

Hackathon Day 2

DAY 2 PLANNING THE TECHNICAL FOUNDATION

Day 2: Activities

1. Define Technical Requirements

Technical requirements are break into 3 parts: Frontend, Sanity CMS as backend and Third-party APIs. Here are Structured explanation.

1: Frontend Requirements:(Nextjs)

- **Goal:**

Make a user-friendly, attractive user interface where user can interact with marketplace.

- **Responsive:**

Make Interface fully responsive that help user to interact with marketplace. For make marketplace responsive use tailwind classes.

- **Essential Pages:**

Develop a Following pages:

DAY 2 PLANNING THE TECHNICAL FOUNDATION

- **Homepage:**

Show all featured product, Show more button, Room inspiration, our proper setup images etc.

- **Shop page:**

Show all the product, new arrivals, filter icon, search bar, Add to cart button, comparison page link, our features etc.

- **Blog page:**

Categories option, different blogs, our recent blogs etc.

- **Contact page:**

Contact information, our email, address, contact number.

- **Add to cart page:**

Cart product display with quantity, checkout option with total amount of Purchase.

- **Checkout page:**

collect user details for shipment, payment.

- **Order Confirmation page:**

Display that order is confirmation message if order is rejected then also display a rejected message.

2: Sanity CMS as Backend:

- **Goals:**

Use sanity to manage or add data dynamically for the marketplace.

- **Requirement:**

DAY 2 PLANNING THE TECHNICAL FOUNDATION

1. **Product Data management:**

Store all product information (such as; name, Id, price, description, stock, image etc.)

2. **Customer Data management:**

Store all Customer Data (such as; Customer name, Id, email, address)

3. **Order record:**

keep all track about order (such as, tracking Id, Delivery status, payment status, shipment status, product Id, order details)

- **Schema Design in Sanity:**

Design schemas that align with business goals.

Product Schema: Fields for name, price, description, images, and stock.

Customer Schema: Fields for name, email, address, and contact details.

Order Schema: Fields for customer reference, product list, payment status, and shipment status.

3: **Third-party API:**

- **Goal: use** External API for advance functionality.
- **Requirement:**
 1. Product API (mock Api, Free Api, sanity CMS)
 2. Shipment tracking API (Shippo, Easy port, DHL, FedEx)
 3. Payment Gateway API (PayPal, stripe, Square)
 4. Authentication Api (NextAuth.js, Auth0)

5. Search and recommendation Api (Elasticsearch)
6. Geolocation and Address Api (Google map)

2. Design System Architecture

1. Key components:

- **Frontend (Next.js):** The user interface where customers interact.
- **Sanity CMS:** Manages product data, orders, and content.
- **Product Data API:** Powers the frontend with dynamic product details.
- **Third-Party APIs:** Used for shipment tracking or other services.
- **Payment Gateway:** Processes payments securely.

2. Key Workflow:

1. General E-Commerce Workflows

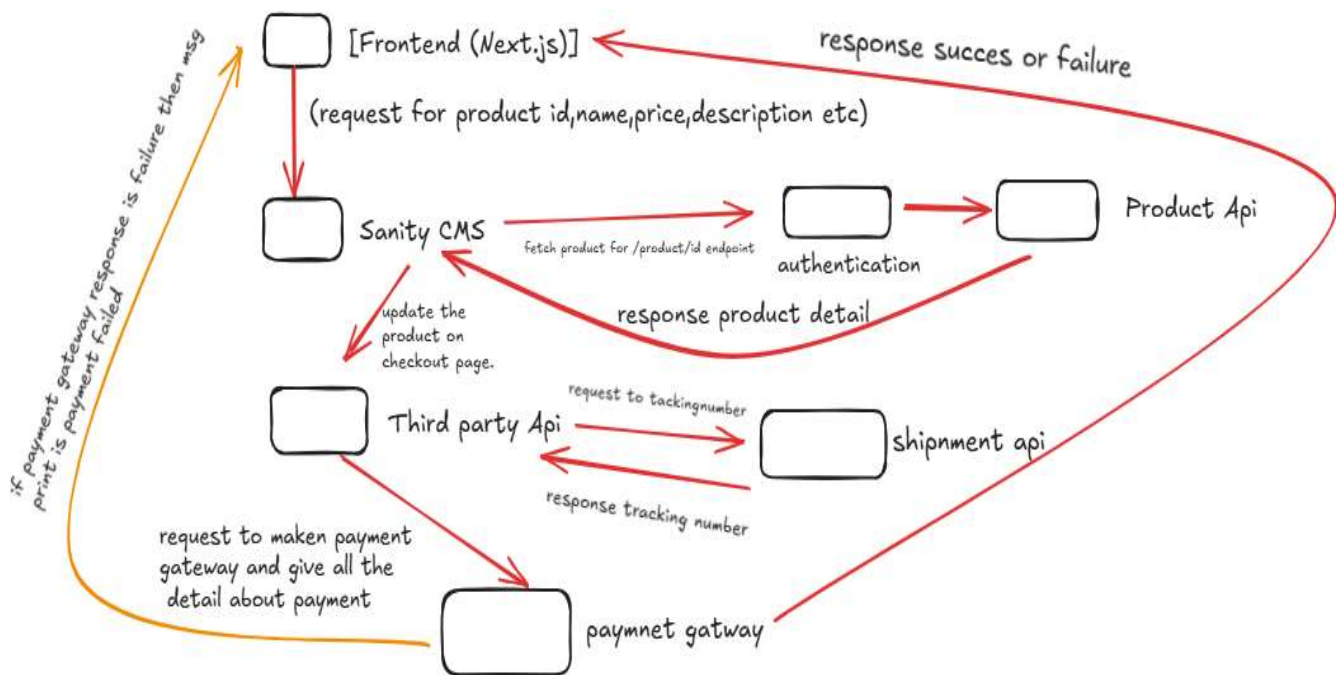
- **Product Browsing:**
 - User visits the site.
 - Frontend sends a request to **/products** endpoint.
 - Sanity CMS fetches data and sends it back to the frontend.
 - Products are displayed dynamically.
- **Cart Management:**
 - User adds products to the cart.
 - Cart data is stored temporarily on the frontend.
 - Checkout triggers a call to save the cart to Sanity CMS.

DAY 2 PLANNING THE TECHNICAL FOUNDATION

- **Order Placement:**

- User proceeds to checkout.
- Frontend sends order details to Sanity CMS.
- Payment Gateway processes the payment securely.
- Order confirmation is stored in Sanity CMS.

3. System Architecture Visualized:



System Architecture Visualized

Conclusion

Day 2 focused on establishing the technical foundation for the marketplace project. The frontend, developed using Next.js and Tailwind CSS, will ensure a user-friendly, responsive interface with essential pages like Home, Shop, Blog, Contact, Cart, and Checkout. Sanity CMS will serve as the backend to manage dynamic product data, customer records, and orders through well-structured schemas. Integration of third-party APIs, such as payment gateways, shipment

DAY 2 PLANNING THE TECHNICAL FOUNDATION

tracking, and authentication, will enable advanced functionalities. The system architecture emphasizes seamless workflows, including product browsing, cart management, and secure order placement, ensuring an efficient and scalable e-commerce platform.

Prepared by: Alishba Naseem

Slot: Tuesday 2 to 5

Sir: Sir Ali Aftab & Sir Fahad Sheikh