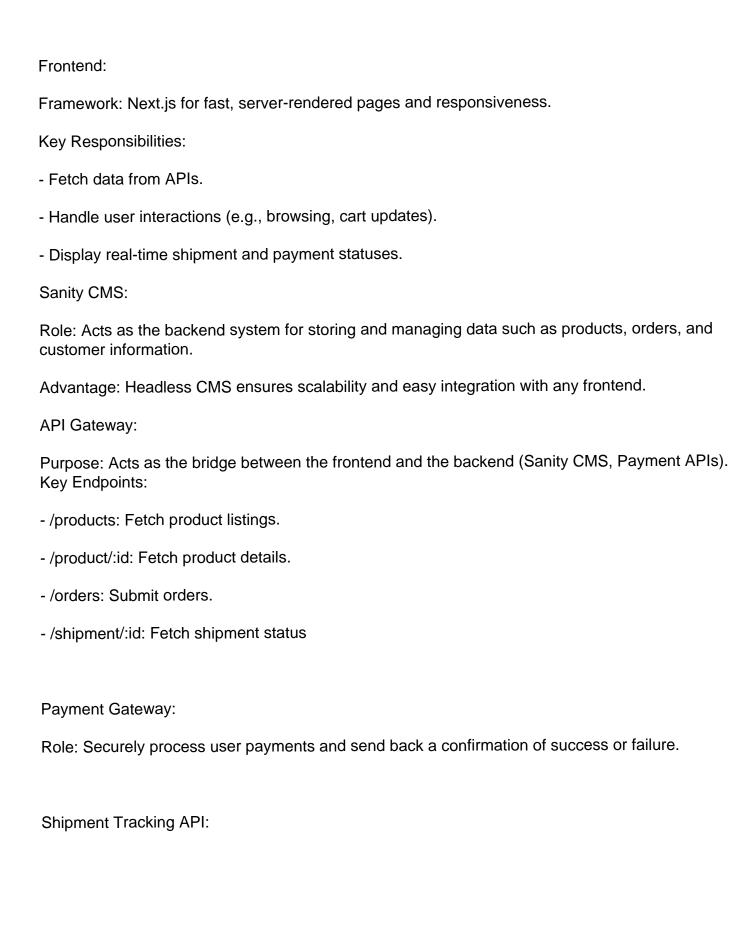
- User (Browser)
- Frontend (Next.js)
- API Gateway (Custom API Endpoints)
- CMS (Sanity for data storage)
- Payment Gateway (Third-party for secure transactions)
- Shipment Tracking API (Real-time updates)

Visual Flow:

- 1. User interacts with the Frontend (Next.js).
- 2. Frontend communicates with:
 - Sanity CMS for product and order data.
 - Custom APIs for handling orders and shipment status.
 - Third-party APIs for payments and shipment tracking.

3. Component Overview

System Architecture Document



System Architecture Document

Role: Provides shipment tracking information for orders.

Example: Integrates with logistics services like DHL or FedEx.

4. Data Flow

Detail how data moves through the system:

- 1. User Action: Browses products -> Frontend fetches data from /products.
- 2. Product Details: User selects a product -> Frontend fetches details via /product/:id.
- 3. Cart & Checkout: User adds items to cart and checks out -> Order data is sent to /orders -> Payment processed by the Payment Gateway.
- 4. Shipment Tracking: User tracks an order -> Frontend fetches status from /shipment/:id.