**Hackathon 3**

**DAY 2**

**E-commerce Skills Strategy for Furniture**

# Specifications for technique

**Frontend :**

An optimised user experience comes next. For styling with Tailwind CSS and Ul Components Library, such as State Management, EventHandlers, and ShadCn Ul.

**Pages :**

Homepage:

Emphasise collections, special bargains, and furniture (such as tables and sofas).

Furniture Categories:

A page containing categories such as office, bedroom, living room, etc.

Product Information Page:

View in detail including material, measurements, and personalisation choices (e.g

., colour, size)

Cart & Checkout:

An easier way to add products and finish transactions

Customisation Page: Interactive features that let customers see furniture arranged in

various ways

**The Sanity CMS backend:**

Keep track of product information such as categories, sizes, and materials , Control customisation choices and inventories, Manage client information and order processing.

**APIs from third parties:**

Payment Gateway: Transactions using Stripe

Tracking : Shippo is used for shipment tracking and delivery updates.

**Description of the System Architecture Plan**

Frontend:

Allows for customisation and displays the furniture.

Sanity CMS:

Holds dynamic data, including orders, customisations, and product inventory.

Third-Party APIS:

Combines shipping, payments, and other services.

**Diagram :**

[Next.js Frontend]

[CMS Sanity]

1

[API for Product Data]

1

[APIs from Third Parties]

Shipment Monitoring (Shippo )

Payment Gateway (Stripe)

**API Conditions**

Purchase furniture

Endpoint: /furniture

Method: GET

Target: Fetch every piece of furniture.

For Example:

[

{"id": 1, "name": "chair", "price": 500, "stock": 10, "material": "foam"),

"id": 2; "name": "Table"; "price": 300; "stock": 5; "material": "Wood"}

]

**Furniture Details**

Endpoint:/furniture/:id

Method: GET

Purpose: Fetch details of a specific item (dimensions, materials, customizations).

Create Order

Endpoint:/orders

Method: POST

Purpose: Save customer orders.

**Draft Furniture Schema Sanity Schema export default**

name: furniture,

type: "document,

title: 'Furniture',

fields:

[

(name: 'name', type:'string', title: "Furniture Name'),

(name: 'price', type: 'number', title: 'Price'),

(name: 'description', type: 'text', title: 'Description'),

(name:'stock, type: 'number', title: 'Stock'),

(name: "material", type:'string', title: 'Material),

(name: 'dimensions', type:'string', title: "Dimensions'),

(name: "customisations', type: 'array', of: [{ type:'string')]title: 'Available Customisations'),

(name: "Image', type: 'image', title: 'Furniture Image')

],

**Export Order Schema by default**

type: "document",

name: "order,"

title: Order,

fields:

[

(name: "customerName', type:'string', title: 'Customer Name'),

(name: 'email', type:'string', title: 'Customer Email'),

(name: "furnitureltems', type: 'array', of: [{ type:'reference', to: [{ type: "furniture }}}}, title: 'Furniture Items'),

(name: "Total, type: 'number', title: 'Total Price'),

(name:'status', type:'string', title: 'Order Status')

]

**Architecture of the System**

Simply state the interactions between each part. For instance: "Next js retrieves furniture data from Sanity CMS using the/furniture API and renders it dynamically."

**documentation for APIs**

Include endpoints together with their intended use and anticipated input and output.

furniture: Receives every piece of furniture.

orders: Stores information about orders in the backend.

**Diagrams of The workflow**

***a customer explores furniture > customises options > adds to cart > check out.***

**Schemas**

Furniture, orders, and customers are all included in sanity schemas.

1. A technical strategy customised for your furniture marketplace before today's end.

2. A system diagram showing the relationships between components.

3. Comprehensive documentation for the API.

4. Sanity schema drafts for ordering and inventory management of furniture.