Hackathon Day 3: API Integration and Data Migration

On Day 3 of the hackathon, the primary focus was on API integration and migrating data into Sanity CMS to build a functional marketplace backend. Below is a detailed account of the steps I followed to complete the task successfully.

Step 1: Understanding the Task and APIs

I started by thoroughly reviewing the provided API documentation and endpoints. The key endpoints included data for product listings, categories, and other relevant details. This step was crucial to understand the data structure and its compatibility with my existing Sanity CMS schema.

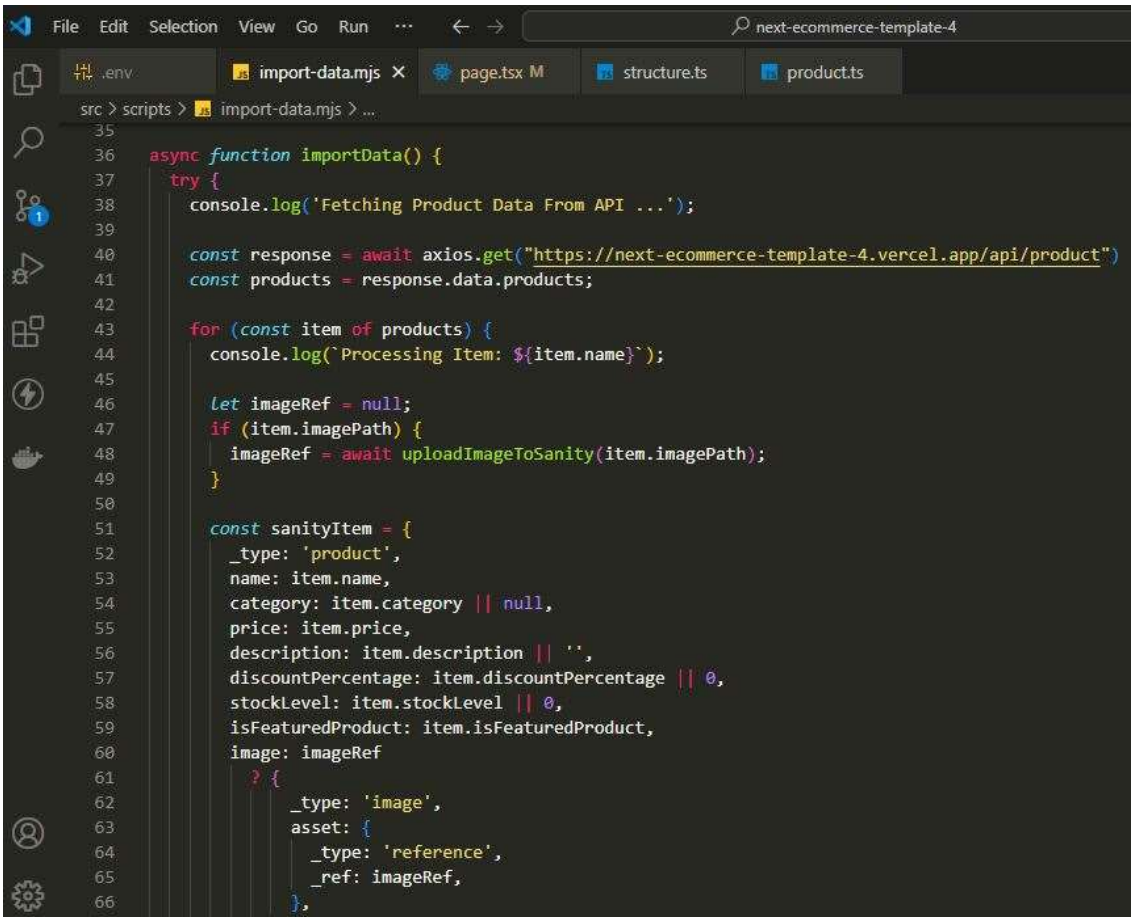
Step 2: Schema Validation and Adjustments

Next, I validated my Sanity CMS schema against the API data. This involved comparing field names and data types to ensure compatibility. For instance, fields like `name` in the API were mapped to `product_title` in my schema. Adjustments were made to align the schema with the incoming data.

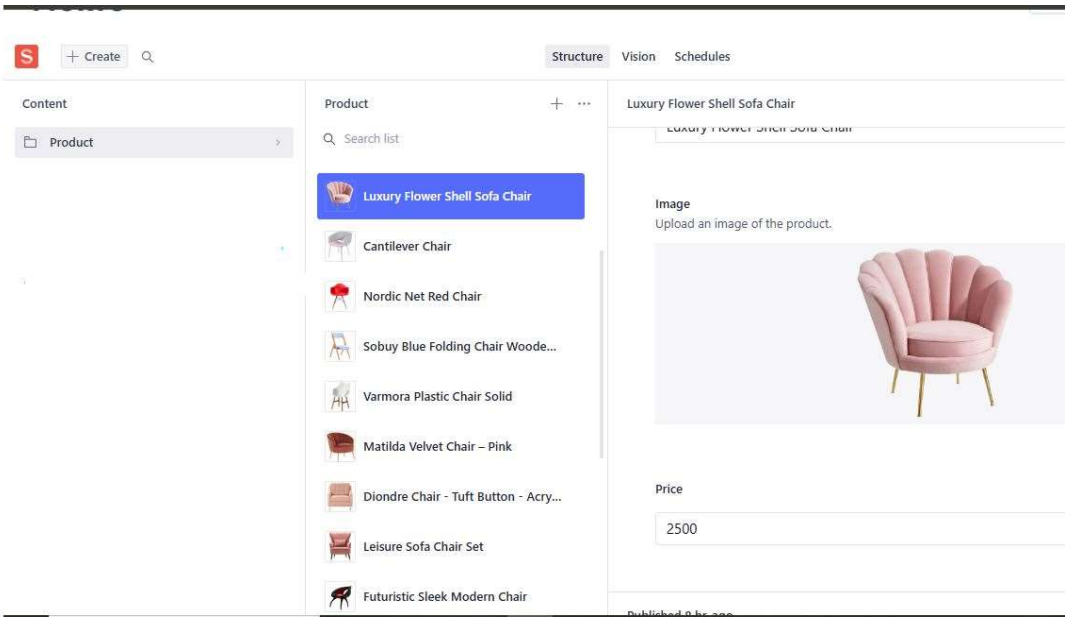
Step 3: Data Migration into Sanity CMS

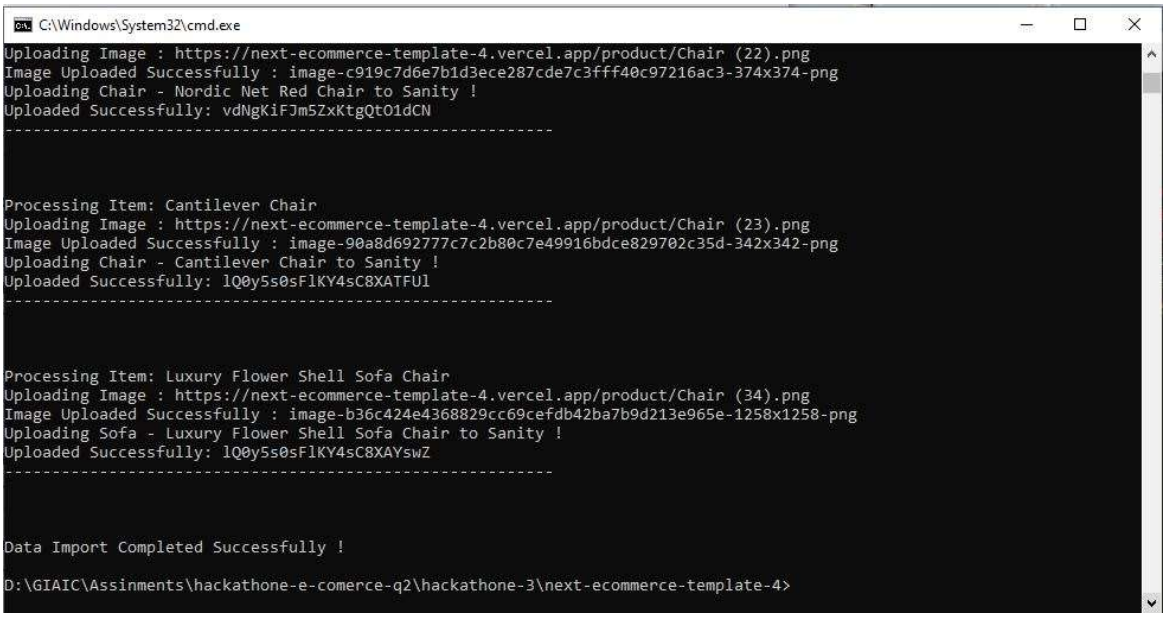
I migrated the data into Sanity CMS using the following approach:

- **.env file**: I copy my project Id and Token to .env file for security reasons that no one can access my Id or Token, this will ensure my data is safe.
- **Script-based Migration**: I clone the repo which was provided by our faculty there was a script to fetch data from the provided API and transform it to match my schema. This script ensured the data was imported accurately and efficiently.

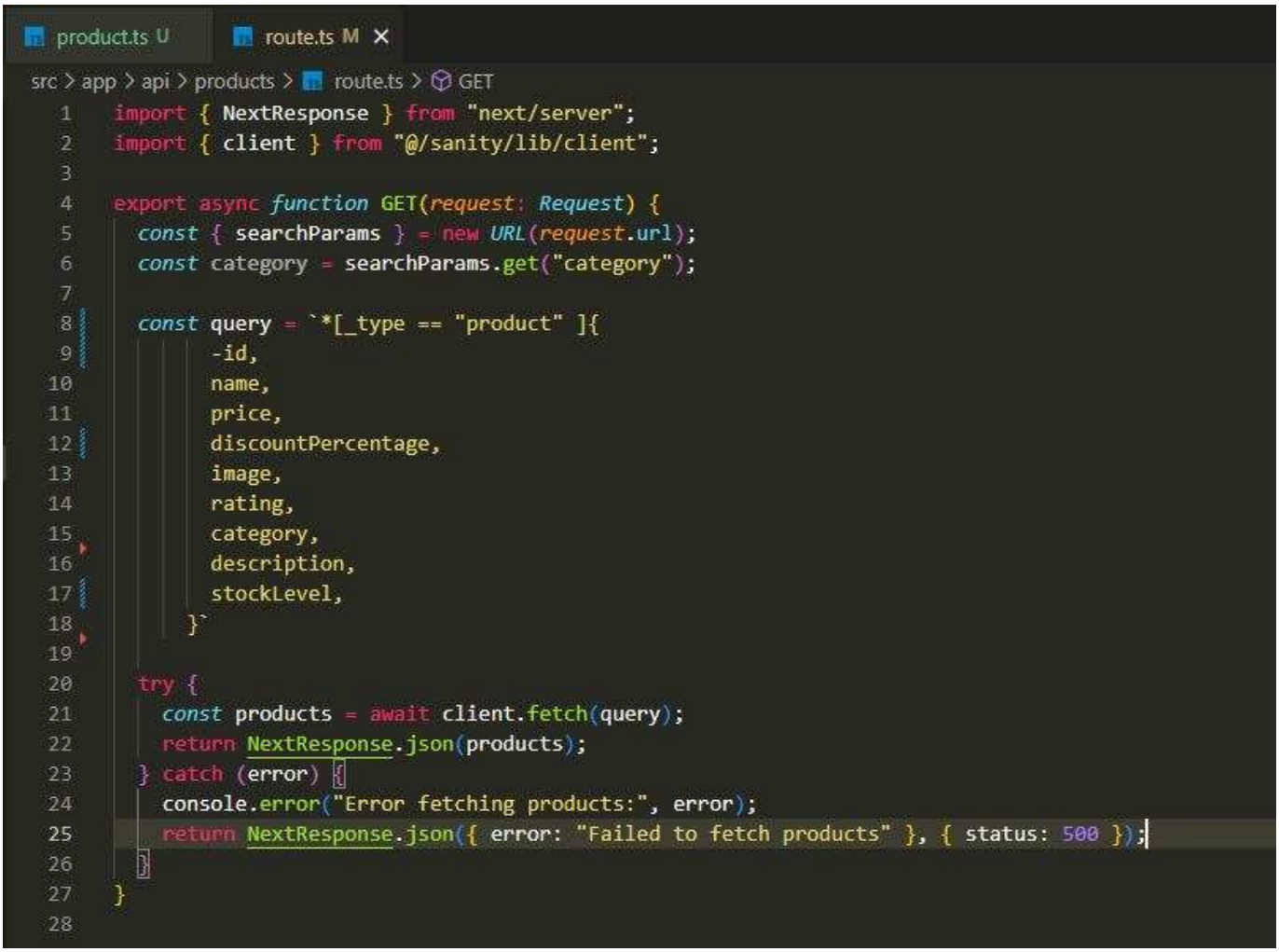


- **Validation**: After importing the data, I validated it within CMD and Sanity Studio to ensure all fields were correctly populated and aligned with the schema.


Create
Structure
Vision
Schedules
Content
Product
Product
Search list
Luxury Flower Shell Sofa Chair
Cantilever Chair
Nordic Net Red Chair
Sobuy Blue Folding Chair Woode...
Varmora Plastic Chair Solid
Matilda Velvet Chair – Pink
Diondre Chair - Tuft Button - Acry...
Leisure Sofa Chair Set
Futuristic Sleek Modern Chair
Luxury Flower Shell Sofa Chair
Image
Upload an image of the product.
Price
2500


C:\Windows\System32\cmd.exe
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: vdNgKiFJm5ZxKtgQtO1dCN
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: lQ0y5s0sFlKY4sC8XATFUl
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: lQ0y5s0sFlKY4sC8XAYswZ
Data Import Completed Successfully !
D:\GIAIC\Assinments\hackathone-e-comerce-q2\hackathone-3\next-ecommerce-template-4>

Step 4: API Integration in Next.js After completing the data migration, I focused on integrating the API into my Next.js project. This included:



1. **Creating Utility Functions**: I created reusable functions to fetch data from the API and Sanity CMS.
2. **Rendering Data**: The fetched data was displayed on the frontend, including product listings and categories.
3. **Error Handling**: Robust error-handling mechanisms were implemented to ensure a smooth user experience.

Step 5: Testing and Validation

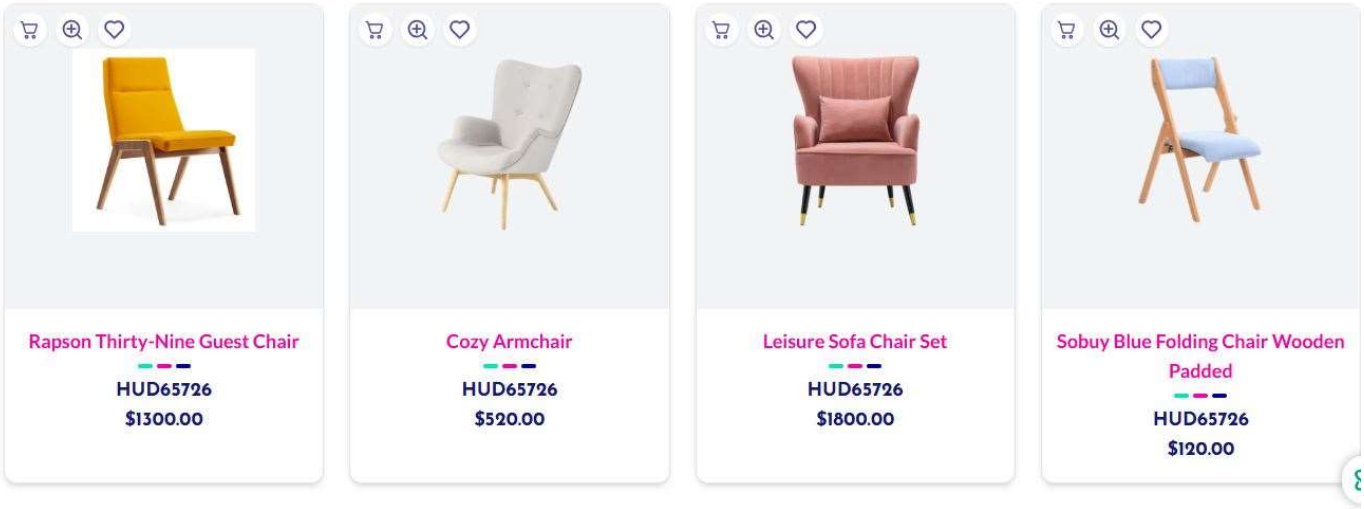
To ensure everything was working as expected:

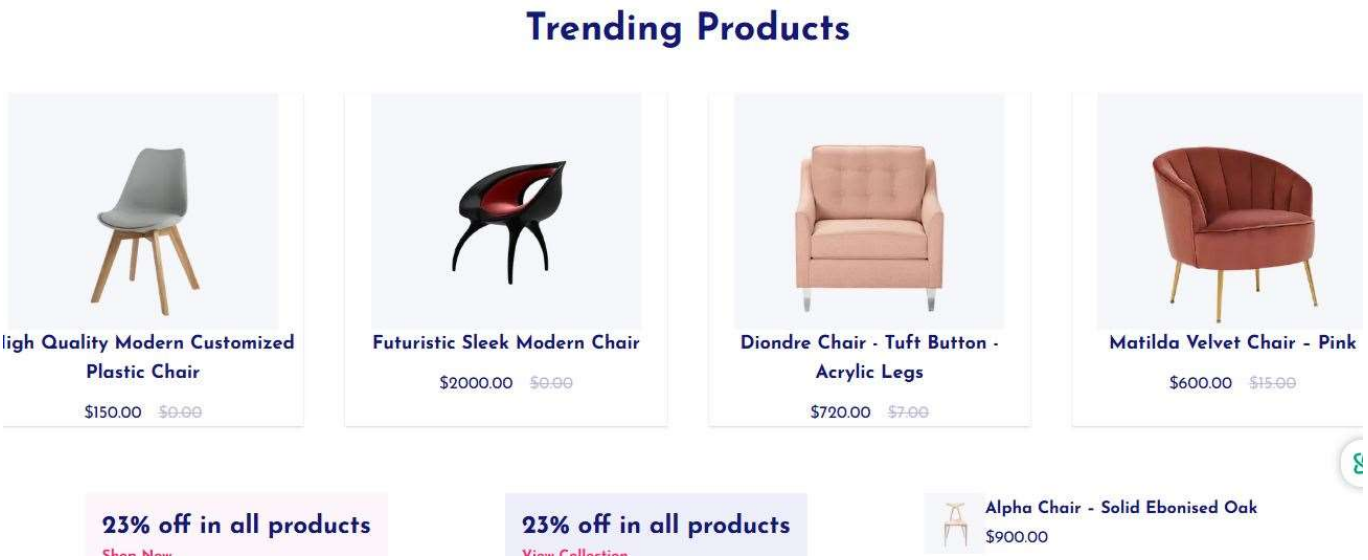
- I used tools like Postman to test API endpoints.
- I logged responses to debug issues and confirm data accuracy.
- I tested the frontend to verify the correct display of data.

Final Outcome

The project resulted in:

1. A fully populated Sanity CMS with imported data from the API.
2. A functional API integration in Next.js, with data dynamically rendered on the frontend.
3. Proper documentation of the entire process, including scripts, screenshots, and validation steps.





This exercise provided hands-on experience in integrating APIs and managing data migration, preparing me to handle real-world challenges in marketplace development.

Self-Validation Checklist

Tasks	✔	✗
API Understanding	✔	
Schema Validation	✔	
Data Migration	✔	
API Integration in Next.js	✔	
Submission Preparation	✔	