Marketplace Technical Foundation - Furniture Marketplace

1. System Architecture Overview

A high-level diagram showing interactions between the frontend (Next.js), Sanity CMS, and third-party APIs (e.g., shipment tracking) will be included.

Architecture Components:

- Frontend (Headless UI): Built with Next.js, using Tailwind CSS, DaisyUI, and Font Awesome for styling.

- Sanity CMS: Manages product data, orders, and customer details.

- Third-Party APIs: Handles shipping and real-time tracking integration.

2. Key Workflows

User Registration:

1. User visits the registration page.

2. Fills in details (name, email, password, etc.).

3. Data is sent to authentication service (if applicable).

4. Upon successful registration, user can log in.

Product Browsing & Viewing:

1. Fetches product data dynamically from Sanity CMS.

2. Displays product details using Tailwind CSS and DaisyUI.

3. Users can filter and search products.

Order Placement:

1. Users select products and add them to the cart.

2. Proceed to checkout by providing shipping details.

3. Order details (product IDs, quantities, and user info) are sent to Sanity CMS.

4. API triggers order confirmation.

3. API Requirements

Endpoint	Method	Payload	Response
/products	GET	-	List of products
/product/{id}	GET	Product ID	Product details
/order	POST	User info, product IDs, quantities	Order confirmation
/shipment/{orderId}	GET	Order ID	Tracking details

4. Sanity Schema Design

Product Schema:

- title: string

- description: text

- price: number

- category: string

- image: URL

- stock: number

Order Schema:

- user: reference

- products: array

- totalPrice: number

- status: string

5. Error Handling in Sanity Queries

- Implement try...catch for API requests.

- Show user-friendly error messages in case of failures.

6. Sanity Product Upload Script

- Develop a Node.js script to bulk upload product data to Sanity CMS.

7. Collaboration Notes

- Worked with peers to finalize schema and UI components.

- Faced challenges in API integration, resolved by debugging request payloads.

- Incorporated feedback to improve UX/UI elements.

8. File Format & Submission

- The document will be submitted in PDF format.

- High-resolution diagrams will be embedded.

- Uploaded to the Documentation folder in the repository with proper naming conventions.