

# Day 04: Building Dynamic Frontend Components For Marketplace

## Key Components to Build:

### Product Listing Component:

### Code Snippet:

```
import { useEffect, useState } from "react";
import { client } from "@sanity/lib/client";
import Image from "next/image";
import Link from "next/link";
import { Product } from "../../types/products";
import { addToCart } from "../../actions/actions";
import Swal from "sweetalert2";

const ProductList: React.FC = () => {
  const [products, setProducts] = useState<Product[]>([]);

  useEffect(() => {
    const fetchProducts = async () => {
      const query = `[_type == "product"]{
        _id,
        productName,
        slug,
        category,
        price,
        "imageUrl": image.asset->url
      }`;

      try {
        const data: Product[] = await client.fetch(query);
        setProducts(data);
      } catch (error) {
        console.error("Error fetching products:", error);
      }
    };

    fetchProducts();
  }, []);

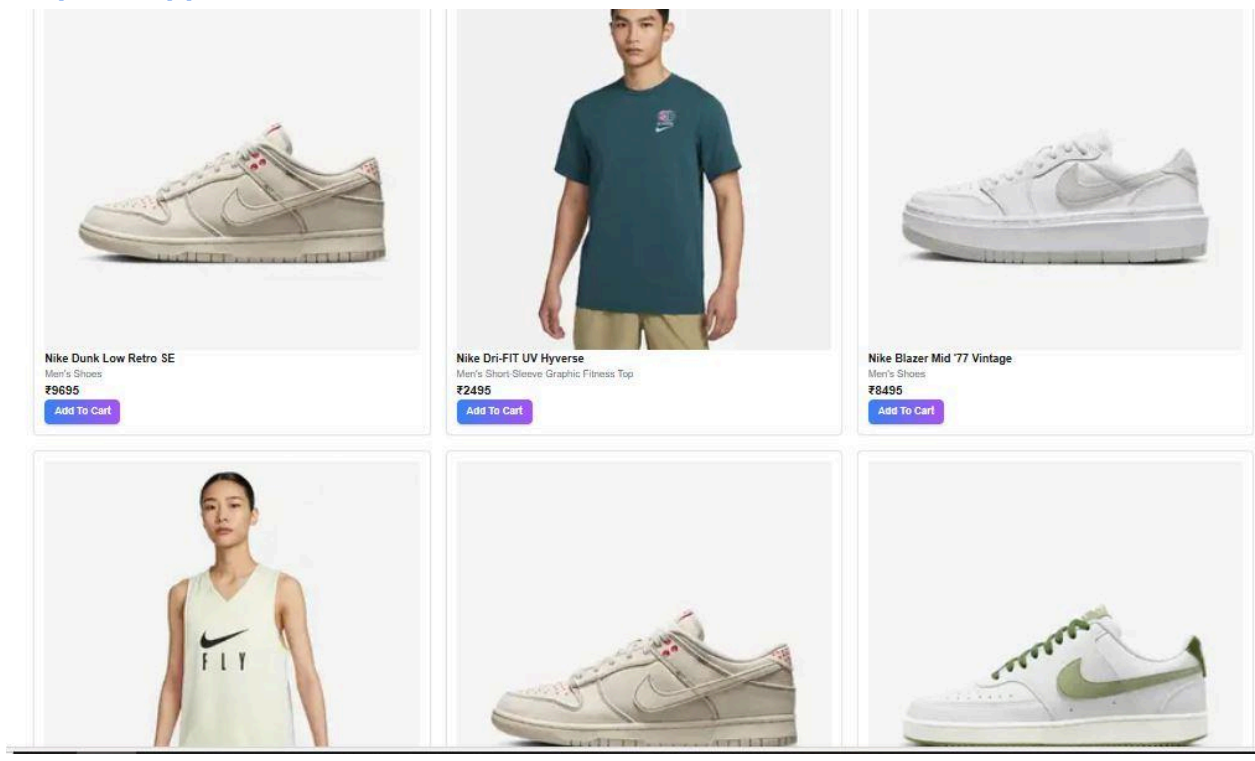
  const handleAddToCart = (e: React.MouseEvent, product: Product) => {
    e.preventDefault();
    Swal.fire({
      position: "top-right",
      icon: "success",
      title: `${product.productName} added to cart`,
      showConfirmButton: false,
      timer: 1000
    });

    addToCart(product)
  }

  return (
```

```
return (  
  <div className="grid grid-cols-1 md:grid-cols-2 lg:grid-cols-3 gap-6">  
    {products.map((product) => (  
      <Link key={product.id} href={` /product/${product.slug.current}`}>  
        <div className="border p-4 rounded-lg shadow cursor-pointer">  
          {product.imageUrl && (  
            <Image  
              src={product.imageUrl}  
              alt={product.productName}  
              width={200}  
              height={200}  
              className="w-full h-auto"  
            />  
          )}  
          <h2 className="text-lg font-semibold">{product.productName}</h2>  
          <p className="text-gray-500">{product.category}</p>  
          <p className="text-xl font-bold">₹{product.price}</p>  
          <button className="bg-gradient-to-r from-blue-500 to-purple-500  
            text-white font-semibold py-2 px-4 rounded-lg shadow-md hover:shadow-lg  
            hover:scale-110 transition-transform duration-200 ease-in-out"  
            onClick={(e) => handleAddToCart(e, product)}>  
            Add To Cart  
          </button>  
        </div>  
      </Link>  
    )}  
  </div>  
);  
};  
export default ProductList;
```

Output Snippet:



## Product Detail Component:

### Code Snippet:

```
import { client } from "@sanity/lib/client";
import { Product } from "../../types/products";
import { groq } from "next-sanity";
import Image from "next/image";
import { urlFor } from "@sanity/lib/image";
import { notFound } from "next/navigation";
import Link from "next/link";

interface ProductPageProps {
  params: { slug: string };
}

// 📄 Fetch Product Data from Sanity
async function getProduct(slug: string): Promise<Product | null> {
  const product = await client.fetch(
    groq`
    *[_type == "product" && slug.current == $slug][0] {
      _id,
      productName,
      category,
      description,
      slug,
      price,
      image
    }
  `,
    { slug }
  );
  return product;
}

// 🏠 Product Page Component
export default async function ProductPage({ params }: ProductPageProps) {
  const product = await getProduct(params.slug);

  // ⚠️ Handle 404 Not Found
  if (!product) {
    notFound();
  }

  return (
    <div className="max-w-6xl mx-auto px-4 py-10">
      <div className="grid grid-cols-1 md:grid-cols-2 gap-8">
```

```
<div className="grid grid-cols-1 md:grid-cols-2 gap-8">
  {/* 🖼️ Product Image */}
  <div className="relative aspect-square w-full max-w-md mx-auto">
    {product.image && (
      <Image
        src={urlFor(product.image).url()}
        alt={product.productName}
        width={500}
        height={500}
        className="rounded-lg shadow-md object-cover w-full h-full"
      />
    )}
  </div>

  {/* 📄 Product Details */}
  <div className="flex flex-col gap-6">
    <h1 className="text-3xl font-bold">{product.productName}</h1>
    <p className="text-lg text-gray-500">{product.category || "Uncategorized"}</p>
    <p className="text-gray-700">{product.description || "No description available"}</p>
    <p className="text-2xl font-semibold text-green-600">
      {product.price ? `₹${product.price}` : "Price not available"}
    </p>

    {/* 🛒 Add to Cart Button */}
    <button className="bg-blue-600 text-white px-6 py-2 rounded-lg hover:bg-blue-700 transition">
      <Link href="/cart"> Add to Cart</Link>
    </button>
  </div>
</div>
</div>
);
}
```

Output Snippet:



Nike Dri-FIT UV Hyverse

Men's Short-Sleeve Graphic Fitness Top

The Nike Dri-FIT UV Hyverse graphic top offers comfort and UV protection in style. Made from soft, breathable fabric, this fitness top keeps you cool and dry during workouts or casual outings.

₹2495

Add to Cart

## Cart Component:

### Code Snippet:

```
const CartPage = () => {
  const [cartItems, setCartItems] = useState<Product[]>([]);

  useEffect(() => {
    setCartItems(getCartItems());
  }, []);

  const handleRemove = (id: string) => {
    Swal.fire({
      title: "Are you sure?",
      text: "You won't be able to undo this action!",
      icon: "warning",
      showCancelButton: true,
      confirmButtonColor: "#d33",
      cancelButtonColor: "#3085d6",
      confirmButtonText: "Yes, remove it!",
    }).then((result) => {
      if (result.isConfirmed) {
        removeFromCart(id);
        setCartItems(getCartItems());
        Swal.fire(
          "Removed!",
          "Item has been removed from your cart.",
        );
      }
    });
  };
}
```

### Output Snippet:

<b>Nike Court Vision Low</b> Price: \$5695 - 1 +	Remove
<b>Nike Air Force 1 Mid '07</b> Price: \$10795 - 1 +	Remove
<b>Nike Dri-FIT UV Hyverse</b> Price: \$2495 - 2 +	Remove
<b>Total:</b>	<b>\$21480.00</b>
Proceed	

Cart > Checkout

### Order Summary

Nike Court Vision Low Quantity: 1	\$5695
<hr/>	
Nike Air Force 1 Mid '07 Quantity: 1	\$10795
<hr/>	
Nike Dri-FIT UV Hyverse Quantity: 2	\$4990
<hr/>	
Subtotal: \$21480	
Discount: -\$0	
<b>Total: \$21480.00</b>	

## Checkout Flow Component:

## Code Snippet:

```
export default function CheckoutPage() {
  const [discount, setDiscount] = useState<number>(0);
  const [formValues, setFormValues] = useState({
    firstName: "",
    lastName: "",
    address: "",
    city: "",
    zipCode: "",
    phone: "",
    email: ""
  });

  const [formErrors, setFormErrors] = useState({
    firstName: false,
    lastName: false,
    address: false,
    city: false,
    zipCode: false,
    phone: false,
    email: false
  });

  useEffect(() => {
    setCartItems(getCartItems());
    const appliedDiscount = localStorage.getItem("appliedDiscount");
    if (appliedDiscount) {
      setDiscount(Number(appliedDiscount));
    }
  }, []);

  const subtotal = cartItems.reduce(
    (total, item) => total + item.price * item.inventory,
    0
  );
  const total = Math.max(subtotal - discount, 0);

  const handleInputChange = (e: React.ChangeEvent<HTMLInputElement>) => {
    setFormValues({
      ...formValues,
      [e.target.id]: e.target.value
    });
  };

  const validateForm = () => {
    const errors = {
      firstName: !formValues.firstName,
      lastName: !formValues.lastName,
      address: !formValues.address,
    };
  };
}
```



```

      <input
        id="firstName"
        placeholder="Enter your first name"
        value={formValues.firstName}
        onChange={handleInputChange}
        className="border"
      />
      {formErrors.firstName && (
        <p className="text-sm text-red-500">
          First name is required.
        </p>
      )}
    </div>
    <div>
      <label htmlFor="lastName">Last Name </label>
      <input
        id="lastName"
        placeholder="Enter your last name"
        value={formValues.lastName}
        onChange={handleInputChange}
      />
      {formErrors.lastName && (
        <p className="text-sm text-red-500">
          Last name is required.
        </p>
      )}
    </div>
  </div>
  <div>
    <label htmlFor="address">Address </label>
    <input
      id="address"
      placeholder="Enter your address"
      value={formValues.address}
      onChange={handleInputChange}
    />
    {formErrors.address && (
      <p className="text-sm text-red-500">Address is required.</p>
    )}
  </div>
  <div>
    <label htmlFor="city">City</label>
    <input
      id="city"
      placeholder="Enter your city"
      value={formValues.city}
      onChange={handleInputChange}
    />
  </div>

```

## Output Snippet:

### Billing Information

First Name

Last Name

Address

City

Zip Code

Phone

Email

Place Order



## Header Components:

## Code Snippet:

```
Header.tsx
src > app > components > Header.tsx ...
1 import Link from "next/link";
2 import Image from "next/image";
3 import { FaSearch, FaHeart, FaShoppingBag } from "react-icons/fa"; // Import the icons
4
5
6 const Header = () => {
7   return (
8     <div>
9       {/* Upper Header */}
10      <div className="flex justify-between items-center px-6 py-2 md:text-[11px] sm:text-[9px] text-[8px] font-medium text-[#111111] font-Helvetica Neue ">
11        <Image src="/images/frame.png" alt="Upperheader Logo" width={24} height={24} />
12        <div className="flex md:gap-4 sm:gap-3 gap-2">
13          <Link href="#" className="hover:text-[#111111]">
14            Find Link Store
15          </Link>
16          <span className="text-[#111111]"></span>
17          <Link href="/Help" className="hover:text-[#111111]">
18            Help
19          </Link>
20          <span className="text-[#111111]"></span>
21          <Link href="/JoinUs" className="hover:text-[#111111]">
22            Join Us
23          </Link>
24          <span className="text-[#111111]"></span>
25          <Link href="/SignIn" className="hover:text-[#111111]">
26            Sign In
27          </Link>
28        </div>
29      </div>
30    </div>
31
32    {/* Main navigation */}
33    <div className="flex justify-between items-center px-6 py-4">
34      {/* Left section (Logo) */}
35      <div className="flex items-center">
36        <Image src="/images/Vector.png" alt="Nike Logo" className="md:w-[78px] sm:w-[60px] w-[30px]" width={78} height={78} />
37      </div>
38
39      {/* Center section (Navigation Links) */}
40      <nav className="flex gap-4 md:gap-6 text-gray-700 font-bold md:text-[16px] sm:text-[14px] text-[10px]">
41        <ul className="flex space-x-6">
42          <li>
43            <Link href="/New & Featured" />
44          </li>
45          <li>
46            <Link href="/all-products" />Men</Link>
47          </li>
48        </ul>
49      </nav>
50    </div>
51  );
52}
```

## Output Snippet:



[Find Link Store](#) | [Help](#) | [Join Us](#) | [Sign In](#)



[New & Featured](#) [Men](#) [Women](#) [Kids](#) [Sale](#) [SNKRS](#)

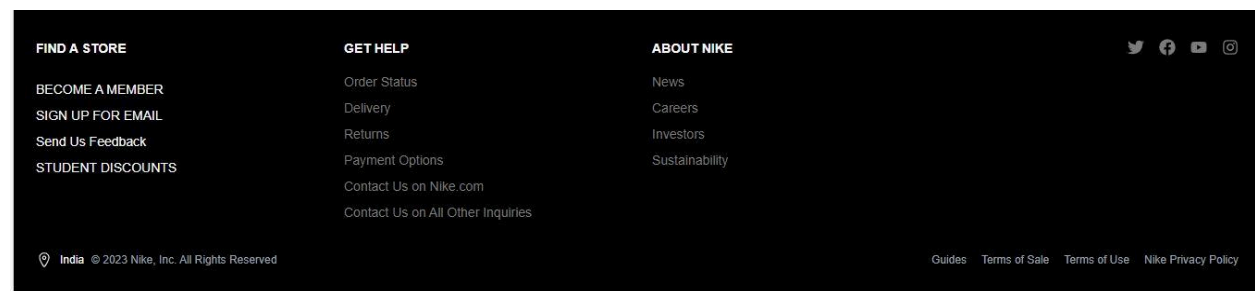


## Footer Components:

### Code Snippet:

```
Footertsx X
src > app > components > Footertsx > ...
1 import Link from 'next/link';
2 import React from 'react';
3 import { FaFacebook, FaInstagram, FaTwitter, FaYoutube } from 'react-icons/fa';
4 import { IoLocationOutline } from 'react-icons/io5';
5
6 const Footer = () => {
7   return (
8     <div className="bg-black text-[#FFFFFF] py-8 px-6">
9       { /* Top Section */ }
10      <div className="grid grid-cols-1 md:grid-cols-4 gap-8">
11        { /* Column 1 */ }
12        <div>
13          <h3 className="text-sm font-semibold mb-4">FIND A STORE</h3>
14          <ul className="space-y-2 text-sm mt-6">
15            <li><Link href="#" className="hover:underline mt-6">BECOME A MEMBER</Link></li>
16            <li><Link href="#" className="hover:underline mt-6">SIGN UP FOR EMAIL</Link></li>
17            <li><Link href="#" className="hover:underline mt-6">Send Us Feedback</Link></li>
18            <li><Link href="#" className="hover:underline mt-6">STUDENT DISCOUNTS</Link></li>
19          </ul>
20        </div>
21
22        { /* Column 2 */ }
23        <div>
24          <h3 className="text-sm font-semibold mb-4">GET HELP</h3>
25          <ul className="space-y-2 text-sm text-[#7E7E7E]">
26            <li><Link href="#" className="hover:underline">Order Status</Link></li>
27            <li><Link href="#" className="hover:underline">Delivery</Link></li>
28            <li><Link href="#" className="hover:underline">Returns</Link></li>
29            <li><Link href="#" className="hover:underline">Payment Options</Link></li>
30            <li><Link href="#" className="hover:underline">Contact Us on Nike.com</Link></li>
31            <li><Link href="#" className="hover:underline">Contact Us on All Other Inquiries</Link></li>
32          </ul>
33        </div>
34
35        { /* Column 3 */ }
36        <div>
37          <h3 className="text-sm font-semibold mb-4">ABOUT NIKE</h3>
38          <ul className="space-y-2 text-sm text-[#7E7E7E]">
39            <li><Link href="#" className="hover:underline">News</Link></li>
40            <li><Link href="#" className="hover:underline">Careers</Link></li>
41            <li><Link href="#" className="hover:underline">Investors</Link></li>
42            <li><Link href="#" className="hover:underline">Sustainability</Link></li>
43          </ul>
44        </div>
45      </div>
46    </div>
47  );
48}
```

### Output Snippet:



## Social Media Sharing Component:

### Code Snippet:

```
{/* Column 4 (Social Media Icons) */}  
<div className="flex justify-start lg:justify-end items-start gap-4" text-["#7E7E7E"]>  
  <a href="#" aria-label="Twitter"><FaTwitter className="text-lg cursor-pointer" hover:text-gray-400 /></a>  
  <a href="#" aria-label="Facebook"><FaFacebook className="text-lg cursor-pointer" hover:text-gray-400 /></a>  
  <a href="#" aria-label="YouTube"><FaYoutube className="text-lg cursor-pointer" hover:text-gray-400 /></a>  
  <a href="#" aria-label="Instagram"><FaInstagram className="text-lg cursor-pointer" hover:text-gray-400 /></a>  
</div>  
</div>
```

### Output Snippet:



### API Integration:

**1. Sanity API Fetching Logic:** Since you're using Sanity CMS, you need to ensure data fetching is properly managed.

- **Sanity Client Setup** (if not already set up): In your lib folder, configure the Sanity client:

```
NEXT_PUBLIC_SANITY_PROJECT_ID=  
NEXT_PUBLIC_SANITY_DATASET="production"  
SANITY_API_TOKEN=
```

## Dynamic Routing:

Dynamic routing is used to handle routes like /Productlist for individual product pages.

### 1. Create Dynamic Route File:

In the pages directory, create a [slug] file inside the Product/[slug]

```
import { useEffect, useState } from "react";
import { client } from "@sanity/lib/client";
import Image from "next/image";
import Link from "next/link";
import { Product } from "../../types/products";
import { addToCart } from "../../actions/actions";
import Swal from "sweetalert2";

const ProductList: React.FC = () => {
  const [products, setProducts] = useState<Product[]>([]);

  useEffect(() => {
    const fetchProducts = async () => {
      const query = `*[_type == "product"]{
        _id,
        productName,
        slug,
        category,
        price,
        "imageUrl": image.asset->url
      }`;

      try {
        const data: Product[] = await client.fetch(query);
        setProducts(data);
      } catch (error) {
        console.error("Error fetching products:", error);
      }
    };

    fetchProducts();
  }, []);
```

```
// 🐛 Product Page Component
export default async function ProductPage({ params }: ProductPageProps) {
  const product = await getProduct(params.slug);

  // ⚠️ Handle 404 Not Found
  if (!product) {
    notFound();
  }

  return (
    <div className="max-w-6xl mx-auto px-4 py-10">
      <div className="grid grid-cols-1 md:grid-cols-2 gap-8">
        {/* 📷 Product Image */}
        <div className="relative aspect-square w-full max-w-md mx-auto">
          {product.image && (
            <Image

```

## 2. Get Product Data Dynamically:



### Nike Metcon 8

Men's Training Shoes

The Nike Metcon 8 is built for intense training sessions. Featuring a stable base, durable materials, and excellent traction, it's the perfect companion for weightlifting, CrossFit, or HIIT workouts.

₹10295

Add to Cart

## Final Checklist:

1. Frontend Component Development : ☒
2. Styling and Responsiveness : ☒
3. Code Quality : ☒
4. Documentation and Submission: ☒
5. Final Review : ☒