

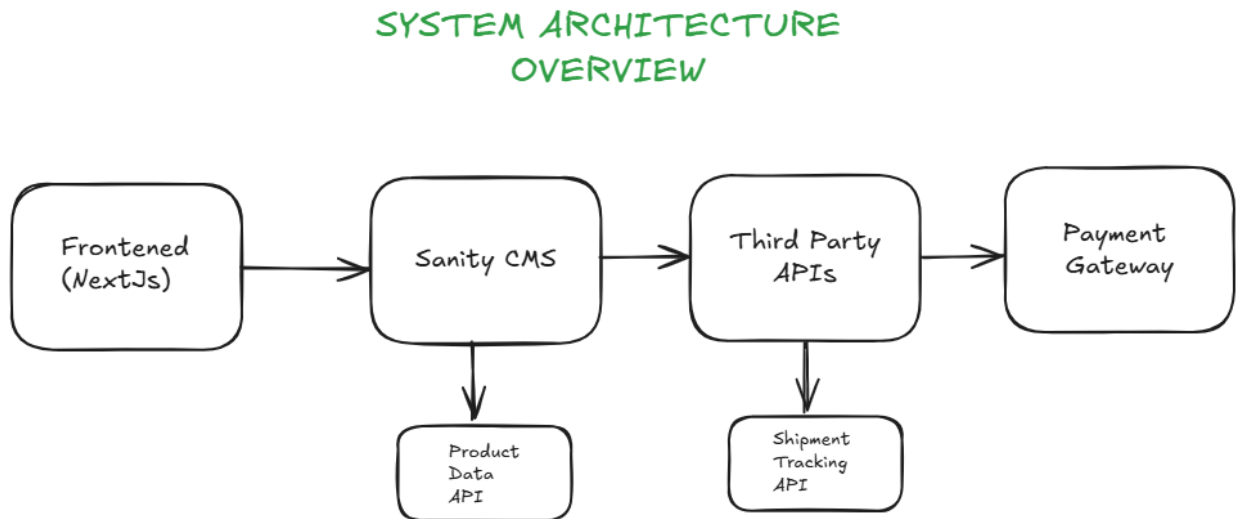
Task for Day 2

Marketplace Technical Foundation - FURNIRO

1. System Architecture Overview

Diagram

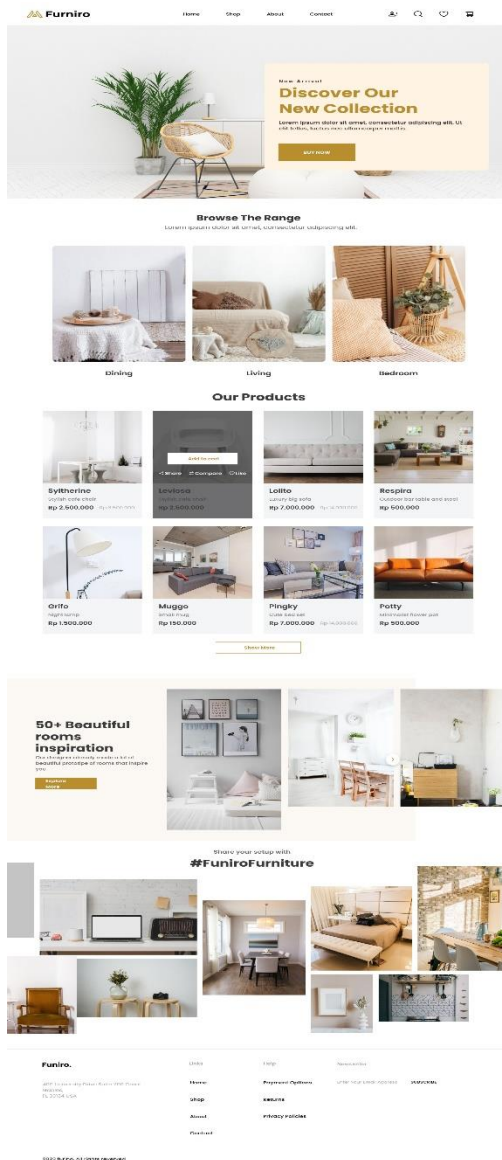
Here is a high-level system architecture diagram:



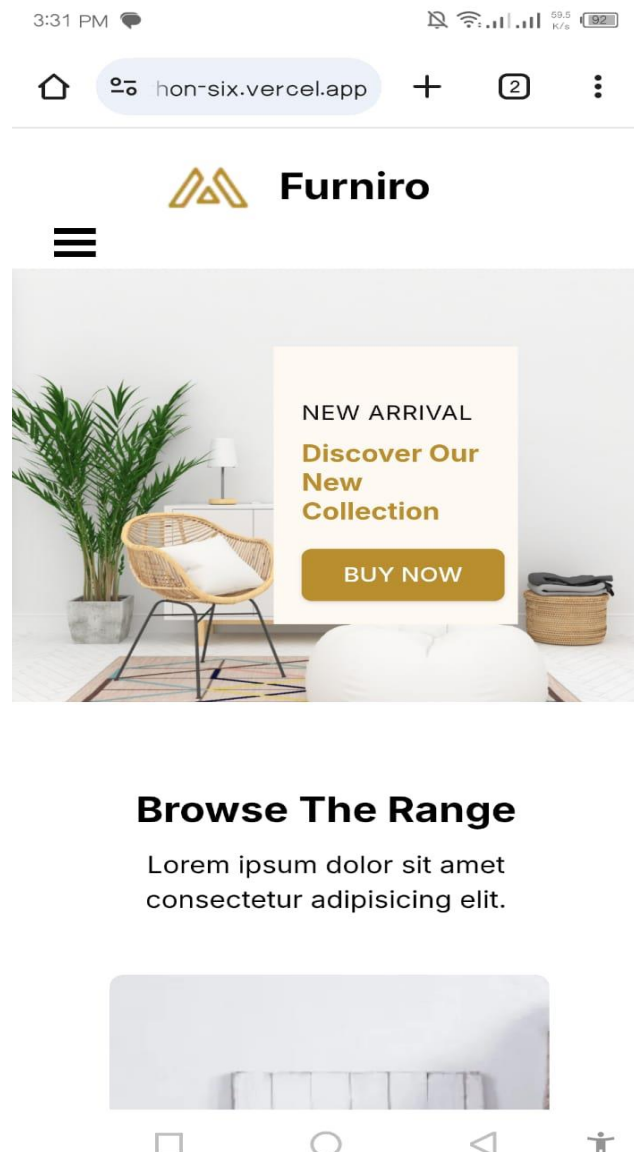
Description of Components

1. Frontend (Next.js)

- Provides a user-friendly interface for browsing, searching, and purchasing products.
- Integrates with APIs for dynamic content rendering and user interactions.



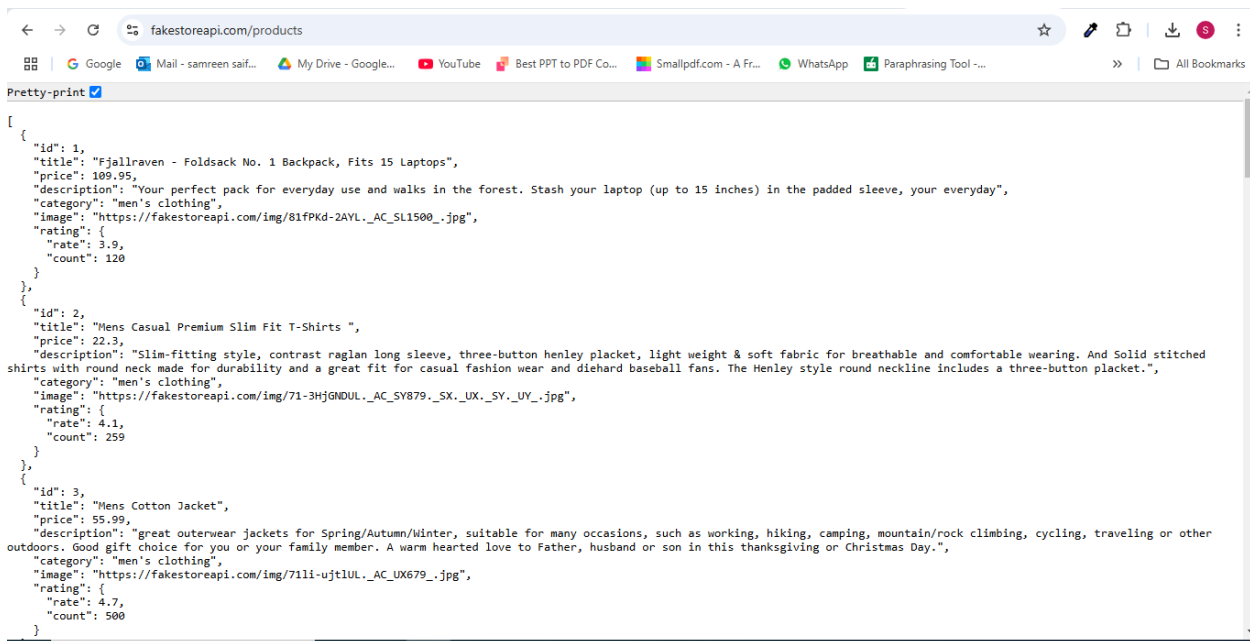
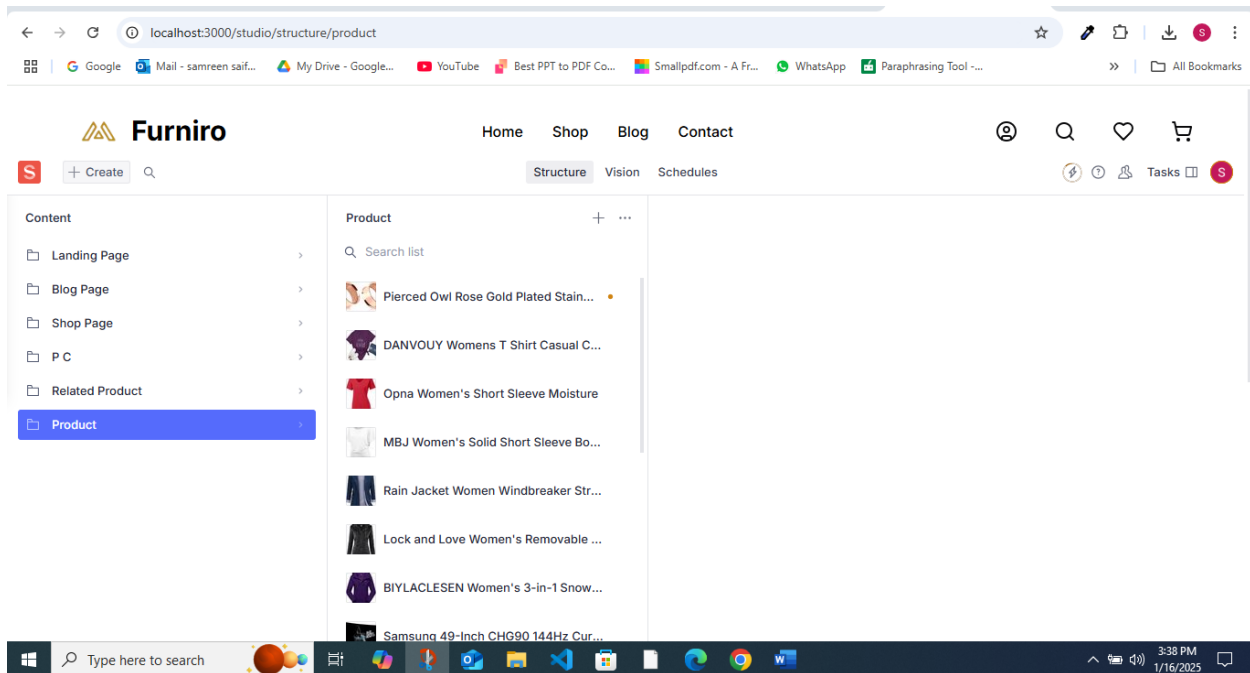
Laptop View



Mobile view

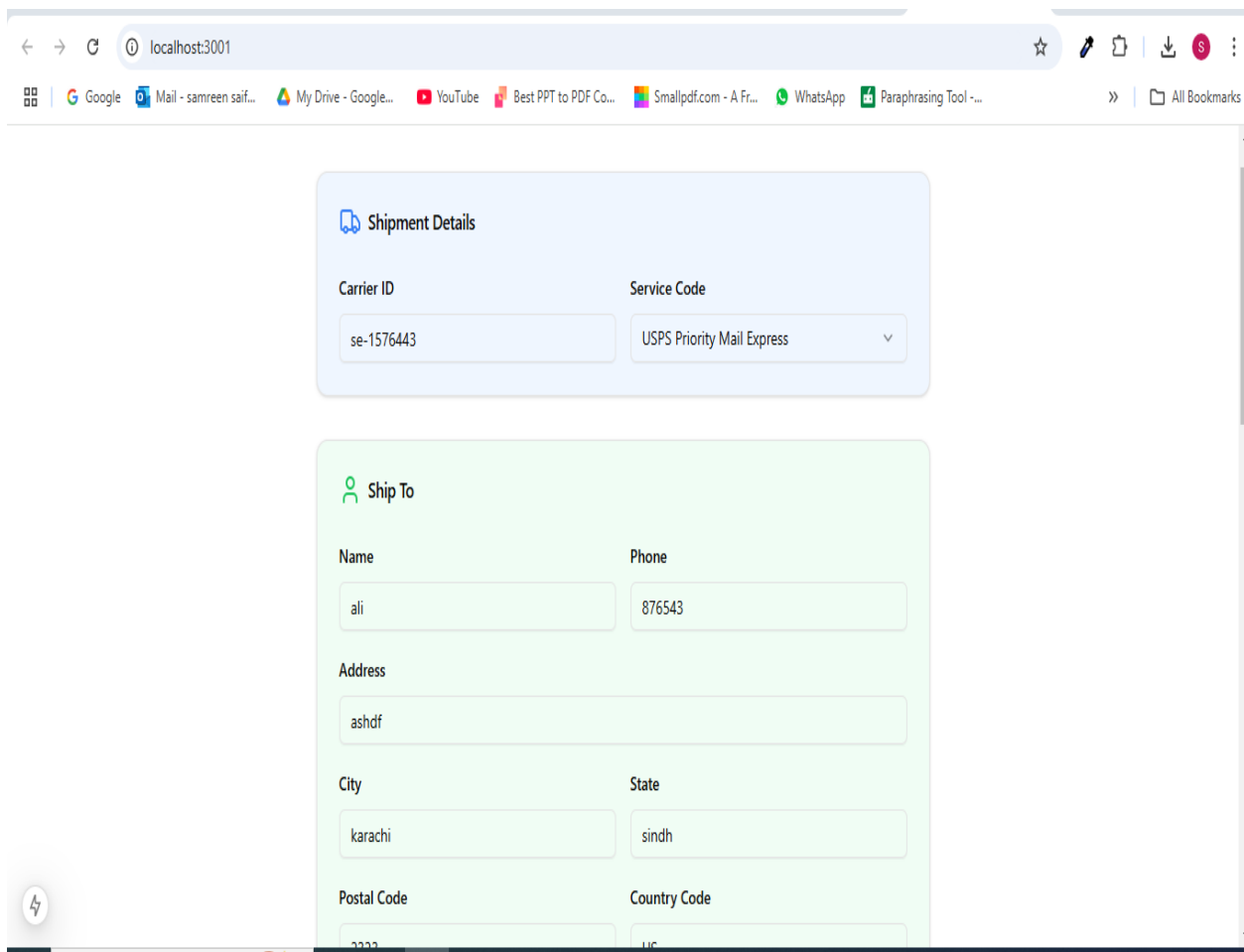
2. Sanity CMS

- Acts as the backend to store and manage product, customer, and order data.
- Provides APIs to fetch and update data dynamically.
- <https://fakestoreapi.com/products>



3. Third-Party APIs

- Services like payment gateways (e.g., Stripe) and shipping APIs that are integrated for order placement, payment processing, and delivery .
- <https://api.shipengine.com/v1/carriers>



The screenshot shows a web browser window with the address bar displaying 'localhost:3001'. The browser's bookmark bar includes links to Google, Mail, My Drive, YouTube, and several other services. The main content area features two distinct form sections. The first section, titled 'Shipment Details' with a truck icon, contains two input fields: 'Carrier ID' with the value 'se-1576443' and 'Service Code' with a dropdown menu currently showing 'USPS Priority Mail Express'. The second section, titled 'Ship To' with a person icon, contains several input fields: 'Name' (ali), 'Phone' (876543), 'Address' (ashdf), 'City' (karachi), 'State' (sindh), 'Postal Code' (3333), and 'Country Code' (US).

Shipment Details

Carrier ID: se-1576443

Service Code: USPS Priority Mail Express

Ship To

Name: ali

Phone: 876543

Address: ashdf

City: karachi

State: sindh

Postal Code: 3333

Country Code: US

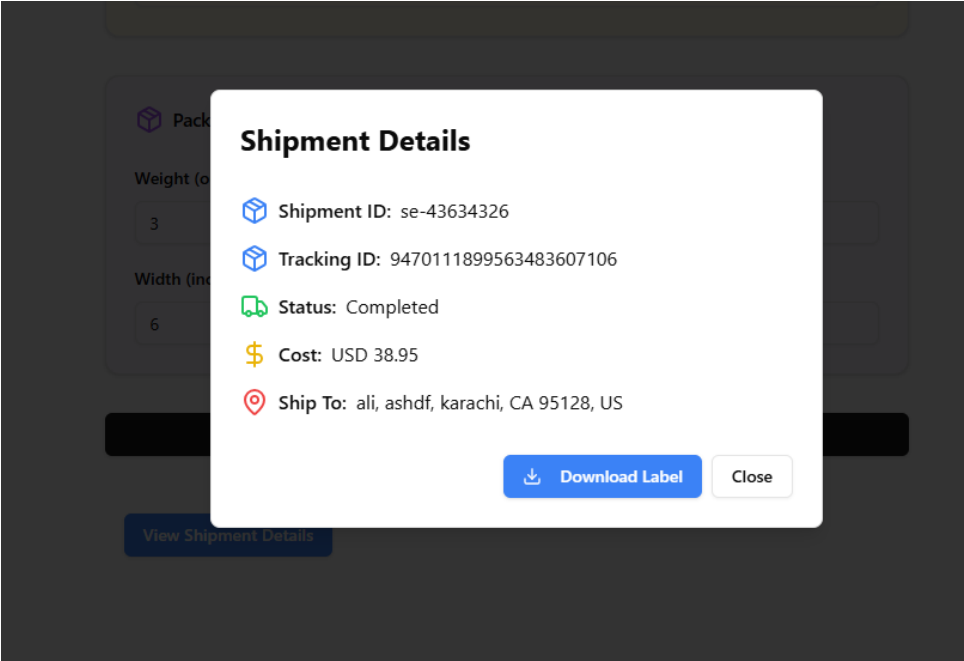
Ship From

Name	Company Name
<input type="text" value="Omar"/>	<input type="text" value="abc"/>
Phone	
<input type="text" value="2123"/>	
Address	
<input type="text" value="dheyg"/>	
City	State
<input type="text" value="lahore"/>	<input type="text" value="punjab"/>
Postal Code	Country Code
<input type="text" value="32323"/>	<input type="text" value="US"/>
Residential?	
<input type="text" value="No"/>	

Package Details

Weight (ounce)	Height (inch)
<input type="text" value="3"/>	<input type="text" value="4"/>
Width (inch)	Length (inch)
<input type="text" value="6"/>	<input type="text" value="7"/>

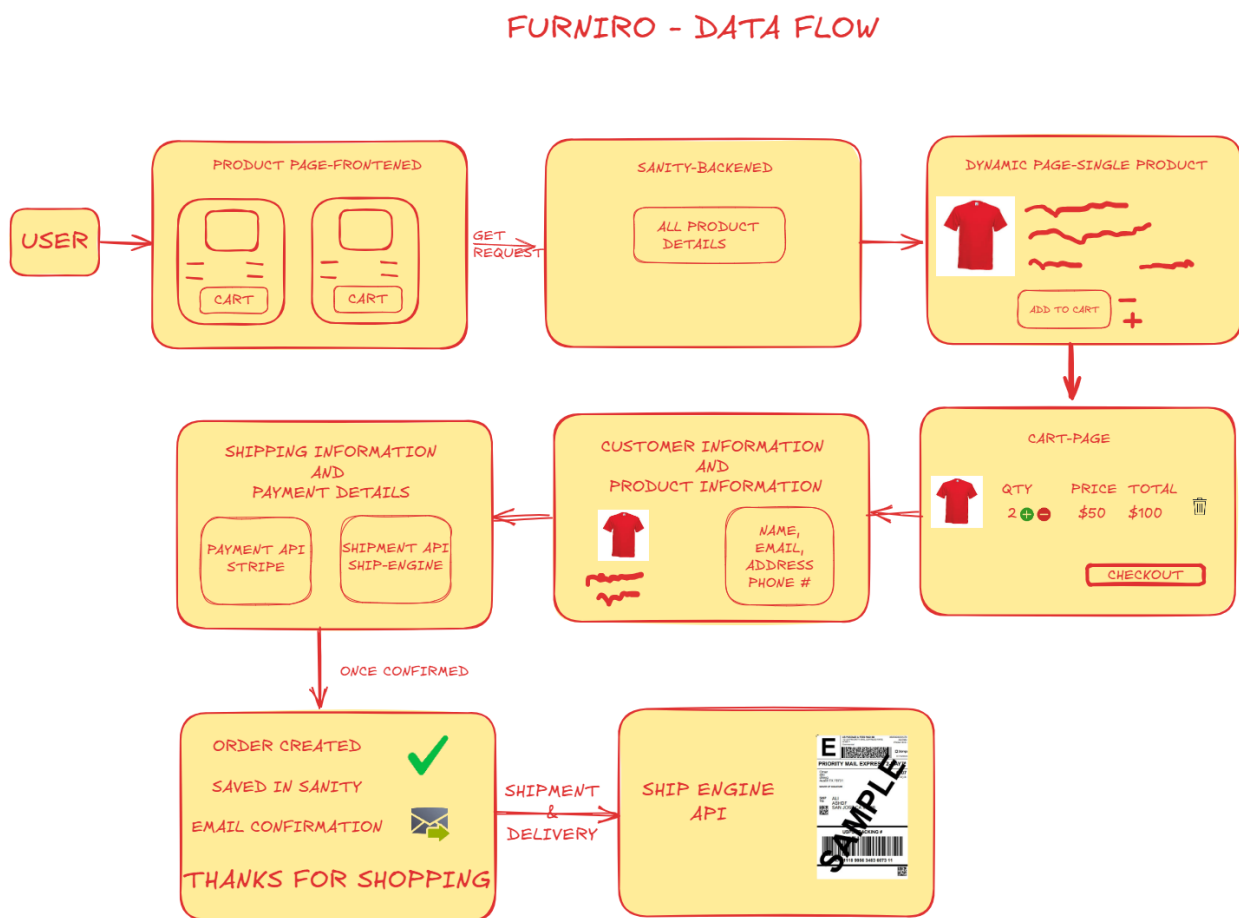
Submit Shipment



4. Payment Gateway

- Handles secure payment processing for customer transactions.
- Ensures data security with encrypted communication.

2. Key Workflows



Workflow 1: User Registration

1. User signs up via the frontend.
2. User data is sent to Sanity CMS via an API.
3. A confirmation email is sent to the user via a third-party API (e.g., SendGrid).

Workflow 2: Product Browsing

1. User visits the homepage.
2. Frontend sends a `GET` request to the Product Data API in Sanity CMS.
3. Sanity API responds with a list of products.
4. Frontend dynamically displays the products with their details.

Workflow 3: Order Placement

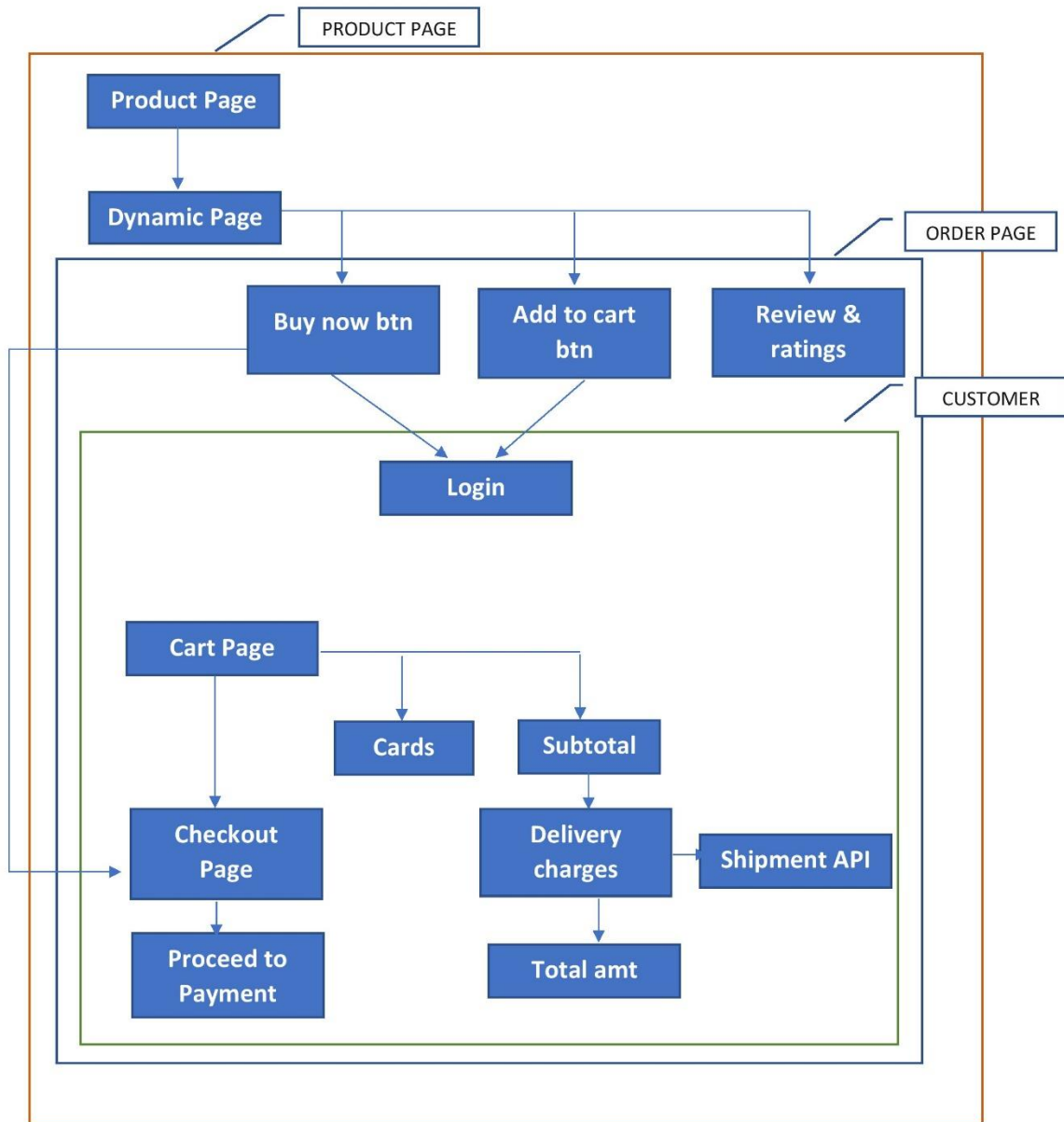
1. User adds products to the cart and proceeds to checkout.
2. Frontend sends a `POST` request to the Sanity CMS API to create a new order.
3. Payment details are processed via the Payment Gateway.
4. Sanity CMS stores the payment confirmation and order details.

Workflow 4: Shipment Tracking

1. After placing an order, the system sends a GET request to the shipment tracking API.
2. Shipment tracking details are fetched and displayed to the user.

General eCommerce Workflow:

Furniro-Workflow



- **Product Browsing:**

- **Step 1:** The user visits the homepage or product listing page.
- **Step 2:** The frontend sends a request to the **Sanity CMS** (e.g., `/products`) to fetch all products in the selected category or general catalog.
- **Step 3:** The products are displayed, and the user can click to view individual product details.
- **Step 4:** Upon selecting a product, detailed information is fetched and displayed (e.g., description, price, availability).

- **Cart Management:**

- **Step 1:** The user adds products to their cart by clicking an “Add to Cart” button.
- **Step 2:** The cart is updated with the product and quantity, and the total price is recalculated.
- **Step 3:** The user can review, edit (increase/decrease quantity), or remove products from the cart.

- **Order Placement:**

- **Step 1:** After reviewing the cart, the user proceeds to checkout.
- **Step 2:** Shipping details are entered, and the total cost (including shipping, taxes, etc.) is displayed.
- **Step 3:** Payment is processed via a third-party API (Stripe, PayPal, etc.).
- **Step 4:** The order is confirmed, and the user receives a confirmation email and/or order details in the UI.

3. Technical Roadmap

Technical Roadmap

week#1

1. Set up Next.js frontend
2. Configure Sanity CMS

week#2

1. Integrate Sanity CMS APIs with the frontend for dynamic product and order data.
2. User registration & login overflows

week#3

1. Add Payment Gateway
2. Integrate Shipment tracking

week#4

Finalize workflows, optimize the frontend, and test the complete system.

Milestones

1. Week 1:

- Set up Next.js frontend with a basic product listing page.
- Configure Sanity CMS for product and order management.

2. Week 2:

- Integrate Sanity CMS APIs with the frontend for dynamic product and order data.
- Implement user registration and login workflows.

3. Week 3:

- Add payment gateway integration and test secure transactions.
- Integrate shipment tracking with a third-party API.

4. Week 4:

- Finalize workflows, optimize the frontend, and test the complete system.