

Day 3 API Integration Report –[PERFUMA]

API Integration Process

Step 1: Setting Up the API Endpoint

- **API Endpoint:** `https://template6-six.vercel.app/api/products`
- **Data Structure:** The API returns product data with various fields, including `productName`, `price`, `inventory`, `status`, `description`, `tags` and `image`.

Step 2: Fetching Data from the API

- An API request was made to retrieve an array of product objects. The response is parsed to ensure proper data serialization and integrity.

Adjustments to Product Schema

1. Field Updates:

- **Images:** The image field was configured as a Sanity image type with hotspot functionality enabled, ensuring optimal image cropping and scaling.
- **Category:** A category field was configured as tags based on their type.

Environment Setup & Dependencies

- **Dependencies Installed:** `@sanity/client`,
- A `.env.local` file was created to securely store sensitive environment variables.

Data Fetching Process

1. Retrieving Data:

- The response data was logged and parsed to verify its structure and ensure integrity.

2. Image Handling:

- Images from the API's `images` field were downloaded using `Axios`.
- These images were then uploaded to the Sanity Asset Manager using the `Sanity client`.

Image Upload Code Example:

```
const query=await client.fetch(`*[_type=='product']{title,_id,tags,
'image':productImage.asset->url,price,
isNew,description,_createdAt,dicountPercentage}` )
```

Migration Steps & Tools

1. Document Creation:

- Sanity documents for each product were created by combining the fetched API data with uploaded image references.

2. Migration Tools:

- **Sanity Client:** Used for interacting with the Sanity CMS.
- **Context Api:** Used for storing fetched data to access globally.

Code Snippets

```
"use client"
import { ReactNode, useEffect, useState } from "react";
import { CartContext, Product } from "../Context";
import { client } from "@sanity/lib/client";

const CartProvider=({children}:{children: React.ReactNode})=>{

  >   interface obj{ ...
  >   }
  >   interface CartProviderProps { ...
  >   }
  const [cart,setcart]=useState<obj[]>([])
  const [products,setProducts]=useState<Product[]>([])
  const [filteredProducts, setFilteredProducts] = useState<Product[]>([]);
  const [filteredProducts2, setFilteredProducts2] = useState<Product[]>([]);
  >   const [filters, setFilters] = useState<{ priceRange: number[]; sortBy: string;catagory:string }>({ ...
  >   });
  const [pagination, setPagination] = useState({ currentPage: 1, itemsPerPage: 16 });
  const [searchQuery,setsearchQuery]=useState<string>('')

  const [wish,setWish]=useState<obj[]>([])
  const addwish:any=(obj:obj)=>setWish([...wish,obj])
  const delwish=(id:any)=>{
    const newCArt=wish.filter((val)=>val.id !== id)
    return setWish([...newCArt])
  }

  const [compare,setcompare]=useState<obj[]>([])
```

API Integration:

javascript

Copy

```
const response = await axios.get('https://677cca8c4496848554c7aa31.mockapi.io/products');
```

```
const products = response.data.data;
```

Migration Script:

```

const [compare,setcompare]=useState<obj[]>([])
const addcompare:any=(obj:obj)=>setcompare([...compare,obj])
const delcompare=(id:any)=>{
  const newcompare=compare.filter((val)=>val.id !== id)
  return setcompare([...newcompare])}

useEffect(()=>{
  const fetchproduct=async()=>{
    const query=await client.fetch(`*[_type=='product']{title,_id,tags,
'image':productImage.asset->url,price,
isNew,description,_createdAt,dicountPercentage}`)
    setProducts(query)
  }
  fetchproduct()
},[])

```

- The migration script was developed to populate Sanity with product data and image references.

Final Checklist

- **Schema Validation:** Confirmed the correct data structure was in place.
- **API Integration:** Ensured smooth interaction between the API and the application.
- **Data Migration:** Successfully migrated the data into the Sanity CMS.
- **Frontend Display:** Ensured the frontend displayed the data properly.

Screenshots:

1. API Call Output

```

use client
import React, { useState, useEffect, useContext } from 'react';
import Image from 'next/image';
import { GitCompareArrows, Heart, Share2 } from 'lucide-react';
import Link from 'next/link';
import { client } from '@sanity/lib/client';
import { CartContext } from './Context';

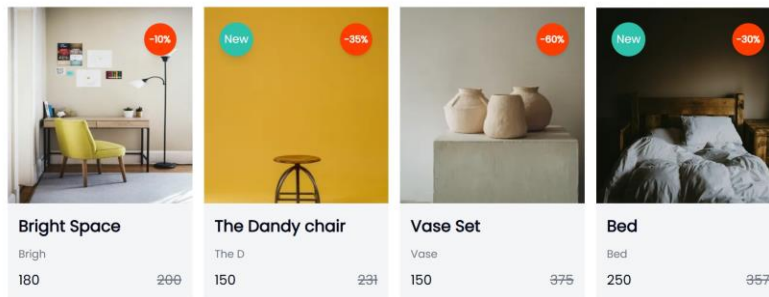
const Products = () => {
  const obj:any=useContext(CartContext)

  const handlewishlistitem=(item:any)=>{
    const isinwishlist=obj.wish.some((inwishlist:any)=>inwishlist.id === item._id)
    if(isinwishlist){
      obj.delwish(item._id)
    }else{
      obj.addwish({id:item._id,name:item.title,image:item.image,price:item.price,qnty:1,shipping:20})
    }
  }

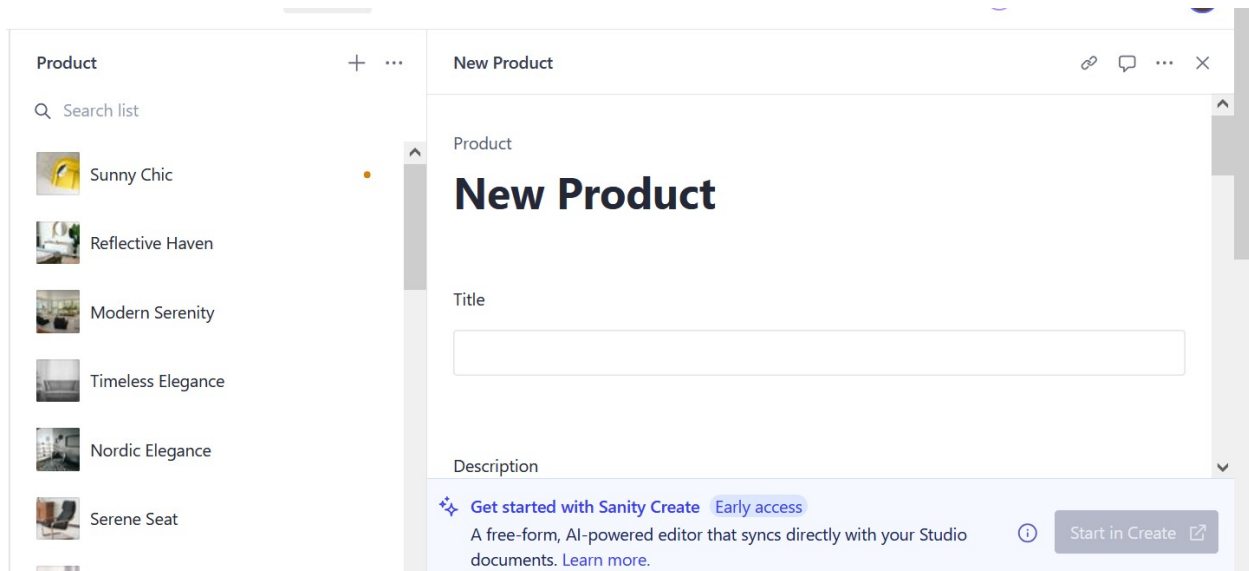
  const handlecompare=(item:any)=>{
    if(obj.compare.length < 2){
      const isincomparelist=obj.compare.some((incompare:any)=>incompare.id === item._id)
      if(isincomparelist){
        obj.delcompare(item._id)
      }else{
        obj.addcompare({description:item.description,id:item._id,compare:title,item.image,item.image.asset->url})
      }
    }
  }
}

```

2. Frontend Display



3. Populated Sanity CMS Fields



Name: Samiullah

Roll No: 00340796

Day / Time: Tuesday -7 pm to 10 pm