DAY 2 PLANNING THE TECHNICAL FOUNDATION

Define Technical Requirements

- **Frontend Requirements:**
  - ***User-friendly interface for browsing products:***
    - Implement intuitive navigation.
    - High-quality product images and detailed descriptions.
    - Clear calls to action (e.g., "Add to Cart," "View Details").
  - ***Responsive design for mobile and desktop users:***
    - Ensure the website adapts seamlessly to different screen sizes and resolutions.
    - Optimize for mobile devices to enhance user experience on smartphones and tablets.
  - ***Essential pages:***
    - **Home Page:**
      - Showcase featured products, promotions, and user testimonials.
      - Provide easy access to key categories and search functionality.
    - **Product Listing Page:**
      - Display products in a visually appealing and organized manner.
      - Allow users to filter and sort products based on various criteria.
    - **Product Detail Page:**
      - Display high-resolution images, detailed descriptions, customer reviews, and seller information.
      - Include options for selecting variations (e.g., color, size).
    - **Cart Page:**
      - Allow users to review their selected items, modify quantities, and apply discounts.
    - **Checkout Page:**
      - Securely collect customer information (shipping address, billing address, payment details).

DAY 2 PLANNING THE TECHNICAL FOUNDATION

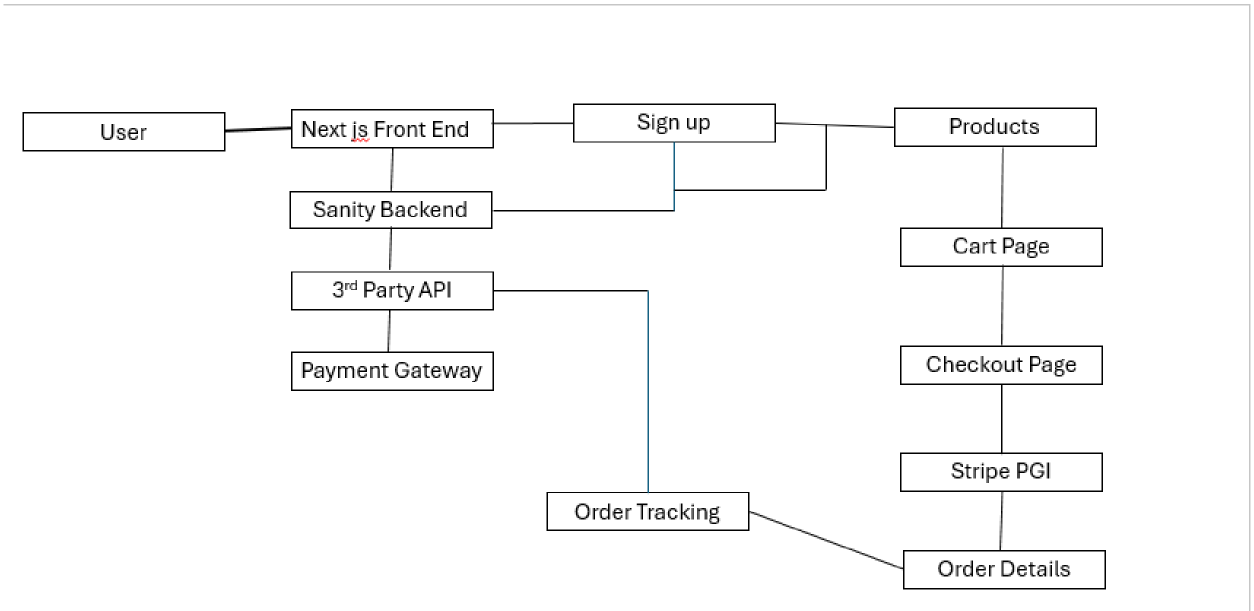
            - Offer various payment options (e.g., credit card, debit card, online wallets).
            - Display shipping costs and estimated delivery times.
        - **Order Confirmation Page:**
            - Display order summary, order number, and tracking information.
            - Provide options for order history and tracking.

- <u>**Sanity CMS as Backend:**</u>
    - ***Manage product data, customer details, and order records:***
            - Create Sanity schemas to structure and store all relevant data within the CMS.
    - ***Focus on designing schemas in Sanity to align with the business goals from Day 1:***
        - **Products:**
            - Product ID, Name, Description, Price, Images, Dimensions, Material, Color, Seller ID, Category ID, Stock Availability, Reviews.
        - **Customers:**
            - Customer ID, Name, Email Address, Phone Number, Shipping Address, Order History.
        - **Orders:**
            - Order ID, Customer ID, Product IDs, Order Date, Order Status, Shipping Address, Billing Address, Payment Details, Tracking Information.
        - **Delivery Zone:**
            - Category ID, Zone Name, Description, Assigned Drivers.
        - **Shipment:**
            - Shipment ID, Order ID, Seller ID, Company Name, Contact Information, Delivery Time.
    - ***Utilize Sanity's features:***
            - Leverage Sanity's content editing interface for easy content management.
            - Utilize Sanity's GraphQL API to fetch data for the frontend.

DAY 2 PLANNING THE TECHNICAL FOUNDATION

- **Third-Party APIs:**
  - ***Integrate APIs for shipment tracking:***
    - Integrate with shipping providers (e.g., FedEx, UPS, DHL) to provide real- time tracking information to customers.
  - ***Integrate APIs for payment gateways:***
    - Integrate with payment gateways (e.g., Stripe, PayPal, Razorpay) to facilitate secure online transactions.
  - ***Other required backend services:***
    - Consider APIs for:
      - **Email notifications:** Send order confirmations, shipping updates, and promotional emails.
      - **SMS notifications:** Send order updates and reminders.
      - **Image and video hosting:** Store and deliver high-quality product images and videos.
      - **Search and recommendation engines:** Enhance product discovery and improve user experience.

- **System Architecture:**



DAY 2 PLANNING THE TECHNICAL FOUNDATION

- ***Diagram:***
    - [Include a simple diagram here. Example using a tool like Lucidchart or draw.io:
        - **Frontend (Next.js)**
            - Connects to:
                - **Sanity CMS** (for product data, customer data, order data)
                - **Third-Party APIs:**
                    - **Shipment Tracking API (e.g., Shippo)**
                    - **Payment Gateway API (e.g., Stripe)**
                    - **Email/SMS API**
- ***Component Roles:***
    - **Frontend:** User interface for interaction.
    - **Sanity CMS:** Content management system for data storage and retrieval.
    - **Shipment Tracking API:** Handles shipping logistics and provides tracking updates.
    - **Payment Gateway API:** Processes online payments securely.
    - **Email/SMS API:** Facilitates communication with users (order confirmations, notifications).

<u>Key Workflows</u>

- ***User Registration:***
    - User enters details on the frontend.
    - Frontend sends data to Sanity CMS.
    - Sanity CMS creates a new customer record.
    - Welcome email sent to user via Email/SMS API.
- ***Product Browsing:***
    - User navigates through product categories on the frontend.
    - Frontend requests product data from Sanity CMS.
    - Sanity CMS provides product information.
    - Frontend displays products to the user.

DAY 2 PLANNING THE TECHNICAL FOUNDATION

- ***Order Placement:***
  - User selects products, adds them to the cart.
  - Frontend sends order details (products, customer, shipping, payment) to Sanity CMS.
  - Sanity CMS creates an order record.
  - Order confirmation is sent to the user via Email/SMS API.
- ***Order Fulfillment:***
  - Order information (shipping address) is sent from Sanity CMS to the Shipment Tracking API.
  - Shipment API generates a shipping label and tracking number.
  - Tracking number is updated in the order record within Sanity CMS.
- ***Shipment Tracking:***
  - Frontend requests tracking information for an order from Sanity CMS.
  - Frontend sends the tracking number to the Shipment Tracking API.
  - Shipment API provides real-time tracking updates.
  - Frontend displays tracking updates to the user.

Category-Specific Instructions

- ***General eCommerce:***
  - **Workflows:** Product browsing, cart management, order placement (as described above).
  - **API Endpoint:**
    - **/products** (GET): Fetch all product details (ID, name, price, image, description, etc.)