HACKATHONE DAY 2

DAY 2 PLANNING THE TECHNICAL FOUNDATION

Define Technical Requirements

We need to plan the technical requirements for our marketplace. These are divided into three parts:

1. **Frontend Requirements**
    - The frontend is what users see and interact with.
    - It should be **easy to use**, **work on mobile and desktop**, and have these pages:
        - **Home:** Show featured products.
        - **Product Listing:** List products with filters and sorting.
        - **Product Details:** Show product info like price, images, and reviews.
        - **Cart:** Let users see and edit their cart.
        - **Checkout:** Where users enter payment and shipping details.
        - **Order Confirmation:** Show order details and tracking info after purchase.
2. **Backend (Sanity CMS)**
    - Sanity CMS will manage all the data for our marketplace.
    - It will store:
        - **Product Data:** Name, price, description, images, stock, etc.
        - **Order Data:** Order ID, customer info, products, payment status, etc.
        - **Customer Data:** Name, email, address, phone number, order history, etc.
3. **Third-Party APIs**
    - We'll use APIs for extra features:
        - **Payment Gateway API:** To process payments (e.g., credit Card, Paypal).
        - **Shipment Tracking API:** To give real-time order tracking.
        - **Other APIs:** For SMS notifications, email alerts, or reviews (optional).
    - These APIs will help the frontend show things like product details, payment confirmations, and tracking info.

Example Workflow

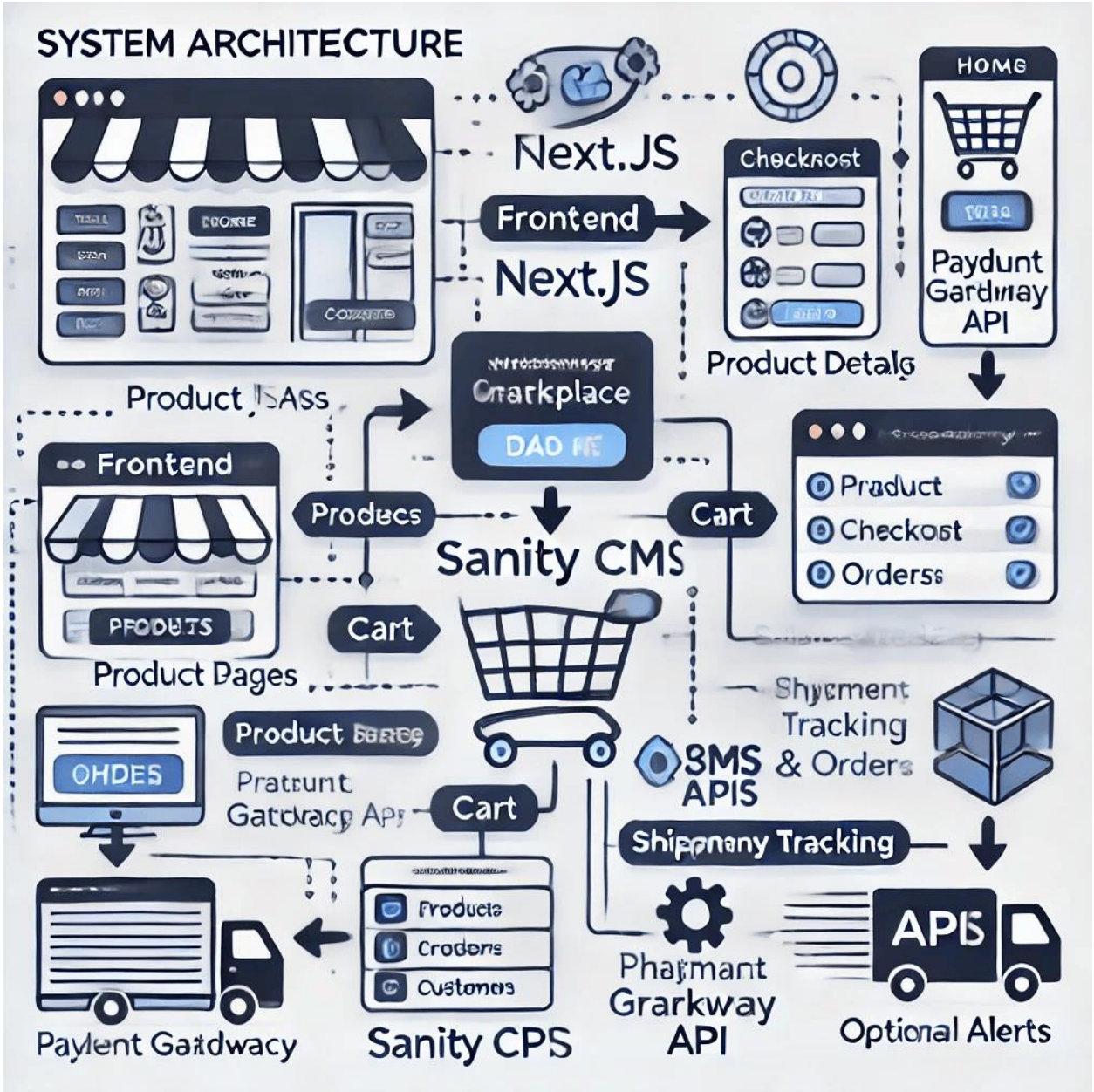
1. **User Browsing Products:**
    - User visits the website and browses products.
    - Frontend fetches product data from Sanity CMS and displays it.
2. **Add to Cart:**
    - User adds products to the cart.

- Cart data is saved locally or in Sanity.

3. **Checkout:**
   - User enters payment and shipping details.
   - Payment is processed via Payment Gateway API.
4. **Order Confirmation:**
   - After payment, order details are saved in Sanity CMS.
   - User sees order confirmation with tracking info.
5. **Shipment Tracking:**
   - User can track their order in real-time using the Shipment Tracking API.

Design System Architecture

The system architecture shows how all the parts of our marketplace work together. Here's what it includes:



1. **Frontend (Next.js):**
    - The user interface where customers browse products, add to cart, and checkout.
2. **Backend (Sanity CMS):**
    - Stores and manages product, customer, and order data.
    - Frontend fetches data from Sanity CMS.
3. **Third-Party APIs:**
    - **Payment Gateway API:** Processes payments.
    - **Shipment Tracking API:** Provides real-time tracking.
    - **Other APIs:** For SMS, email, or reviews (optional).

How Components Interact

1. **User Browsing Products:**
    - Frontend fetches product data from Sanity CMS and displays it.
2. **User Adds to Cart:**
    - Cart data is saved locally or in Sanity.
3. **User Checks Out:**
    - Frontend sends payment details to Payment Gateway API.
    - After payment, order details are saved in Sanity CMS.
4. **Order Confirmation and Tracking:**
    - Frontend fetches tracking info from Shipment Tracking API.
    - User sees order confirmation with tracking details.

5. Sanity Schema Example:

Field Name	Type	Title	Description
name	string	Product Name	The name of the product.
price	number	Price	The price of the product.
stock	number	Stock Level	The available stock level of the product.