

Day 4 -

Dynamic Frontend Components

AVION

NAME:ANOUSHA BAIG

ROLLNO:00296842

Key Components Built:

Product Listing Component:

- Fetching data from the API (or Sanity CMS).
- Dynamic routing for product details.

figma > src > app > AllProducts > page.tsx > Products > products.map() callback

```
import Navbar from './header';

interface Product {
  _id: string;
  name: string;
  price: number;
  imageUrl: string;
}

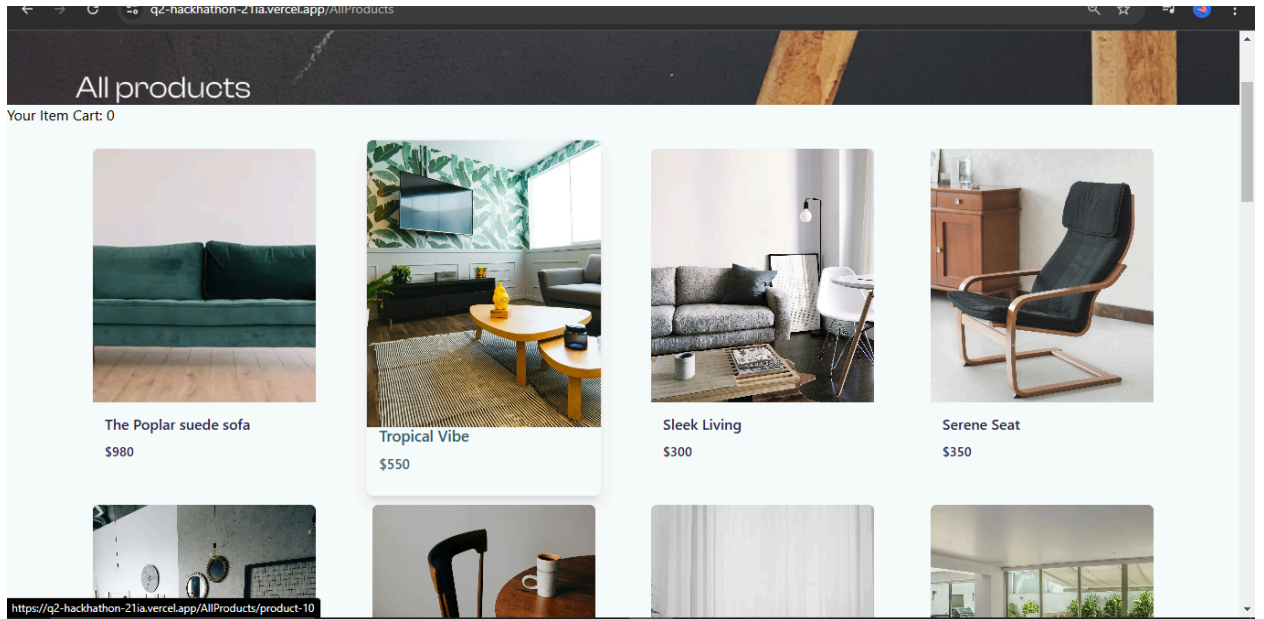
export default function Products() {
  const [products, setProducts] = useState<Product[]>([]);
  const [, setError] = useState<Error | null>(null);

  useEffect(() => {
    const fetchProducts = async () => {
      try {
        const query = `*_type == "product"{
          _id,
          name,
          price,
          imageUrl:
            image.asset->url
        }`;
        const products = await client.fetch(query);
        setProducts(products);
      } catch (error) {
        setError(error as Error);
      }
    };
    fetchProducts();
  }, []);

  return (
```

figma > src > app > AllProducts > page.tsx > Products > products.map() callback

```
export default function Products() {
  return (
    <div className="bg-gray-50 min-h-screen">
      <header>
        <Navbar />
      </header>
      <div className="container mx-auto px-4 mt-8">
        { /* Grid Container */ }
        <div className="grid grid-cols-1 sm:grid-cols-2 lg:grid-cols-4 gap-6 justify-items-center">
          {products.map((product) => (
            <Link
              key={product._id}
              href={` /AllProducts/${product._id}`}
              className="relative block group w-full max-w-[305px] h-[462px] rounded-lg overflow-hidden transform transition duration-200 ease-in
            >
              { /* Product Image */ }
              <div className="relative w-full h-[75%] bg-gray-200 group-hover:scale-105 transition-transform duration-300">
                <Image
                  src={product.imageUrl}
                  alt={product.name}
                  fill
                  className="object-cover transition duration-300 transform group-hover:scale-110"
                />
              </div>
              { /* Product Details */ }
              <div className="p-4 flex flex-col gap-2 transition-transform duration-300 group-hover:translate-y-2">
                <h4 className="text-xl font-semibold text-[#2A254B] group-hover:text-[#38626e] transition duration-300">
                  {product.name}
                </h4>
                <p className="text-lg font-medium text-[#2A254B] group-hover:text-[#38626e] transition duration-300">
                  ${product.price}
                </p>
              </div>
            </Link>
          ))}
        </div>
      </div>
    </div>
  );
}
```



Product Detail Component:

- Display of the dynamic grid layout and data fetching logic.

```

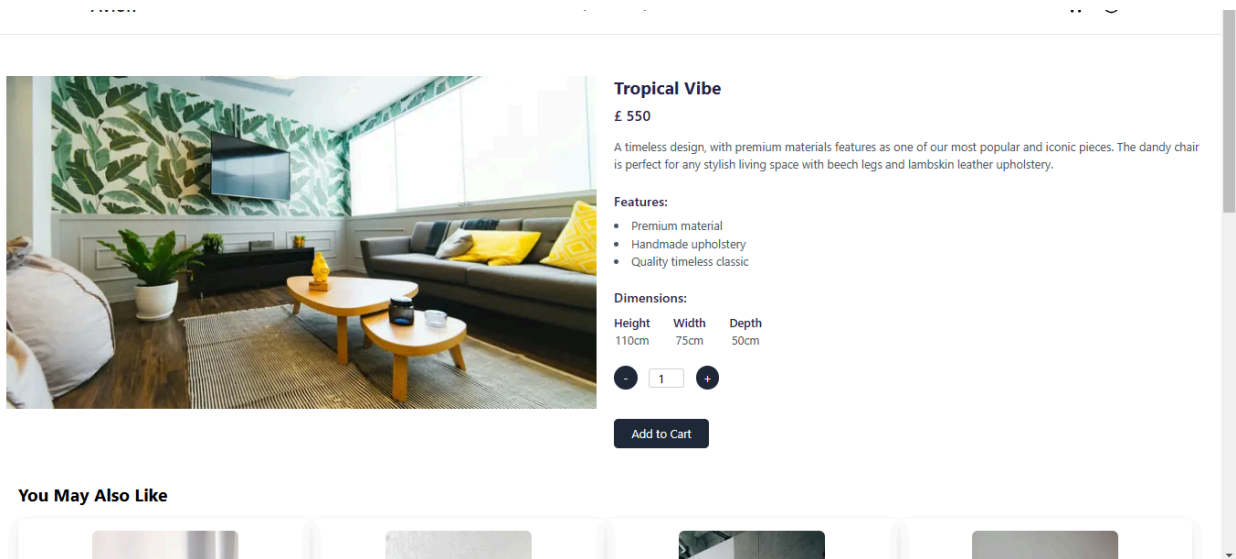
useEffect(() => {
  const fetchProducts = async () => {
    try {
      const features = await client.fetch(
        `*[_type == "product"][2..5]{
          id,
          name,
          price,
          description,
          image {
            asset->{
              _id,
              url
            }
          }
        }`
      );
      setFeaturedProduct(features);
    } catch (error) {
      console.error("Error fetching featured products:", error);
    }
  }
}, []);

```

Add to Cart Component:

- **Description:**

I implemented an Add to Cart functionality where users can add products to the shopping cart. The cart displays the product name, quantity, and total price, and it dynamically updates as items are added.



Category Component:

- categories are displayed and filtered dynamically from the backend.

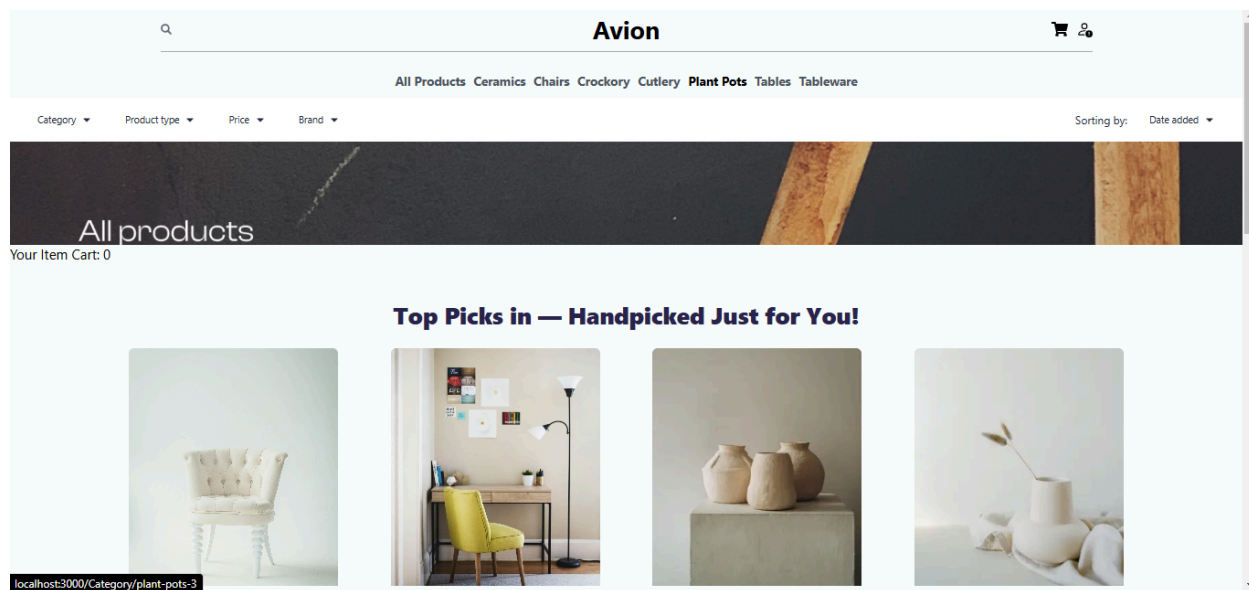
```

useEffect(() => {
  const fetchCategory = async () => {
    try {
      const categoryData = await client.fetch(
        `*_type == "category" && _id == ${categoryId}{
          _id,
          name,
          slug
        }`,
        { categoryId }
      );
      setCategory(categoryData[0]);
    } catch (error) {
      console.error("Error fetching category:", error);
    }
  };
  fetchCategory();
}, [categoryId]);

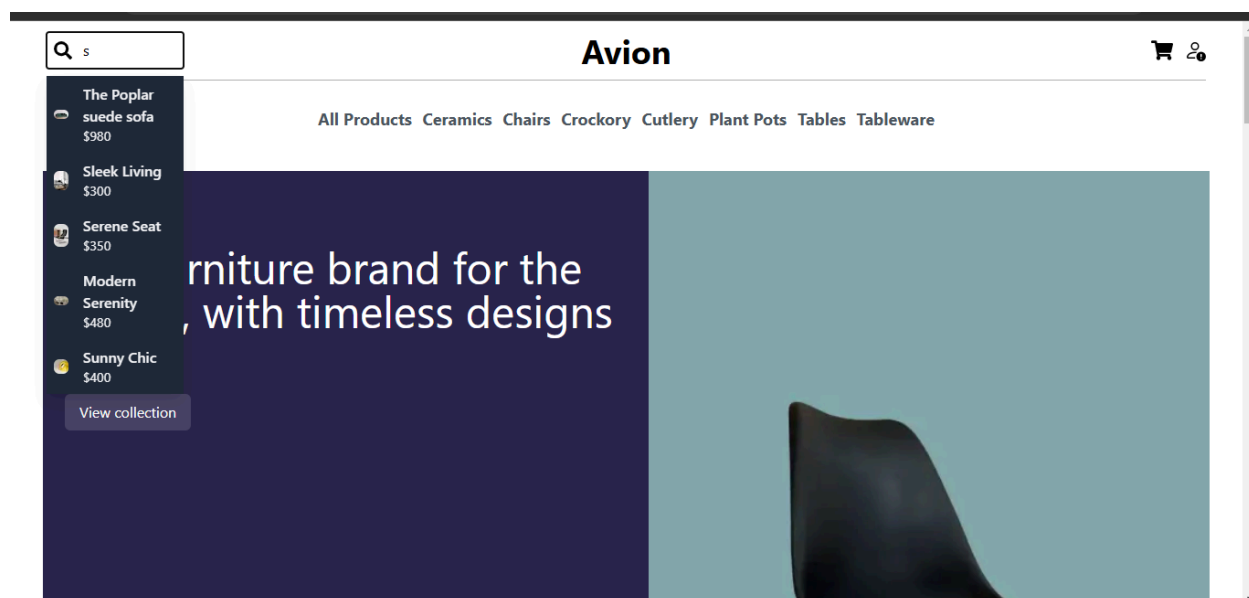
// Fetch products of the category based on category ID
useEffect(() => {
  const fetchProducts = async () => {
    try {
      const productsData = await client.fetch(
        `*_type == "product" && category_id == ${categoryId}{
          _id,
          name,
          slug,
          price,
          image
        }`,
        { categoryId }
      );
      setProducts(productsData);
    } catch (error) {
      console.error("Error fetching products:", error);
    }
  };
  fetchProducts();
}, [categoryId]);

```

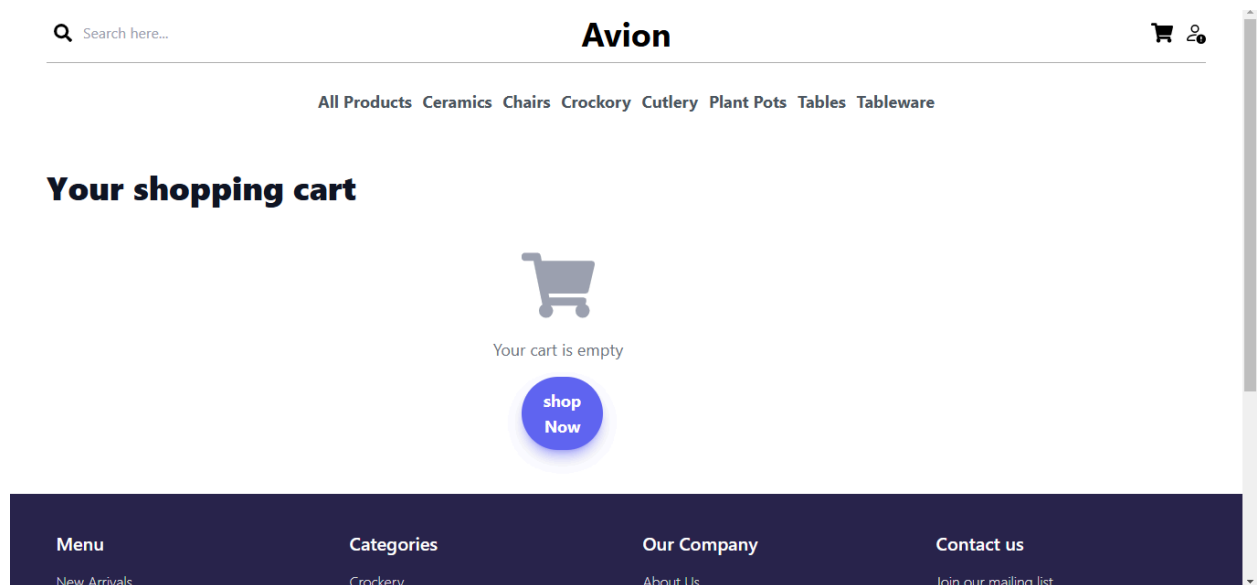
- Product by selected Categories



● Advance Search Bar



- Cart Component



Search here...

Avion

All Products

Ceramics

Chairs

Crockory

Cutlery

Plant Pots

Tables

Tableware

Your shopping cart

Product

Quantity

Total

Modern Serenity

A timeless design, with premium materials features as one of our most popular and iconic pieces. The dandy chair is perfect for any stylish living space with beech legs and lambskin leather upholstery.

£ 480

1

£ 480.00

The Dandy chair

A timeless design, with premium materials features as one of our most popular and iconic pieces. The dandy chair is perfect for any stylish living space with beech legs and lambskin leather upholstery.

£ 150

1

£ 150.00

PROFESSIONAL PRACTICE EMPHASIZED

Frontend Component Development:

Styling and Responsiveness:

Documentation and Submission:	<input checked="" type="checkbox"/>
-------------------------------	-------------------------------------

Code Quality:	<input checked="" type="checkbox"/>
---------------	-------------------------------------

Final Review:	<input checked="" type="checkbox"/>
---------------	-------------------------------------