Day 3 - API Integration Report - [Hekto Marketplace Name]

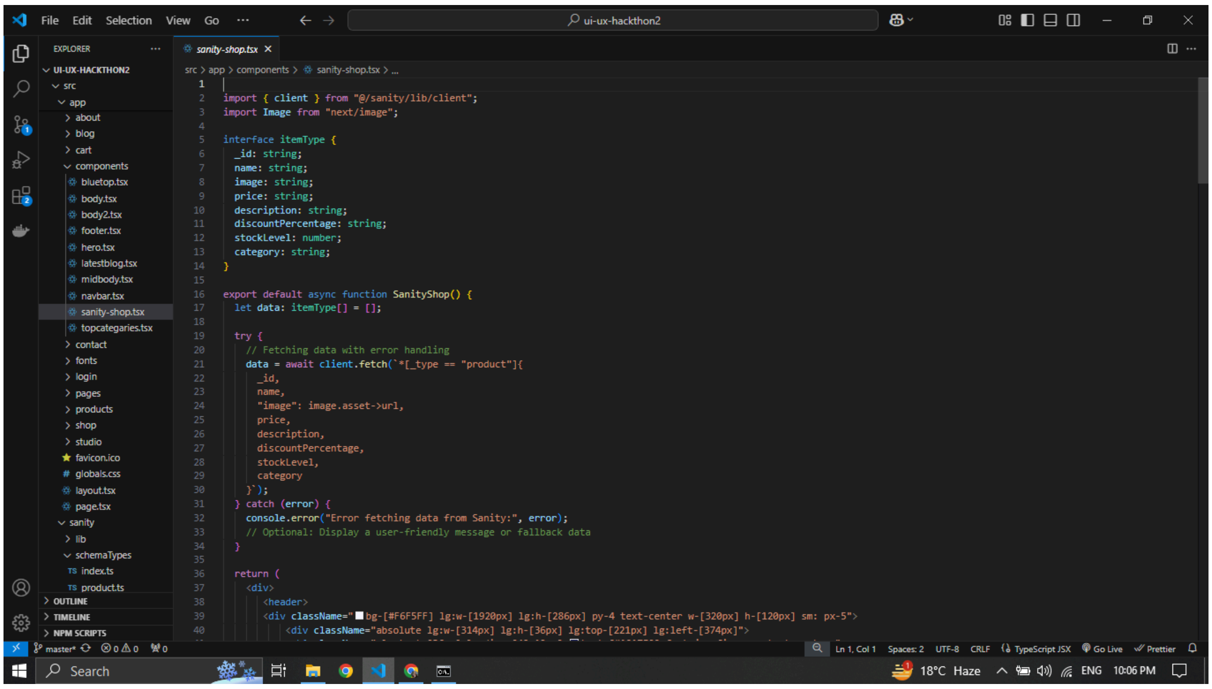
API Integration Process

1. Understanding the Provided API

- **API Documentation:**
  - I reviewed the API documentation to understand the available endpoints and their response structures.
  - Example Endpoint: `/api/products` for fetching product data.
- **Tools Used:**
  - **Postman:** Tested API endpoints to verify responses and check for errors.
  - **Browser Developer Tools:** Validated API calls and responses directly in the application.
- **Key Findings:**
  - The API returned product data including `id`, `name`, `description`, `price`, and `image` fields.

2. API Integration Steps

- **Utility Function:** I created a utility function to fetch data from the API:



```tsx

import { client } from "@/sanity/lib/client";
import Image from "next/image";

interface itemType {
  _id: string;
  name: string;
  image: string;
  price: string;
  description: string;
  discountPercentage: string;
  stockLevel: number;
  category: string;
}

export default async function SanityShop() {
  let data: itemType[] = [];

  try {
    // Fetching data with error handling
    data = await client.fetch(`*[_type == "product"]{
      _id,
      name,
      "image": image.asset->url,
      price,
      description,
      discountPercentage,
      stockLevel,
      category
    }`);
  } catch (error) {
    console.error("Error fetching data from Sanity:", error);
    // Optional: Display a user-friendly message or fallback data
  }

  return (
    <div>
      <header>
      <div className="bg-[#F6F5FF] lg:w-[1920px] lg:h-[286px] py-4 text-center w-[320px] h-[120px] sm: px-5">
        <div className="absolute lg:w-[314px] lg:h-[36px] lg:top-[221px] lg:left-[374px]">
```

Frontend Integration:

- Implemented a product listing component in the frontend to display the fetched data.
- Used Tailwind CSS for styling and ensured responsive design.

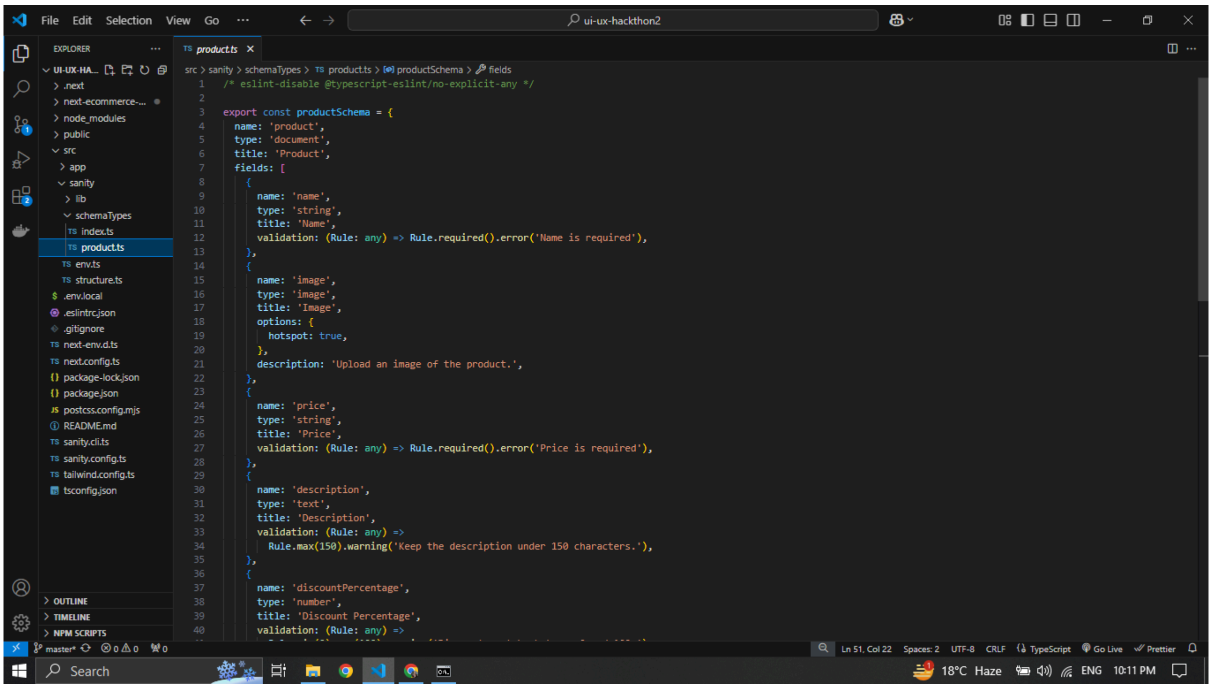
Adjustments Made to Schemas

1. Understanding Schema Requirements

- **Provided API Data:**
    - Example field from the API: `product_title`.
- **Sanity Schema Field:**
    - Adjusted field name to `name` in the Sanity schema for consistency.

2. Schema Modifications

- Modified `product` schema in Sanity:



```
/* eslint-disable @typescript-eslint/no-explicit-any */

export const productSchema = {
  name: 'product',
  type: 'document',
  title: 'Product',
  fields: [
    {
      name: 'name',
      type: 'string',
      title: 'Name',
      validation: (Rule: any) => Rule.required().error('Name is required'),
    },
    {
      name: 'image',
      type: 'image',
      title: 'Image',
      options: {
        hotspot: true,
      },
      description: 'Upload an image of the product.',
    },
    {
      name: 'price',
      type: 'string',
      title: 'Price',
      validation: (Rule: any) => Rule.required().error('Price is required'),
    },
    {
      name: 'description',
      type: 'text',
      title: 'Description',
      validation: (Rule: any) =>
        Rule.max(150).warning('Keep the description under 150 characters.'),
    },
    {
      name: 'discountPercentage',
      type: 'number',
      title: 'Discount Percentage',
      validation: (Rule: any) =>
```

Reasoning:

- Adjusted field names for compatibility with API data.
- Ensured proper field types (`string`, `number`, `image`) for smooth migration.

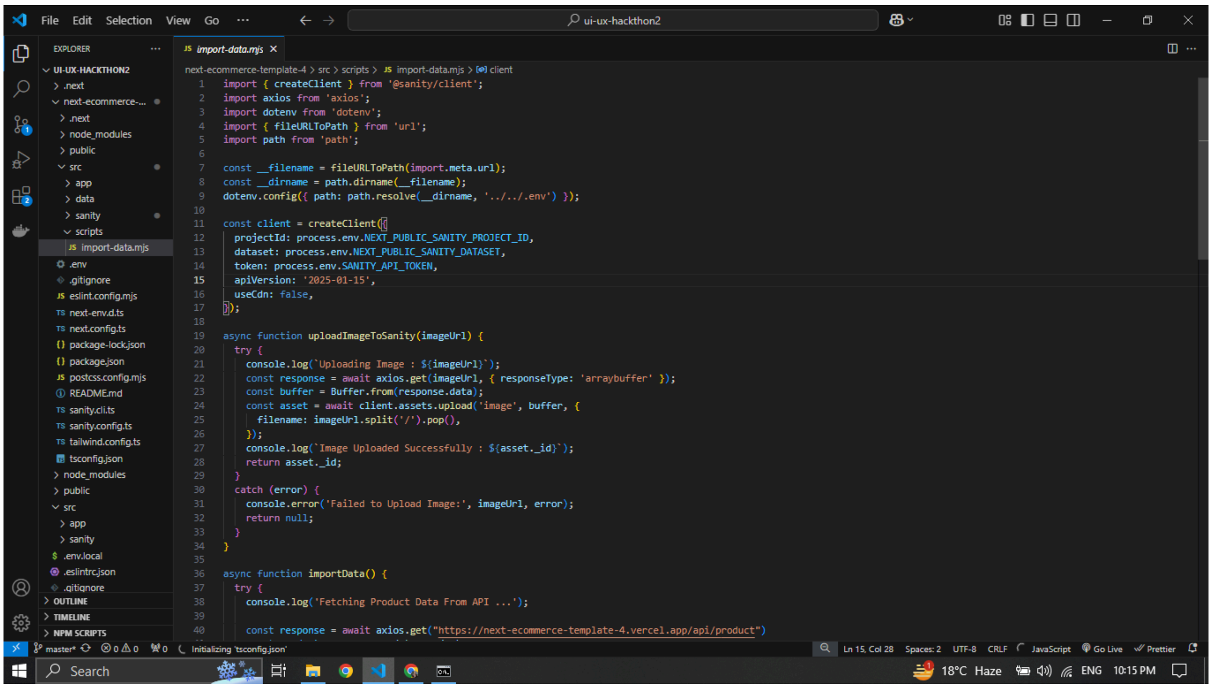
Migration Steps and Tools Used

1. Tools Used

- **Sanity CLI:** For setting up the project and importing data.
- **Custom Scripts:** Wrote scripts to transform and migrate API data into Sanity CMS

2. Migration Script

- Example migration script used to import API data:



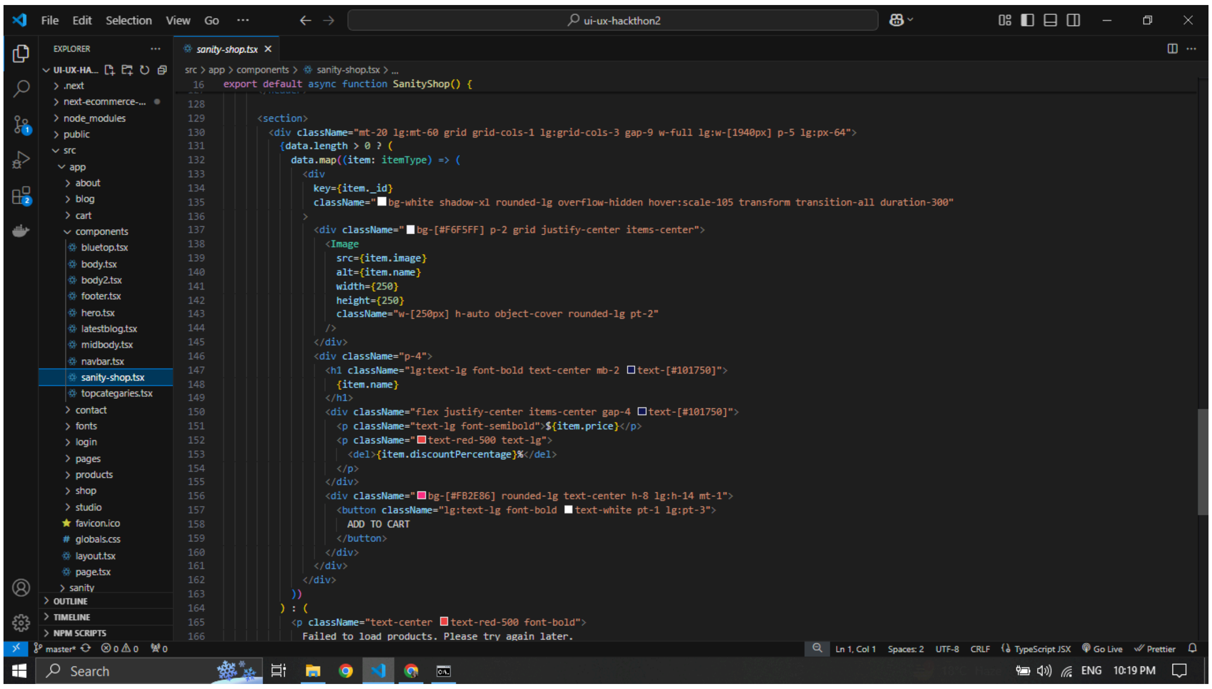
File Edit Selection View Go
ui-ux-hackthon2
EXPLORER
import-data.mjs
next-ecommerce-template-4 > src > scripts > import-data.mjs > client

```javascript
import { createClient } from '@sanity/client';
import axios from 'axios';
import dotenv from 'dotenv';
import { fileURLToPath } from 'url';
import path from 'path';

const __filename = fileURLToPath(import.meta.url);
const __dirname = path.dirname(__filename);
dotenv.config({ path: path.resolve(__dirname, '../../.env') });

const client = createClient({
  projectId: process.env.NEXT_PUBLIC_SANITY_PROJECT_ID,
  dataset: process.env.NEXT_PUBLIC_SANITY_DATASET,
  token: process.env.SANITY_API_TOKEN,
  apiVersion: '2025-01-15',
  useCdn: false,
});

async function uploadImageToSanity(imageUrl) {
  try {
    console.log(`Uploading Image : ${imageUrl}`);
    const response = await axios.get(imageUrl, { responseType: 'arraybuffer' });
    const buffer = Buffer.from(response.data);
    const asset = await client.assets.upload('image', buffer, {
      filename: imageUrl.split('/').pop(),
    });
    console.log(`Image Uploaded Successfully : ${asset._id}`);
    return asset._id;
  }
  catch (error) {
    console.error('Failed to Upload Image:', imageUrl, error);
    return null;
  }
}

async function importData() {
  try {
    console.log('Fetching Product Data From API ...');

    const response = await axios.get("https://next-ecommerce-template-4.vercel.app/api/product")
```

3. Migration Process

- Fetched data from the API.
- Transformed the data to match the Sanity schema.
- Imported data into Sanity CMS using the migration script.

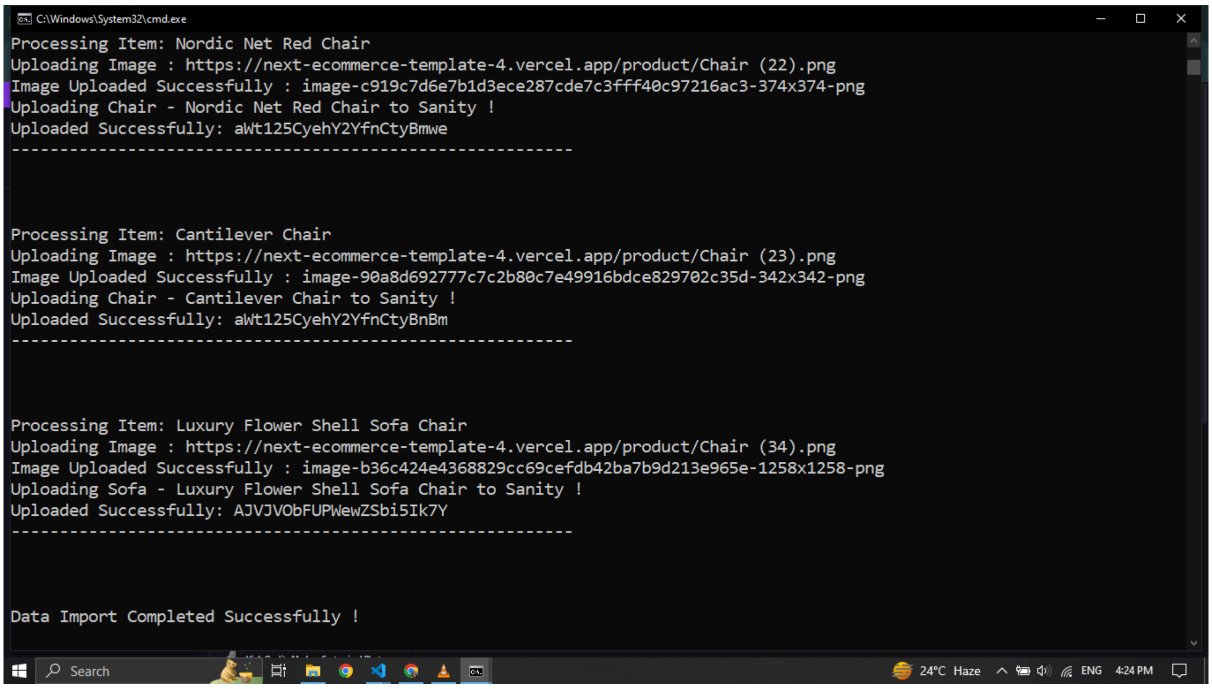
Frontend Product Display Code:



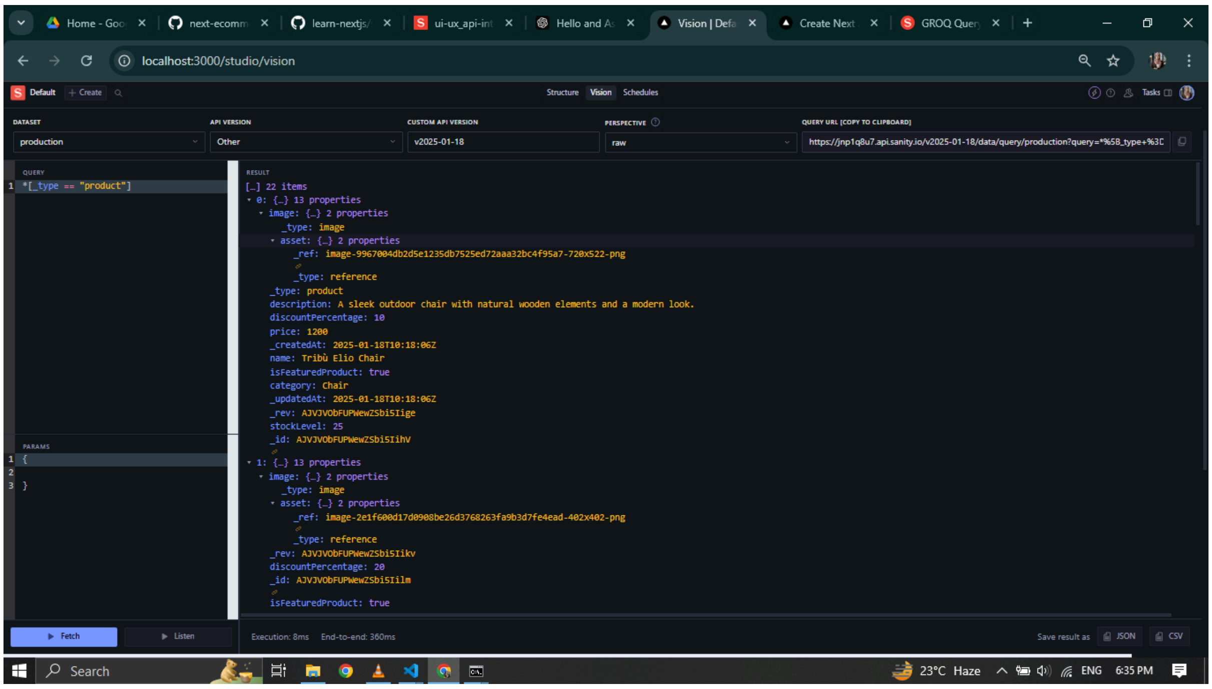
```
export default async function SanityShop() {

    <section>
      <div className="mt-20 lg:mt-60 grid grid-cols-1 lg:grid-cols-3 gap-9 w-full lg:w-[1940px] p-5 lg:px-64">
        {data.length > 0 ? (
          data.map((item: itemType) => (
            <div
              key={item._id}
              className="bg-white shadow-xl rounded-lg overflow-hidden hover:scale-105 transform transition-all duration-300"
            >
              <div className="bg-[#F6F5FF] p-2 grid justify-center items-center">
                <Image
                  src={item.image}
                  alt={item.name}
                  width={250}
                  height={250}
                  className="w-[250px] h-auto object-cover rounded-lg pt-2"
                />
              </div>
              <div className="p-4">
                <h1 className="lg:text-lg font-bold text-center mb-2 text-[#101750]">
                  {item.name}
                </h1>
                <div className="flex justify-center items-center gap-4 text-[#101750]">
                  <p className="text-lg font-semibold">${item.price}</p>
                  <p className="text-red-500 text-lg">
                    <del>{item.discountPercentage}%</del>
                  </p>
                </div>
                <div className="bg-[#FB2E86] rounded-lg text-center h-8 lg:h-14 mt-1">
                  <button className="lg:text-lg font-bold text-white pt-1 lg:pt-3">
                    ADD TO CART
                  </button>
                </div>
              </div>
            </div>
          ))
        ) : (
          <p className="text-center text-red-500 font-bold">
            Failed to load products. Please try again later.
```

Screenshots

- **API Calls:** Include screenshots from Postman or browser developer tools.
- **Frontend Display:** Show products displayed on the frontend.
- **Sanity CMS:** Capture populated fields in the Sanity dashboard.



```
C:\Windows\System32\cmd.exe
Processing Item: Nordic Net Red Chair
Uploading Image : https://next-ecommerce-template-4.vercel.app/product/Chair (22).png
Image Uploaded Successfully : image-c919c7d6e7b1d3ece287cde7c3fff40c97216ac3-374x374-png
Uploading Chair - Nordic Net Red Chair to Sanity !
Uploaded Successfully: aWt125CyehY2YfnCtyBmwe
----------------------------------------------------------



Processing Item: Cantilever Chair
Uploading Image : https://next-ecommerce-template-4.vercel.app/product/Chair (23).png
Image Uploaded Successfully : image-90a8d692777c7c2b80c7e49916bdce829702c35d-342x342-png
Uploading Chair - Cantilever Chair to Sanity !
Uploaded Successfully: aWt125CyehY2YfnCtyBnBm
----------------------------------------------------------



Processing Item: Luxury Flower Shell Sofa Chair
Uploading Image : https://next-ecommerce-template-4.vercel.app/product/Chair (34).png
Image Uploaded Successfully : image-b36c424e4368829cc69cefdb42ba7b9d213e965e-1258x1258-png
Uploading Sofa - Luxury Flower Shell Sofa Chair to Sanity !
Uploaded Successfully: AJVJVObFUPWewZSbi5Ik7Y
----------------------------------------------------------



Data Import Completed Successfully !
```



localhost:3000/studio/vision

DATASET: production
API VERSION: Other
CUSTOM API VERSION: v2025-01-18
PERSPECTIVE: raw
QUERY URL [COPY TO CLIPBOARD]: https://jnp1q8u7.api.sanity.io/v2025-01-18/data/query/production?query=*%5B_type+%3C

QUERY

```
*[_type == "product"]
```

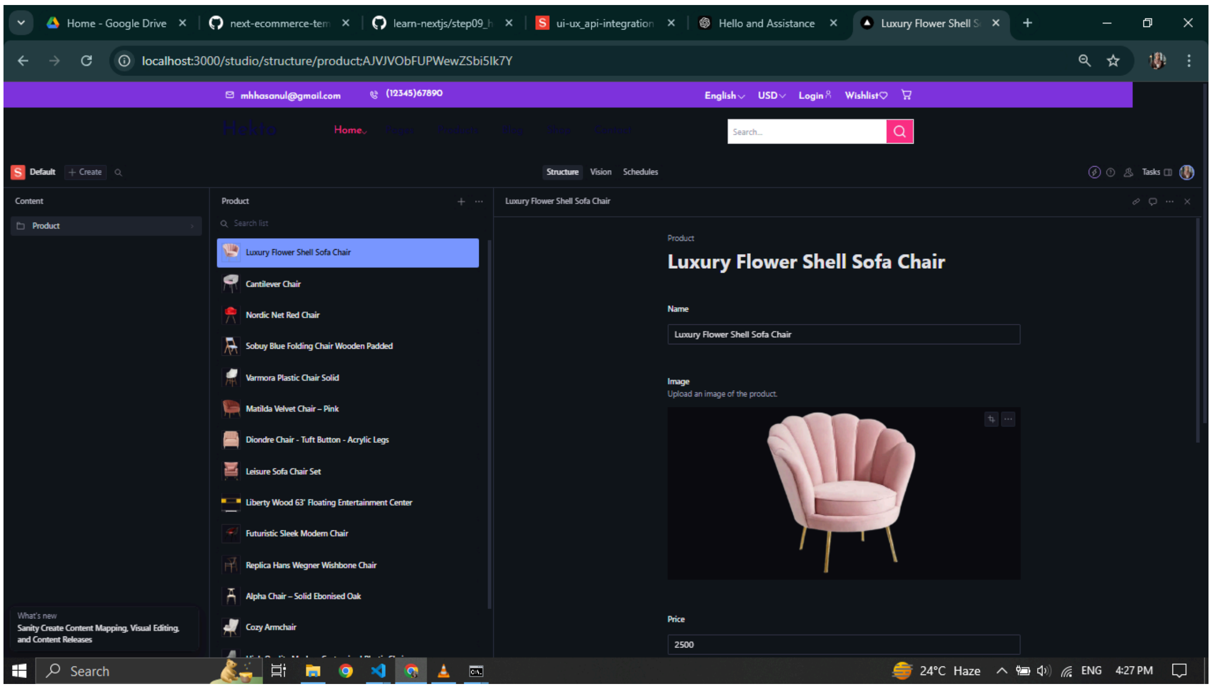
PARAMS

```
{

}
```

RESULT

```
[…] 22 items
▾ 0: {…} 13 properties
  ▾ image: {…} 2 properties
      _type: image
    ▾ asset: {…} 2 properties
        _ref: image-9967004db2d5e1235db7525ed72aaa32bc4f95a7-720x522-png
        _type: reference
    _type: product
    description: A sleek outdoor chair with natural wooden elements and a modern look.
    discountPercentage: 10
    price: 1200
    _createdAt: 2025-01-18T10:18:06Z
    name: Tribù Elio Chair
    isFeaturedProduct: true
    category: Chair
    _updatedAt: 2025-01-18T10:18:06Z
    _rev: AJVJVObFUPWewZSbi5Iige
    stockLevel: 25
    _id: AJVJVObFUPWewZSbi5IihV
▾ 1: {…} 13 properties
  ▾ image: {…} 2 properties
      _type: image
    ▾ asset: {…} 2 properties
        _ref: image-2e1f600d17d0908be26d3768263fa9b3d7fe4ead-402x402-png
        _type: reference
    _rev: AJVJVObFUPWewZSbi5Iikv
    discountPercentage: 20
    _id: AJVJVObFUPWewZSbi5Iilm
    isFeaturedProduct: true
```

Execution: 8ms End-to-end: 360ms



localhost:3000/studio/structure/product;AJVJVObFUPWewZSbi5Ik7Y

mhhasanul@gmail.com (12345)67890 English USD Login Wishlist

Content
- Product

Product
- Luxury Flower Shell Sofa Chair
- Cantilever Chair
- Nordic Net Red Chair
- Sobuy Blue Folding Chair Wooden Padded
- Varmora Plastic Chair Solid
- Matilda Velvet Chair – Pink
- Diondre Chair - Tuft Button - Acrylic Legs
- Leisure Sofa Chair Set
- Liberty Wood 63' Floating Entertainment Center
- Futuristic Sleek Modern Chair
- Replica Hans Wegner Wishbone Chair
- Alpha Chair – Solid Ebonised Oak
- Cozy Armchair

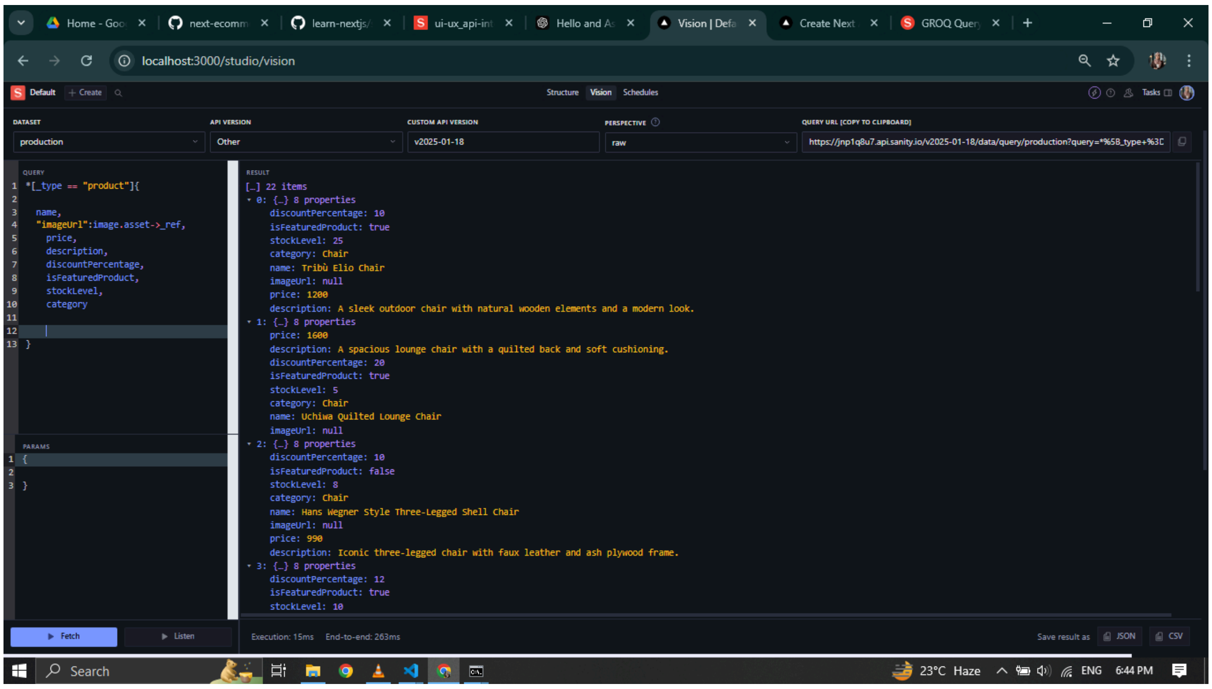
Product
Luxury Flower Shell Sofa Chair

Name: Luxury Flower Shell Sofa Chair

Image
Upload an image of the product.

Price: 2500

What's new
Sanity Create Content Mapping, Visual Editing, and Content Releases


localhost:3000/studio/vision

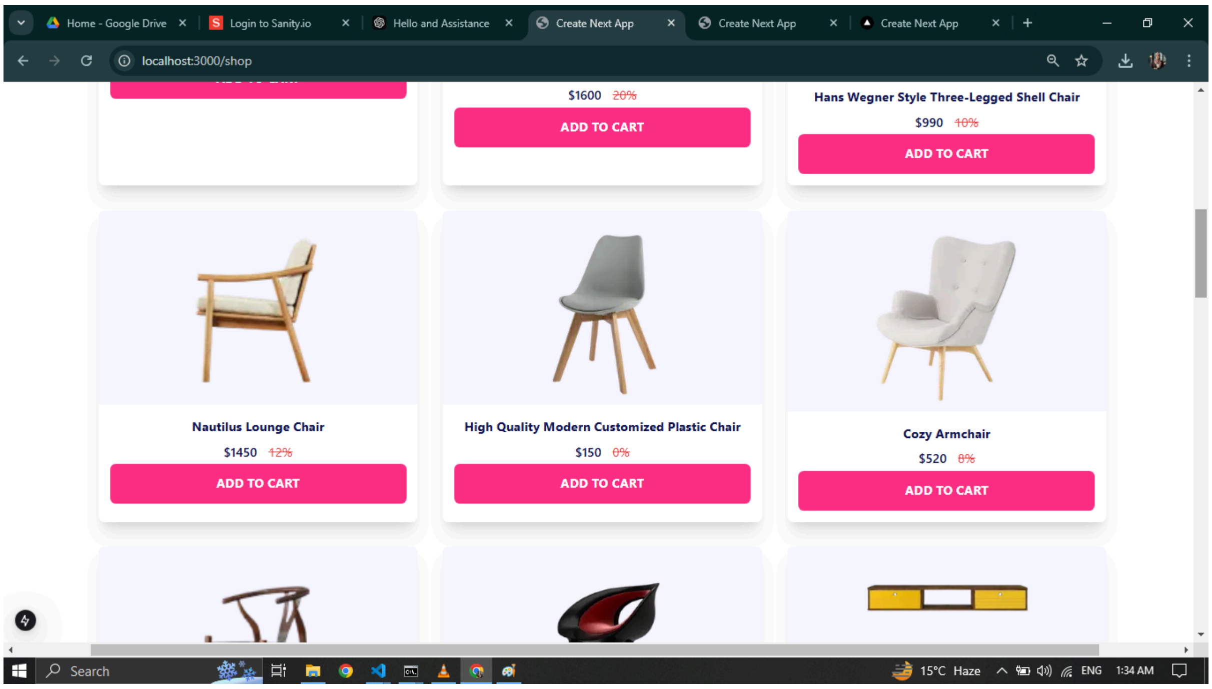
DATASET: production
API VERSION: Other
CUSTOM API VERSION: v2025-01-18
PERSPECTIVE: raw
QUERY URL [COPY TO CLIPBOARD]: https://jnp1q8u7.api.sanity.io/v2025-01-18/data/query/production?query=*%5B_type+%3D

QUERY
```
*[_type == "product"]{

  name,
  "imageUrl":image.asset->_ref,
    price,
    description,
    discountPercentage,
    isFeaturedProduct,
    stockLevel,
    category


}
```

PARAMS
```
{

}
```

RESULT
```
[…] 22 items
▾ 0: {…} 8 properties
    discountPercentage: 10
    isFeaturedProduct: true
    stockLevel: 25
    category: Chair
    name: Tribù Elio Chair
    imageUrl: null
    price: 1200
    description: A sleek outdoor chair with natural wooden elements and a modern look.
▾ 1: {…} 8 properties
    price: 1600
    description: A spacious lounge chair with a quilted back and soft cushioning.
    discountPercentage: 20
    isFeaturedProduct: true
    stockLevel: 5
    category: Chair
    name: Uchiwa Quilted Lounge Chair
    imageUrl: null
▾ 2: {…} 8 properties
    discountPercentage: 10
    isFeaturedProduct: false
    stockLevel: 8
    category: Chair
    name: Hans Wegner Style Three-Legged Shell Chair
    imageUrl: null
    price: 990
    description: Iconic three-legged chair with faux leather and ash plywood frame.
▾ 3: {…} 8 properties
    discountPercentage: 12
    isFeaturedProduct: true
    stockLevel: 10
```

Execution: 15ms End-to-end: 263ms



Conclusion

- The API integration and data migration tasks have been successfully completed.
- Data is now properly displayed on the frontend and managed in Sanity CMS.