Day 3 - API Integration Report General E-commerce

Overview:

On day three of the marketplace hackathon the task was to inject data into sanity CMS and make our frontend live. We were given APIs and data migration documentation to make our frontend live.This was one of the best practices to prepare us for the international market.

API Integration:

For the API integration process I used The API Provided in the comprehensive guide:
1-The API integration involved Populating sanity CMS with data.
2- The data was formatted according to the schema.
3-I ensured that The data was compatible with my template due to which I committed some changes In the schema.

Migration steps and tools used.

The API integration process was challenging but not difficult. I myself faced errors and bugs but in the end I finally did it.

Installation:

First of all I Started by the installation of next.js and sanity CMS in my project.

Sensitive data:

I added sensitive data into the .env.local file like project ID ,dataset, and my token. I also made sure that I didn't push it into the github repo for safety reasons.

Making necessary folders and data importation:

By the guide Provided to me, I created a scripts folder in the root of my project. Within the scripts folder I Created a file named import-data.mjs after adding the provided code into the file. I added some code to package.json to tell it from where to get the data. I ran the file by using the command 👇
npm run import-data.
Hence as a result the data was imported into sanity.

Adjustments made to schema:

The product schema I originally designed was mostly compatible with my data Which I imagined But related to The API I made some drastic changes.

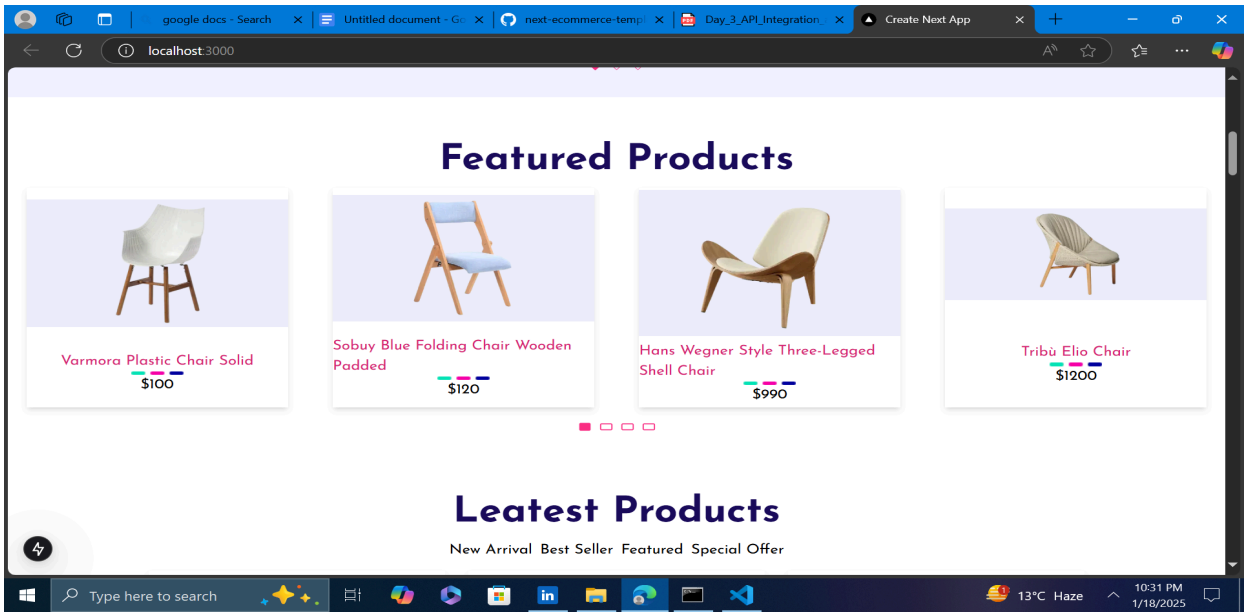
1-First of all I changed the tags part. I created categories using boolean Buttons so that there are no spelling errors and I would be able to save time.

2-I removed the size tags as my data was about chairs and sofas so there's no urgent need of sizes but I will add the functionality according to the time management.

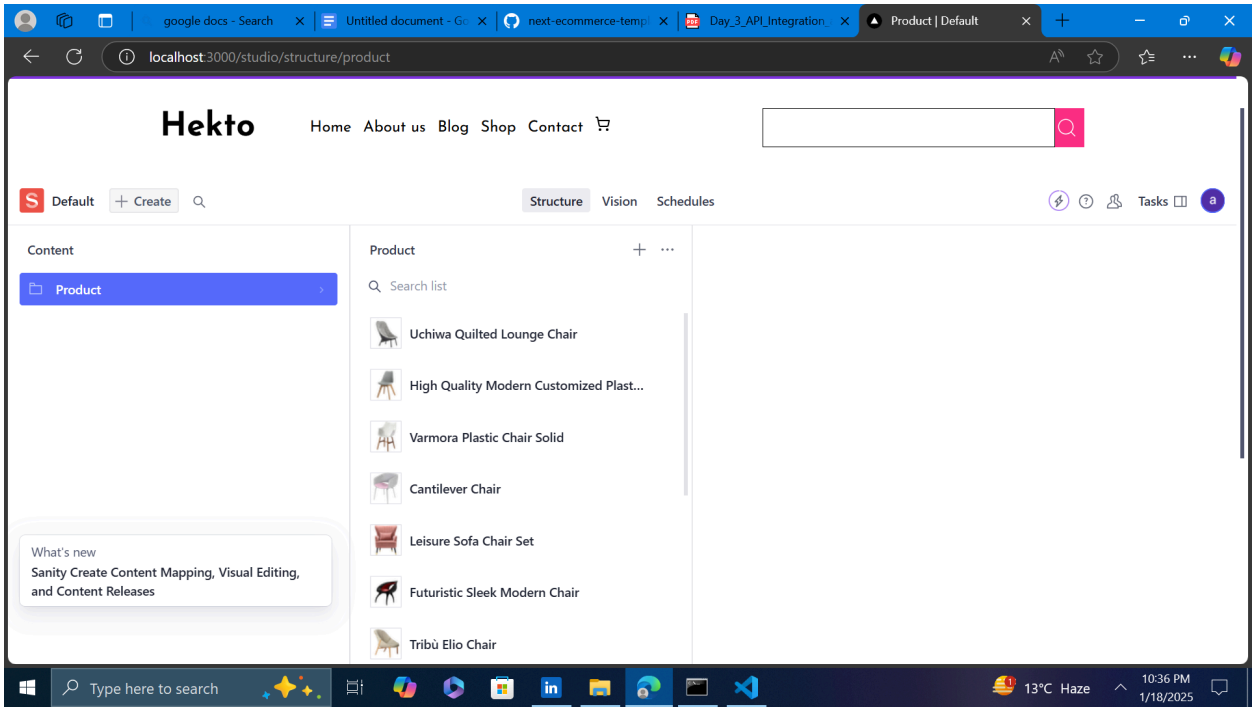
API Calls:

```
const fetchproductdata:ProductsDetails[]= await client.fetch(`
    *[_type=="product" && LatestProducts == true]{
      _id,
      name,
      image,
      price,
     description,
     discountPercentage,
     stockLevel,
     category
 }`)
console.log("fetched product",fetchproductdata);
```

Data successfully displayed in the frontend.



Data successfully displayed In sanity CMS:



Code sinnepts For API Integration and migration scripts:

```
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.local') });

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")
    const products = response.data.products;

    for (const item of products) {
      console.log(`Processing Item: ${item.name}`);

      let imageRef = null;
      if (item.imagePath) {
        imageRef = await uploadImageToSanity(item.imagePath);
      }

      const sanityItem = {
        _type: 'product',
        name: item.name,
        category: item.category || null,
        price: item.price,
        description: item.description || '',
        discountPercentage: item.discountPercentage || 0,
        stockLevel: item.stockLevel || 0,
        isFeaturedProduct: item.isFeaturedProduct,
        image: imageRef
          ? {
              _type: 'image',
              asset: {
                _type: 'reference',
                _ref: imageRef,
              },
            }
          : undefined,
      };

      console.log(`Uploading ${sanityItem.category} - ${sanityItem.name}
to Sanity !`);
      const result = await client.create(sanityItem);
      console.log(`Uploaded Successfully: ${result._id}`);
```

```
console.log("------------------------------------------------------------")
      console.log("\n\n")
    }

    console.log('Data Import Completed Successfully !');
  } catch (error) {
    console.error('Error Importing Data : ', error);
  }
}

importData();
```

Conclusion:

Day Three Was filled with an exciting challenge. But I am happy that I have finally completed It on my own and each and every day I am getting closer to achieving my marketplace goal.