

Hackathon Day 3

API Integration Process Using Sanity

Sanity, a headless CMS, was employed to streamline content management and ensure smooth API integration. Below is an overview of the process:

1. Project Setup:

- Created a new Sanity project via the Sanity CLI.
- Configured the sanity. Json file with project-specific details.
- This project involves developing an e-commerce store.
- The administrator has the capability to add products using the Sanity content management system.
- The /shop/[item] Route, situated within the app directory, is specifically designed to handle the management and display of product data.

2. Adjustments Made to Schemas

- Schema modifications were implemented to align with the website's content structure and ensure optimal performance. The Structure is showed in screenshots

Screenshots

Schema: product page

```
1  export const products={
2    name: 'Products',
3    title: 'Product Category',
4    type: 'document',
5    fields: [
6      {
7        name: 'ProductCategory',
8        type: 'string',
9        title: 'Category Name',
10     },
11     {
12       name: 'ProductCategoryImage',
13       type: 'image',
14       title: 'Category Image',
15     },
16   ],
17 },
18
19
20 }
```

Schema Shop page:

```
1  export const shop={
2    name: 'Shop',
3    title: 'Shop Products',
4    type: 'document',
5    fields: [
6      {
7        name: 'ProductID',
8        type: 'string',
9        title: 'Shop Product ID',
10     },
11     {
12       name: 'ProductName',
13       type: 'string',
14       title: 'Shop Product Name',
15     },
16     {
17       name: 'ProductImage',
18       type: 'image',
19       title: 'Shop Product Image',
20     },
21     {
22       name: 'ProductPrice',
23       type: 'string',
24       title: 'Shop Product Price',
25     },
26     {
27       name: 'ProductDescription',
28       type: 'string',
29       title: 'Shop Product Description',
30     },
31     {
32       name: 'ProductDiscount',
33       type: 'string',
34       title: 'Shop Product Discount',
35     },
36     {
37       name: 'ProductFeatured',
38       type: 'boolean',
39       title: 'Shop Product Is Featured',
40     },
41     {
42       name: 'Productcategory',
43       type: 'string',
44       title: 'Shop Product Category',
45     },
46   ],
47 },
48
49
50 }
```

3. Migration Steps and Tools Used

- Content migration was a critical phase to ensure the smooth transition of existing data into Sanity.
- Making Schema in Sanity
- Creating Documents in Sanity Including Product Category and Shop products
- Using sanity client and declare some functions to fetch data from the Sanity Studio

API Calls:

Product page:

```
1 // components/ProductGrid.tsx
2 import Link from 'next/link';
3 import Header from '../Components/component1';
4 import Footer from '../Components/component2';
5 import Image from 'next/image';
6 import { client } from '@sanity/lib/client';
7 import imageUrlBuilder from '@sanity/image-url';
8 import Bottom1 from '../Components/component3';
9
10 const builder = imageUrlBuilder(client);
11
12 function urlFor(source: any) {
13   return builder.image(source);
14 }
15
16 async function getData() {
17   const query = `*[type == "Products"]{
18     _id,
19     ProductCategory,
20     ProductCategoryImage
21   }`;
22   const data = await client.fetch(query);
23   return data;
24 }
25
26 export default async function ProductGrid() {
27   const data = await getData();
28
29   return (
30     <div>
31       <Header />
32       <div className="bg-gray-50">
33         <section className="bg-purple-100 text-center py-10">
34           <h1 className="text-3xl font-bold text-gray-800">Products</h1>
35           <p className="mt-2 text-gray-600">Browse our product categories</p>
36         </section>
37       </div>
38       <div className="p-6">
39         <h1 className="text-2xl font-bold text-center mb-6">Ecommerce Items</h1>
40         <div className="text-center text-gray-500 mb-6">
41           {data.length} Product Categories
42         </div>
43         <div className="grid grid-cols-1 sm:grid-cols-2 md:grid-cols-3 lg:grid-cols-4 gap-6">
44           {data.map((val: any) => (
45             <Link
46               key={val._id}
47               href="/shop"
48               passHref
49             >
50               <div className="group block border border-gray-200 rounded-lg overflow-hidden hover:shadow-lg transition-shadow 300">
51                 <div className="bg-gray-100 aspect-w-1 aspect-h-1 w-full">
52                   <Image
53                     src={urlFor(val.ProductCategoryImage).url()}
54                     alt={val.ProductCategory}
55                     className="object-cover w-full h-full group-hover:scale-105 transition-transform duration-300"
56                     width={250}
57                     height={183}
58                   />
59                 </div>
60                 <div className="p-4">
61                   <h2 className="font-semibold text-lg text-gray-800">
62                     {val.ProductCategory}
63                   </h2>
64                 </div>
65               </div>
66             </Link>
67           ))}
68         </div>
69       </div>
70       <Bottom1 />
71       <Footer />
72     </div>
73   );
74 }
75
```

API Call Shop Page

```
1 import Link from 'next/link';
2 import Header from '../Components/component1';
3 import Footer from '../Components/component2';
4 import Bottom1 from '../Components/component3';
5 import Image from 'next/image';
6 import { client } from '@sanity/lib/client';
7
8 // Fetch data from Sanity
9 async function fetchShopProducts() {
10   const query = `*[ _type == 'Shop' ]{
11     ProductID,
12     ProductName,
13     "imageUrl": ProductImage.asset->url,
14     ProductPrice,
15     ProductDescription,
16     ProductDiscount
17   }`;
18   const products = await client.fetch(query);
19   return products;
20 }
21
22 export default async function Shop() {
23   const products = await fetchShopProducts();
24   return (
25     <div>
26       <Header/>
27       <div className="bg-gray-50">
28         <section className="bg-purple-100 text-center py-10">
29           <h1 className="text-3xl font-bold text-gray-800">Shop</h1>
30           <p className="mt-2 text-gray-600"></p>
31         </section>
32       </div>
33       <div className="flex flex-col md:flex-row container mx-auto px-4 py-8">
34         /* Left Sidebar */
35         <aside className="w-full md:w-1/4 bg-gray-50 p-4 rounded-md shadow">
36           <h2 className="text-lg font-semibold text-gray-700 mb-4">Filters</h2>
37
38           <div className="mb-6">
39             <h3 className="text-md font-medium text-gray-600 mb-2">Product Brand</h3>
40             <ul className="space-y-1">
41               <li><input type="checkbox" /> Coaster Furniture</li>
42               <li><input type="checkbox" /> Fusion Dot High Fashion</li>
43               <li><input type="checkbox" /> Unique Furniture</li>
44               <li><input type="checkbox" /> Dream Furniture</li>
45               <li><input type="checkbox" /> Young Repurposed</li>
46               <li><input type="checkbox" /> Green Furniture</li>
47             </ul>
48             /* Add more brands */
49           </div>
50
51           <div className="mb-6">
52             <h3 className="text-md font-medium text-gray-600 mb-2">Categories</h3>
53             <ul className="space-y-1">
54               <li><input type="checkbox" /> Accessories</li>
55               <li><input type="checkbox" /> Watches</li>
56               <li><input type="checkbox" /> Magento</li>
57               <li><input type="checkbox" /> clothes</li>
58               <li><input type="checkbox" /> Furniture</li>
59             </ul>
60             /* Add more categories */
61           </div>
62
63           <div className="mb-6">
64             <h3 className="text-md font-medium text-gray-600 mb-2">Price Filter</h3>
65             <ul className="space-y-1">
66               <li><input type="radio" name="price" /> $0 - $150</li>
67               <li><input type="radio" name="price" /> $150 - $350</li>
68               <li><input type="radio" name="price" /> $350 - $550</li>
69               <li><input type="radio" name="price" /> $550 - $650</li>
70               <li><input type="radio" name="price" /> $650 - $1000</li>
71               <li><input type="radio" name="price" /> above $1000</li>
72             </ul>
73           </div>
74         </aside>
75
76         /* Products Section */
77         <main className="flex-1 md:ml-8">
78           <h1 className="text-xl font-semibold text-gray-800 mb-4">Shop Products</h1>
79
80           <div className="grid grid-cols-1 sm:grid-cols-2 lg:grid-cols-3 gap-6">
81             {products.map((product: any) => (
82               <div
83                 key={product.ProductID}
84                 className="bg-white rounded-lg shadow p-4 hover:shadow-lg transition flex flex-col">
85               <Link href={`/shop/${product.ProductID}`} passHref>
86                 <div>
87                   <img
88                     src={product.imageUrl}
89                     alt={product.ProductName}
90                     className="rounded-md w-full object-cover"
91                     width={300}
92                     height={200}
93                   />
94                 </div>
95                 <div>
96                   <div className="mt-4">
97                     <h2 className="text-lg font-semibold text-gray-700">
98                       {product.ProductName}
99                     </h2>
100                     <p className="text-gray-500 text-sm">{product.ProductDescription}</p>
101                     <div className="mt-2 flex justify-between items-center">
102                       <p className="text-gray-800 font-bold">
103                         ${product.ProductDiscount || product.ProductPrice}
104                       </p>
105                       {product.ProductDiscount && (
106                         <p className="text-gray-500 line-through">
107                           ${product.ProductPrice}
108                         </p>
109                       )}
110                     </div>
111                   </div>
112                 </div>
113             ))}
114           </div>
115         </main>
116       </div>
117       <div>
118         <Bottom1/>
119         <Footer/>
120       </div>
121     );
122   }
123 }
124
```

Frontend Display:

Products

Browse our product categories

Ecommerce Items

8 Product Categories



Phones



Electronics



Sport goods



Cameras



Furniture



Beauty



Mobile Accessories



Watches

Live

Sunshine

Handcrafted

Mestonix

Pure

Hekto

Enter Email Address

Contact Info
17 Princess Road, London, Greater London
NW1 8JR, UK

Categories

- Laptops & Computers
- Cameras & Photography
- Smart Phones & Tablets
- Video Games & Consoles
- Waterproof Headphones

Customer Care

- My Account
- Discount
- Returns
- Orders History
- Order Tracking

Pages

- Blog
- Browse the Shop
- Category
- Pre-Built Pages
- Visual Composer Elements
- WooCommerce Pages

