

Building Dynamic Frontend Components for Marketplace

Dynamic Products Listing Component

The Dynamic Component for product listing page fetching data from sanity and displaying on UI through mapping each product with unique ID using map method and each product containing Product Image, Product Name and



```
1 export default async function Products() {
2
3   const products = await client.fetch(allProductsQuery, {}, {cache: "no-store"})
4
5   return (
6     <>
7       <Image className="w-full" src={productbanner} alt="product banner"></Image>
8
9       <div className="flex flex-col gap-5 md:mx-20 sm:mx-10">
10
11         <div className="flex gap-6 my-10">
12           <div className="flex flex-wrap sm:justify-start justify-center gap-6">
13             {products.map((product:ProductsCardType)=>(
14
15               <Link key={product._id} href={`/productsListing/${product.slug.current}`}>
16                 <ProductsProp
17                   image={product.image}
18                   title={product.name}
19                   price={product.price}
20                 />
21               </Link>
22             ))}
23           </div>
24         </div>
25       </div>
26     </>
27   );
}
```

The screenshot shows a product listing page for 'All products'. The top navigation bar includes links for Ceramics, Chairs, Crockery, Cutlery, Plant Pots, Tables, and Tableware. Below the header, there's a section titled 'All products' with a search bar. The main content area displays a grid of 20 products, each with a thumbnail image, name, and price. The products are categorized into several groups:

- Sofas:** The Poplar suede sofa (Price: \$280), Tropical Vibe (Price: \$550), Sleek Living (Price: \$300), Timeless Elegance (Price: \$320), Bold Nest (Price: \$260), and Retro Vibe (Price: \$340).
- Chairs:** Serene Seat (Price: \$350), Nordic Elegance (Price: \$280), TimberCraft (Price: \$320), The Steel Chair (Price: \$250), Bright Space (Price: \$180), The Dandy chair (Price: \$180), Wood Chair (Price: \$100), Pure Aura (Price: \$280), and Zen Table (Price: \$250).
- Tables:** Avion (Ceramic), Cloud Haven Chair (Price: \$230), Sunny Chio (Price: \$400), Modern Serenity (Price: \$480), Reflective Haven (Price: \$300), and The Lucky Lamp (Price: \$200).
- Cards:** Vase Set (Price: \$150), Marble Edge (Price: \$412), Rustic Vase Set (Price: \$210), and Avion (Ceramic).
- Bedding:** Bed (Price: \$260).
- Decor:** Pure Aura (Price: \$280) and Zen Table (Price: \$250).

At the bottom of the page, there are links for 'Menu', 'Categories', 'Our Company', and 'Join our mailing list'.

Dynamic Data Explanation

This Website once a static before but after integration of sanity data in project all data is fetched and displayed in this platform are coming from sanity and now it's a fully functioned dynamic marketplace.

This website displays the data which are stored and managed from sanity which admin can update the data like products, categories, stocks and prices from sanity studio without integrating the code.

Dynamic Products Details Component

The Dynamic Component for Product Details page fetching data from sanity and displaying on UI through mapping each products detail with products slug using map method and each product containing Product Image, Name, Price, Description and Features, When click on any product it redirect on details page and also have a add to cart button



Avion

[Ceramics](#)[Chairs](#)[Crockery](#)[Cutlery](#)[Plant Pots](#)[Tables](#)[Tableware](#)

Modern Serenity

\$480

A timeless design, with premium materials features as one of our main pieces. The dandy chair is perfect for any stylish living space with leather upholstery.

Premium material
Handmade upholstery
Quality timeless classic

Dimensions

Height	Width	Depth
110cm	75cm	50cm

Amount: - 1 +

Dynamic Category Pages Component

The Dynamic Category page is displaying the data based on different categories of products and also have dedicated slugs data where each slug is representing a seprate category page but its dynamic so there is ony 1 dynamic page app/category/[slug]/page.tsx and the data is fetched from sanity to code.



```
1 import ProductsProp from "@/components/ProductsProp";
2 import { client } from "@sanity/lib/client";
3 import Link from "next/link";
4 import { ProductsCardType } from "@sanity/schemaTypes/productTyp
5 e";
6 // Fetch data for the category page
7 async function getCategoryData(slug: string) {
8   const query = `
9     *[_type == "category" && slug.current == $slug][0] {
10       name,
11       slug,
12       "products": *[_type == "product" && references(^._id)] {
13         name,
14         slug,
15         price,
16         "image": image.asset->url
17       }
18     }
19   `;
20   const data = await client.fetch(query, { slug });
21   return data;
22 }
23
24 // Fetch all category slugs for static generation
25 export async function generateStaticParams() {
26   const query = `*[_type == "category"] { "slug": slug.current }`;
27   const categories = await client.fetch(query);
28   return categories.map((category: { slug: string }) => ({
29     slug: category.slug,
30   }));
31 }
```



```
1 interface Category {
2   name: string;
3   products: ProductsCardType[];
4 }
5
6 interface Props {
7   params: { slug: string };
8 }
9
10 // Dynamic category page
11 const CategoryPage = async ({ params }: Props) => {
12   const { slug } = params;
13
14   // Fetch category data
15   const category: Category | null = await getCategoryData(slug);
16
17   if (!category) {
18     return <div className="container mx-auto p-4">Category not found.</div>;
19   }
20
21   return (
22     <div className="container mx-auto py-10">
23       <h1 className="text-4xl font-semibold uppercase mb-5 text-center">{category.name}</h1>
24       <div className="flex flex-wrap sm:justify-start justify-center gap-6">
25         {category.products.map((product) => (
26           <Link href={`/productsListing/${product.slug.current}`} key={product.slug.current} className="border rounded p-4 shadow-md">
27             <ProductsProp
28               image={product.image}
29               title={product.name}
30               price={product.price}
31             />
32             <Link>
33             ))
34           </div>
35         </div>
36       );
37     );
38
39 export default CategoryPage;
40
```

Q

Avion

Ceramics Chairs Crockery Cutlery Plant Pots Tables Tableware

Ceramics



Modern Serenity
Price: \$480

Reflective Haven
Price: \$300

The Lucky Lamp
Price: \$200

Q

Avion

Ceramics Chairs Crockery Cutlery Plant Pots Tables Tableware

Tableware



The Poplar suede sofa
Price: \$980

Tropical Vibe
Price: \$550

Sunny Chic
Price: \$400

Bold Nest
Price: \$260

Q

Avion

Ceramics Chairs Crockery Cutlery Plant Pots Tables Tableware

Chairs



TimberCraft
Price: \$320

The Steel Chair
Price: \$250

The Dandy chair
Price: \$150

Wood Chair
Price: \$100

Dynamic Component Based on Tags

The Dynamic Components based on tags are fetched in a component using props and the tags are passed in props like best selling , new arrival etc , we call a component and pass a tag then display the data based on that tag



```
1 "use client";
2 import { useEffect, useState } from "react";
3 import { client } from "@/sanity/lib/client";
4 import { ProductsCardType } from "@/sanity/schemaTypes/productTyp
5 import Link from "next/link";
6 import ProductsProp from "./ProductsProp";
7
8
9 const fetchProductsByTag = async (tag: string) => {
10   const query = `
11     *[_type == "product" && "${tag}" in tags] {
12       _id,
13       name,
14       slug,
15       "image": image.asset->url,
16       price,
17       tags,
18       description,
19       dimensions
20     }
21   `;
22   return await client.fetch(query, {}, {cache: "no-store"});
23 };
24
25 const ProductsByTag = ({ tag }: { tag: string }) => {
26   const [products, setProducts] = useState([]);
27
28   useEffect(() => {
29     const fetchData = async () => {
30       const fetchedProducts = await fetchProductsByTag(tag);
31       setProducts(fetchedProducts);
32     };
33     fetchData();
34   }, [tag]);

```



```
1 if (products.length === 0) {
2   return <p className="text-center text-4xl text-[#2a254b] my-32">No products found for {tag}</p>;
3 }
4
5 return (
6   <div className="flex flex-col gap-10 my-20 sm:mx-20">
7     <h1 className="text-[32px] leading-[39.36px] lg:text-start text-center capitalize">{tag}</h1>
8     <div className="flex gap-6 lg:flex-wrap flex-wrap lg:justify-start justify-center">
9       {products.map((product: ProductsCardType) => (
10         <Link key={product._id} href={`/productsListing/${product.slug.current}`}>
11           <ProductsProp
12             image={product.image}
13             title={product.name}
14             price={product.price}
15           />
16           </Link>
17         )));
18       </div>
19
20       <Link className="flex justify-center items-center mx-auto my-5 sm:w-[170px] w-full h-[56px] bg-[#f9f9f9]" href={'/productsListing'}>View collection</L
21 ink></div>
22 );
23 );
24
25 export default ProductsByTag;
26
```



```
1 import ProductsByTag from "./ProductsByTag";
2
3 function PopularProducts() {
4   return (
5     <div>
6       <ProductsByTag tag="popular products"/>
7     </div>
8   );
9 }
10
11 export default PopularProducts;
12
```

Dynamic Component by Tag name

New Ceramics Tag

New Ceramics

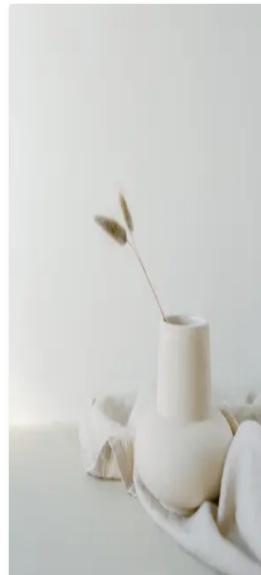


The Dandy chair



Vase Set

Price: \$150



Rustic Vase Set

Price: \$210



The Lucky Lamp

Price: \$200

Popular Product Tag

Popular Products



The Poplar suede sofa

Price: \$980



TimberCraft

Price: \$320



The Dandy chair

Price: \$150

Complete Dynamic Marketplace

The Complete dynamic market place is ready where all the data are fetched from sanity and managed in sanity.

All the pages and components now fetched and displayed data using Groq query of sanity CMS with clean code and fast performance and best practices approach.