

Day 4 - Building Dynamic Frontend Components for my Marketplace

Objective

On Day 4, we focus on designing and developing dynamic frontend components to display marketplace data fetched from Sanity CMS or APIs. The goal is to create **modular, reusable, and scalable components** while ensuring a **responsive user experience**.

Key Learning Outcomes

1. Build dynamic frontend components that display data from Sanity CMS or APIs.
 2. Implement reusable and modular components.
 3. Apply state management techniques.
 4. Ensure responsive design and follow UI/UX best practices.
 5. Prepare for real-world client projects by replicating professional workflows.
-

Key Components to Build

1. Product Listing Component

- Dynamically render product data in a grid layout.
- Include fields such as:
 - Product Name
 - Price
 - Image
 - Stock Status
- Implement a card-based layout displaying essential product details.

2. Product Detail Component

- Create individual product detail pages using **Next.js dynamic routing**.
- Include detailed fields such as:
 - Product Description
 - Price
 - Available Sizes/Colors
- Enable interactive features like image zoom and product variations.



Jacket Dress

Department

\$23 \$12.5



Designer Dress

Department

\$25.5 \$19.5



Formal Gown

Department

\$25 \$15



Summer Dress

Department

\$30 \$20

```
src
├── component
├── about-components
├── cart-components
│   └── CartContext.tsx
├── contact-components
├── home-components
│   ├── editors-pics.tsx
│   ├── feature-posts.tsx
│   ├── feature-products.tsx
│   ├── footer.tsx
│   ├── header.tsx
│   ├── hero.tsx
│   ├── loader.tsx
│   ├── product-card.tsx
├── pricing-component
├── productList-components
├── products-components
│   ├── best-seller.tsx
│   ├── details.tsx
│   ├── hero.tsx
└── team-components
```

```
22 type CartContextType = {
30   clearWishlist: () => void;
31   decreaseQuantity: (id: string) => void;
32 };
33
34 const CartContext = createContext<CartContextType | undefined>(undefined);
35
36 export const CartProvider = ({ children }: { children: ReactNode }) => {
37   const [cartItems, setCartItems] = useState<CartItem[]>([]);
38   const [wishlist, setWishlist] = useState<CartItem[]>([]);
39
40   // Load cart and wishlist from localStorage on initial render
41   useEffect(() => {
42     const storedCart = localStorage.getItem("cartItems");
43     const storedWishlist = localStorage.getItem("wishlistItems");
44     if (storedCart) {
45       setCartItems(JSON.parse(storedCart));
46     }
47     if (storedWishlist) {
48       setWishlist(JSON.parse(storedWishlist));
49     }
50   }, []);
51 }
```

1. About Components

AboutUs

This component introduces the company, providing a brief overview of its mission and values. It may include an image and a CTA button for further engagement.

ABOUT COMPANY

ABOUT US

We know how large objects will act,
but things on a small scale

Get Quote Now



```
1 import Image from "next/image";
2 import React from "react";
3
4 const AboutUs = () => {
5   return (
6     <div className="flex flex-col md:flex-row items-center justify-between px-4 py-8 wrapper">
7       <div className="w-full text-center md:text-left md:w-1/2 md:ml-16">
8         <h3 className="text-[#252B42] font-bold text-[16px] mb-6">
9           ABOUT COMPANY
10        </h3>
11        <h1 className="text-[#252B42] font-bold text-[32px] sm:text-[48px] md:text-[58px] mb-6">
12          ABOUT US
13        </h1>
14        <p className="text-[#737373] text-[16px] sm:text-[18px] md:text-[20px] mt-5">
15          We know how large objects will act, <br />
16          but things on a small scale
17        </p>
18
19        <div className="mt-7 flex justify-center md:justify-start">
20          <button className="px-4 py-3 bg-[#23A6F0] text-[#FAFAFA] rounded-md hover:bg-blue-500">
21            Get Quote Now
22          </button>
23        </div>
24      </div>
25    </div>
26  );
27}
```

Companies

Displays partner companies or affiliates associated with the marketplace. It helps build credibility by showcasing collaborations.

MeetTeam

Highlights key team members with their roles and brief bios. This enhances transparency and trust among users.

2. Cart Components

CartContext

Manages the state of the shopping cart. It tracks items added to the cart, handles removal, and calculates the total price dynamically.



Precision Meets Style

This product is crafted with precision and care to meet your needs. It combines functionality with a sleek design, making it a perfect addition

to your daily essentials. Our commitment to quality ensures long-lasting durability. Each detail is meticulously designed to deliver a seamless and enjoyable user experience.

Whether for personal or professional use, this product stands out in style and performance. Choose from multiple options to suit your preferences.

Product Details

- > What materials are used
- > Does this product come
- > Are there any color options
- > Is this product eco-friendly

Order & Shipping

- > How long does delivery take
- > Can I track my order?
- > What are the shipping costs

```
src > component > cart-components > CartContext.tsx > ...
22  type CartContextType = {
30    clearWishlist: () => void;
31    decreaseQuantity: (id: string) => void;
32  };
33
34  const CartContext = createContext<CartContextType | undefined>(undefined);
35
36  export const CartProvider = ({ children }: { children: ReactNode }) => {
37    const [cartItems, setCartItems] = useState<CartItem[]>([]);
38    const [wishlist, setWishlist] = useState<CartItem[]>([]);
39
40    // Load cart and wishlist from localStorage on initial render
41    useEffect(() => {
42      const storedCart = localStorage.getItem("cartItems");
43      const storedWishlist = localStorage.getItem("wishlistItems");
44      if (storedCart) {
45        setCartItems(JSON.parse(storedCart));
46      }
47      if (storedWishlist) {
48        setWishlist(JSON.parse(storedWishlist));
49      }
50    }, []);
51  }
```

3. Contact Components

ContactUs

Provides a contact form for users to reach out to the company. It may include fields like name, email, message, and a submit button.

CONTACT US

Get in touch today!



```
import Image from "next/image";
import Link from "next/link";
import React from "react";
import { FaTwitter, FaFacebook, FaInstagram, FaLinkedin } from "react-icons/fa";

const ContactUs = () => {
  return (
    <div className="flex flex-col md:flex-row items-center justify-between px-4 py-8 wrapper">
      <div className="w-full text-center md:text-left md:w-1/2 md:ml-16">
        <h3 className="text-[#252B42] font-bold text-[16px] mb-6">CONTACT US</h3>
        <h1 className="text-[#252B42] font-bold text-[32px] sm:text-[48px] md:text-[58px] mb-6">
          Get in touch <br />
          today!
        </h1>
        <p className="text-[#737373] text-[16px] sm:text-[18px] md:text-[20px] mt-5">
          We know how large objects will act, <br />
          but things on a small scale
        </p>

        <div className="text-[#252B42] font-bold text-[24px] mt-8">
```

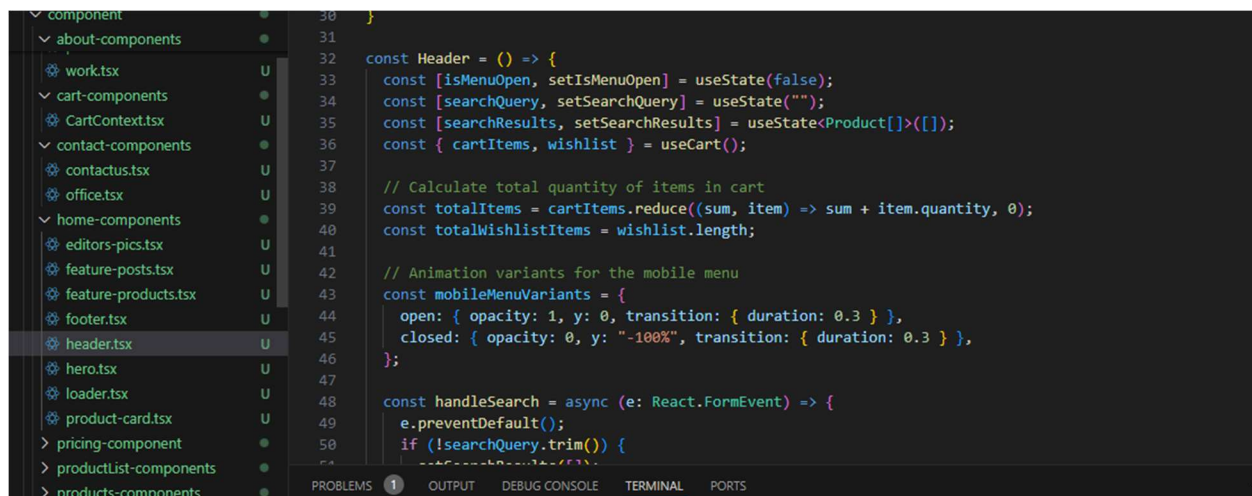
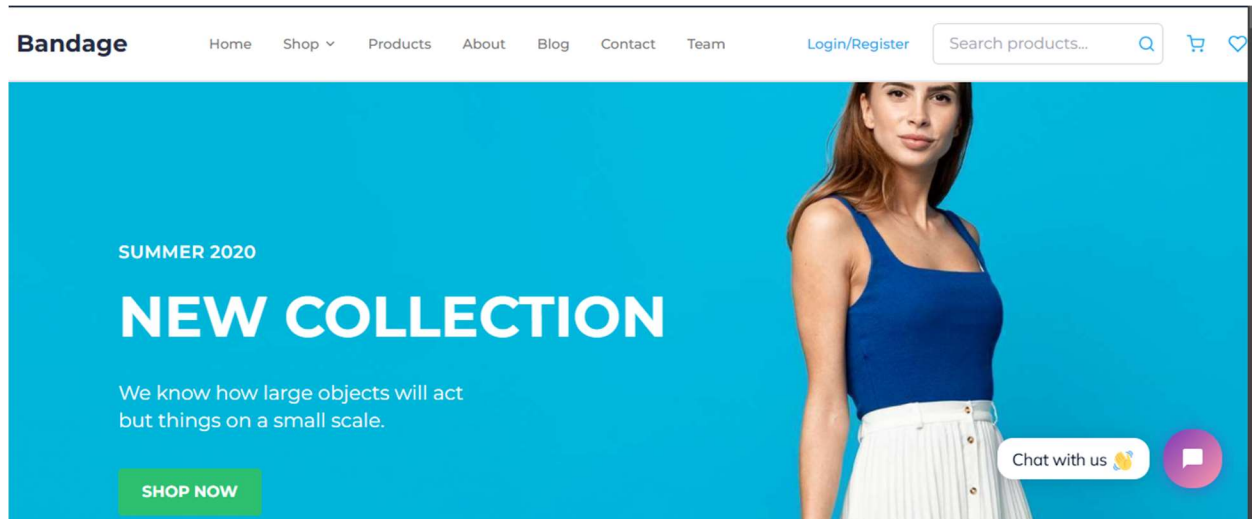
Office

Displays company office locations, including addresses, phone numbers, and maps for easier navigation.

4. Home Components

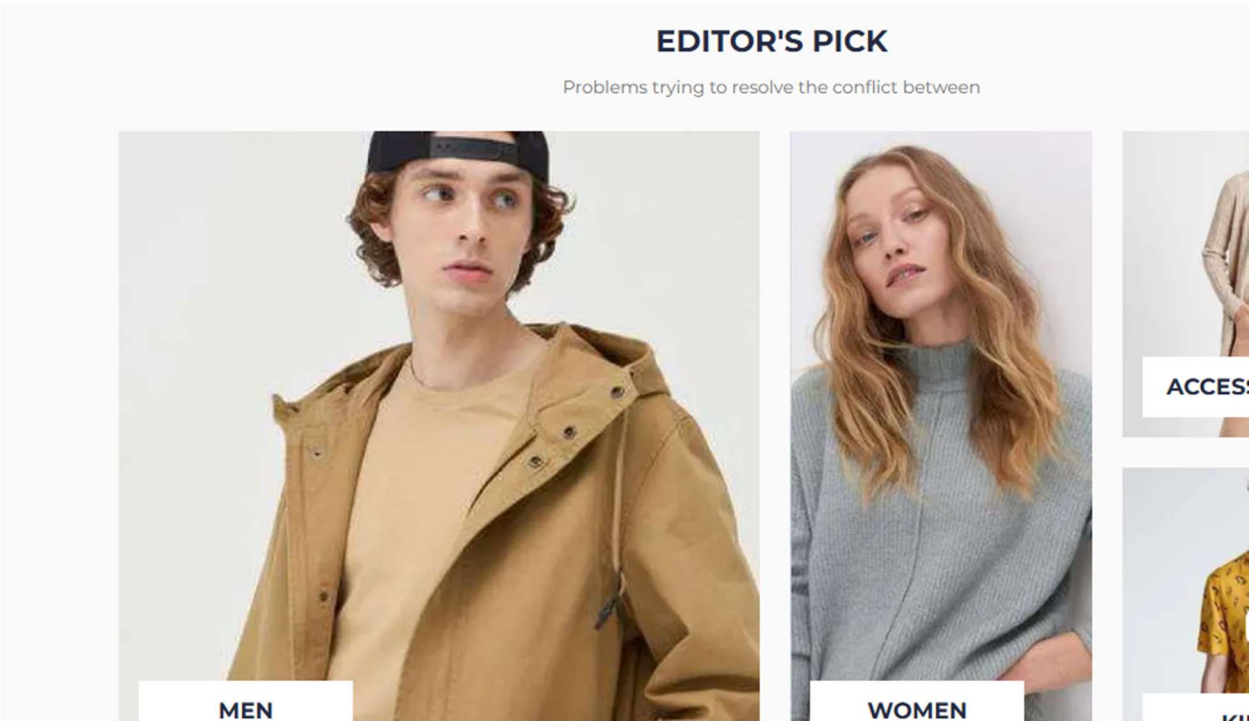
Header

Contains the website's navigation bar, including links to different sections, a search bar, and possibly a login/signup button.



Hero

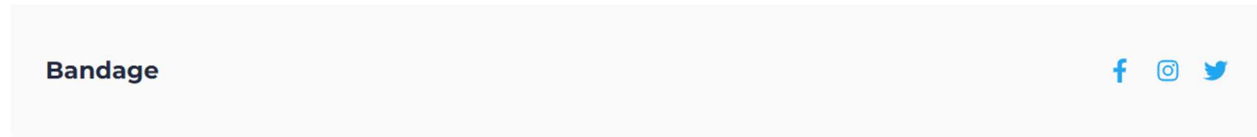
The main banner section of the homepage, featuring a compelling call-to-action (CTA), promotional content, and a background image.



```
1  import Link from "next/link";
2  import React from "react";
3  import { motion } from "framer-motion";
4
5
6  const Hero = () => {
7    // Animation variants for text content
8    const textVariants = {
9      hidden: { opacity: 0, y: 20 },
10     visible: { opacity: 1, y: 0, transition: { duration: 0.8, ease: "easeOut" } },
11   };
12
13   // Animation variants for the button
14   const buttonVariants = {
15     hidden: { opacity: 0, scale: 0.8 },
16     visible: { opacity: 1, scale: 1, transition: { duration: 0.8, delay: 0.5, ease: "easeOut" } },
17     hover: { scale: 1.05, transition: { duration: 0.2 } },
18     tap: { scale: 0.95 },
19   };
20
21   return (
```

Footer

Displays company information, useful links, social media handles, and other essential links at the bottom of the page.



- [About Us](#)
- [Carrier](#)
- [We are hiring](#)
- [Blog](#)

[Terms of Service](#)
[Privacy Policy](#)
[Cookie Policy](#)
[Compliance](#)

- Business Marketing
- User Analytics
- Live Chat
- Unlimited Support

- iOS & Android
- Watch a Demo
- Customers
- API

Your Email [Subscribe](#)

PricingComponent

Displays different pricing plans or packages available in the marketplace. It includes feature comparisons and CTA buttons for selecting a plan.

PRICING

Simple Pricing

[Home](#) > [Pricing](#)

Conclusion

These components are designed to create a modular, reusable, and dynamic frontend structure for the marketplace, aligning with Day 4's objectives of **state management, modular design, and responsive UI development**.

”Develop By”

Name : ”Muhammad Owais Shah”

Class : ”Saturday 2pm to 5pm”

Roll : 00183255

