

Day 3 - API Integration Report - Meubel House.

1. Introduction

This report outlines the process of integrating an external API into a marketplace project. The objective was to fetch product data, including images, from an API and store it in a Sanity CMS. It also includes adjustments made to Sanity's schema, migration steps, and tools used.

2. API Integration Process

1. Fetching Data from API:

- I fetched product data from an external API using Axios, a promise-based HTTP client.
- The API returned a list of products, each containing product details like name, image URL, price, stock level, and category.

2. Handling Images:

- The images were hosted externally (Unsplash). For each product, the image URL was fetched.
- I used Sanity's asset pipeline to upload the images to the Sanity media library by converting the image URL into a binary buffer using Axios.

3. Importing Data into Sanity:

- Once the images were uploaded, the product data (including image references) was created in the Sanity CMS using the Sanity client.
- Each product document was created with fields like name, description, price, category, and an image reference.

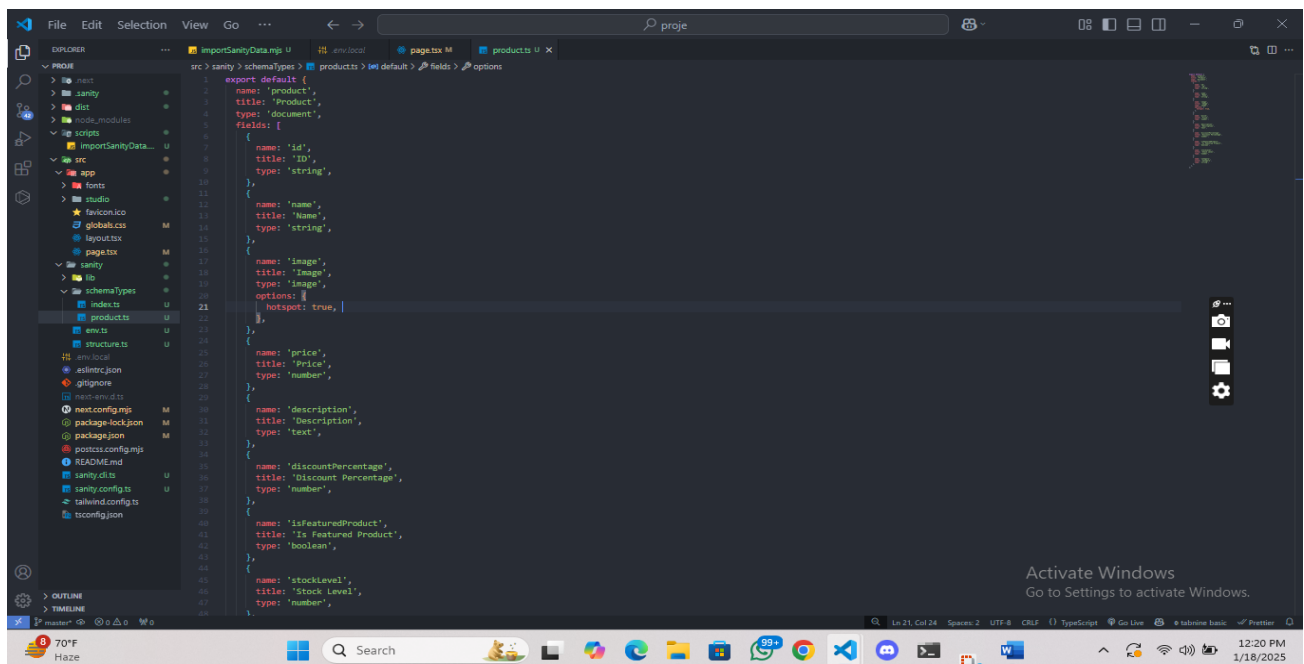
4. Sanity Schema:

- The Sanity schema was updated to support the product model, which included fields like id, name, image, price, description, etc.
- An image field was specifically added to reference the uploaded image in Sanity.

3. Adjustments Made to Schemas

Schema Adjustments

The original schema did not include a field for image references. I modified it to accommodate an image upload via Sanity's asset pipeline. Below is the updated schema:



```
export default {
  name: 'product',
  title: 'Product',
  type: 'document',
  fields: [
    {
      name: 'id',
      title: 'ID',
      type: 'string',
    },
    {
      name: 'name',
      title: 'Name',
      type: 'string',
    },
    {
      name: 'image',
      title: 'Image',
      type: 'image',
      options: {
        hotspot: true,
      },
    },
    {
      name: 'price',
      title: 'Price',
      type: 'number',
    },
    {
      name: 'description',
      title: 'Description',
      type: 'text',
    },
    {
      name: 'discountPercentage',
      title: 'Discount Percentage',
      type: 'number',
    },
    {
      name: 'isFeaturedProduct',
      title: 'Is Featured Product',
      type: 'boolean',
    },
    {
      name: 'stockLevel',
      title: 'Stock Level',
      type: 'number',
    },
  ],
}
```

Changes to Sanity Documents

- **Image Field:** Added a new field image of type image to handle product images. This field stores the reference to the uploaded image.

- **Hotspot Option:** Enabled the hotspot option to allow for better image cropping and manipulation within Sanity Studio.

4. Migration Steps and Tools Used

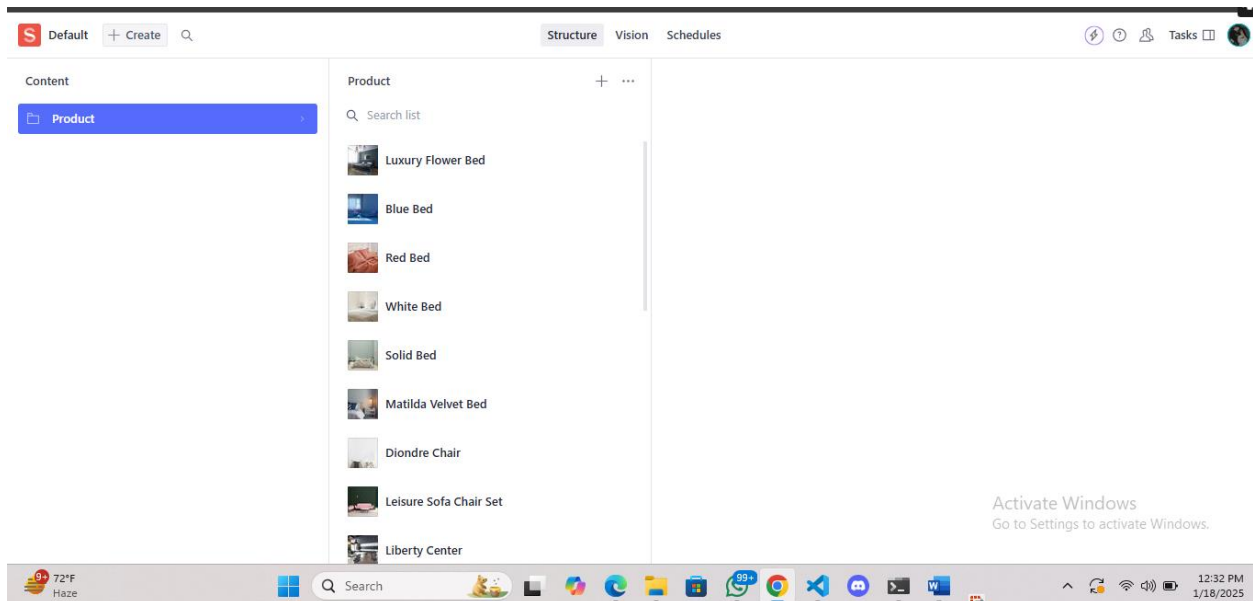
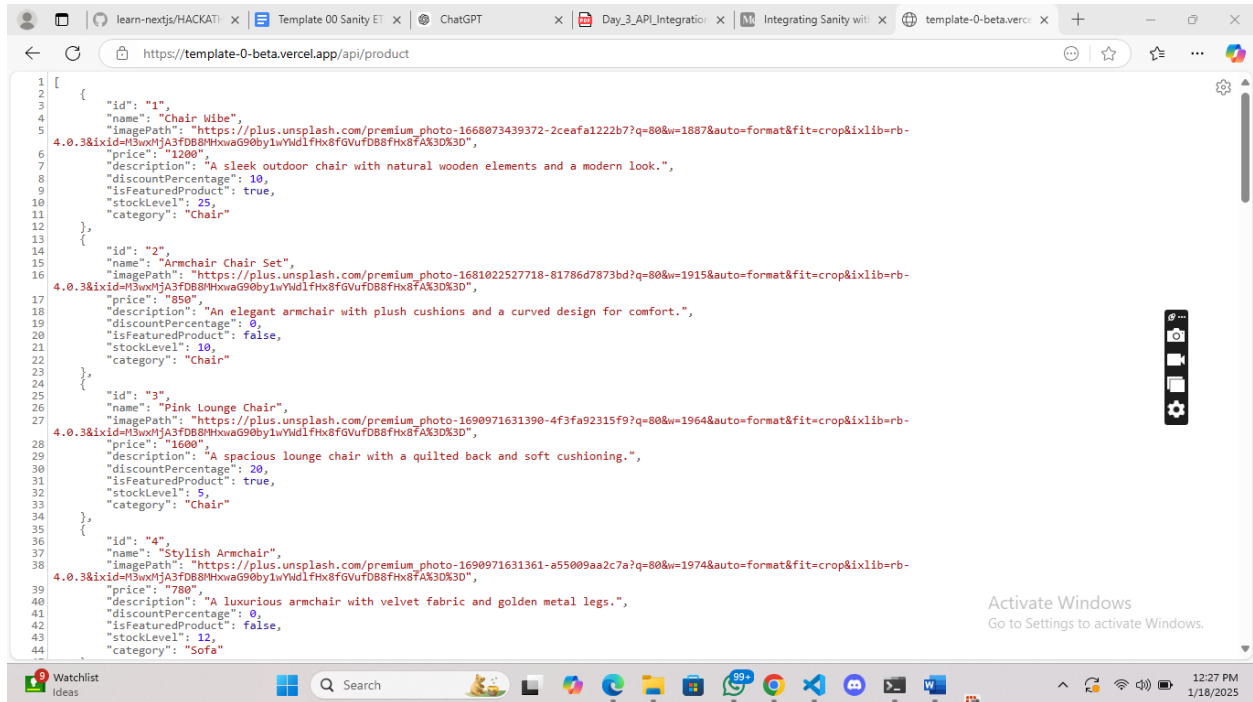
Migration Steps

1. **Fetching Product Data:** Used Axios to fetch product data from the API endpoint.
2. **Uploading Images to Sanity:** Images were fetched via the URL and uploaded to Sanity using `client.assets.upload()`.
3. **Creating Product Documents:** Once images were uploaded, I created product documents in Sanity using `client.create()`, linking each product to its respective image.
4. **Validation:** After the data was imported, I validated that each product's data was correctly displayed in the Sanity CMS and on the frontend.

Tools Used

- **Sanity Client:** To interact with the Sanity API and create product documents.
- **Axios:** For making HTTP requests to the external API and downloading images.
- **dotenv:** For managing environment variables securely (like Sanity API token).
- **Next.js:** For rendering the products on the frontend.

5. Screenshots



Products



Chair Wibe
A sleek outdoor chair with natural wooden elements and a modern look.
Price: \$1200
Discount: 10%
Featured: Yes
Stock Level: 25
Category: Chair



Armchair Chair Set
An elegant armchair with plush cushions and a curved design for comfort.
Price: \$850
Discount: 0%
Featured: No
Stock Level: 10
Category: Chair



Nautilus Lounge Chair
A lounge chair with a shell-inspired design and premium upholstery.
Price: \$1450
Discount: 12%
Featured: Yes
Stock Level: 10
Category: Sofa

Activate Windows
Go to Settings to activate Windows.

6. Code Snippets for API Integration and Migration Scripts

API Integration Code Snippet (Fetching Data and Uploading Images)

```
File Edit Selection View Go ... projc
EXPLORER
  PROJE
    > next
    > .sanity
    > dist
    > node_modules
    > scripts
    > importSanityData.mjs
    > src
    > app
    > fonts
    > studio
    > favicon.ico
    > global.asax
    > layouts
    > page.txt
    > sanity
    > lib
    > schemaTypes
    > index.ts
    > products
    > env.ts
    > structure.ts
    > env.local
    > aslinter.json
    > glgignore
    > next-env.d.ts
    > next.config.mjs
    > package-lock.json
    > package.json
    > postcss.config.mjs
    > README.md
    > sanity.d.ts
    > sanity.config.ts
    > tailwind.config.ts
    > tsconfig.json
  OUTLINE
  TIMELINE
  master
  72°F
  Haze
  12:35 PM
  1/18/2025

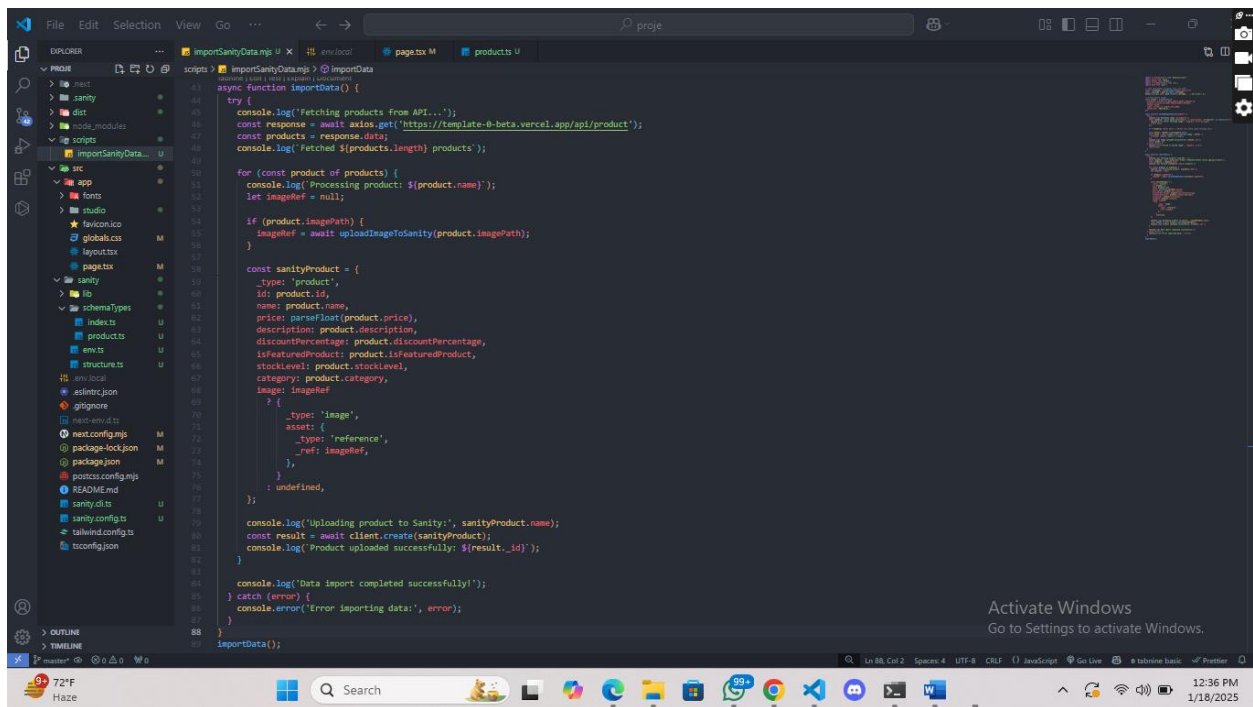
import { createClient } from '@sanity/client';
import axios from 'axios';
import dotenv from 'dotenv';
import { fileURLToPath } from 'url';
import path from 'path';

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

// Create Sanity client
const client = createClient({
  projectId: process.env.NEXT_PUBLIC_SANITY_PROJECT_ID,
  dataset: process.env.NEXT_PUBLIC_SANITY_DATASET,
  useCdn: false,
  token: process.env.SANITY_API_TOKEN,
  apiVersion: '2021-08-31',
});

// async function uploadImageToSanity(imageUrl) {
//   try {
//     console.log('Uploading image: ' + imageUrl);
//     const response = await axios.get(imageUrl, { responseType: 'arraybuffer' }).catch((error) => {
//       console.error('Error fetching image: ' + imageUrl, error.message);
//       return null;
//     });
//     if (!response) return null; // Handle case where image fetching fails
//     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 products from API...');
//     const response = await axios.get('https://template-0-beta.vercel.app/api/product');
//     const products = response.data;
//     for (const product of products) {
//       const imageUrl = product.imageUrl;
//       const assetId = await uploadImageToSanity(imageUrl);
//       if (assetId) {
//         console.log('Image ID: ' + assetId);
//       }
//     }
//   } catch (error) {
//     console.error('Error importing data: ' + error.message);
//   }
// }
```



This report documents the API integration process, schema adjustments, migration steps, and tools used for importing product data into Sanity CMS. It also provides code snippets for fetching API data and uploading product information. The project successfully integrates product data from an external API, handles image uploads, and renders the data in a Next.js frontend.