

DAY 4 - BUILDING DYNAMIC FRONTEND COMPONENTS FOR YOUR MARKETPLACE

Key Components to Build

1. Product Listing Component.
2. Product Detail Component.
3. Cart Component.
4. Checkout WorkFlow Component.
5. Header Components.

1. Product listing Component:

The Product Listing Component displays all available products in an organized manner, allowing users to browse efficiently. 🔍 It fetches data dynamically and ensures a smooth shopping experience with clear visuals and essential details like price, name, and image.

```
import { Product } from "../../types/products";
import { groq } from "next-sanity";

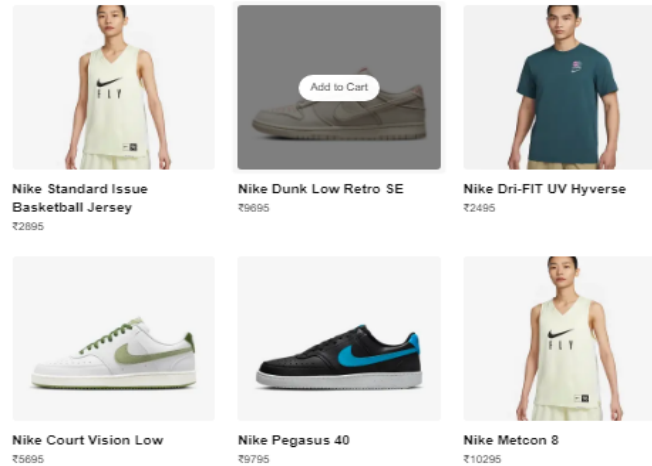
Codeium: Refactor | Explain
interface ProductCard {
  image: string;
  name: string;
  title: string;
  description: string;
  description2?: string;
  color: string;
  price: string;
}

Codeium: Refactor | Explain | Generate JSDoc
export default function ProductCard() {

const [product, setProduct] = useState<Product[]>([])
useEffect(() => {
  async function productData() {
    Codeium: Refactor | Explain | Generate JSDoc
    const productFetchData: Product[] = await client.fetch(groq`*[_type == "product"]{
      productName,
      price,
      image,
      color,
      description,
      inventory,
      category,
      status,
      _id
    }`);
    setProduct(productFetchData);
  }
  productData();
}, []);

const dispatch = useAppDispatch();
```

Output:



2. Product Detail Component:

The Product Detail Component provides in depth information about a specific product, including images, price, description, and available options. 🛒 It enhances the user experience by displaying key details, customer reviews, and an "Add to Cart" button for seamless purchasing.

```
"use client"
import { client } from "@sanity/lib/client";
import { fetchProduct } from "@sanity/lib/queries";
import { groq } from "next-sanity";
import { useEffect, useState } from "react";
import { Product } from "../../types/products";
import Image from "next/image";
import { useRouter } from "next/router";
import { useAppDispatch } from "../../hooks/redux";
import { addToCart } from "../../redux/cartSlice";

const Page = (props: { params: { id: string } }) => {
  const dispatch = useAppDispatch();

  const [product, setProduct] = useState<Product | null>(null);
  const [count, setCount] = useState(1);

  const handleIncrement = () => {
    setCount(count + 1);
  };

  const handleDecrement = () => {
    setCount(count - 1);
  };

  useEffect(() => {
    const fetchProductData = async () => {
      try {
        // Unwrap the params promise if needed
        const resolvedParams = await props.params;
        const id = resolvedParams.id;
        console.log("Product ID:", id);

        // Fetch product data using Sanity client
        const productFetchData: Product[] = await client.fetch(
          groq`*[_type == "product" && _id == $id]{
            _id,
            productName,
            _type,
            image,
            price,
            description
          }`,
          { id } // Pass id as a parameter to avoid string interpolation for security
        );
      } catch (error) {
        console.error("Error fetching product data:", error);
      }
    };

    fetchProductData();
  }, [props.params]);

  console.log(product);

  if (!product) {
    return (
      <div className="loading-container">
        <div className="spinner"></div>
        <p>Loading...</p>
      </div>
    );
  }
}
```

```
if (productFetchData.length > 0) {
  setProduct(productFetchData[0]); // Assuming only one product matches the ID
} else {
  console.error("No product found for the given ID.");
}

catch (error) {
  console.error("Error fetching product data:", error);
}

fetchProductData();
}, [props.params]);
console.log(product);

if (!product) {
  return (
    <div className="loading-container">
      <div className="spinner"></div>
      <p>Loading...</p>
    </div>
  );
}
```

Output:




Nike Court Vision Low


The Nike Court Vision Low blends basketball-inspired style with everyday comfort. Featuring a sleek design and durable materials, this versatile shoe is perfect for any casual wardrobe.

₹ 5695

Total: ₹ 5695

 Add To Cart

3. Cart Component:

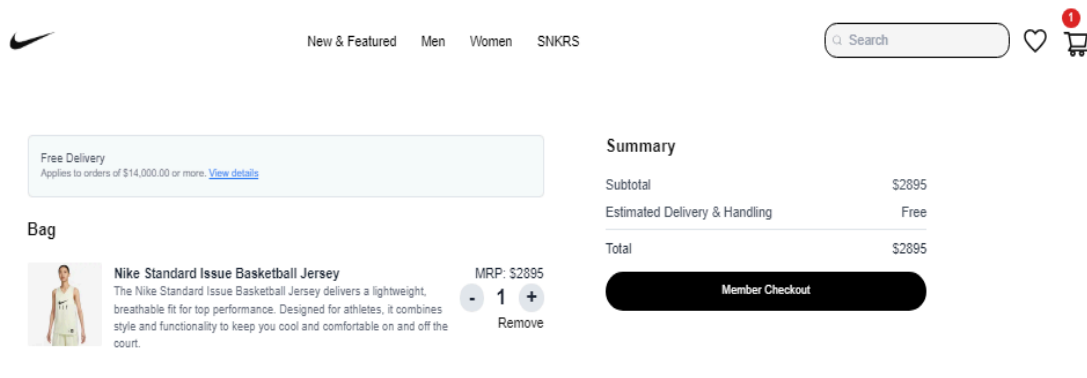
The Cart Component allows users to view and manage the products they have added for purchase.  It displays item details, quantities, total price, and provides options to update or remove items before proceeding to checkout.

```
"use client";
import Link from "next/link";
import Image from "next/image";
import { useState } from "react";
import { useAppDispatch, useAppSelector } from "../../hooks/redux";
import { decrementQuantity, getCart, incrementQuantity, removeFromTheCart } from "../../redux/cartSlice";
import { Product } from "../../types/products";
import { urlFor } from "@sanity/lib/image";
```

Codeium: Refactor | Explain | Generate JSDoc

```
export default function Bag() {
  const productCart = useAppSelector(getCart);
  const dispatch = useAppDispatch();
  let totalPrice = 0;
  productCart.forEach((item: Product) => {
    totalPrice += item.price * item.quantity;
  });
}
```

Output:



4. Checkout WorkFlow Component:

The Checkout WorkFlow Component streamlines the purchasing process by guiding users through payment, delivery details, and order confirmation. It ensures a smooth and secure transaction, integrating multiple payment options and real time order validation.

```
"use client"
import Image from "next/image";
import { useAppSelector } from "../../hooks/redux";
import { getCart } from "../../redux/cartSlice";
import { Product } from "../../types/products";
import { urlFor } from "@sanity/lib/image";

Codeium: Refactor | Explain | Generate JSDoc
export default function Checkout() {
  const productCart = useAppSelector(getCart);
  let totalPrice = 0;
  productCart.forEach((item: Product) => {
    totalPrice += item.price * item.quantity;
  });
}
```

Output:

Order Summary

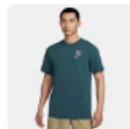
Subtotal	\$21580
Delivery/Shipping	Free
<hr/>	
Total	\$21580

(*The total reflects the price of your order, including all duties and taxes)

Arrives Mon, 27 Mar - Wed, 12 Apr



Nike Standard Issue Basketball Jersey
Qty 2
\$2895



Nike Dri-FIT UV Hyverse
Qty 1
\$2495



Nike Air Force 1 React
Qty 1
\$13295

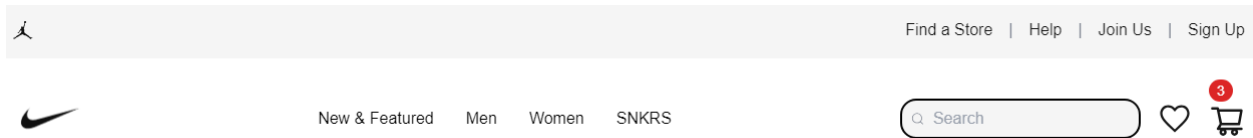
5. Header Component:

The Header Component serves as the navigation bar of the website, providing quick access to essential sections like Home, Categories, Cart, and User Profile. It enhances user experience by ensuring easy and intuitive navigation across the platform.

```
"use client";
import Image from "next/image";
import Link from "next/link";
import { CiSearch } from "react-icons/ci";
import { AiOutlineShoppingCart } from "react-icons/ai";
import { IoMdHeartEmpty } from "react-icons/io";
import { useAppSelector } from "../../hooks/redux";
import { getCart } from "../../redux/cartSlice";

Codeium: Refactor | Explain | Generate JSDoc
export default function Header() {
  const cartItem = useAppSelector(getCart);
```

Output:



Conclusion:

In summary, the Marketplace Website Structure is built with essential components like **Product Listing, Product Detail, Cart, Checkout Workflow, and Header** to ensure a seamless shopping experience. Each component plays a crucial role in enhancing user interaction, from browsing products to completing purchases. With a well defined structure, the platform ensures smooth navigation, easy product discovery, and a hassle free checkout process, making online shopping more efficient and enjoyable.