EGovernor Sindh Initiative Ai course

Name: Mehwish Sheikh
Roll number: 223558

Hackathon Day:3

API INTEGRATION AND DATA MIGRATION

General E-Commerce Website:Comforty

Introduction:

This document provides a detailed overview of the API integration process, schema adjustments, and data migration into Sanity CMS for the Comforty Chair Marketplace. The objective was to establish seamless API interactions, incorporate external product data, and populate the CMS to ensure a functional and robust marketplace.

API Integration Process

1. API Integration

The API integration process involved fetching external product data and populating it into Sanity CMS. The following steps were followed:

1.1 Identifying API Endpoints

- The external API providing product data was analyzed, including:
    - Endpoints for product details (title, price, description, inventory, etc.).
    - Endpoints for product images and category data.

1.2 Creating API Fetch Functions

- Fetch functions were implemented using `fetch()` to retrieve data from the API.
- Example code snippet:
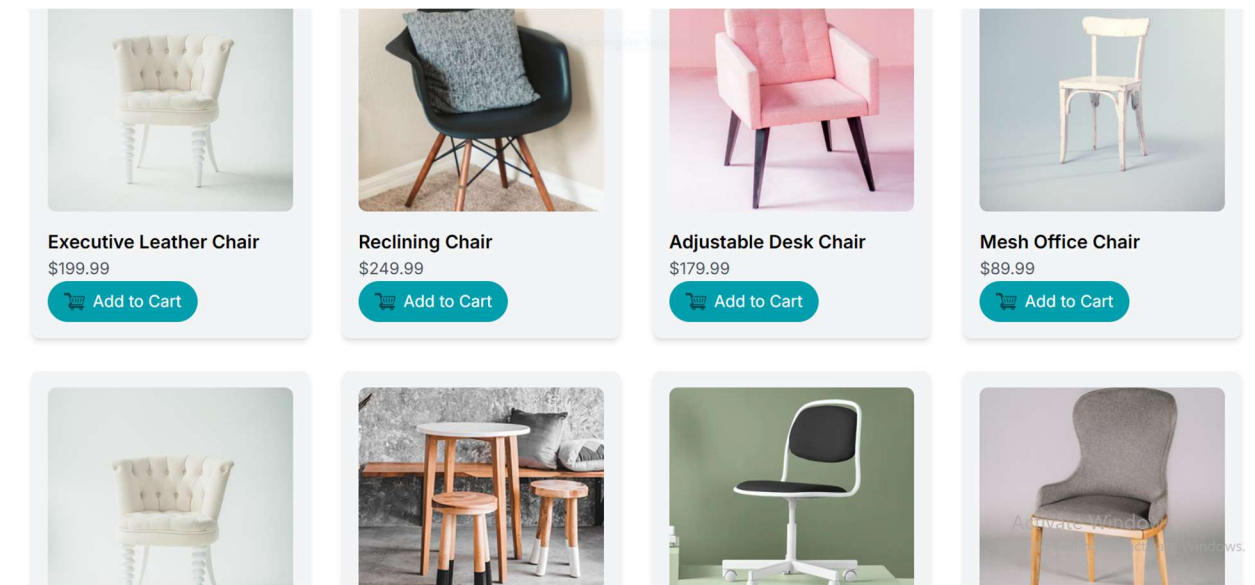
EGovernor Sindh Initiative Ai course **Name: Mehwish Sheikh**

Roll number: 223558

Hackathon Day:3
API INTEGRATION AND DATA MIGRATION
General E-Commerce Website:Comforty

```tsx
const ProductPage: React.FC = () => {
  const { addToCart } = useCart();
  const [products, setProducts] = useState<Product[]>([]);
  const [loading, setLoading] = useState<boolean>(false);
  const [error, setError] = useState<string | null>(null);

  Pieces: Comment | Pieces: Explain
  useEffect(() => {
    const fetchProducts = async () => {
      try {
        const res = await fetch('/api/products');
        if (!res.ok) {
          throw new Error('Failed to fetch products');
        }
        const data = await res.json();
        setProducts(data);
      } catch (error: any) {
        console.error('Error fetching products:', error);
        setError(error.message);
      } finally {
        setLoading(false);
      }
    };

    fetchProducts();
  }, []);
```

1.3 Integrating API Calls in the Project

- API calls were incorporated into Next.js lifecycle methods or React hooks (e.g., `useEffect`) to fetch and display data dynamically.
- Example usage on the **Product Detail Page**:
  - Displayed data such as product title, price, description, and inventory status.
  - Integrated a fallback mechanism for handling errors during API fetch.

Frontend Display of API Data

Hackathon Day:3

API INTEGRATION AND DATA MIGRATION

General E-Commerce Website:Comforty

Handling Data

- Parsed the fetched product data and rendered it dynamically on the frontend.
- Example:

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

Error Handling

- Implemented error handling for API calls to ensure a smooth user experience.
- Example fallback mechanism:

```javascript
CopyEdit
if (loading) return <div>Loading...</div>;
if (error) return <div>Error: Unable to fetch products</div>;
```
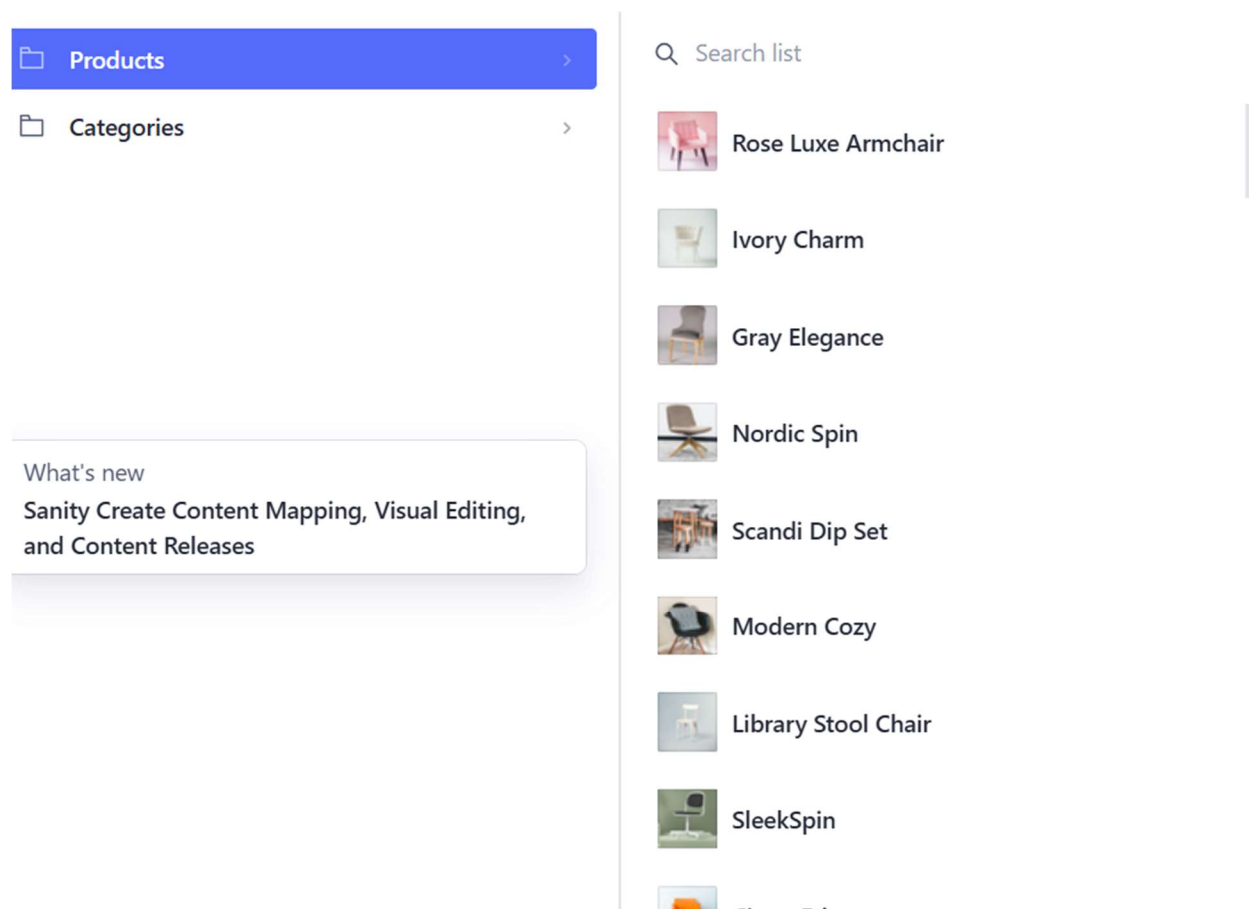
Hackathon Day:3

API INTEGRATION AND DATA MIGRATION

General E-Commerce Website:Comforty

Adjustments Made to Schemas

2. Schema Adjustments

To accommodate the external API data, the following schema changes were implemented in Sanity CMS:

Hackathon Day:3

API INTEGRATION AND DATA MIGRATION

General E-Commerce Website:Comforty

2.1 Product Schema

- **New fields added**:
  - `priceWithoutDiscount`: Numeric field for original product price before discounts.
  - `tags`: Array field for product tags like "featured" or "on sale."

Hackathon Day:3

API INTEGRATION AND DATA MIGRATION

General E-Commerce Website:Comforty

o   `rating`: Number field to store product ratings.

```javascript
import { defineType } from "sanity";

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",
```

EGovernor Sindh Initiative Ai course

Name: Mehwish Sheikh

Roll number: 223558

Hackathon Day:3

API INTEGRATION AND DATA MIGRATION

General E-Commerce Website:Comforty

```
        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: [
            { title: "Featured", value: "featured" },
            {
```

Hackathon Day:3
API INTEGRATION AND DATA MIGRATION
General E-Commerce Website:Comforty

```
      name: "tags",
      title: "Tags",
      type: "array",
      of: [{ type: "string" }],
      options: {
        list: [
          { title: "Featured", value: "featured" },
          {
            title: "Follow products and discounts on Instagram",
            value: "instagram",
          },
          { title: "Gallery", value: "gallery" },
        ],
      },
    },
  ],
});
```

2.2 Category Schema

- Added a **reference field** in the product schema to link each product to its respective category.
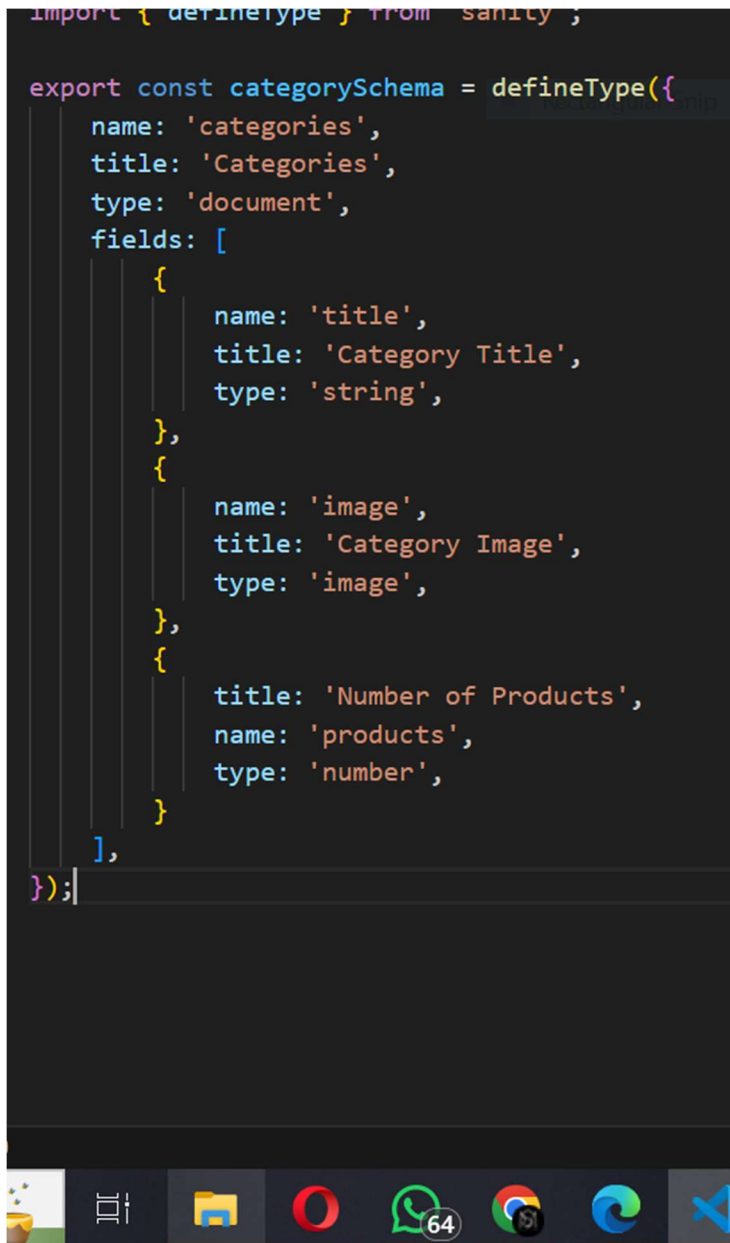- Updated schema snippet:

Hackathon Day:3

API INTEGRATION AND DATA MIGRATION

General E-Commerce Website:Comforty

```javascript
import { defineType } from 'sanity';

export const categorySchema = defineType({
    name: 'categories',
    title: 'Categories',
    type: 'document',
    fields: [
        {
            name: 'title',
            title: 'Category Title',
            type: 'string',
        },
        {
            name: 'image',
            title: 'Category Image',
            type: 'image',
        },
        {
            title: 'Number of Products',
            name: 'products',
            type: 'number',
        }
    ],
});
```
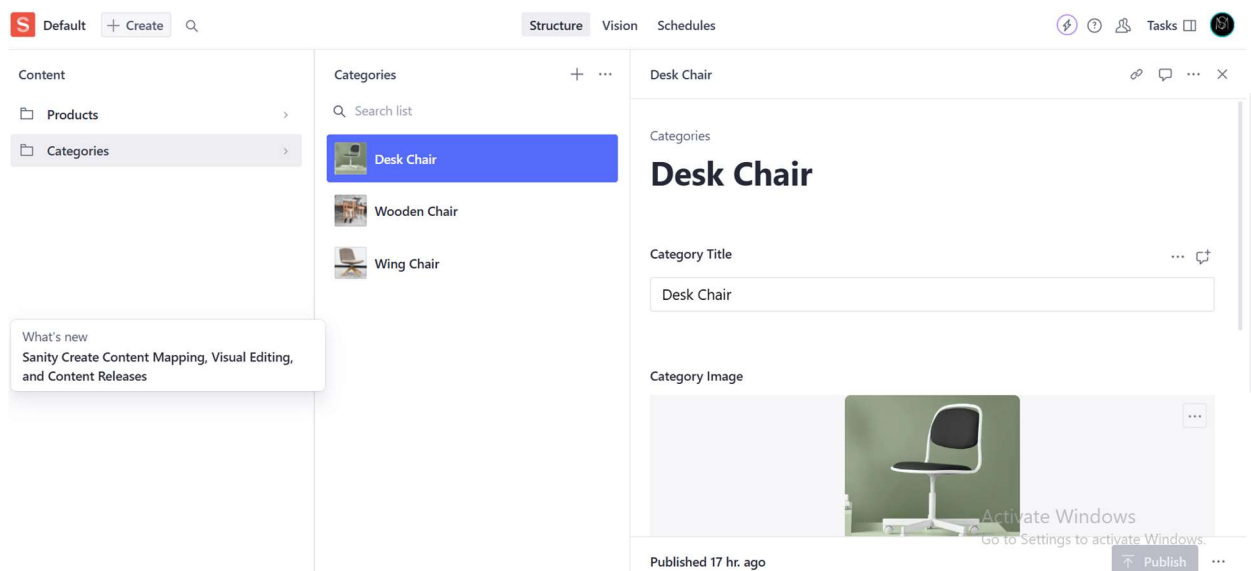
Hackathon Day:3
API INTEGRATION AND DATA MIGRATION
General E-Commerce Website:Comforty

2.3 Image Handling

- Product images fetched from the API were uploaded to Sanity CMS as image assets.
- Sanity's asset reference field was used to associate the images with their respective products.

Migration Steps and Tools Used

3. Data Migration

The migration process involved transferring product data from the external API into Sanity CMS.

3.1 Preparing the Migration Script

- A Node.js script was written to automate the migration process using Sanity's client library.

EGovernor Sindh Initiative Ai course
Name: Mehwish Sheikh

Roll number: 223558

Hackathon Day:3

API INTEGRATION AND DATA MIGRATION

General E-Commerce Website:Comforty

3.2 Script to Migrate Product Data

- The script iterated over the API data and created documents in Sanity CMS.

Sample Migration Script:

```javascript
// Import environment variables from .env.local
import "dotenv/config";

// Import the Sanity client to interact with the Sanity backend
import { createClient } from "@sanity/client";

// 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
}

// Create a Sanity client instance to interact with the target Sanity dataset
const targetClient = createClient({
  projectId: NEXT_PUBLIC_SANITY_PROJECT_ID, // Your Sanity project ID
  dataset: NEXT_PUBLIC_SANITY_DATASET || "production", // Default to "production" if not set
```

EGovernor Sindh Initiative Ai course

Name: Mehwish Sheikh

Roll number: 223558

Hackathon Day:3

API INTEGRATION AND DATA MIGRATION

General E-Commerce Website:Comforty

```javascript
const targetClient = createClient({
  projectId: NEXT_PUBLIC_SANITY_PROJECT_ID, // Your Sanity project ID
  dataset: NEXT_PUBLIC_SANITY_DATASET || "production", // Default to "production" if not set
  useCdn: false, // Disable CDN for real-time updates
  apiVersion: "2023-01-01", // Sanity API version
  token: NEXT_PUBLIC_SANITY_AUTH_TOKEN, // API token for authentication
});

// Function to upload an image to Sanity
// Pieces: Comment | Pieces: Explain
async function uploadImageToSanity(imageUrl) {
  try {
    // Fetch the image from the provided URL
    const response = await fetch(imageUrl);
    if (!response.ok) throw new Error(`Failed to fetch image: ${imageUrl}`);

    // Convert the image to a buffer (binary format)
    const buffer = await response.arrayBuffer();

    // Upload the image to Sanity and get its asset ID
    const uploadedAsset = await targetClient.assets.upload("image", Buffer.from(buffer), {
      filename: imageUrl.split("/").pop(), // Use the file name from the URL
    });

    return uploadedAsset._id; // Return the asset ID
  } catch (error) {
    console.error("Error uploading image:", error.message);
    return null; // Return null if the upload fails
  }
}
```

Activate Windows
Go to Settings to activate Windows.

EGovernor Sindh Initiative Ai course

Name: Mehwish Sheikh

Roll number: 223558

Hackathon Day:3

API INTEGRATION AND DATA MIGRATION

General E-Commerce Website:Comforty

```
    _type: "categories",
    title: category.title,
    image: imageId ? { _type: "image", asset: { _ref: imageId } } : undefined, // Add image if uploaded
  };

  // Save the category to Sanity
  const result = await targetClient.createOrReplace(newCategory);
  categoryIdMap[category._id] = result._id; // Store the new category ID
  console.log(`Migrated category: ${category.title} (ID: ${result._id})`);
}

// Migrate products
for (const product of productsData) {
  console.log(`Migrating product: ${product.title}`);
  const imageId = await uploadImageToSanity(product.imageUrl); // Upload product image

  // Prepare the new product object
  const newProduct = {
    _type: "products",
    title: product.title,
    price: product.price,
    priceWithoutDiscount: product.priceWithoutDiscount,
    badge: product.badge,
    image: imageId ? { _type: "image", asset: { _ref: imageId } } : undefined, // Add image if uploaded
    category: {
      _type: "reference",
      _ref: categoryIdMap[product.category._id], // Use the migrated category ID
    },
```

Hackathon Day:3

API INTEGRATION AND DATA MIGRATION

General E-Commerce Website:Comforty

```
      _type: "categories",
      title: category.title,
      image: imageId ? { _type: "image", asset: { _ref: imageId } } : undefined, // Add image if uploaded
    };

    // Save the category to Sanity
    const result = await targetClient.createOrReplace(newCategory);
    categoryIdMap[category._id] = result._id; // Store the new category ID
    console.log(`Migrated category: ${category.title} (ID: ${result._id})`);
  }

  // Migrate products
  for (const product of productsData) {
    console.log(`Migrating product: ${product.title}`);
    const imageId = await uploadImageToSanity(product.imageUrl); // Upload product image

    // Prepare the new product object
    const newProduct = {
      _type: "products",
      title: product.title,
      price: product.price,
      priceWithoutDiscount: product.priceWithoutDiscount,
      badge: product.badge,
      image: imageId ? { _type: "image", asset: { _ref: imageId } } : undefined, // Add image if uploaded
      category: {
        _type: "reference",
        _ref: categoryIdMap[product.category._id], // Use the migrated category ID
      },
```

Activate Windows
Go to Settings to activate Windows.

Hackathon Day:3

API INTEGRATION AND DATA MIGRATION

General E-Commerce Website:Comforty

```
// Prepare the new product object
const newProduct = {
  _type: "products",
  title: product.title,
  price: product.price,
  priceWithoutDiscount: product.priceWithoutDiscount,
  badge: product.badge,
  image: imageId ? { _type: "image", asset: { _ref: imageId } } : undefined, // Add image if uploaded
  category: {
    _type: "reference",
    _ref: categoryIdMap[product.category._id], // Use the migrated category ID
  },
  description: product.description,
  inventory: product.inventory,
  tags: product.tags,
};

// Save the product to Sanity
const result = await targetClient.create(newProduct);
console.log(`Migrated product: ${product.title} (ID: ${result._id})`);
}

console.log("Data migration completed successfully!");
} catch (error) {
  console.error("Error during migration:", error.message);
  process.exit(1); // Stop execution if an error occurs
}
}
```

Activate Windows
Go to Settings to activate Windows.

Conclusion

The API integration and data migration processes for the Comforty Chair Marketplace were successfully completed. The adjustments made to schemas ensured compatibility with the external API data, while the migration script effectively populated Sanity CMS. The marketplace is now ready to display and manage products dynamically, providing a seamless experience for users.