

DAY 3 - API INTEGRATION AND DATA MIGRATION

Objective

This report documents the integration of an external API into the e-commerce website built with Next.js and the migration of product data into Sanity CMS.

1. API Integration Process

The external API was integrated to fetch product data, with the following steps:

- **API Connection:** Configured API endpoints to retrieve product data such as names, descriptions, prices, and images.
- **Data Fetching:** Utilized Next.js for fetching API data and rendering it dynamically on the frontend.
- **Frontend Display:** Ensured that the fetched product data is displayed correctly on product pages.

Top selling



Casual Green Bomber Jacket
4.5/5
\$300 ~~\$375.00~~



Classic Black Straight-Leg Jeans
4.5/5
\$170 ~~\$202.30~~



Gray Slim-Fit Jogger Pants
4.5/5
\$170 ~~\$200.00~~



Sleeve Stripe T-Shirt
4.5/5
\$130 ~~\$216.67~~

[View All](#)

2. Schema Adjustments

The following schema adjustments were made in Sanity to accommodate the external API data:

- **Product Schema:** Added fields such as product name, price, description, and image to match the structure of the external API data.
- **User Schema:** Adjusted to handle user data linked with products, as necessary.

```
src > sanity > schemaTypes > TS product.ts > [x] default > fields
1  import { defineType } from "sanity"
2
3  export default defineType({
4    name: 'products',
5    title: 'Products',
6    type: 'document',
7    fields: [
8      {
9        name: 'name',
10       title: 'Name',
11       type: 'string',
12     },
13     {
14       name: 'price',
15       title: 'Price',
16       type: 'number',
17     },
18     {
19       name: 'description',
20       title: 'Description',
21       type: 'text',
22     },
23     {
24       name: 'image',
25       title: 'Image',
26       type: 'image',
27       options: {
28         hotspots: true
29       }
30     }
31   ]
32 })
```

3. Migration Steps and Tools

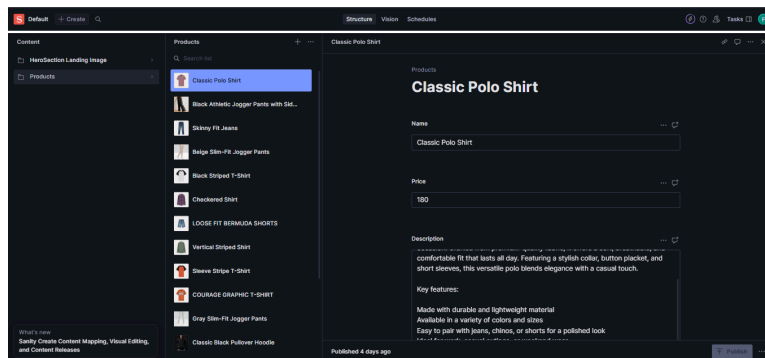
The migration of product data from the external API into Sanity CMS was performed with the following steps:

1. **Data Fetching:** API data was fetched dynamically in Next.js during the build or at runtime (depending on the requirements).
2. **Sanity Integration:** Utilized Sanity's JavaScript client to programmatically populate the Sanity CMS fields with the fetched API data.

3. **Data Population:** The fetched data was inserted into the corresponding product fields in Sanity.

Tools Used:

- **Sanity API Client** (for interacting with Sanity CMS)
- **Next.js** (for API integration and server-side data fetching)



```
src > sanity > lib > TS client.ts > getProducts
1  import { createClient } from "next-sanity";
2
3  export const client = createClient({
4    projectId: process.env.NEXT_PUBLIC_SANITY_PROJECT_ID,
5    dataset: process.env.NEXT_PUBLIC_SANITY_DATASET,
6    apiVersion: "2023-01-01",
7    useCdn: false,
8    token: process.env.SANITY_API_TOKEN,
9  });
10
11  export async function getProducts() {
12    return client.fetch(`*[_type == "products"]{
13      _id,
14      name,
15      price,
16      description,
17      "imageUrl": image.asset->url,
18      category,
19      discountPercent,
20      "isNew": new,
21      colors,
22      sizes
23    }`);
24  }
25
```

Conclusion

The integration of the external API and the migration of product data into Sanity CMS using Next.js was successfully completed. The marketplace is now dynamically pulling product data and managing it efficiently in Sanity for easy content updates.