HACKATHON DAY 2 :
MarketPlace Technical Foundation - Furniture Website

System Architecture Overview:

Overview

This document outlines the architecture and interaction between components of the **Furniture E-commerce Marketplace**, ensuring scalability, performance, and maintainability.

1. Frontend (Client-Side):

Technology: Next.js (React framework)

- **Features**:
    - Server-Side Rendering (SSR) for SEO
    - Responsive UI with Tailwind CSS
    - Dynamic routing for product pages
    - Image optimization with Next.js

Pages & Their Features

Home Page

- Hero section with featured furniture & promotions
- Category-based navigation (Living Room, Bedroom, Office, etc.)

Product Listing Page

- Dynamic grid layout , Quick view option
- Filters & sorting (price, material, rating)

Product Details Page

- High-resolution images with zoom & 360° view , Customer reviews & Q&A
- Description, dimensions, materials, and stock availability

Cart Page

- Displays selected items with live price updates
- Quantity adjustments & item removal , Discount code and estimated shipping

Checkout Page

- Secure payment options with real-time validation
- Guest checkout & login option , Order summary & final pricing

2. Backend (Headless CMS & APIs)

Technology: Sanity CMS (Headless CMS)

- **Features**:
  - Product data storage
  - Orders & customer data management
  - API endpoints for frontend integration

3. Third-Party API's Purpose:

1. **Shipment Tracking API:**
   - Provides real-time tracking information for orders.
   - Allows customers to view the status of their orders directly on the website.
2. **Payment Gateway API:**
   - Processes transactions securely by integrating payment providers like PayPal, Stripe, or credit/debit card services.
   - Ensures smooth and safe online payments for customers.
3. **Mock API:**
   - **JSONPlaceholder** or **MockAPI** for simulating backend data during development and testing, useful for working on the frontend without a real backend.

System Workflow

User Registration

- User signs up on the frontend.
- Data is stored in Sanity CMS.
- Confirmation is sent via email API.

Product Browsing

- Next.js frontend fetches product categories from **Sanity CMS**.
- Products are displayed dynamically.
- Filters and sorting options enhance the browsing experience.

Order Placement

- User adds items to the cart.
- Proceeds to checkout and submits order.
- **Order details are saved in Sanity CMS.**

Payment Processing

- User selects a payment method (Stripe, PayPal, etc.).

- Secure API request is sent to **Payment Gateway**.
- Payment is processed, and confirmation is recorded in **Sanity CMS**.

API Endpoints

1. Product Management API

- **Endpoint Name**: /products
- **Method:** GET
- **Description**: Fetch all products.
- **Response:** [
- {"id": "123", "name": "Wooden Table", "price": 299, "stock": 15, "image": "image_url"},
- {"id": "124", "name": "Leather Sofa", "price": 499, "stock": 20, "image": "image_url"}
-]

2. Order Management API

- **Endpoint Name**: /orders
- **Method**: POST
- **Description**: Create a new order.
- **Response**:
- {"orderId": "ORD123456", "status": "created", "delivery": "2025-02-10"}

3. Shipment Tracking API

- **Endpoint Name**: /shipment
- **Method**: GET
- **Description**: Track order status.
- **Response:** {"shipmentId": "SHIP12345", "status": "shipped", "deliveryDate": "2025-02-10"}

4. Cart API

- **Endpoint Name**: /cart
- **Method**: POST
- **Description**: Add items to the cart.
- **Response:** {"cartId": "CART567", "status": "Item added"}

5. Review API

- **Endpoint Name**: /reviews
- **Method:** POST
- **Description:** Submit a product review.

- **Response:** {"reviewId": "REV456", "status": "Review submitted"}

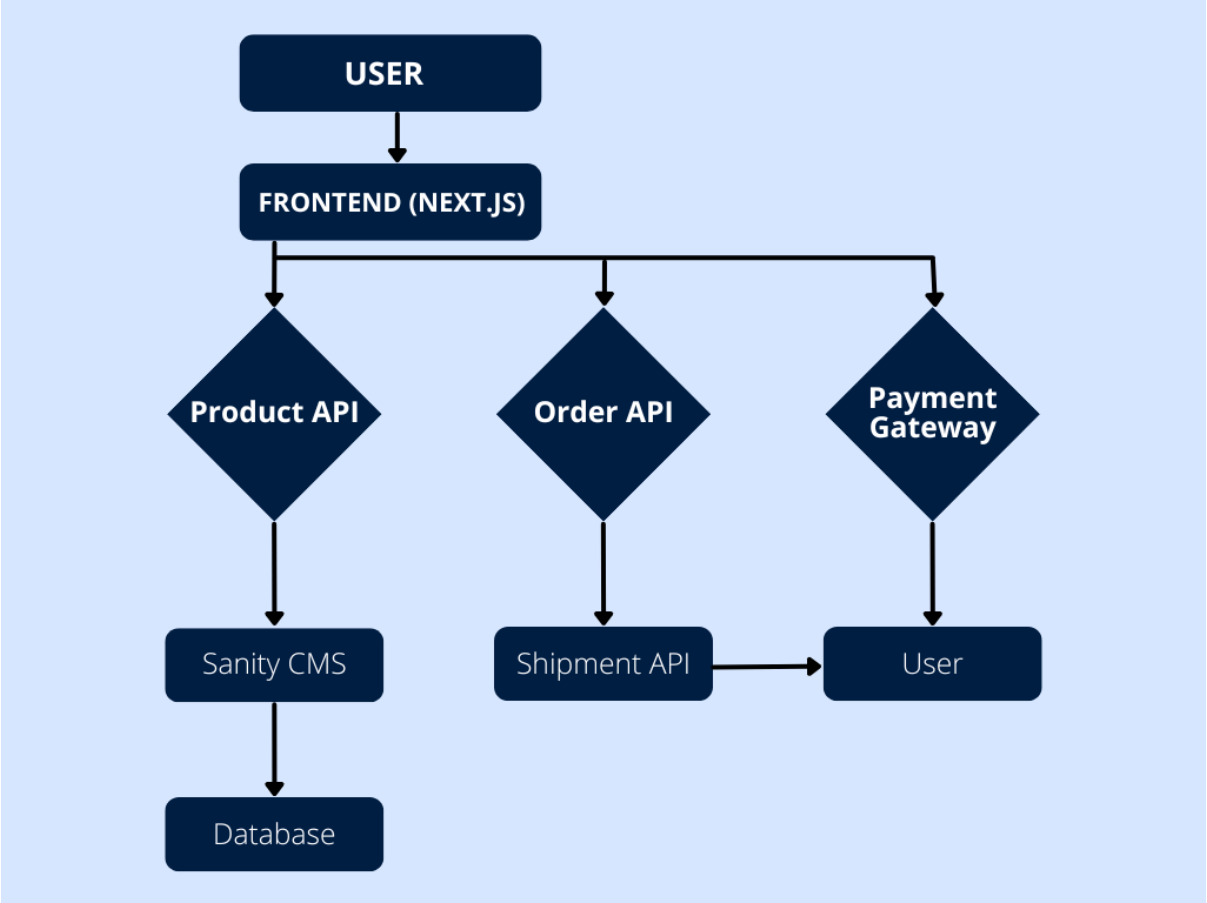
Sanity Schema Example

```
export default {
  name: 'product',
  title: 'Product',
  type: 'document',
  fields: [
    {
      name: 'name',
      title: 'Product Name',
      type: 'string',
      validation: Rule => Rule.required().min(3).max(50),
    },
    {
      name: 'slug',
      title: 'Slug',
      type: 'slug',
      options: {
        source: 'name',
        maxLength: 96,
      },
      validation: Rule => Rule.required(),
    },
    {
      name: 'description',
      title: 'Description',
      type: 'text',
      validation: Rule => Rule.required().max(300),
    },
    {
      name: 'price',
      title: 'Price',
      type: 'number',
      validation: Rule => Rule.required().positive(),
    },
    {
```

```
  name: 'category',
  title: 'Category',
  type: 'string',
  options: {
    list: [
      { title: 'Sofas', value: 'sofas' },
      { title: 'Tables', value: 'tables' },
      { title: 'Chairs', value: 'chairs' },
      { title: 'Beds', value: 'beds' },
    ],
  },
},
{
  name: 'stock',
  title: 'Stock Quantity',
  type: 'number',
  validation: Rule => Rule.min(0),
},
{
  name: 'images',
  title: 'Product Images',
  type: 'array',
  of: [{ type: 'image' }],
  options: { hotspot: true },
},
{
  name: 'dimensions',
  title: 'Dimensions (L x W x H)',
  type: 'string',
},
{
  name: 'materials',
  title: 'Materials',
  type: 'string',
},
{
  name: 'ratings',
```

```
      title: 'Customer Ratings',
      type: 'number',
      validation: Rule => Rule.min(0).max(5),
    },
  ],
};
```

Design System Architecture



Prepared by: Alishba Rafiq

Slot: Saturday 9-12 am

Teacher: Bilal Muhammad Khan & Aneeq Khatri