Day-3 API Integration Report – Comforty

1. Introduction

The purpose of this report is to document the process of integrating external data from a REST API into the Sanity CMS, using the Sanity client for seamless content management. The integration process is aimed at automating the addition and management of products and categories in the backend, which will then be displayed in the Next.js frontend.

The core steps include:

- Fetching product and category data from an external API.
- Migrating this data to Sanity.
- Ensuring smooth synchronization for the frontend.

2. API Integration Overview

- **API Name**: The data is fetched from the external marketplace API hosted at the base URL: `https://giaic-hackathon-template-08.vercel.app`.
- **Data Fetched**: The API provides two key pieces of information:
  - **Products**: Product titles, prices, descriptions, categories, images, inventory, etc.
  - **Categories**: Categories associated with products.
- **Sanity Client**: The Sanity client is used to interact with the Sanity CMS and upload the fetched product and category data. This data is then structured and stored in the corresponding schemas within Sanity.

Code Snippet of Api.

```
// Load required environment variables
const {
  NEXT_PUBLIC_SANITY_PROJECT_ID, // Sanity project ID
  NEXT_PUBLIC_SANITY_DATASET, // Sanity dataset (e.g., "production")
  NEXT_PUBLIC_SANITY_AUTH_TOKEN, // Sanity API token
  BASE_URL = "https://giaic-hackathon-template-08.vercel.app", // API base URL for products and categories
} = process.env;

// Check if the required environment variables are provided
if (!NEXT_PUBLIC_SANITY_PROJECT_ID || !NEXT_PUBLIC_SANITY_AUTH_TOKEN) {
  console.error("Missing required environment variables. Please check your .env.local file.");
  process.exit(1); // Stop execution if variables are missing
}
```

1. Schema Structure Changes

The original `productSchema` was more basic and contained only a few essential fields like name, price, stock, and description. The updated schema expands significantly, adding more fields to capture detailed product information and enhance product management within the system.

The new schema includes the following additional fields:

- **title** (string): The product's title (formerly `name` in the old schema).
- **priceWithoutDiscount** (number): Price of the product without any discounts.
- **badge** (string): This could be used for displaying tags like "New", "Sale", or other promotional labels.
- **image** (image): Allows uploading a product image for each product.
- **category** (reference): A reference to the category document, which helps in organizing products within specific categories.
- **inventory** (number): Manages stock levels for the product (replaces the `stock` field in the old schema).
- **tags** (array): Allows tags to be associated with products. The tags include predefined options like "Featured", "Instagram", and "Gallery".

Snippet of new Product Schema.

```
export const productSchema = defineType({
  name: "products",
  title: "Products",
  type: "document",
  fields: [
    {
      name: "title",
      title: "Product Title",
      type: "string",
    },
    {
      name: "price",
      title: "Price",
      type: "number",
    },
    {
      title: "Price without Discount",
      name: "priceWithoutDiscount",
      type: "number",
    },
    {
      name: "badge",
      title: "Badge",
      type: "string",
    },
    {
      name: "image",
      title: "Product Image",
      type: "image",
    },
    {
      name: "category",
      title: "Category",
      type: "reference",
      to: [{ type: "categories" }],
    },
    {
      name: "description",
      title: "Product Description",
      type: "text",
    },
    {
      name: "inventory",
      title: "Inventory Management",
      type: "number",
    },
    {
      name: "tags",
      title: "Tags",
      type: "array",
      of: [{ type: "string" }],
      options: {
        list: [
```

Migration Steps and Tools Used

Tools Used

1. **Sanity Client**: Used for interacting with the Sanity CMS (via `@sanity/client`).
2. **Node.js & Fetch API**: Used for fetching data from an external API and uploading it to Sanity.
3. **Environment Variables**: For storing sensitive data like project ID, dataset, and API token (`dotenv`).
4. **Add Sanity domain for images**:

```
import type { NextConfig } from "next";

const nextConfig: NextConfig = {
  images: {
    domains: ['cdn.sanity.io'], // Add Sanity's image CDN
  },
};

export default nextConfig;
```

Migration Steps

1. **Set Up Environment**:
   - Install necessary dependencies (`@sanity/client`, `dotenv`).
   - Configure environment variables (`.env.local`) for secure access to API keys.
2. **Sanity Client Setup**:
   - Create a `targetClient` instance using the Sanity project ID and API token.
   - Configure it to interact with the Sanity dataset.
3. **Fetch Data from API**:
   - Use `fetch` to get product and category data from the external API.
4. **Upload Images to Sanity**:
   - If the API provides image URLs, use a helper function to upload images to Sanity and retrieve their asset IDs.
5. **Migrate Categories**:
   - Iterate through categories and upload them to Sanity, linking images (if available).
6. **Migrate Products**:
   - Iterate through products, upload their images, and reference their categories in Sanity.
7. **Verify Data**:
   - After migration, manually verify data in Sanity to ensure the migration is successful.

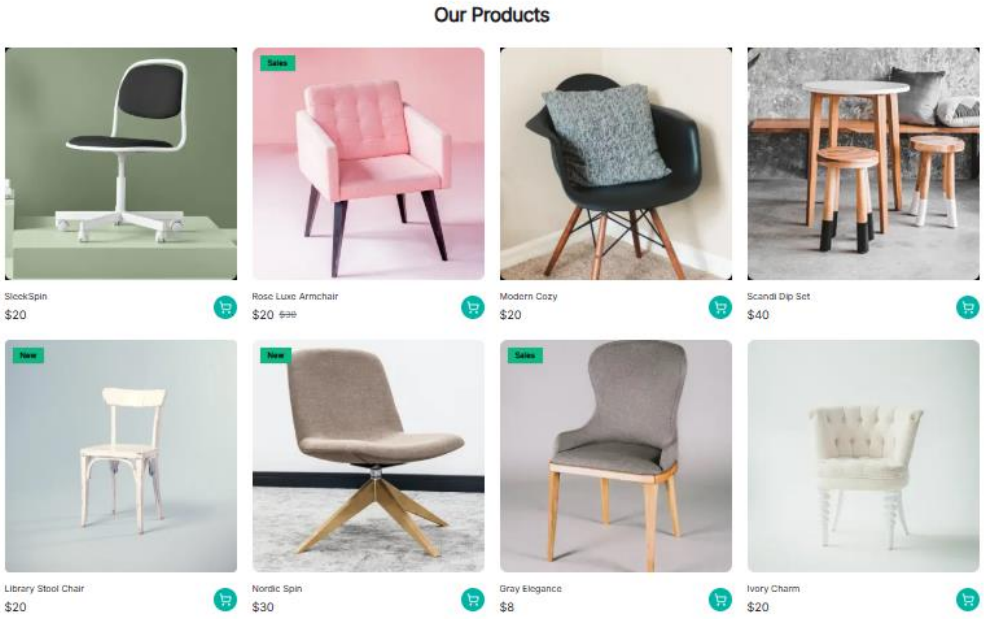
Api call response in console.

Snippet:


```
    _id: '9pIJ0O0PMKhFhzCfbMQLwZ'
  },
  {
    imageUrl: 'https://cdn.sanity.io/images/cqnnj0a4/production/f49ec164e3b62fbf8ad5d3b7cf0e5622478ee0ba-312x312.png',
    category: {
      _id: '407a8583-6203-4f61-becf-8e8b4c5461b6',
      title: 'Wooden Chair'
    },
    description: 'Lorem ipsum dolor sit amet, consectetur adipiscing elit. Nullam tincidunt erat enim. Lorem ipsum dolor
sit amet, consectetur adipiscing',
    tags: [ 'gallery' ],
    _id: '9pIJ0O0PMKhFhzCfbMQMFW',
    title: 'Scandi Dip Set',
    price: 40,
    priceWithoutDiscount: null,
    badge: null,
    inventory: 10
  },
  {
    priceWithoutDiscount: null,
    imageUrl: 'https://cdn.sanity.io/images/cqnnj0a4/production/81a5b7de166f930870a82f8f3e661b38a70de9f4-312x312.png',
    description: 'Lorem ipsum dolor sit amet, consectetur adipiscing elit. Nullam tincidunt erat enim. Lorem ipsum dolor
sit amet, consectetur adipiscing',
    inventory: 10,
    title: 'SleekSpin',
    price: 20,
    badge: null,
    category: {
      _id: 'b5710116-09af-4d0e-aa9a-dcd02fe919a9',
      title: 'Desk Chair'
```

Products fetched in next.js

Snippet of Products.



Category fetched in next.js

Snippet of Category.