Marketplace Builder Hackathon 2025

Objective

Comforty will focus on selling comfortable stools, chairs, and sofas. The design and features will emphasize comfort and quality, making it easy for users to find and purchase the perfect furniture pieces for their homes or offices.

My Furniture Website:-

I'm planning to build a website focused on selling comfortable stools, chairs, and sofas. Here's what it will include,

- ## Frontend Requirements:

1. **User-Friendly Interface**:
   The website will have an intuitive layout that makes it easy for customers to browse and find the perfect furniture, with clear categories like **Stools**, **Chairs**, and **Sofas**.

2. **Responsive Design**:
   It will feature a **mobile-first design**, ensuring the site looks and works great on both **mobile** and **desktop** devices, with flexible grids and responsive images.

3. **Essential Pages**:
   - ✓ **Home Page**: A welcoming page showcasing key categories and featured products.
   - ✓ **Product Listing Page**: Displays products in a clean,

organized way with filters for easy browsing.

- ✓ **Product Details Page**: Offers in-depth information about each item, including comfort features, dimensions, and high-quality images.

- ✓ **Cart Page**: Lets users view, modify, or remove items before proceeding to checkout.

- ✓ **Checkout Page**: A simple, secure process to review the order, select payment, and finalize the purchase.

- ✓ **Order Confirmation Page**: Displays a summary of the order, delivery details.

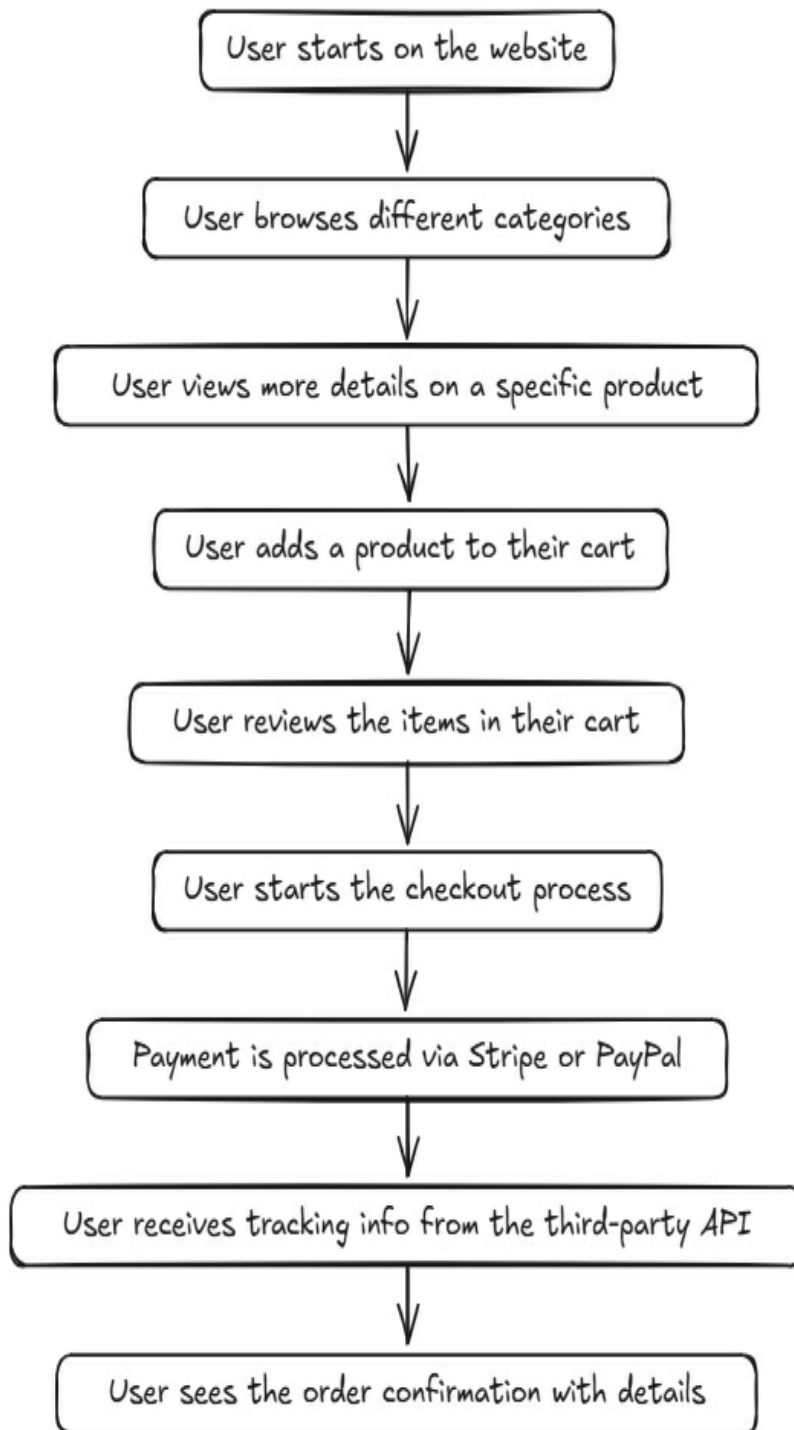**Backend Requirements with Sanity CMS**

1. **Product Management**:

- ✓ Store and manage product details (e.g., comfort features, product,materials)

- ✓ Easy updates to product catalog via Sanity CMS

2. **Customer and Order Management**:

- ✓ Track customer data (name, address, order history)
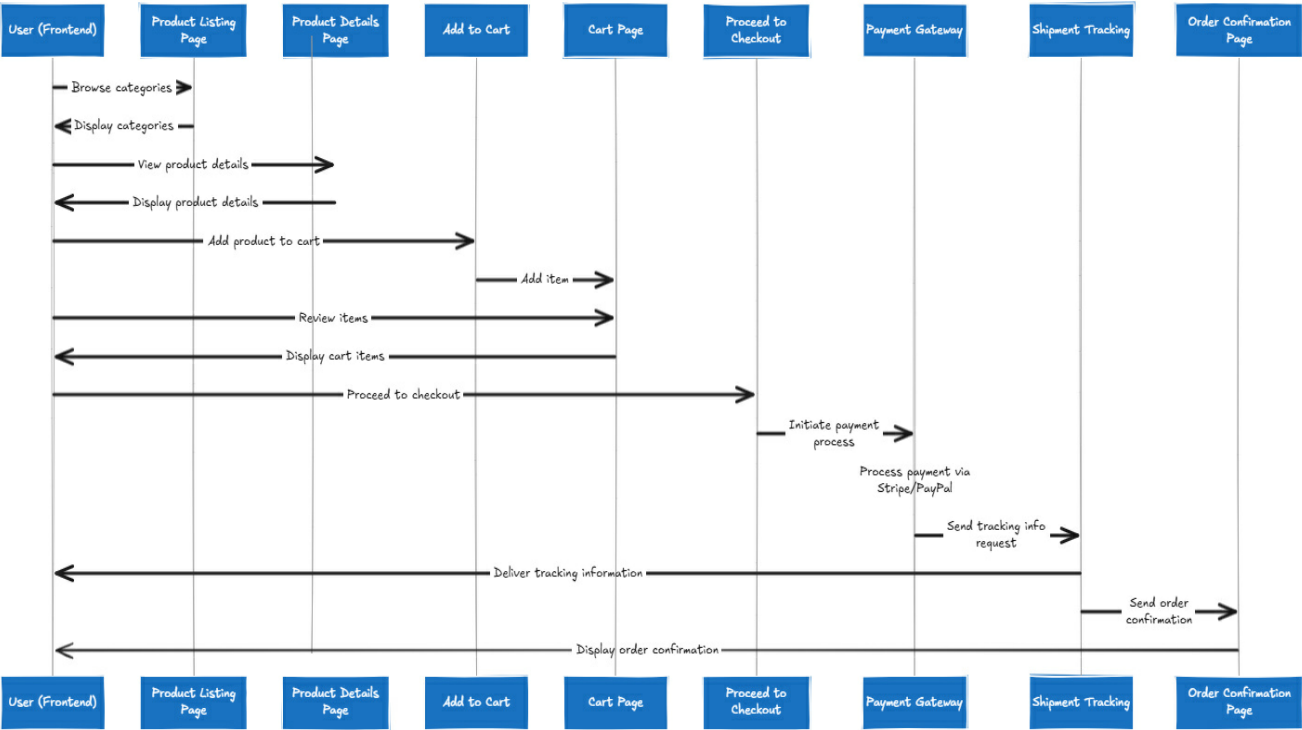
- ✓ Manage order and shipping details

System Architecture:

Diagram 1:

User starts on the website

↓

User browses different categories

↓

User views more details on a specific product

↓

User adds a product to their cart

↓

User reviews the items in their cart

↓

User starts the checkout process

↓

Payment is processed via Stripe or PayPal

↓

User receives tracking info from the third-party API

↓

User sees the order confirmation with details

Diagram 2:

APIs Requirements We Provide:

Endpoint	Method	Description
/products	GET	Retrieves a list of all available products.
/product/{id}	GET	Retrieves detailed information about a specific product.
/orders	POST	Creates a new order with customer and product details.
/shipment/{orderId}	GET	Retrieves the shipment status of a specific order.

Explanations:

1. **/products (GET):**

   ✓ **This API will list all the products available on your website, making it easier for users to view what is available to purchase.**

2. **/product/{id} (GET):**

   ✓ **This API allows the user to click on a product and view more detailed information such as the description, material, and dimensions.**

3. **/orders (POST):**

   ✓ **When a customer completes their shopping, this API will record their order by collecting their details, including products, quantities, and shipping information.**

4. **/shipment/{orderId} (GET):**

   ✓ **Once an order is placed, this API can be used to check the status of the shipment and tell the user where their order is in the delivery process (e.g., shipped, in transit, or delivered).**

WorkFlow:-

Clerk (Authentication)

- **Purpose:** Clerk handles user authentication and authorization for your marketplace. It provides features like sign-up, login, and user profile management, ensuring secure access to your platform.

- **Why Use It:** Simplifies the implementation of secure authentication and ensures smooth user registration and login experiences.

Sanity CMS (Content Management)

- **Purpose:** Sanity CMS serves as the backend for managing content such as product data, customer details, and orders. It allows for flexible and scalable management of product listings and other dynamic content.

- **Why Use It:** Provides a powerful content management system that integrates seamlessly with your frontend, allowing easy updates and real-time data management.

ShipEngine API (Shipping and Tracking)

- **Purpose:** ShipEngine API provides shipping label creation, real-time tracking, and rate calculation across multiple carriers (e.g., FedEx, UPS, USPS).

- **Why Use It:** Simplifies logistics by allowing you to easily integrate shipment tracking and management, offering your customers real-time updates on their orders.

Stripe (Payment Gateway)

- **Purpose:**

    Stripe is a secure payment processing platform that enables you to accept payments from customers through various methods (credit cards, PayPal, etc.).

- **Why Use It:** Offers a seamless and secure checkout experience, with robust support for different currencies and payment methods, while also handling fraud prevention and chargebacks.

React Context API (Cart Functionality)

- **Purpose:**

   The React Context API is used to manage the state of the shopping cart, including adding/removing items, displaying the total, and maintaining cart details across pages.

- **Why Use It:**

   Allows you to efficiently manage and share cart data across different components in your React app, creating a smoother user experience with real-time updates.

Tools & Libraries

- ✓ **Clerk**: Authentication.
- ✓ **Sanity CMS:** Content management.
- ✓ **ShipEngine API:** Shipping and tracking.
- ✓ **Stripe**: Payment gateway.
- ✓ **React Context API:** Cart functionality.