Day 3 of Hackathon: API Integration and Data Migration for Bandage Fashion Marketplace

Name: Muhammad Zaki

Overview

This document outlines the approach taken to integrate external API data, modify schemas, and migrate data into Sanity CMS for the Bandage fashion marketplace. The primary aim was to effortlessly import product data, optimize the CMS framework, and enhance marketplace functionality.

External API Integration

1. Identifying Key API Endpoints

We started by evaluating the external API providing product data, which included the following critical endpoints:

- **Product Information**: Contains essential details like title, price, description, and availability.
- **Product Images**: Endpoint dedicated to fetching product images.
- **Product Categories**: Includes details about the categorization of products like men's and women's clothing.

2. Developing API Fetch Functions

To retrieve data, we implemented fetch functions using JavaScript's `fetch()` method, enabling dynamic product data loading from the external API.

Sample Code:

```
useEffect(() => {
  fetchProducts()
    .then((data) => setProducts(data))
    .catch((error) => console.error('Error fetching products:', error));
}, []);
```

3. Embedding API Calls into the Project

API requests were integrated within the Next.js lifecycle methods, utilizing React hooks like `useEffect` to dynamically render product details on the frontend.

Example for the Product Details Page:

- Displays product title, price, and inventory status.
- Adds a fallback mechanism for handling errors during data fetching.

Error Handling:

To improve the user experience, we implemented robust error handling during API calls:

```
if (loading) return <div>Loading...</div>;
if (error) return <div>Error: Unable to fetch products</div>;
```

Schema Modifications

Several schema updates were necessary within Sanity CMS to accommodate data from the external API.

1. Changes to the Product Schema

We added the following fields to the Product schema:

- **`priceWithoutDiscount`**: Numeric field to capture the original price before any discounts.
- **`tags`**: An array field for storing product labels like "featured" or "on sale."
- **`rating`**: A number field to store product ratings.

2. Enhancing the Category Schema

A reference field was added to the product schema to ensure that products are linked to their relevant categories, facilitating accurate product categorization.

3. Image Management

Product images were fetched from the external API and uploaded as assets to Sanity CMS. We utilized an asset reference field to link these images to corresponding products in the CMS.

Data Migration Process

The data migration involved transferring product data from the external API into Sanity CMS for centralized management.

1. Writing the Migration Script

A Node.js script was written using the Sanity client library to automate the transfer of product data into the CMS.

2. Migration Script Example

The script iterated through the product data from the external API, creating new documents in Sanity CMS for each product.

Sample Migration Script:

```
const sanityClient = require('@sanity/client');
const client = sanityClient({
  projectId: 'your-project-id',
  dataset: 'your-dataset',
  useCdn: false
});

const products = fetchProductData(); // Hypothetical function to fetch data

products.forEach(product => {
  client.create({
    _type: 'product',
    title: product.title,
    price: product.price,
    description: product.description,
    category: product.category,
    rating: product.rating,
    images: product.images
  });
});
```

Conclusion

The API integration and data migration process for Bandage’s fashion marketplace was successfully completed. The schema updates ensured full compatibility with the external product data, while the migration script effectively populated Sanity CMS with essential product information. As a result, the marketplace is now fully capable of dynamically displaying and managing product data, offering a streamlined shopping experience to its users.