Day 3 - API Integration Report - Nike Shoes Store

Areeba Irfan   18-1-2025

Overview of Marketplace Website Integration and API Setup

This document outlines the process of integrating an API into the marketplace website, specifically designed to function as a dynamic e-commerce platform, inspired by Nike's website.

Objective and Approach

The website's core functionality involves dynamically managing product data, such as names, images, and prices, which were initially hardcoded. To address this limitation, we implemented Sanity as the content management system, allowing us to store and retrieve product data efficiently. By using Groq queries, we fetch data from Sanity to update the website's UI in real-time, ensuring that changes to product details are easily reflected on the site without requiring manual code updates.

In this document, I will provide a step-by-step guide on how the API is integrated into the website, showcasing the seamless transition from static data to a dynamic, scalable solution. This approach not only enhances flexibility but also improves the user experience by making it easier to manage product information.
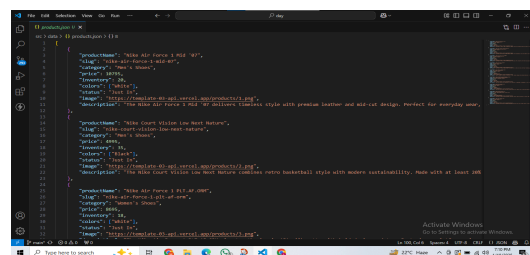

Objective:

- Integrating APIs into your Nike Shoes Website
- .Migrating data into Sanity CMS.
- Ensuring compatibility between APIs and the CMS schema.
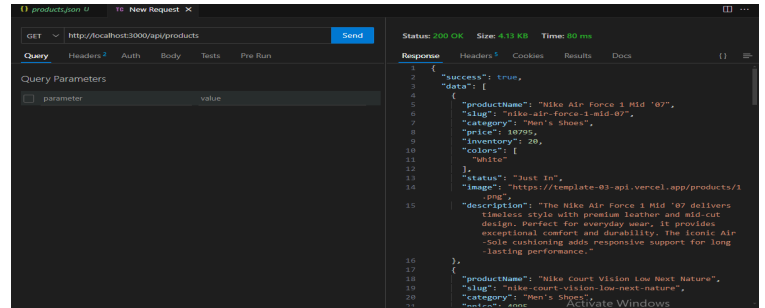
---

3. API Integration Process

Folder Structure and API Setup:

- Open the website project directory.
- Create a `data` folder.
- Add a `product.ts` file inside the `data` folder.
- Configure the API inside the `product.ts` file.

Testing the API:

Test the API endpoints using Postman to verify the responses and data integrity.

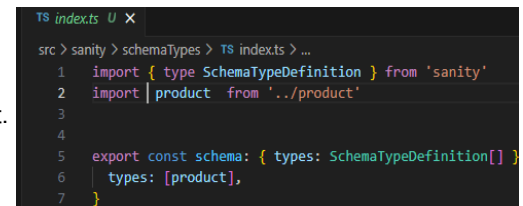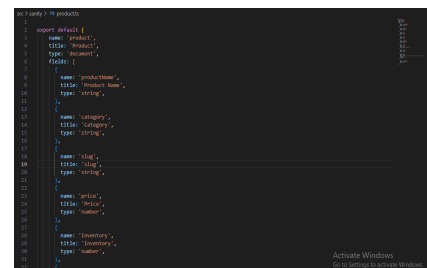## Schema Adjustments

Modifications:

- Created a schema in Sanity CMS.
- Adjusted the schema to match the API data structure.

Additions:

- Added fields like `slug` to enable dynamic data fetching.

Importing Schema:

imported the `products.ts` file into the main schemafile in the Sanity project.

Data Migration

Method Used:

- Created a script named `importSanityData.js` to migrate data into Sanity.

Steps for Migration:

Install required dependencies:

npm install @sanity/client axios dotenv

Run the import script:
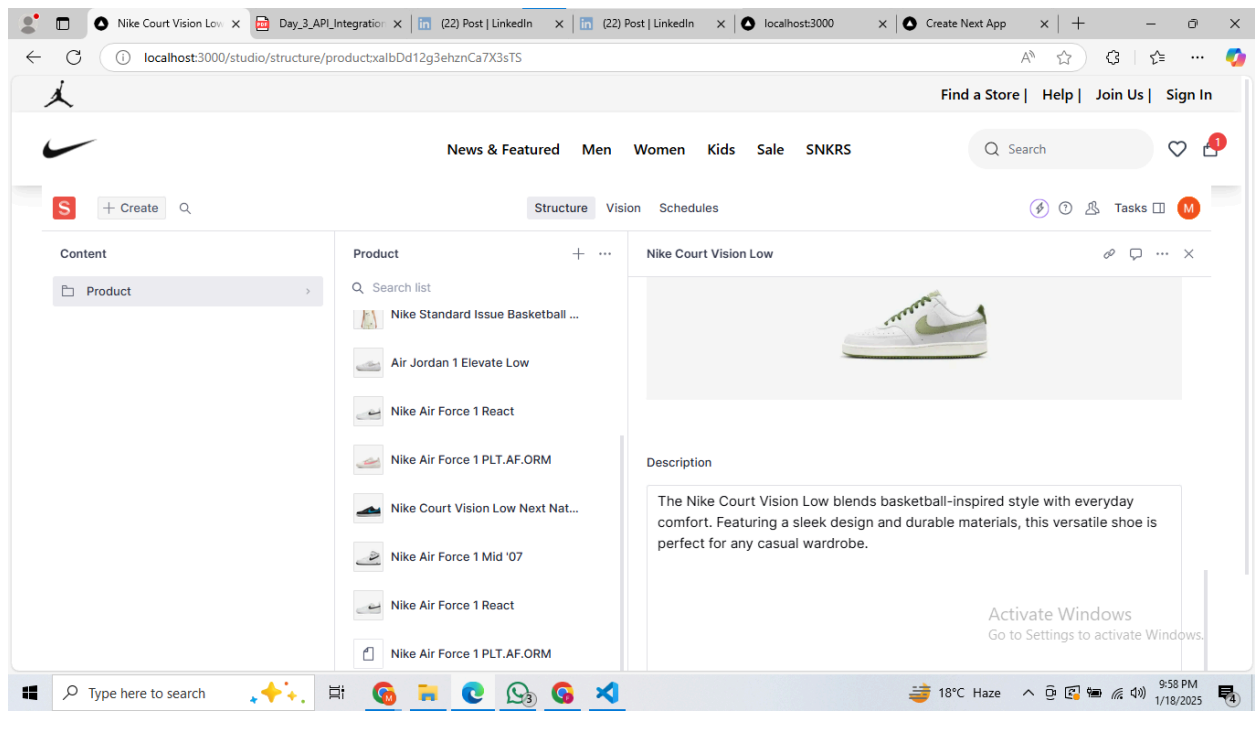
"npm run import-data"
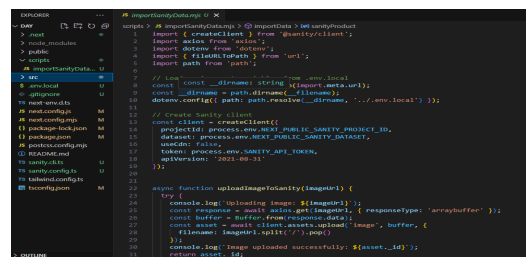
Verify the data is imported into Sanity CMS.

Result:

The product data is successfully migrated into Sanity CMS, and GROQ queries are used to fetch the data in the project.
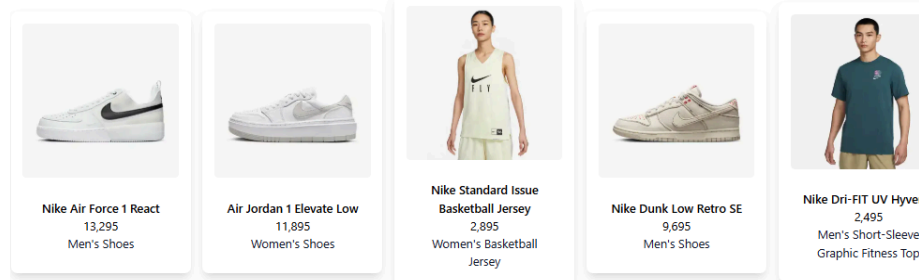

1. Fetched product data using the API.

After you run the command on the terminal your api data will be import on the sanity Through Groq query you will be able to fetch the data in to the project

6. Testing and Validation

API Testing:

- Tools Used: Postman.
- Verified successful API calls and responses.

Validation Checklist:

- Populated Sanity CMS? ✔️
- Data correctly displayed on the frontend? ✔️
- Errors handled gracefully? ✔️

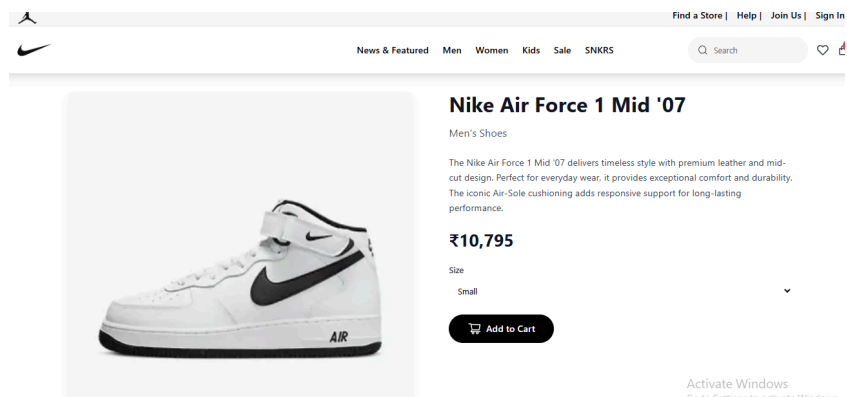## 7. Result and Output

Expected Output:

1. **Data in Sanity CMS:**
   - Products with fields like:
     - Name
     - Price
     - Category
     - Image
2. **Frontend Display:**
   - Product listings, categories, and details pages working as intended.

Best Of Air Max

Shop ‹ ›

Nike Court Vision Low Next Nature
4,995
Men's Shoes

Nike Air Force 1 Mid '07
10,795
Men's Shoes

Nike Air Force 1 Mid '07
10,795
Men's Shoes

Nike Air Force 1 PLT.AF.ORM
8,695
Women's Shoes

Nike Air Force 1 Reac
13,295
Men's Shoes