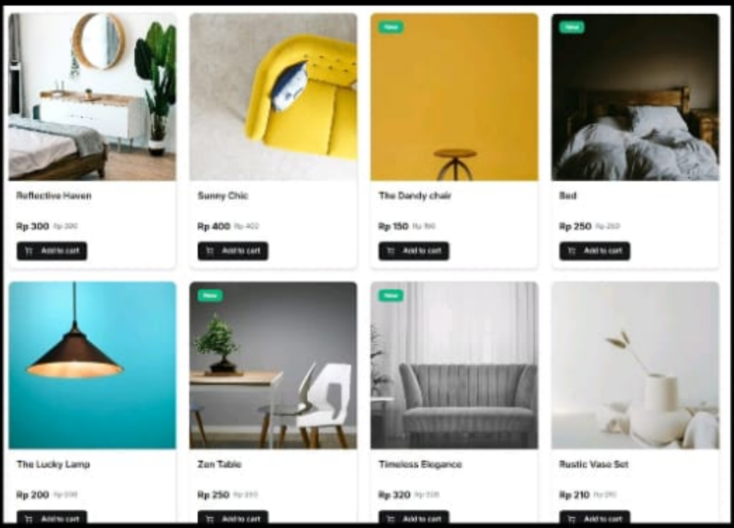


New

New

New

New

Reflective Haven

Rp 300 ~~Rp 300~~

Add to cart

Sunny Chic

Rp 400 ~~Rp 400~~

Add to cart

The Dandy chair

Rp 150 ~~[illegible]~~

Add to cart

Bed

Rp 250 ~~Rp 250~~

Add to cart

The Lucky Lamp

Rp 200 ~~[illegible]~~

Add to cart

Zen Table

Rp 250 ~~[illegible]~~

Add to cart

Timeless Elegance

Rp 320 ~~[illegible]~~

Add to cart

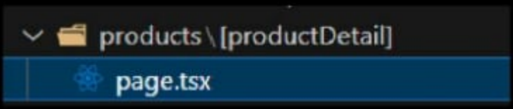
Rustic Vase Set

Rp 210 ~~[illegible]~~

Add to cart

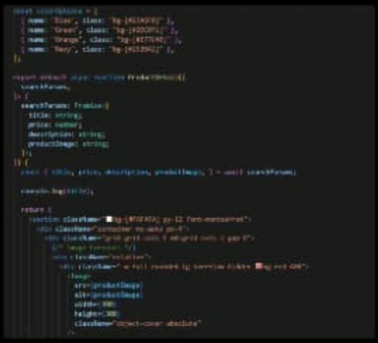
Fetch Data Successfully

I successfully fetched data from the API and displayed it dynamically on my website. This integration ensures real-time updates and seamless synchronization of data with the user interface

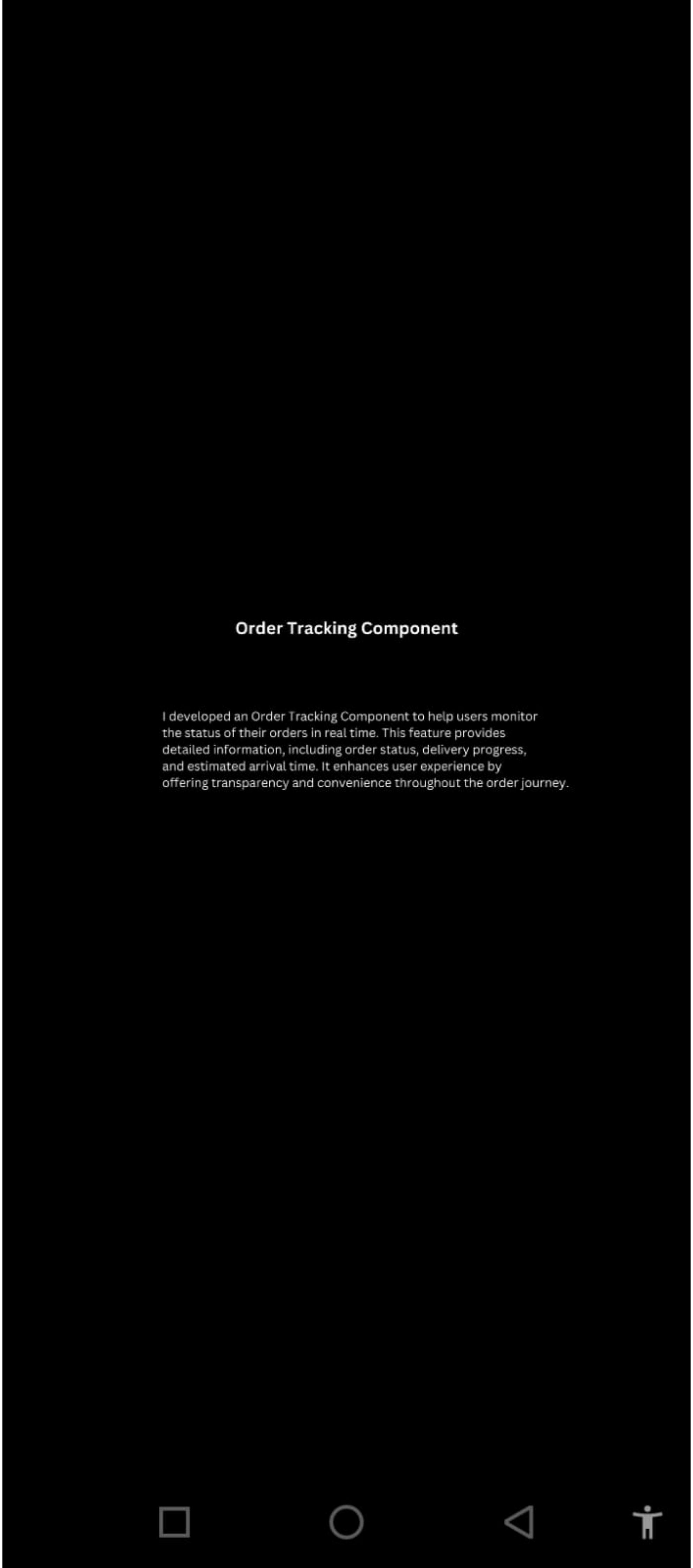


products \ [productDetail]

page.tsx

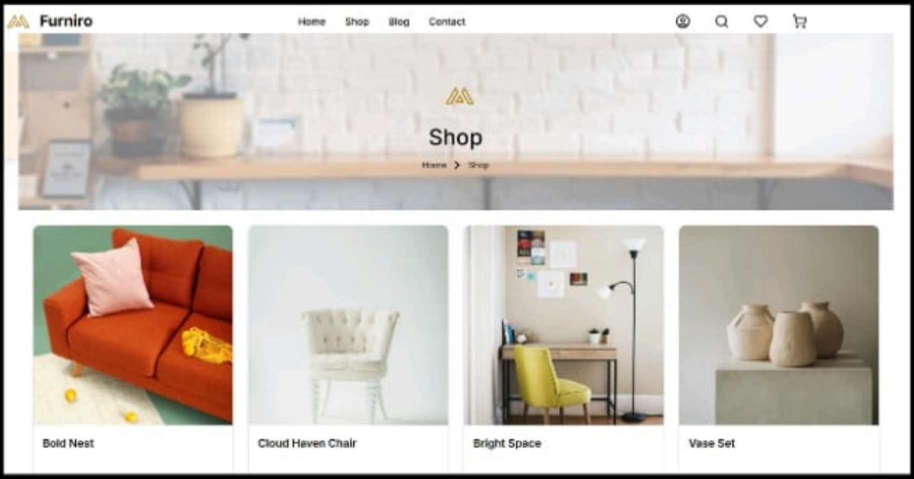


```
const colorOptions = [
  { name: 'Blue', class: "bg-[#23A6F0]" },
  { name: 'Green', class: "bg-[#2DC071]" },
  { name: 'Orange', class: "bg-[#E77C40]" },
  { name: 'Navy', class: "bg-[#252B42]" },
];

export default async function ProductDetail({
  searchParams,
}: {
  searchParams: Promise<{
    title: string;
    price: number;
    description: string;
    productImage: string;
  }>;
}) {
  const { title, price, description, productImage, } = await searchParams;

  console.log(title);

  return (
    <section className="bg-[#FAFAFA] py-12 font-montserrat">
      <div className="container mx-auto px-4">
        <div className="grid grid-cols-1 md:grid-cols-2 gap-8">
          {/* Image Carousel */}
          <div className="relative">
            <div className="w-full rounded-lg overflow-hidden bg-red-400">
              <Image
                src={productImage}
                alt={productImage}
                width={300}
                height={300}
                className="object-cover absolute"
              />
```



Order Tracking Component

I developed an Order Tracking Component to help users monitor the status of their orders in real time. This feature provides detailed information, including order status, delivery progress, and estimated arrival time. It enhances user experience by offering transparency and convenience throughout the order journey.


Furniro

Home Shop Blog Contact

Shop

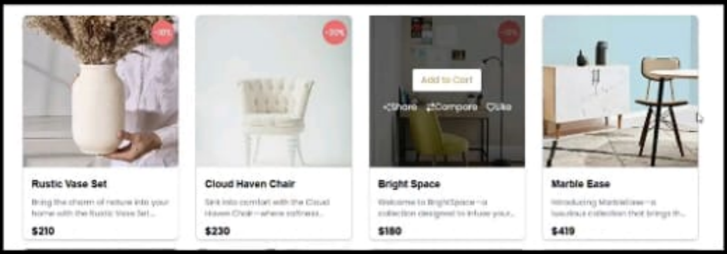
Home > Shop

Bold Nest

Cloud Haven Chair

Bright Space

Vase Set

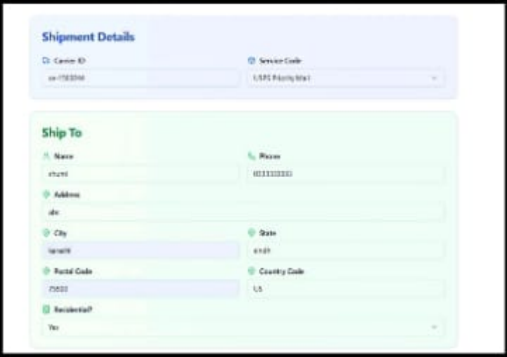


Rustic Vase Set
Bring the charm of nature into your home with the Rustic Vase Set...
$210

Cloud Haven Chair
Sink into comfort with the Cloud Haven Chair—where softness...
$230

Bright Space
Welcome to BrightSpace—a collection designed to infuse your...
$180

Marble Ease
Introducing MarbleEase—a luxurious collection that brings th...
$419

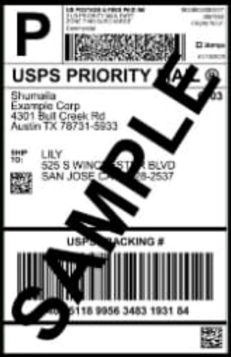


Shipment Details

Carrier ID	Service Code
se-1583046	USPS Priority Mail

Ship To

Name	Phone
shumi	03333333333
Address	
abc	
City	State
karachi	sindh
Postal Code	Country Code
75500	US
Residential?	
Yes	



P

USPS PRIORITY MAIL ®

Shumaila
Example Corp
4301 Bull Creek Rd
Austin TX 78731-5933

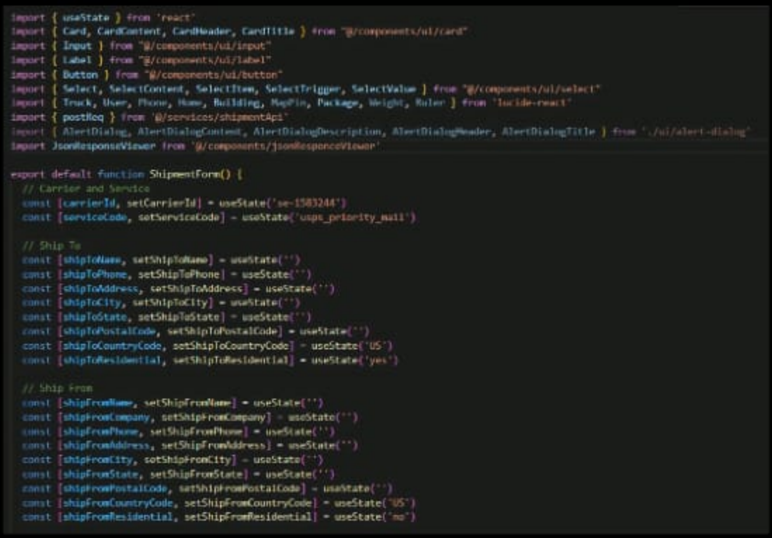
03

SHIP TO: LILY
525 S WINC ST R BLVD
SAN JOSE C 8-2537

USPS ACKING #

5118 9956 3483 1931 84

SAMPLE



```
import { useState } from 'react'
import { Card, CardContent, CardHeader, CardTitle } from "@/components/ui/card"
import { Input } from "@/components/ui/input"
import { Label } from "@/components/ui/label"
import { Button } from "@/components/ui/button"
import { Select, SelectContent, SelectItem, SelectTrigger, SelectValue } from "@/components/ui/select"
import { Truck, User, Phone, Home, Building, MapPin, Package, Weight, Ruler } from 'lucide-react'
import { postReq } from '@/services/shipmentApi'
import { AlertDialog, AlertDialogContent, AlertDialogDescription, AlertDialogHeader, AlertDialogTitle } from './ui/alert-dialog'
import JsonResponseViewer from '@/components/jsonResponceViewer'

export default function ShipmentForm() {
  // Carrier and Service
  const [carrierId, setCarrierId] = useState('se-1583244')
  const [serviceCode, setServiceCode] = useState('usps_priority_mail')

  // Ship To
  const [shipToName, setShipToName] = useState('')
  const [shipToPhone, setShipToPhone] = useState('')
  const [shipToAddress, setShipToAddress] = useState('')
  const [shipToCity, setShipToCity] = useState('')
  const [shipToState, setShipToState] = useState('')
  const [shipToPostalCode, setShipToPostalCode] = useState('')
  const [shipToCountryCode, setShipToCountryCode] = useState('US')
  const [shipToResidential, setShipToResidential] = useState('yes')

  // Ship From
  const [shipFromName, setShipFromName] = useState('')
  const [shipFromCompany, setShipFromCompany] = useState('')
  const [shipFromPhone, setShipFromPhone] = useState('')
  const [shipFromAddress, setShipFromAddress] = useState('')
  const [shipFromCity, setShipFromCity] = useState('')
  const [shipFromState, setShipFromState] = useState('')
  const [shipFromPostalCode, setShipFromPostalCode] = useState('')
  const [shipFromCountryCode, setShipFromCountryCode] = useState('US')
  const [shipFromResidential, setShipFromResidential] = useState('no')
```

DAY 4 - BUILDING DYNAMIC FRONTEND COMPONENTS FOR YOUR MARKETPLACE

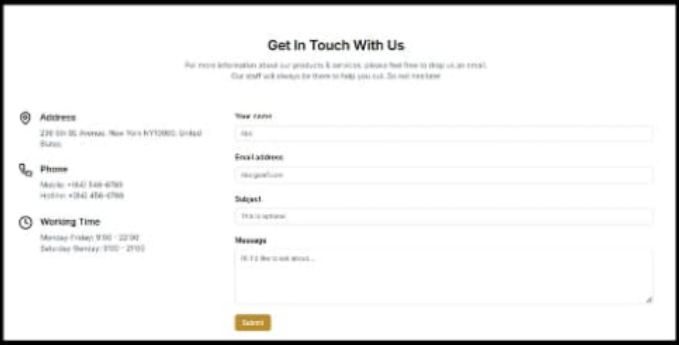
Date: 20.01.2025

Objective:

I focus on designing and developing dynamic frontend components
to display marketplace data fetched from Sanity CMS or APIs.
This step emphasizes modular, reusable component design and real-world
practices for building scalable and responsive web applications.

Contact Component

I have created a Contact component that accepts user requests.
I want to display it in my document along with a picture.



Get In Touch With Us

For more information about our products & services, please feel free to drop us an email.
Our staff will always be there to help you out. Do not hesitate

Address
236 5th SE Avenue, New York NY10000, United States

Phone
Mobile: +(84) 546-6789
Hotline: +(84) 456-6789

Working Time
Monday-Friday: 9:00 - 22:00
Saturday-Sunday: 9:00 - 21:00

Your name: Abc

Email address: Abc@def.com

Subject: This is optional

Message: Hi! I'd like to ask about...

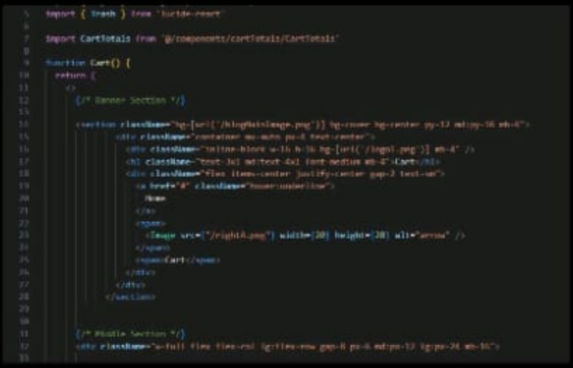
Submit

🔑 **Key Learnings:**

1️⃣ dynamic components data fetched from Sanity CMS and APIs.✓
2️⃣ Designed reusable, modular components for better scalability.✓
3️⃣ Implemented state management techniques for seamless functionality.✓
4️⃣ Focused on responsive design and UX/UI best practices.✓
5️⃣ Prepared for real-world client projects by simulating professional workflows.✓

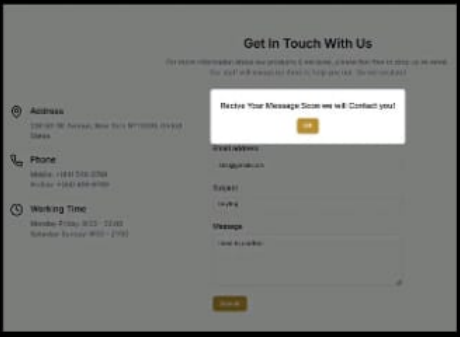
Cart Component

I implemented a cart feature in my website to allow users to add and manage documents easily. This functionality ensures an organized and user-friendly way to store and review selected documents before final actions.



```
import { Trash } from 'lucide-react'

import CartTotals from '@/components/cartTotals/CartTotals'

function Cart() {
  return (
    <>
      {/* Banner Section */}

      <section className="bg-[url('/blogMainImage.png')] bg-cover bg-center py-12 md:py-16 mb-6">
            <div className="container mx-auto px-4 text-center">
              <div className="inline-block w-16 h-16 bg-[url('/logo1.png')] mb-4" />
              <h1 className="text-3xl md:text-4xl font-medium mb-4">Cart</h1>
              <div className="flex items-center justify-center gap-2 text-sm">
                <a href="#" className="hover:underline">
                  Home
                </a>
                <span>
                  <Image src={"/right4.png"} width={20} height={20} alt="arrow" />
                </span>
                <span>Cart</span>
              </div>
            </div>
          </section>


      {/* Middle Section */}
      <div className="w-full flex flex-col lg:flex-row gap-8 px-6 md:px-12 lg:px-24 mb-16">
```

Confimation Reciving Replay as a popUp



Get In Touch With Us

[illegible]

Address

[illegible]

Phone

[illegible]

Working Time

[illegible]

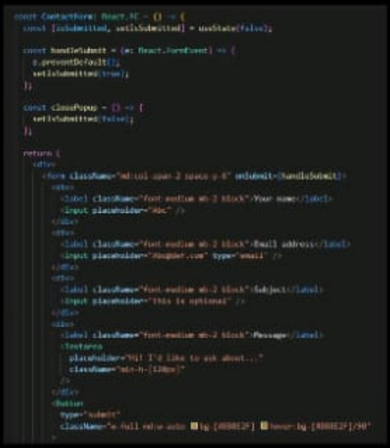
Recive Your Message Soon we will Contact you!

Email address

Subject

Message

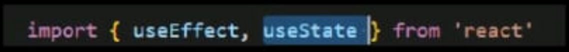
Contact Form Code



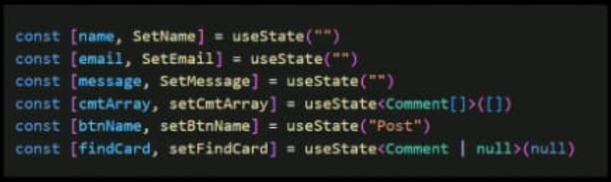
```
const ContactForm: React.FC = () => {
  const [isSubmitted, setIsSubmitted] = useState(false);

  const handleSubmit = (e: React.FormEvent) => {
    e.preventDefault();
    setIsSubmitted(true);
  };

  const closePopup = () => {
    setIsSubmitted(false);
  };

  return (
    <div>
      <form className="md:col-span-2 space-y-6" onSubmit={handleSubmit}>
        <div>
          <label className="font-medium mb-2 block">Your name</label>
          <input placeholder="Abc" />
        </div>
        <div>
          <label className="font-medium mb-2 block">Email address</label>
          <input placeholder="Abc@der.com" type="email" />
        </div>
        <div>
          <label className="font-medium mb-2 block">Subject</label>
          <input placeholder="this is optional" />
        </div>
        <div>
          <label className="font-medium mb-2 block">Message</label>
          <Textarea
            placeholder="Hi! I'd like to ask about..."
            className="min-h-[120px]"
          />
        </div>
        <Button
          type="submit"
          className="w-full md:w-auto bg-[illegible] hover:bg-[illegible]/90"
        >
```

use of useState useEffect

I used useState to manage the component's local state and useEffect to handle side effects, such as fetching data and updating the UI dynamically, in my website. These hooks helped implement efficient and reactive functionalities.



```
import { useEffect, useState } from 'react'
```



```
const [name, SetName] = useState("")
const [email, SetEmail] = useState("")
const [message, SetMessage] = useState("")
const [cmtArray, setCmtArray] = useState<Comment[]>([])
const [btnName, setBtnName] = useState("Post")
const [findCard, setFindCard] = useState<Comment | null>(null)
```

Product Listing

```
[illegible]
export interface Product {
    id: string
    src: string
    name: string
    description: string
    price: number
    originalPrice?: number
    image: string
    tag?: 'new' | 'sale'
    salePercentage?: number
}

export const products: Product[] = [
    {
      id: '1',
      name: [illegible],
      src: [illegible],
      description: [illegible],
      price: [illegible],
      originalPrice: [illegible],
      image: [illegible],
      tag: 'sale',
      salePercentage: [illegible],
    },
    {
      id: '2',
      name: [illegible],
      src: [illegible],
      description: [illegible],
      price: [illegible],
      image: [illegible],
    },
    {
      id: '3',
      name: [illegible],
      src: [illegible],
      description: [illegible],
      price: [illegible],
      originalPrice: [illegible],
      image: [illegible],
      tag: [illegible]
```