

Day 2 Activities: Transitioning to Technical Planning for Furniture Marketplace

Introduction

The "Furniture Marketplace" is an eCommerce platform aimed at providing users with a seamless experience for browsing and purchasing furniture. This technical document serves as a blueprint for the project, covering system architecture, workflows, API requirements, and database schemas. The design ensures scalability, maintainability, and alignment with business goals.

1. Define Technical Requirements

The first step is to translate business goals into clear technical requirements. For each feature identified on Day 1, outline the following:

Frontend Requirements:

- **User-Friendly Interface:**
 - A visually appealing layout for browsing furniture by categories like living room, bedroom, office, etc.
 - Easy-to-navigate menus, search functionality, and filter options for material, size, price, and color.
- **Responsive Design:**
 - Ensure compatibility across mobile, tablet, and desktop devices.
- **Essential Pages:**
 - **Home Page:** Featured furniture collections and promotional banners.
 - **Product Listing:** Display categories with thumbnails and summaries.
 - **Product Details:** Detailed view with high-resolution images, descriptions, dimensions, material, and reviews.
 - **Cart:** Users can view selected items, adjust quantities, and proceed to checkout.
 - **Checkout:** Secure form for billing, shipping, and payment information.
 - **Order Confirmation:** Summarizes purchase details and estimated delivery.

Backend Requirements:

- **Sanity CMS:**

- ## 2. Design System Architecture



- **Frontend:** Handles user interaction and displays dynamic content from APIs.
- **Sanity CMS:** Stores and manages furniture catalog data, customer details, orders, and reviews.
- **Third-Party APIs:**
 - Shipment tracking API for delivery updates.
 - Payment gateway API for processing transactions.

1. Product Browsing:

- a. User visits the marketplace → Frontend requests product data from Sanity → Products displayed dynamically.
2. **Order Placement:**

- a. User adds items to the cart → Completes checkout → Order details saved in Sanity CMS.
- 3. **Shipment Tracking:**
 - a. Order status fetched via third-party shipment API → Displayed to the user in real-time.
- 4. **Payment Processing:**
 - a. Payment details securely handled by a payment gateway → Confirmation sent to both the user and Sanity CMS.

Key Steps with Flowchart

1. User Registration:

[User] -> [Frontend] -> [Sanity CMS] -> Confirmation
Email Sent

2. Product Browsing:

[User] -> [Frontend] -> [Sanity CMS] -> Product Data
Rendered

3. Order Placement:

[User] -> [Frontend] -> [Sanity CMS] -> [Payment Gateway]
-> Order Confirmed

4. Shipment Tracking:

[User] -> [Frontend] -> [Third-Party API] -> Shipment
Status Displayed

3. Plan API Requirements

Based on your furniture marketplace workflows, define essential API endpoints and their specifications.

Example API Requirements:

- **Endpoint:** /products
 - **Method:** GET
 - **Description:** Fetch all available furniture items.

- **Response Example:** {


```

        "id": 101,
        "name": "Modern Sofa",
        "price": 500,
        "stock": 20,
        "dimensions": "80x35x40 inches",
        "material": "Leather",
        "image": "url-to-image"
      
```

}
- **Endpoint:** /orders
 - **Method:** POST
 - **Description:** Save a new order in Sanity.
 - **Payload:** {


```

            "customerName": "John Doe",
            "address": "123 Main St",
            "products": [
              { "id": 101, "quantity": 2 },
              { "id": 202, "quantity": 1 }
            ],
            "paymentStatus": "Paid"
          
```

}
 - **Response Example:** { "orderId": 12345, "status": "Confirmed" }
- **Endpoint:** /shipment-status
 - **Method:** GET
 - **Description:** Fetch real-time delivery updates.
 - **Response Example:** {


```

            "orderId": 12345,
            "status": "In Transit",
            "ETA": "2 days"
          
```

}

4. Write Technical Documentation

Document the system architecture, workflows, and API requirements for professional reference.

Components of the Documentation:

- 1. **System Architecture Overview:**
 - a. Include a diagram illustrating the interaction between the frontend, Sanity CMS, and third-party APIs.
- 2. **Key Workflows:**
 - a. **User Registration:**
 - i. User signs up → Data stored in Sanity → Confirmation email sent.
 - b. **Furniture Browsing:**
 - i. Categories fetched from Sanity → Products dynamically displayed → Filters applied.
 - c. **Order Management:**
 - i. Items added to the cart → Order placed → Payment processed → Shipment tracked.
- 3. **API Endpoints Table:**

Endpoint	Method	Purpose	Response
/products	GET	Fetch all product details	JSON object with products
/orders	POST	Create a new order	Order confirmation
/shipment-status	GET	Track shipment status	Real-time delivery updates

4. Sanity Schema Example:

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

};

5. **Technical Roadmap:**

a. **Milestones:**

- i. Week 1: Finalize design and backend schemas.
- ii. Week 2: Implement product browsing and order workflows.
- iii. Week 3: Integrate payment and shipment APIs.
- iv. Week 4: Testing and deployment.