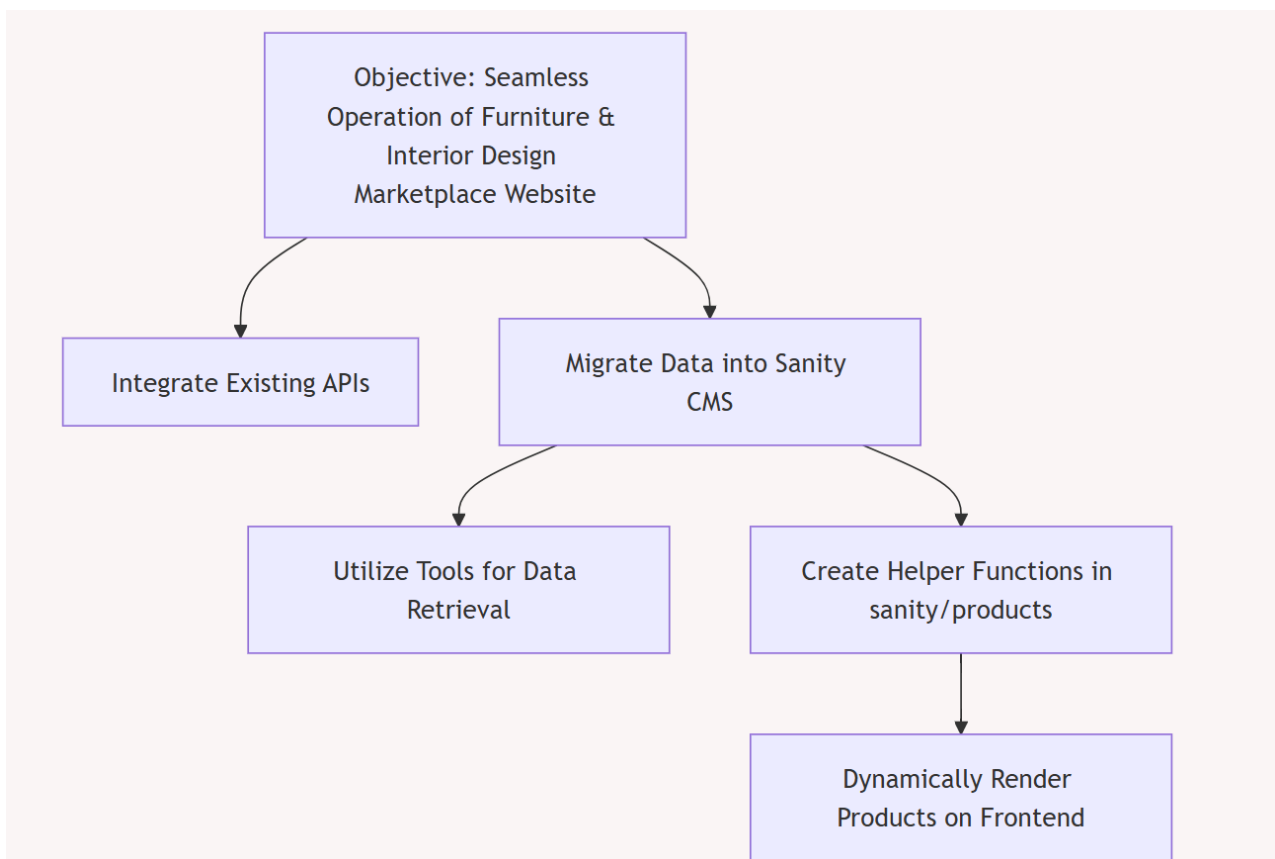
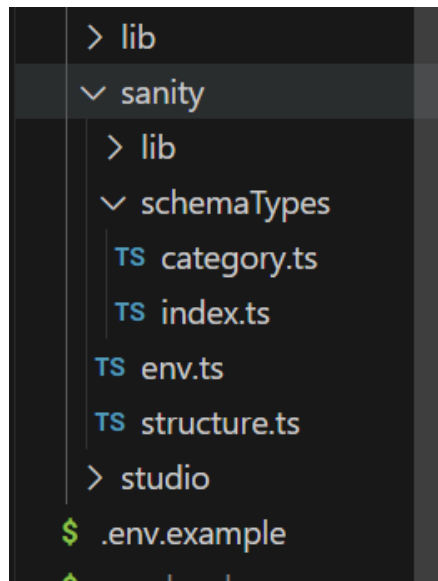


Day 3 - API Integration Report E-Commerce

Objective

To seamlessly integrate existing APIs and migrate data into Sanity CMS, enabling smooth functionality for the dynamic furniture marketplace. This involves utilizing tools for data retrieval, creating helper functions in sanity/products, and displaying products dynamically on the frontend.



Key Activities

1. API Integration - Integrated API to fetch product data - Utilized Typegen to streamline the process of fetching data from Sanity CMS.

2. Data Migration - Migrated product data into Sanity CMS using a migration script. - Validated data integrity and alignment with the defined schema.

```
import { createClient } from '@sanity/client';
import axios from 'axios';
import dotenv from 'dotenv';
import { fileURLToPath } from 'url';
import path from 'path';

// Load environment variables from .env.local
const __filename = fileURLToPath(import.meta.url);
const __dirname = path.dirname(__filename);
dotenv.config({ path: path.resolve(__dirname, '../.env.local') });

// Create Sanity client
const client = createClient({
  projectId: process.env.NEXT_PUBLIC_SANITY_PROJECT_ID,
  dataset: process.env.NEXT_PUBLIC_SANITY_DATASET,
  useCdn: false,
  token: process.env.SANITY_API_TOKEN,
  apiVersion: '2021-08-31',
});

// Upload image to Sanity
async function uploadImageToSanity(imageUrl) {
  try {
    console.log(`Uploading image: ${imageUrl}`);
    const response = await axios.get(imageUrl, { responseType: 'arraybuffer' });
    const buffer = Buffer.from(response.data);
    const asset = await client.assets.upload('image', buffer, {
      filename: imageUrl.split('/').pop(),
    });
    console.log(`Image uploaded successfully: ${asset._id}`);
    return {
      _type: 'image',
      asset: {
        _type: 'reference',
        _ref: asset._id,
      },
    },
  }
}
```

```

35     },
36   },
37 };
38 } catch (error) {
39   console.error('Failed to upload image:', imageUrl, error);
40   return null;
41 }
42 }
43
44 // Import data from API and upload to Sanity
45 async function importData() {
46   try {
47     console.log('Fetching products from API...');
48     const response = await axios.get('https://template-0-beta.vercel.app/api/product');
49     const products = response.data;
50     console.log(`Fetched ${products.length} products`);
51
52     for (const product of products) {
53       console.log(`Processing product: ${product.name}`);
54
55       // Fetch the category from Sanity or create if it doesn't exist
56       const categoryDoc = await client.fetch('*[_type == "category" && name == $name][0]', {
57         name: product.category || 'Uncategorized',
58       });
59
60       const categoryRef = categoryDoc || await client.create({
61         _type: 'category',
62         name: product.category || 'Uncategorized',
63       });
64
65       let imageRef = null;
66       if (product.imagePath) {
67         imageRef = await uploadImageToSanity(product.imagePath);
68       }
69

```

```

70     const sanityProduct = {
71       _type: 'product', // Fixed document type
72       id: product.id,
73       name: product.name,
74       image: imageRef,
75       price: parseFloat(product.price),
76       description: product.description || 'No description provided',
77       discountPercentage: product.discountPercentage || 0,
78       isFeaturedProduct: product.isFeaturedProduct || false,
79       stockLevel: product.stockLevel || 0,
80       category: {
81         _type: 'reference',
82         _ref: categoryRef._id, // Reference to the category document
83       },
84     };
85
86     console.log('Uploading product to Sanity:', sanityProduct.name);
87     const result = await client.create(sanityProduct);
88     console.log(`Product uploaded successfully: ${result._id}`);
89   }
90
91   console.log('Data import completed successfully!');
92 } catch (error) {
93   console.error('Error importing data:', error);
94 }
95 }
96
97 // Call the import function
98 importData();
99

```

Migration Data from Sanity CMS

Data migration from Sanity CMS involved transferring product-related content from external sources into the CMS, ensuring proper alignment with the defined schema and maintaining data integrity. The migration process aimed to populate the CMS with structured and accurate data for seamless API integration and frontend display.

Process Overview:

1. Data Collection:

- Gathered product information from external databases and files, such as JSON or CSV formats.
- Organized the data to match the schema fields defined in Sanity CMS.

2. Data Preparation:

- Preprocessed the data to ensure consistency, such as formatting values, removing duplicates, and validating entries.

3. Data Migration:

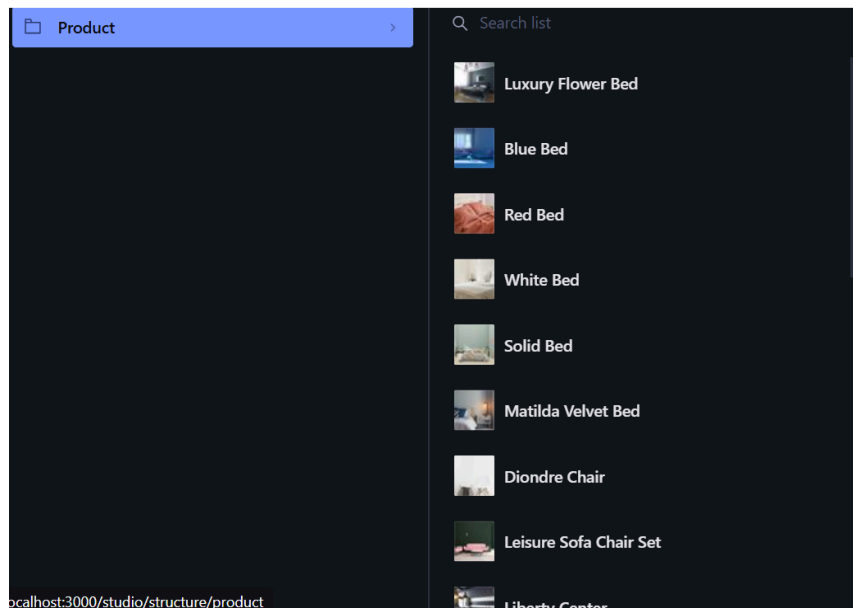
- Imported the prepared data into Sanity CMS using programmatic methods to create or update entries within the CMS.
- Verified that all necessary fields, including product name, description, price, and other metadata, were correctly populated.

4. Validation and Testing:

- Performed checks to confirm data accuracy and completeness in Sanity CMS.
- Validated the alignment of data with the schema structure and tested API queries to ensure smooth data retrieval.

```
9  // CarouselNext,
10  CarouselNext,
11  CarouselPrevious,
12  } from "@components/ui/carousel"
13
14  import { client } from "@sanity/lib/client";
15
16  // Define your product type
17  type Product = {
18    _id: string;
19    name: string;
20    description: string;
21    price: number;
22    imageUrl: string;
23    discountPercentage: number;
24    stockLevel: number;
25    category: string;
26    isFeaturedProduct: boolean;
27  };
28
29
30
31  export default function SecCard() {
32    const [products, setProducts] = useState<Product[]>([]);
33
34    useEffect(() => {
35      const fetchProducts = async () => {
36        // Using client.fetch to get data
37        const data = await client.fetch(
38          `*[_type == 'product']{
39            _id,
40            name,
41            description,
42            price,
43            discountPercentage,
44            "imageUrl": image.asset->url,
45            stockLevel,
46            category,
47            isFeaturedProduct
48          }`
49        );
50        setProducts(data);
51      };
52    });
```

Sanity Dashboard Screenshot Where Our Product is Store



Code Example For Fetching Data in Frontend

```
import { client } from "@sanity/lib/client";

// Define your product type
type Product = {
  _id: string;
  name: string;
  description: string;
  price: number;
  imageUrl: string;
  discountPercentage: number;
  stockLevel: number;
  category: string;
  isFeaturedProduct: boolean;
};
```

Product Display



Alpha Table

A sturdy oak chair with a sleek and minimalist design.

\$900



Liberty Center

Versatile entertainment chair for modern interiors.

\$1100



Leisure Sofa Chair Set

A comfortable set of chairs with soft cushions for relaxation.

\$1800



Diondre Chair

A tufted chair with acrylic legs for a chic, modern touch.

\$720



Replica Table



Sleek Modern Table



Matilda Velvet Bed



Luxury Flower Bed