📝 Day 3 API Integration Report

Avion Furniture and Decor Website

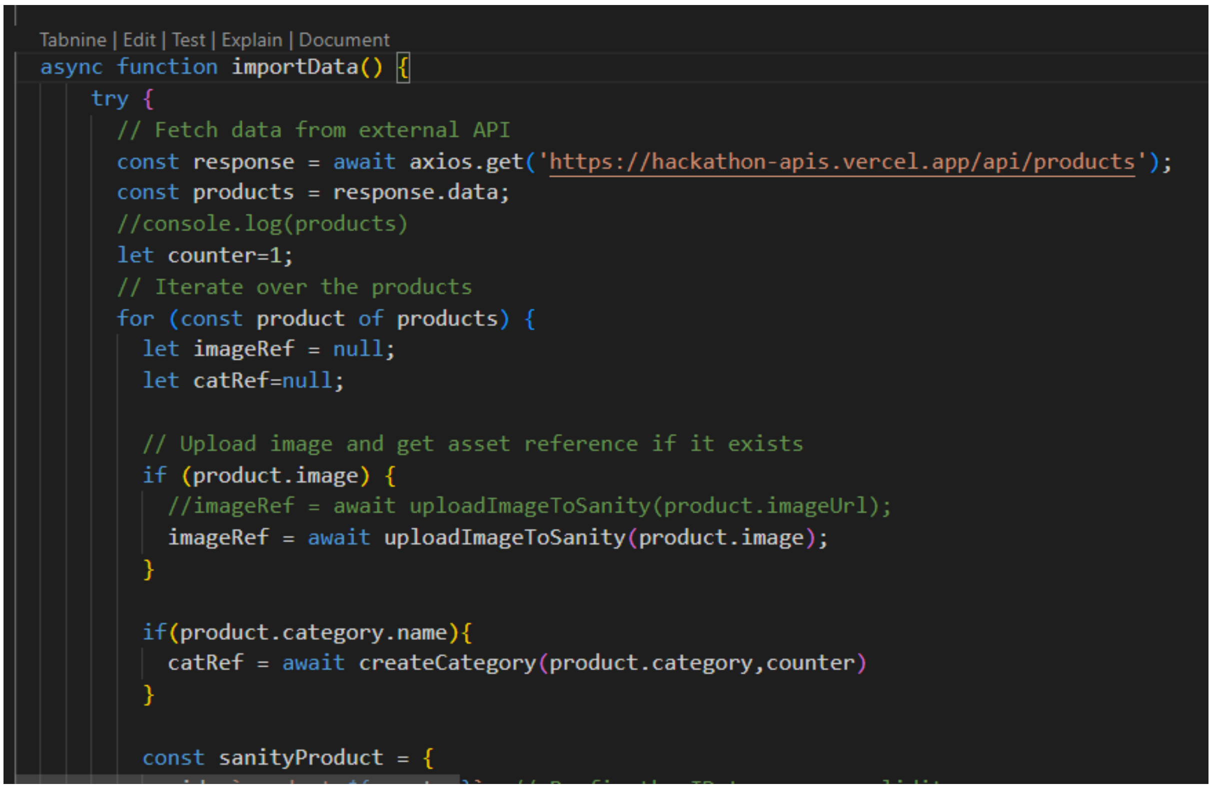
🎨 UI Development

We developed the **Avion Furniture and Decor** website UI with a focus on creating a visually appealing and user-friendly design. The technologies used include **Next.js**, **TypeScript**, and **Tailwind CSS**.

Our team leveraged **VS Code** as the integrated development environment (IDE), ensuring an efficient workflow. The **Thunder Client** extension within VS Code was particularly helpful for testing and validating API responses. This approach enabled us to deliver a modern, responsive interface aligned with the website's branding. ✨

🔗 API Integration Process

The API integration process was completed successfully, enabling dynamic content and efficient data handling for the website. The following steps outline the integration procedure:



```
Tabnine | Edit | Test | Explain | Document
async function importData() {
    try {
      // Fetch data from external API
      const response = await axios.get('https://hackathon-apis.vercel.app/api/products');
      const products = response.data;
      //console.log(products)
      let counter=1;
      // Iterate over the products
      for (const product of products) {
        let imageRef = null;
        let catRef=null;

        // Upload image and get asset reference if it exists
        if (product.image) {
          //imageRef = await uploadImageToSanity(product.imageUrl);
          imageRef = await uploadImageToSanity(product.image);
        }

        if(product.category.name){
          catRef = await createCategory(product.category,counter)
        }

        const sanityProduct = {
```

Migration Setup and Tools Used 🛠️

Environment Setup 🌐

- Installed required dependencies using npm install 📦

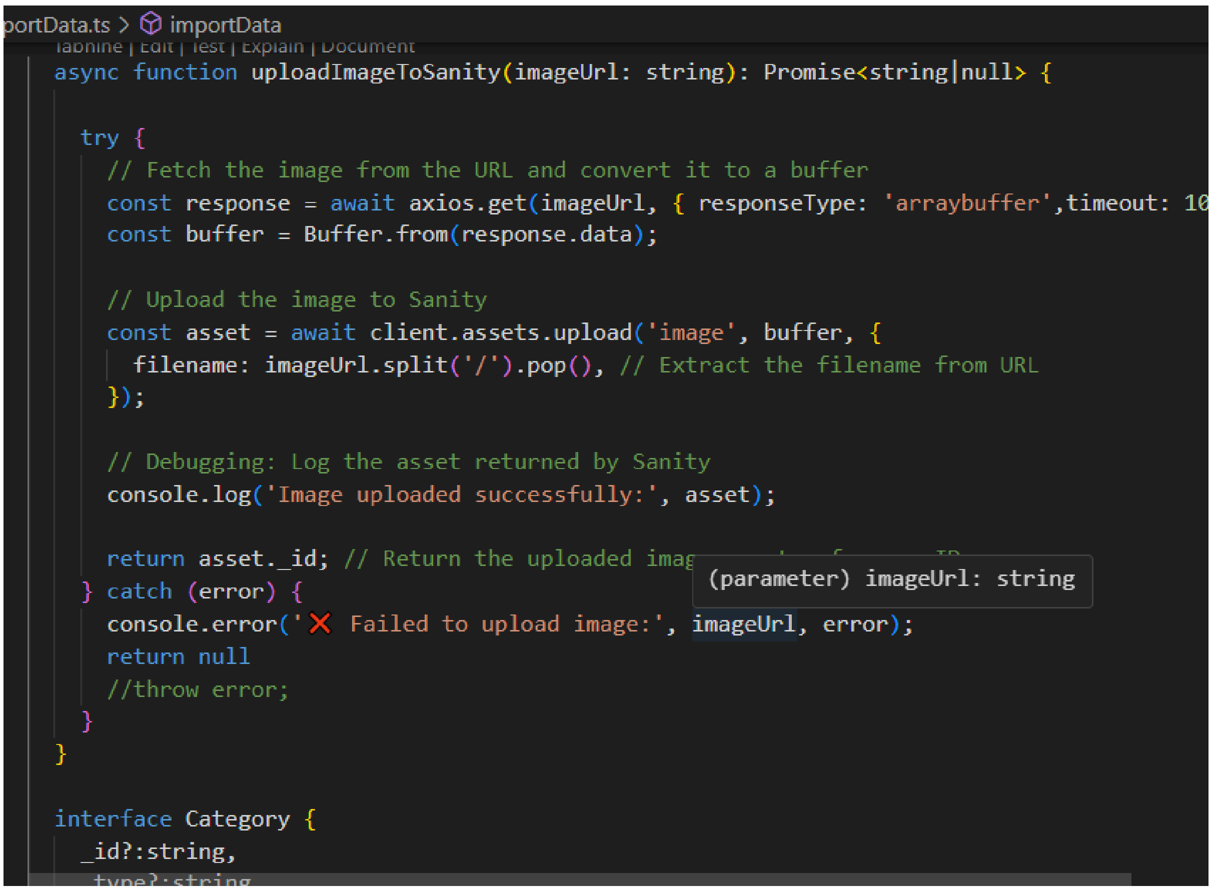
- Installed dotenv package using npm install dotenv 🔒

- Created a .env file to secure environment variables 📝

Data Fetching 📊

- Retrieved product data from API using Axios library 📈

- Parsed and logged the data to confirm its structure and integrity 🔍



```
portData.ts > importData
Tabnine | Edit | Test | Explain | Document
async function uploadImageToSanity(imageUrl: string): Promise<string|null> {

  try {
    // Fetch the image from the URL and convert it to a buffer
    const response = await axios.get(imageUrl, { responseType: 'arraybuffer',timeout: 10
    const buffer = Buffer.from(response.data);

    // Upload the image to Sanity
    const asset = await client.assets.upload('image', buffer, {
      filename: imageUrl.split('/').pop(), // Extract the filename from URL
    });

    // Debugging: Log the asset returned by Sanity
    console.log('Image uploaded successfully:', asset);

    return asset._id; // Return the uploaded imag
                                        (parameter) imageUrl: string
  } catch (error) {
    console.error('❌ Failed to upload image:', imageUrl, error);
    return null
    //throw error;
  }
}

interface Category {
  _id?:string,
  type?:string
```

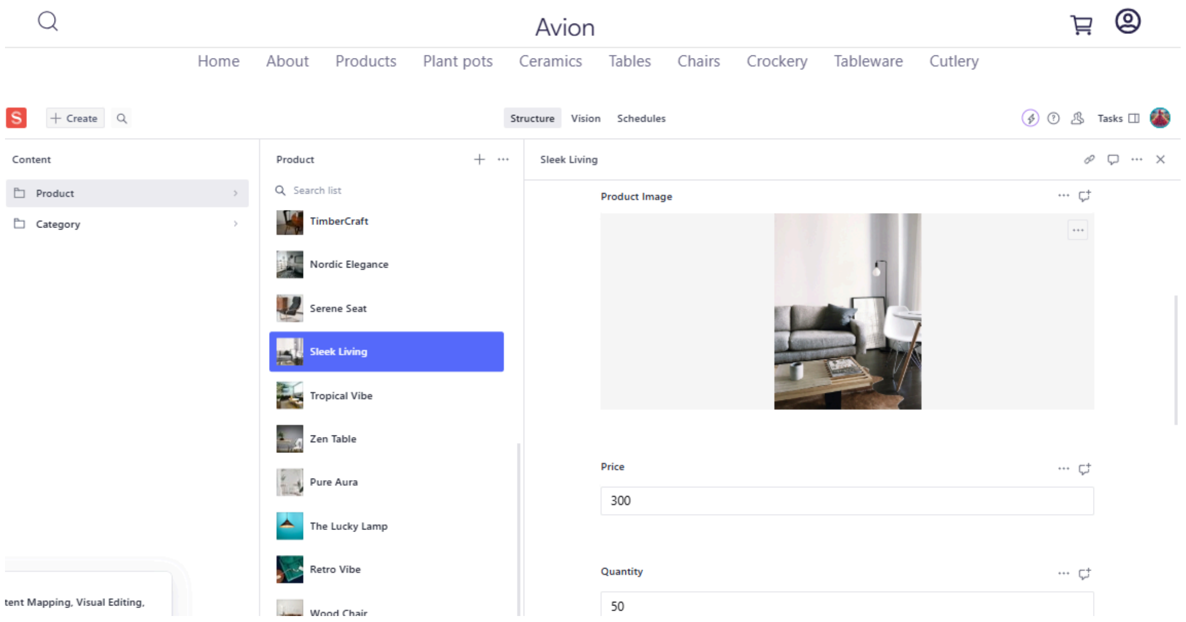
Tools Used 🧰

- Axios Library: Used for interacting with the API 📱

- Thunder Client Extension: Used for checking the API 🔍

- VS Code: Used as the Integrated Development Environment (IDE) 💻

Integrating Fetched Data into Sanity CMS 📊



Product

- TimberCraft
- Nordic Elegance
- Serene Seat
- Sleek Living
- Tropical Vibe
- Zen Table
- Pure Aura
- The Lucky Lamp
- Retro Vibe
- Wood Chair

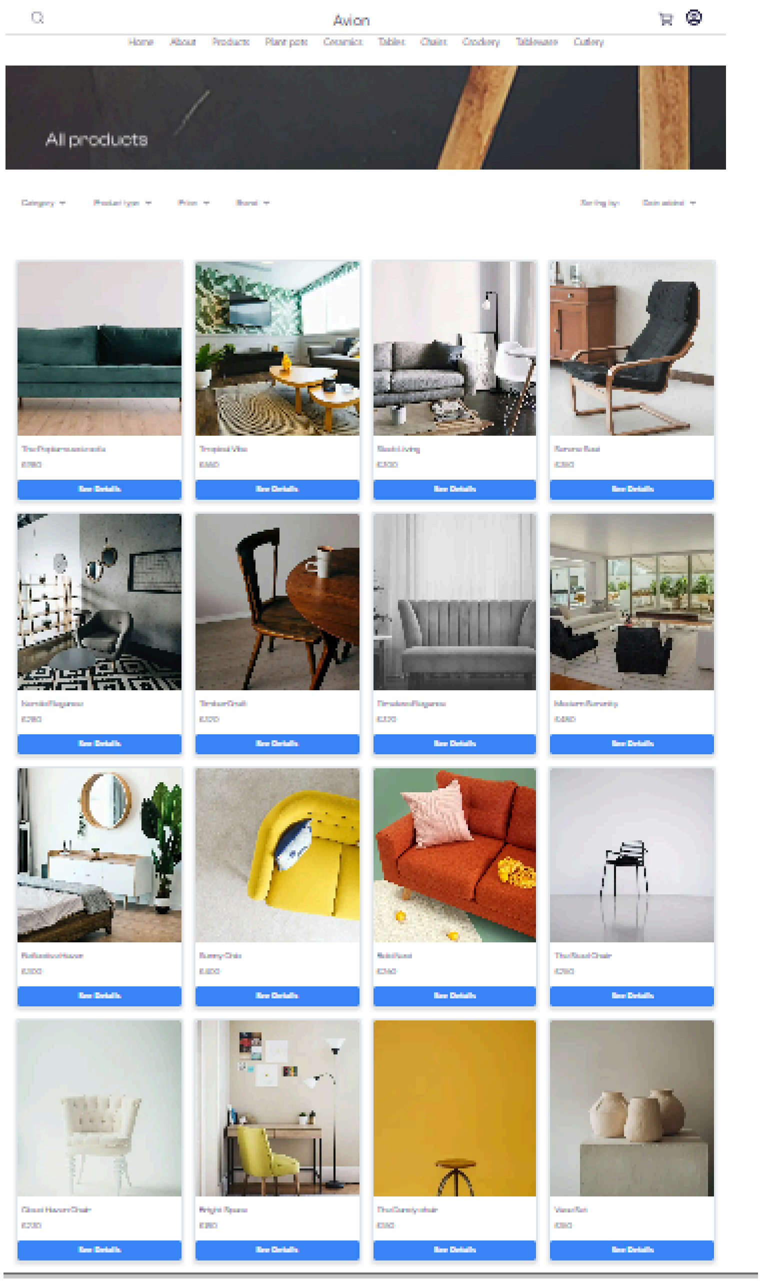
Sleek Living

Product Image

Price: 300

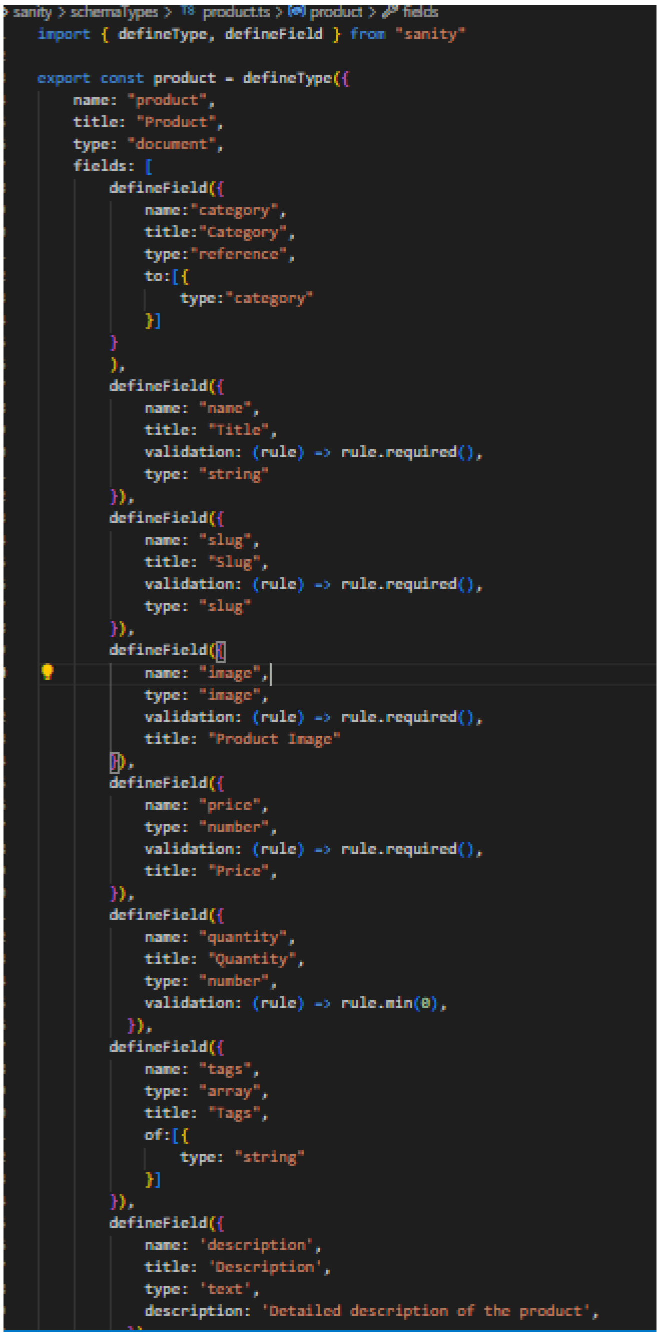
Quantity: 50

Fetched Data Visualization 📊



Define Schemas:

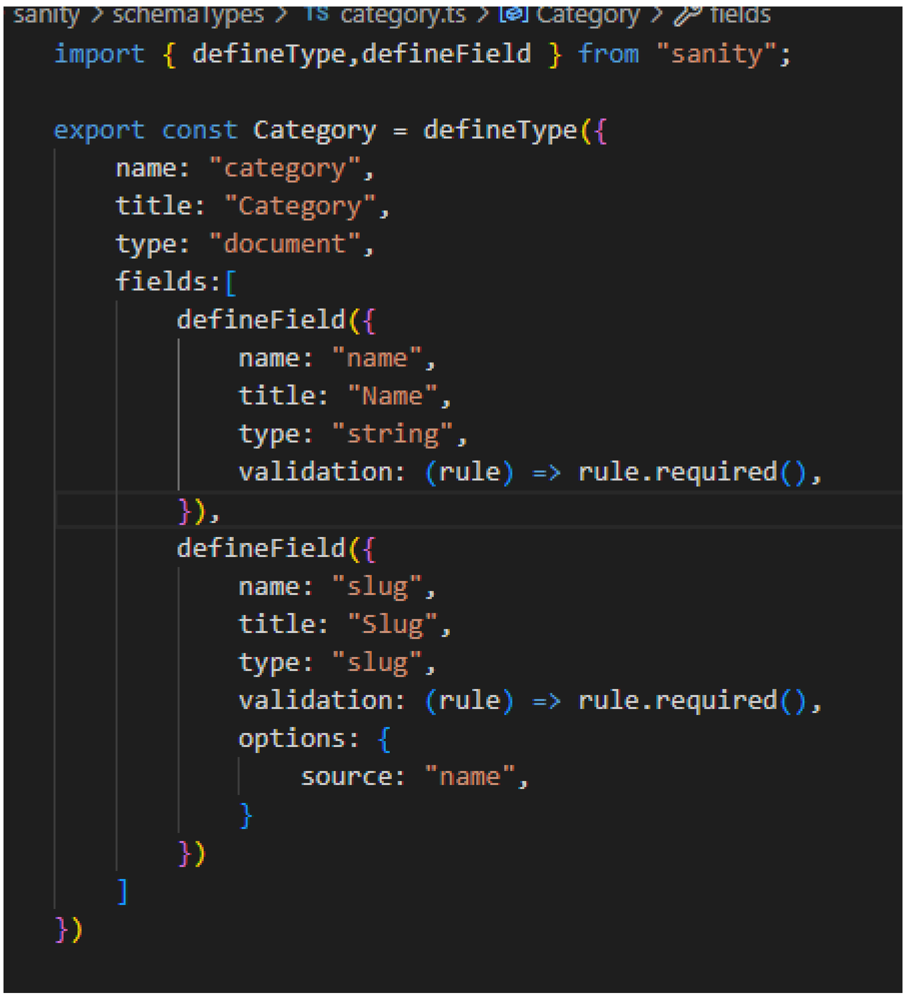
1.Product Schema:



```
import { defineType, defineField } from "sanity"

export const product = defineType({
    name: "product",
    title: "Product",
    type: "document",
    fields: [
        defineField({
            name:"category",
            title:"Category",
            type:"reference",
            to:[{
                type:"category"
            }]
        }
        ),
        defineField({
            name: "name",
            title: "Title",
            validation: (rule) => rule.required(),
            type: "string"
        }),
        defineField({
            name: "slug",
            title: "Slug",
            validation: (rule) => rule.required(),
            type: "slug"
        }),
        defineField({
            name: "image",
            type: "image",
            validation: (rule) => rule.required(),
            title: "Product Image"
        }),
        defineField({
            name: "price",
            type: "number",
            validation: (rule) => rule.required(),
            title: "Price",
        }),
        defineField({
            name: "quantity",
            title: "Quantity",
            type: "number",
            validation: (rule) => rule.min(0),
          }),
        defineField({
            name: "tags",
            type: "array",
            title: "Tags",
            of:[{
                type: "string"
            }]
        }),
        defineField({
            name: 'description',
            title: 'Description',
            type: 'text',
            description: 'Detailed description of the product',
```

1.Category Schema:



sanity > schemaTypes > TS category.ts > Category > fields

```
import { defineType,defineField } from "sanity";

export const Category = defineType({
    name: "category",
    title: "Category",
    type: "document",
    fields:[
        defineField({
            name: "name",
            title: "Name",
            type: "string",
            validation: (rule) => rule.required(),
        }),
        defineField({
            name: "slug",
            title: "Slug",
            type: "slug",
            validation: (rule) => rule.required(),
            options: {
                source: "name",
            }
        })
    ]
})
```

🌟 Conclusion

The successful completion of the UI development, API integration, and migration steps highlights significant progress in the project. By adhering to secure coding practices like using `.env` files and implementing robust error handling, we ensured the system is both scalable and reliable.

This milestone reflects our commitment to delivering high-quality, secure, and efficient solutions. 🚀

Prepared by :

Erum Baby Waris (Senior Student)

Roll No:00422422