

Marketplace Technical Foundation - Day 2

Day 2 Goal

Transition from business-oriented planning to technical preparation for building the marketplace. The focus is on creating a high-level technical plan tailored to a **General E-Commerce platform for lightweight daily-use women's jewelry**, including:

- System architecture.
- API requirements.
- Technical workflows.
- Integration with Sanity CMS and third-party APIs.

Recap of Day 1: Business Focus

Achievements:

1. Business Goals Defined:

- Problem-solving objectives.
- Target audience: Women seeking lightweight, elegant, and affordable jewelry for daily use.
- Unique value proposition (UVP): Affordable prices, high-quality products, and fast delivery.

2. Data Schema Drafted:

- Identified entities: Products, Orders, Customers, and their relationships.

3. Focused Planning:

- Established a strong foundation for technical implementation.

Day 2 Activities: Transitioning to Technical Planning

1. Define Technical Requirements

- **Frontend Requirements:**
 - 1) User-friendly and responsive design for mobile and desktop.
 - 2) Essential pages: Home, Product Listing, Product Details, Cart, Checkout, and Order Confirmation.
- **Backend (Sanity CMS):**
 - 1) Use Sanity CMS to manage product data, customer details, and orders.
 - 2) Design schemas aligned with jewelry business needs.
- **Third-Party APIs:**
 - 1) Integrate APIs for payment gateways and shipment tracking.

2. Design System Architecture

- Create a diagram showing interactions between components:
 - 2) **Frontend (Next.js):** User interface.
 - 3) **Sanity CMS:** Manages products, orders, and customer details.
 - 4) **Third-Party APIs:** Payment gateway and shipment tracking.

Example Workflow:

1. User browses jewelry products → Sanity CMS fetches product data → Displayed on frontend.
2. User places an order → Order details saved in Sanity CMS.
3. Shipment updates fetched via third-party API → Displayed to the user.
4. Payment processed securely → Status updated in the system.

3. Plan API Requirements

- Define endpoints based on workflows:

General E-Commerce Example:

- **/products (GET):** Fetch all product details.
 - Response: { "id": 1, "name": "Earring A", "price": 500, "stock": 20 }
- **/orders (POST):** Create a new order.
 - Payload: Customer info, product details, and payment status.
- **/shipment (GET):** Track order status.
 - Response: { "shipmentId": 123, "status": "In Transit", "ETA": "2 days" }

4. Write Technical Documentation

- Structure documentation into:
 1. **System Architecture Overview:**
 - Diagram with component interactions.
 2. **API Specification:**
 - Endpoints, methods, payloads, and responses.
 3. **Workflow Diagrams:**
 - Visualize user interactions and data flow.
 4. **Sanity Schemas:**
 - Example:

```
export default {  
  name: 'product',  
  type: 'document',  
  fields: [  
    { name: 'name', type: 'string', title: 'Product Name' },  
    { name: 'price', type: 'number', title: 'Price' },  
    { name: 'stock', type: 'number', title: 'Stock Level' }  
  ]  
};
```

5. Collaborate and Refine

- **Peer Review:**
 - Share plans for feedback from teammates and mentors.
- **Version Control:**
 - Use GitHub for tracking changes and collaborating on diagrams or drafts.

Key Outcomes of Day 2

1. **Aligned Technical Plan:**
 - A comprehensive plan reflecting business goals for women's jewelry.
2. **System Architecture Diagram:**
 - Clear illustration of frontend, backend, and API interactions.
3. **Detailed API Requirements:**
 - Endpoints, methods, and response examples tailored for jewelry business workflows.
4. **Sanity Schemas Drafted:**
 - Key data entities designed for scalability.
5. **Refined Documentation:**
 - Professional, portfolio-ready submission.

Industry Best Practices

1. **Plan Before Coding:**
 - Avoid rework by creating a clear roadmap.
2. **Use the Right Tools:**
 - Leverage Sanity CMS and APIs for backend efficiency.
3. **Collaboration:**
 - Seek feedback to enhance quality.
4. **Focus on User Experience:**
 - Ensure a seamless and intuitive user journey.

Submission Guidelines

1. **Title:** Marketplace Technical Foundation – General E-Commerce.
2. **Repository Submission:**
 - Folder: "Documentation" containing diagrams, schemas, and technical plans.
3. **File Naming:**
 - Example: SystemArchitecture_Day2.pdf, APIEndpoints.xlsx.
4. **Quality Check:**
 - Review diagrams, schemas, and written content for accuracy and clarity.