Urban Loom:
API Integration and Data Migration Report

Overview of Urabn Loom :

Urban Loom is a modern e-commerce platform specializing in high-quality furniture for contemporary homes. The website is designed to provide an intuitive user experience where customers can explore various furniture categories, view detailed product information, and make well-informed purchase decisions. By leveraging cutting-edge technology like **Sanity CMS**, the platform ensures a seamless backend system for managing and retrieving data.

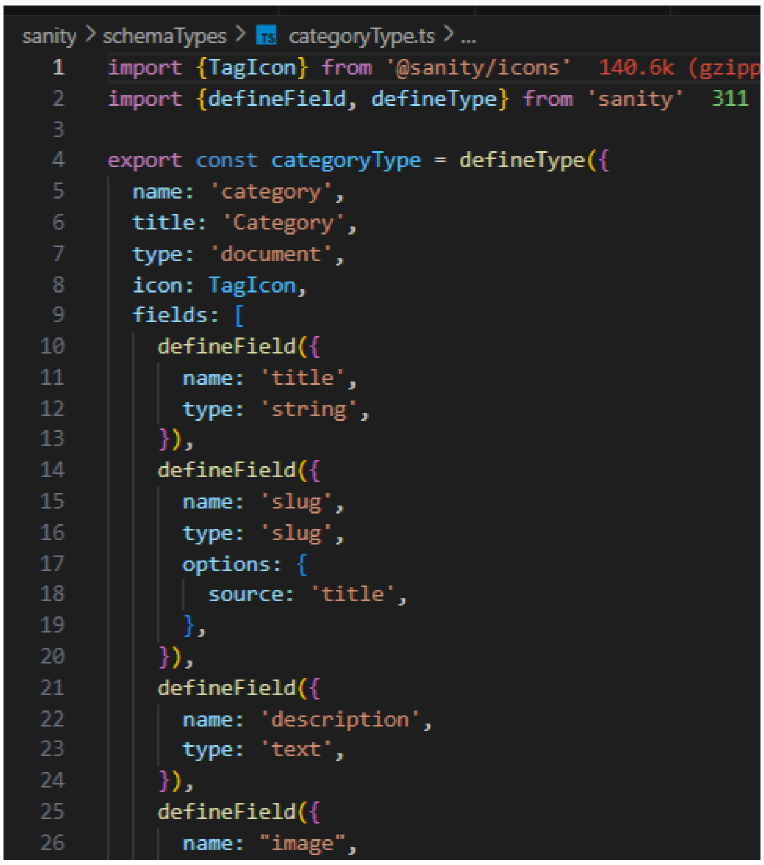
1: Validation and Schema Field Checking :

In Urban Loom, Sanity CMS ensures that all product, category, and sales data is validated before being saved. This prevents errors by checking that all required fields are filled out correctly.

1.0: Field Validation :

- Each schema (product, category, sales) defines mandatory fields like name, price, and category etc .
- Sanity CMS will automatically validate these fields when creating or updating entries, preventing incomplete data from being saved.

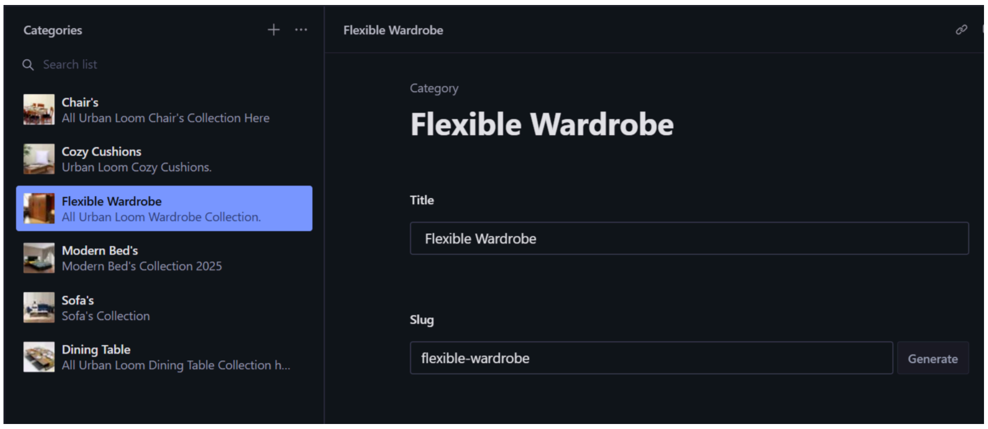
Category Schema Type :



sanity > schemaTypes > categoryType.ts > ...

```
import {TagIcon} from '@sanity/icons'  140.6k (gzipp
import {defineField, defineType} from 'sanity'  311

export const categoryType = defineType({
  name: 'category',
  title: 'Category',
  type: 'document',
  icon: TagIcon,
  fields: [
    defineField({
      name: 'title',
      type: 'string',
    }),
    defineField({
      name: 'slug',
      type: 'slug',
      options: {
        source: 'title',
      },
    }),
    defineField({
      name: 'description',
      type: 'text',
    }),
    defineField({
      name: "image",
```

In Sanity :



Chair's
All Urban Loom Chair's Collection Here

Cozy Cushions
Urban Loom Cozy Cushions.

Flexible Wardrobe
All Urban Loom Wardrobe Collection.

Modern Bed's
Modern Bed's Collection 2025

Sofa's
Sofa's Collection

Dining Table
All Urban Loom Dining Table Collection h...

Flexible Wardrobe

Category

Flexible Wardrobe

Title: Flexible Wardrobe

Slug: flexible-wardrobe

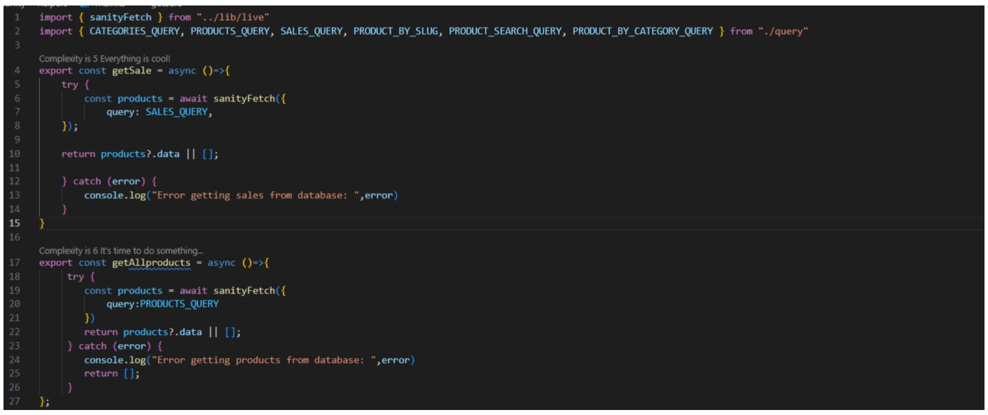
2: Data Management and Queries :

This document details the backend logic and data fetching implementation for the "Urban Loom" furniture e-commerce website, focusing on the usage of Sanity CMS and utility functions.

2.0: Queries and Data Fetching :

The data in "Urban Loom" is fetched from Sanity CMS using custom queries and utility functions. These are structured to handle product data, categories, and search functionalities efficiently.

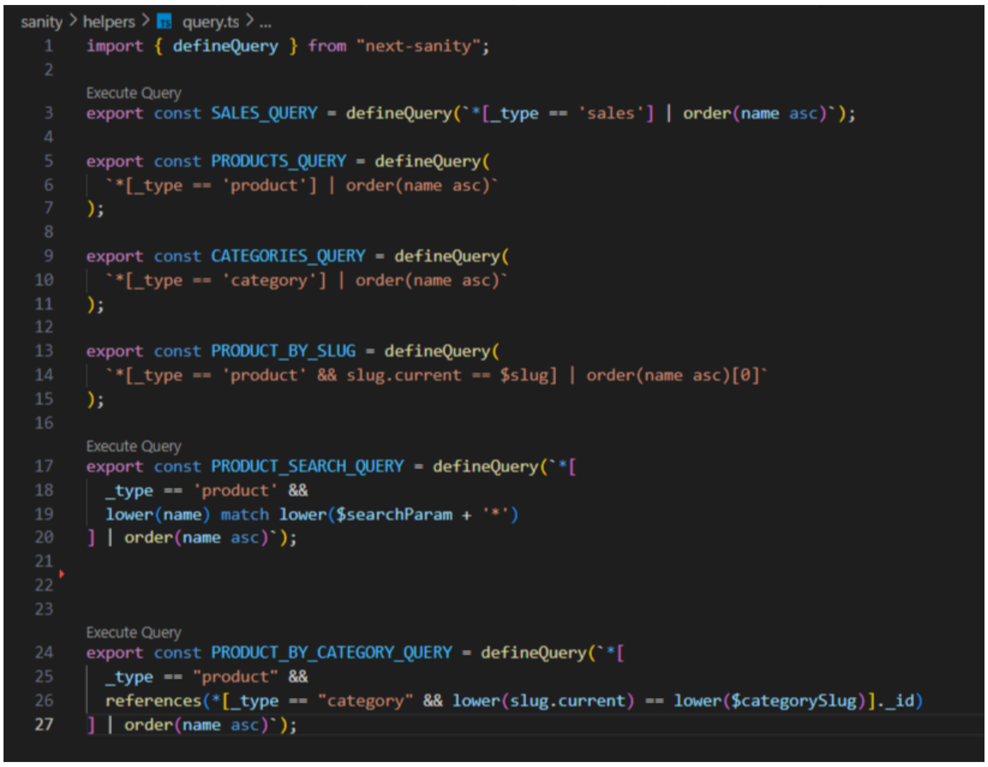
2.1: Fetching Function Like This :



```
import { sanityFetch } from "../lib/live"
import { CATEGORIES_QUERY, PRODUCTS_QUERY, SALES_QUERY, PRODUCT_BY_SLUG, PRODUCT_SEARCH_QUERY, PRODUCT_BY_CATEGORY_QUERY } from "./query"

Complexity is 5 Everything is cool!
export const getSale = async ()=>{
    try {
        const products = await sanityFetch({
            query: SALES_QUERY,
    });

    return products?.data || [];

    } catch (error) {
        console.log("Error getting sales from database: ",error)
    }
}

Complexity is 6 It's time to do something...
export const getAllproducts = async ()=>{
     try {
        const products = await sanityFetch({
            query:PRODUCTS_QUERY
        })
        return products?.data || [];
     } catch (error) {
        console.log("Error getting products from database: ",error)
        return [];
     }
};
```

To be Continued

2.2: Queries Implementation :



```
sanity > helpers > query.ts > ...
import { defineQuery } from "next-sanity";

Execute Query
export const SALES_QUERY = defineQuery(`*[_type == 'sales'] | order(name asc)`);

export const PRODUCTS_QUERY = defineQuery(
  `*[_type == 'product'] | order(name asc)`
);

export const CATEGORIES_QUERY = defineQuery(
  `*[_type == 'category'] | order(name asc)`
);

export const PRODUCT_BY_SLUG = defineQuery(
  `*[_type == 'product' && slug.current == $slug] | order(name asc)[0]`
);

Execute Query
export const PRODUCT_SEARCH_QUERY = defineQuery(`*[
  _type == 'product' &&
  lower(name) match lower($searchParam + '*')
] | order(name asc)`);



Execute Query
export const PRODUCT_BY_CATEGORY_QUERY = defineQuery(`*[
  _type == "product" &&
  references(*[_type == "category" && lower(slug.current) == lower($categorySlug)]._id)
] | order(name asc)`);
```

3: Sanity Integration :

The project uses Sanity CMS for seamless content management and data handling. The integration involves :

3.0: Functional API Implementation:

- Queries defined in query.ts enable dynamic data retrieval.
- These are executed through utility functions in main.ts, ensuring structured and reusable logic.

3.1: Dynamic Content Rendering :

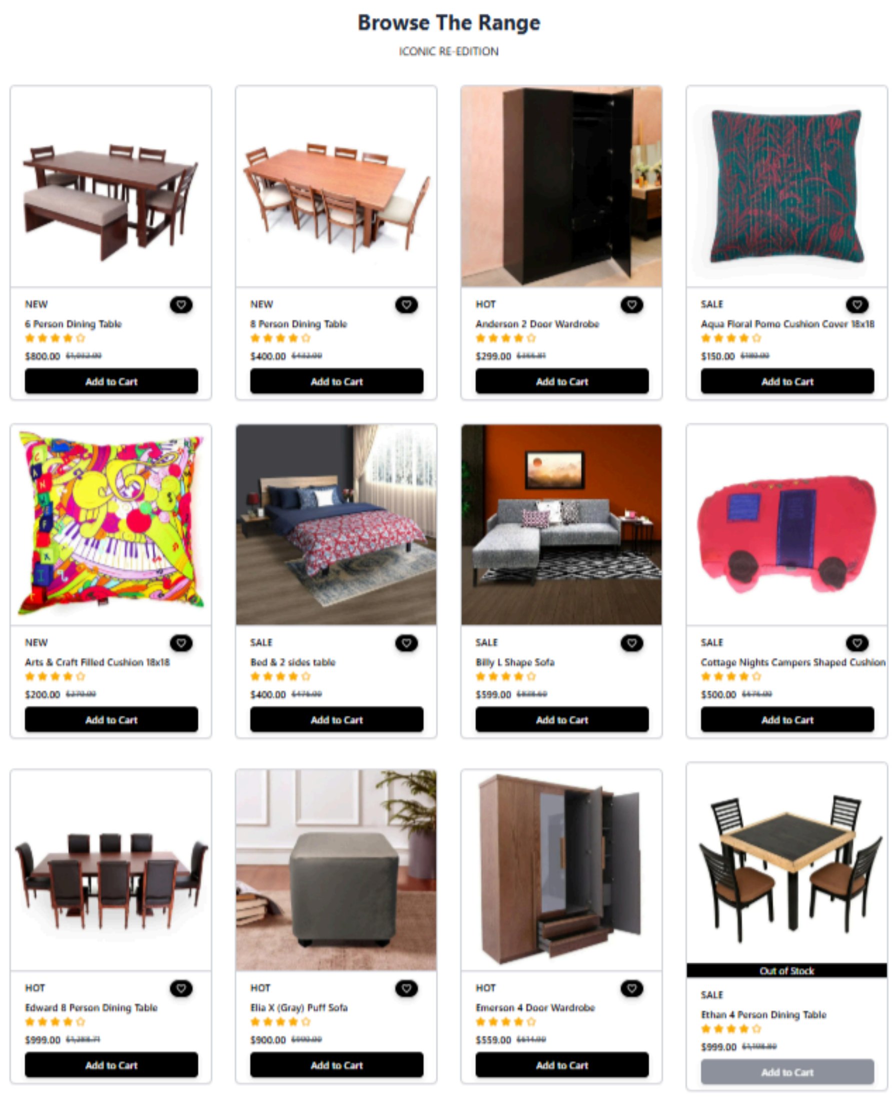
- Fetched data is dynamically rendered on the frontend using JavaScript and TypeScript components.
- Sanity’s flexibility allows for easy updates and scalability.

3.2: Schema Validation :

- Sanity schemas ensure consistent data structure for products, categories, and sales.
- Validation prevents errors during query execution.

Product Listing Image:

In Urban Loom, product images are dynamically displayed on the product listing page. Each product has an associated image, which is fetched from Sanity CMS and rendered on the website.



Browse The Range

ICONIC RE-EDITION

NEW
6 Person Dining Table
★★★★☆
$800.00 ~~$1,032.00~~
Add to Cart

NEW
8 Person Dining Table
★★★★☆
$400.00 ~~$482.00~~
Add to Cart

HOT
Anderson 2 Door Wardrobe
★★★★☆
$299.00 ~~$366.81~~
Add to Cart

SALE
Aqua Floral Pomo Cushion Cover 18x18
★★★★☆
$150.00 ~~$180.00~~
Add to Cart

NEW
Arts & Craft Filled Cushion 18x18
★★★★☆
$200.00 ~~$270.00~~
Add to Cart

SALE
Bed & 2 sides table
★★★★☆
$400.00 ~~$476.00~~
Add to Cart

SALE
Billy L Shape Sofa
★★★★☆
$599.00 ~~$838.60~~
Add to Cart

SALE
Cottage Nights Campers Shaped Cushion
★★★★☆
$500.00 ~~$676.00~~
Add to Cart

HOT
Edward 8 Person Dining Table
★★★★☆
$999.00 ~~$1,288.71~~
Add to Cart

HOT
Elia X (Gray) Puff Sofa
★★★★☆
$900.00 ~~$990.00~~
Add to Cart

HOT
Emerson 4 Door Wardrobe
★★★★☆
$559.00 ~~$614.00~~
Add to Cart

Out of Stock
SALE
Ethan 4 Person Dining Table
★★★★☆
$999.00 ~~$1,108.80~~
Add to Cart

Thanks For Viewing

Ayesha Mughal