MarketPlace Technical Foundation: Task 2:

[General-e-commerce website chair]

Introduction:

The e-commerce landscape is evolving rapidly, requiring a unique blend of technical sophistication and user-centric design to stand out. For a marketplace dedicated to chairs—ranging from ergonomic office chairs to luxury furniture—building a robust, scalable, and secure platform is paramount. This document outlines the foundational technical framework to establish a marketplace that delivers a seamless experience for buyers and sellers alike, prioritizing innovation, functionality, and a dynamic user experience.

Technical Requirements:

Defination:

1:Frontend :

Create an visually aapealing user interface with the following pages.

- ## HomePage:
  - landing page with featured products and categories.
- ## Product Listing Page:
  - Display a products with essential details like name,price,and image.
- ## Cart Page:
  - A summary of selected items with options to update quantities or remove items.
- ## Checkout Page:

- A form to collect user details,shipping address, and payment information.

2: Sanity CMS:

- **Products:**Add ,update,or delete product detail (e.g:chair with properties like name,price,description,image)
- **Orders:**store and manage user orders,including customer information,order status,and payment detail.
- **Third-party APIS:**integrate external services for:
- **Payment Gateways:**services like stripe,paypal to process payment securely 1. Register as a Merchant with Easypaisa
- Visit the Easypaisa Business Portal and sign up for a merchant account.
- Provide necessary details like your business name, registration documents, and bank information.
- Once approved, you'll receive access to Easypaisa's API and integration documentation.
- **Shipment tracking:**

APIS such as shippo to track shipping status for customer order.

Example Product(chair):

- For a chair product,the followingdetails will be managed in sanity CMS and displayed on the frontened.
- **Product Name:**Ergonic chair
- **Price:**$150

- **Description:**A comfortable,adjustable.
- **Image:**Product image.

System Architecture Plan:

Description: This document outline the system architecture for an e-commerce platform focused on selling products like chairs.The diagram and explaination below to illustrate how different components interact,including the frontend,sanity CMS,and third-party APIS.

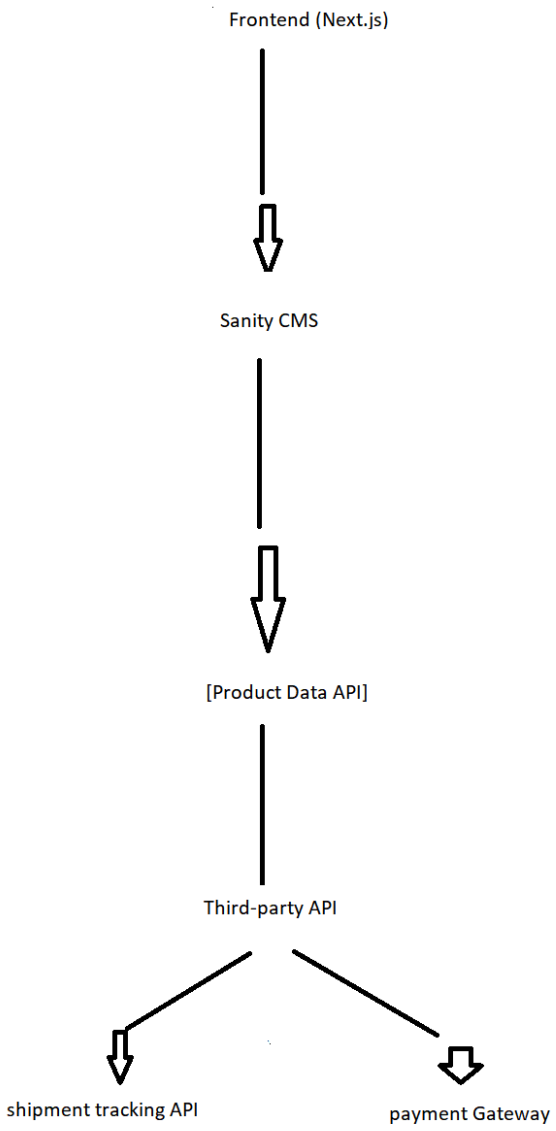


Diagram Explanation:

1: Frontend (Next js): The user interface where customers browse and interact with chair products.

2: Sanity CMS:A backend system for storing and managing chair product content.

3:Product Data API: An API that provides details to the frontend and third-party APIS.

4:Third-party API: External services that add additional functionalities.

5:Shipment Tracking API: Used for live tracking of shipment like ApI key to track an order for location.

6:Payment Gateway: Ensures secure payment processing like stripe,paypal method.

APIS(Application Programming Interface):

are set of rules and protocols that allow two softwere applications to communicate and exchange data with each other.

API Requirements:

1: Endpoint:Name:/products

Method: Get

Purpose: Fetch details of all chair products data.

2: Endpoint Name:/products/:id

Method: Get

Purpose: Fetch details of a specific chair products by its id.

3: Endpoint Name/Products

Method: Post

Purpose: Add a new chair product

4:Endpoint Name:/products/:id

Method: Put

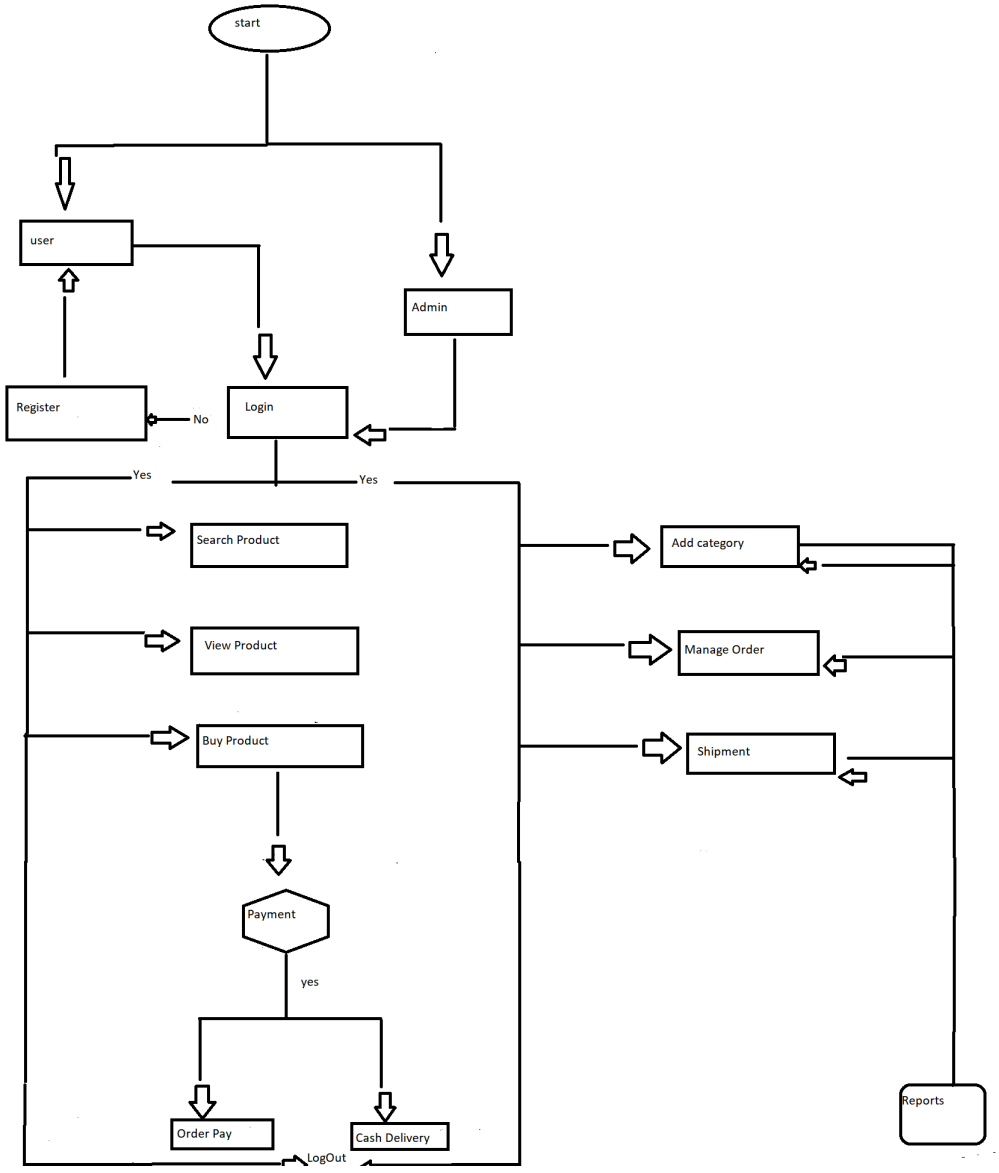
Purpose: Update details of a specific chair product by it ID.

5: EndPoint Name:/products/:id

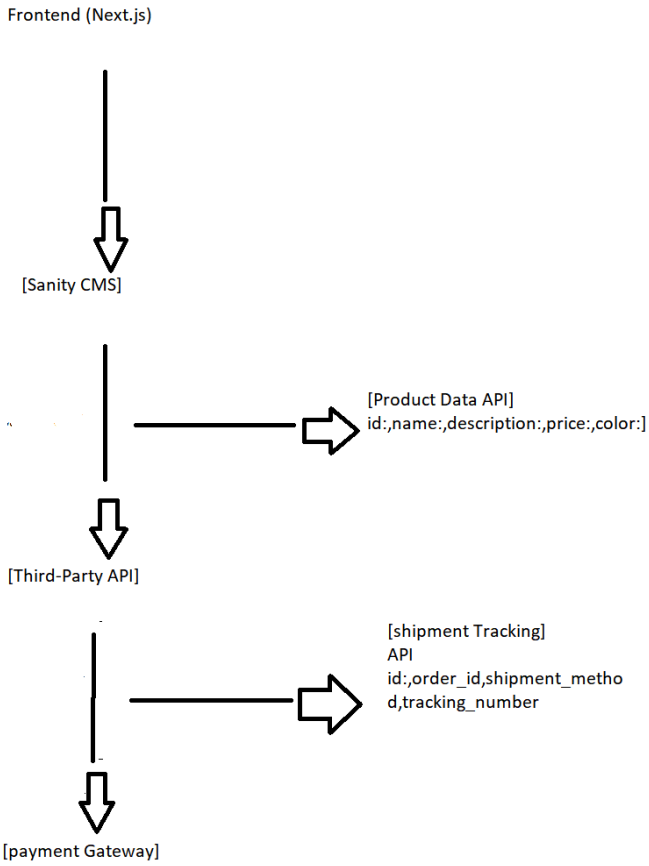
Method: Delete

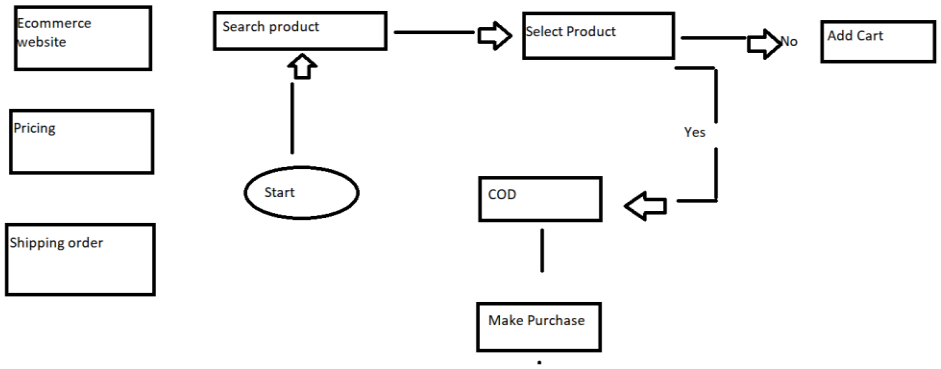
Purpose: Delete a specific chair product by its ID.

FlowChart:

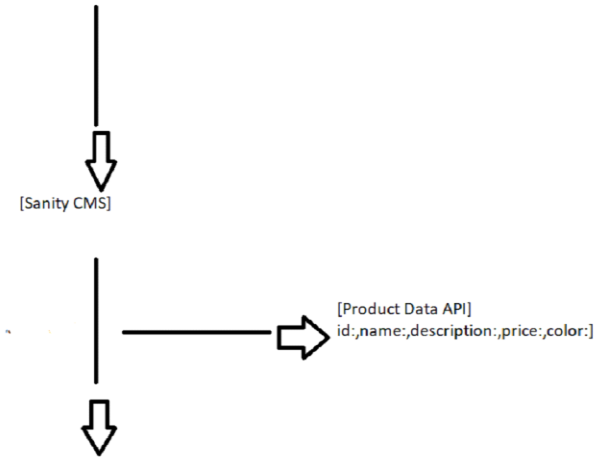


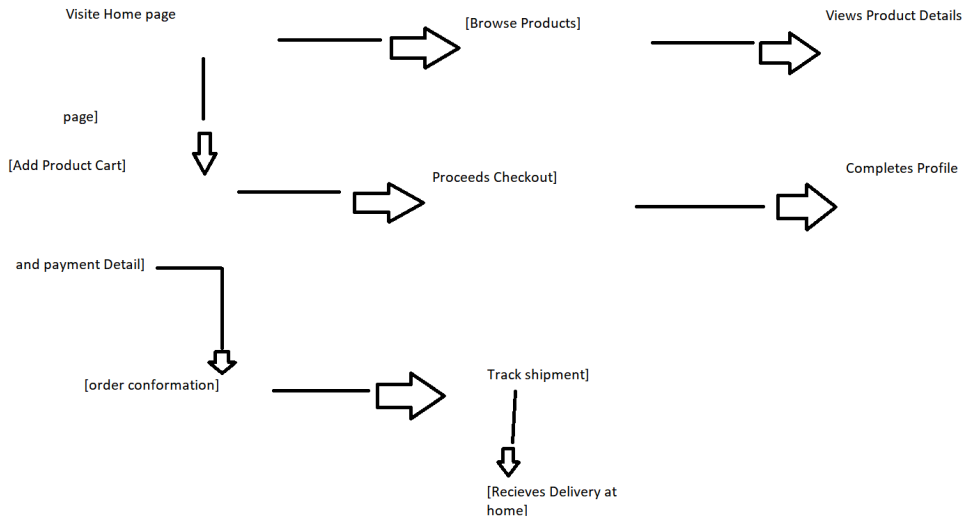
System architecture Oerview Diagram:


Frontend (Next.js)
[Sanity CMS]
[Product Data API]
id:,name:,description:,price:,color:]
[Third-Party API]
[shipment Tracking]
API
id:,order_id,shipment_metho
d,tracking_number
[payment Gateway]

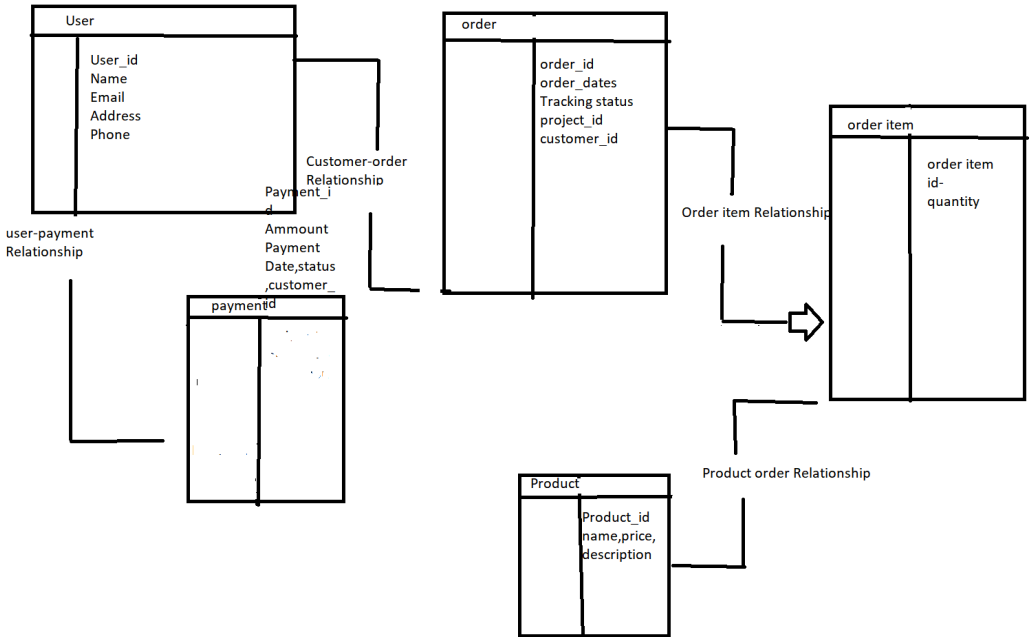


Frontend (Next.js)





Er Diagram:



EndPoint	Method	Description	Parameter	Response
/api/product	Get	Fetch all Product items	None	{id:1,name:"chair"}
/api/product/:id	Get	Fetch a single chair item	id(path)	{id:1,name:"chair"}
/api/product	post	Add a new product item	name,price	{sucess:true,id:5}
/api/product/:id	Put	update a productt item	id(path),name	{sucess:true}
/api/product/:id	Delete	Delete a product item	id (path)	{sucess:true}
/api/categories	Get	Fetch all product categories	None	{categories:["chairs"]

Api Schema:

Here's a schema for a chair product in sanity CMS.

export default{

name:"product",

type:"document",

```
tittle:"Product",
fields:[
{
name:"name",
type:"string",
tittle:"product Name",
},
{name:"price",
type:"number",
tittle:"price",
},
{
name:"description",
type:"text",
tittle:"description",
},
{name:"image",
type:"image",
tittle:"product Image",
},
{mame:"color",
type:"string",
tittle:"color",
description:"Primary color of the chair"
```

```
(e.g:black,white,blue)
,},
{
name:"currency",
type:"string",
description:"currency in which price iis listed (e.g:USD,EUR)
,},
{name:"reviews",
type:"array",
tittle:"reiews",of:[
{type:"object:,
fields:[
{name:"reiewer",type:"string",tittle:"Reviewer name"},
{name:"comment",type:"text",tittle:"Reiew comment"},
{name:"rating",type:"number",tittle:rating",}
],
},
],
},
],
};
```