Day-04 OF ASSINGMENT:

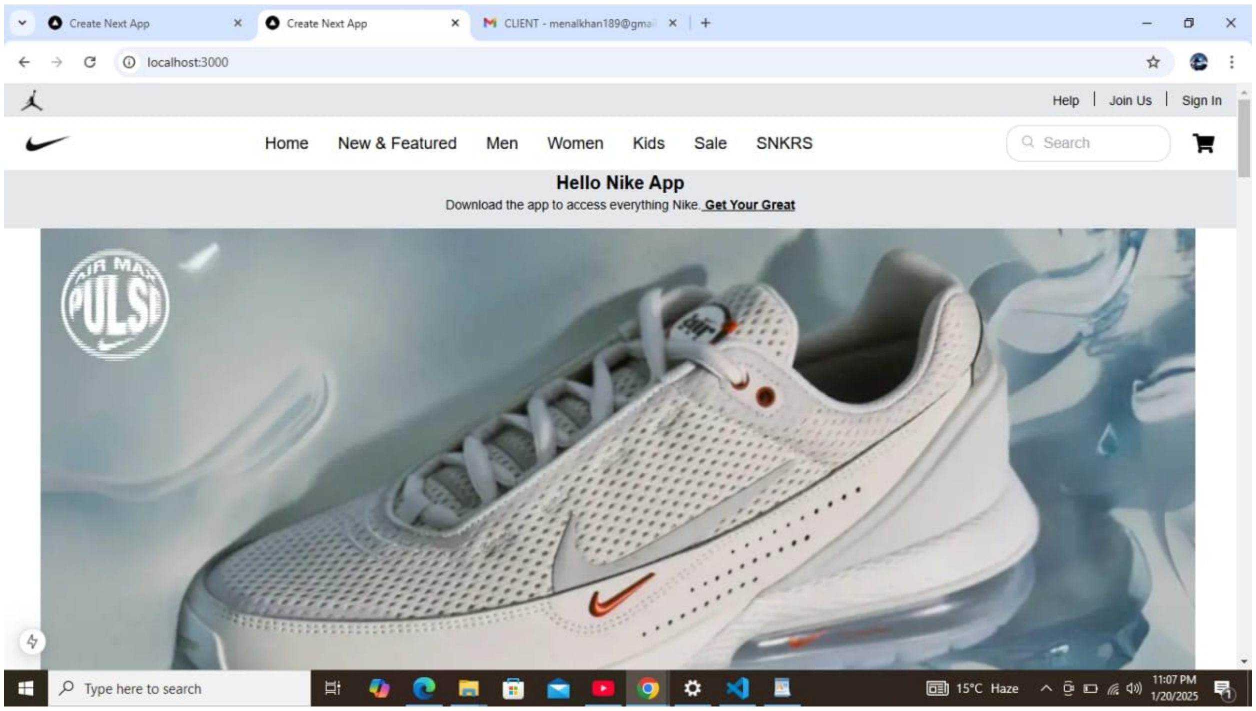
Made by Aimal Khan:

Thid is the Day-04 of hackathon assignment and I mine template is 3.

Frontend Design:

- I have made fronted design as per hackathon requirement.
- My fronted design is fully responsive in small and large screen.
- I put a lot of efforts on responsiveness.

  Here Is MY HOME PAGE Design:



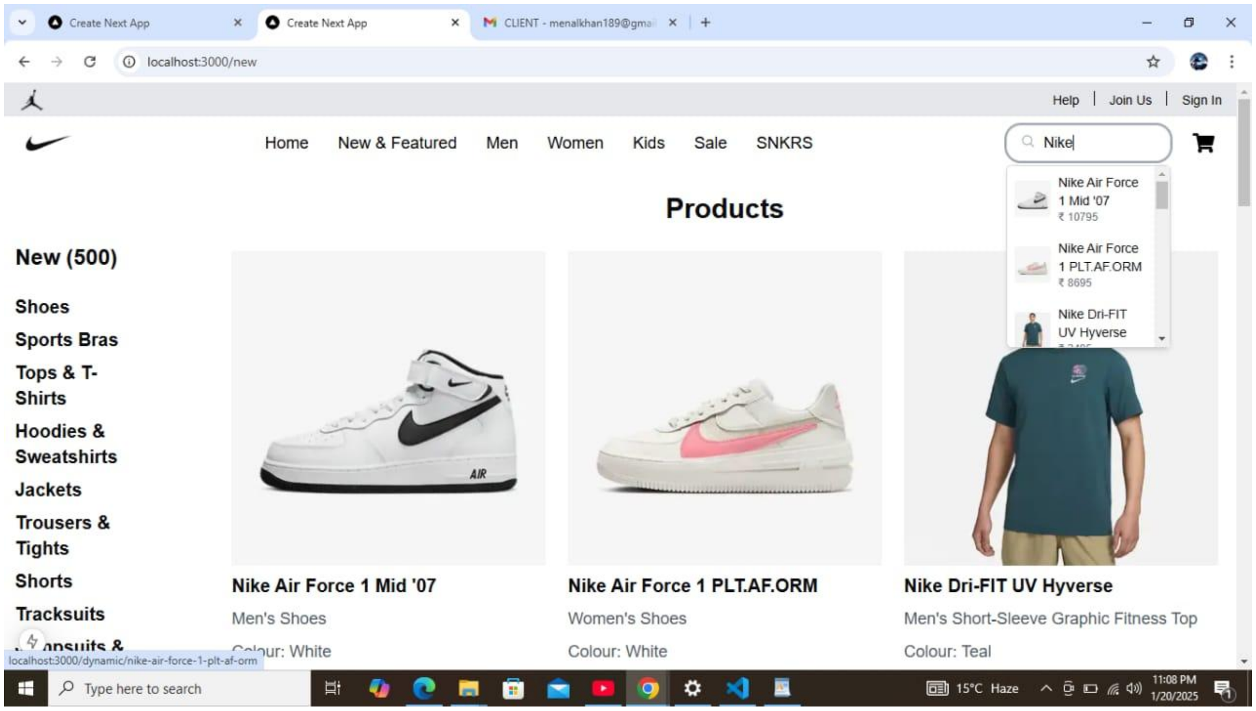
Api Migration:

I used the api which was given in assignment 3 and I have migrated it to sanity and after migration I fetch sanity data and make it visible in my product page which name is New and Feature.

I have added dynamic features in Shop which are.

- Add to Cart Functionality
- Alert Messages
- Added Checkout Page
- Search option (in which products can be searched)

MY PRODUCT PAGE:



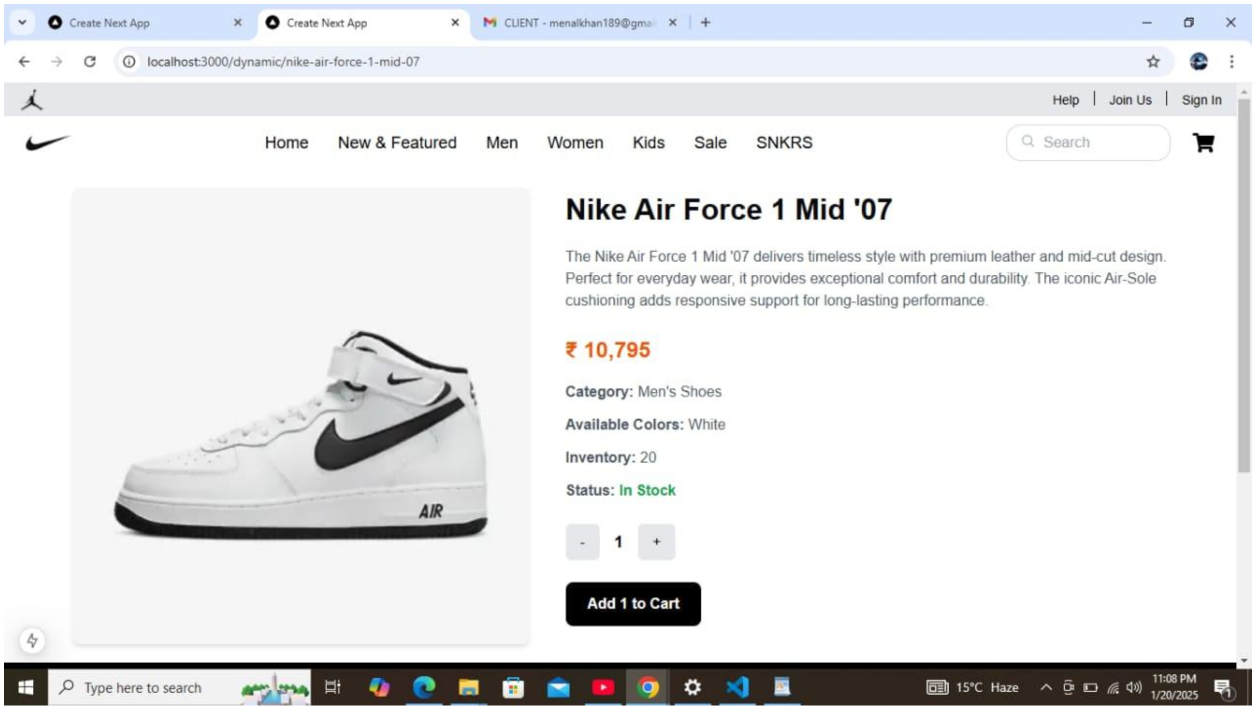
SEARCH OPTION:

I have added functionality in search option when user search any product item will result in showing the item.

After clicking the product user will be redirected to the Product Detail page of the specific item.

PRODUCT DETAILS PAGE OF SHOP:

Here is my product detail page in which user can increase or decrease quantity and also there is Add to Cart Button b clicking on it user will be redirected to cart page.



Nike Air Force 1 Mid '07

The Nike Air Force 1 Mid '07 delivers timeless style with premium leather and mid-cut design. Perfect for everyday wear, it provides exceptional comfort and durability. The iconic Air-Sole cushioning adds responsive support for long-lasting performance.

₹ 10,795

Category: Men's Shoes

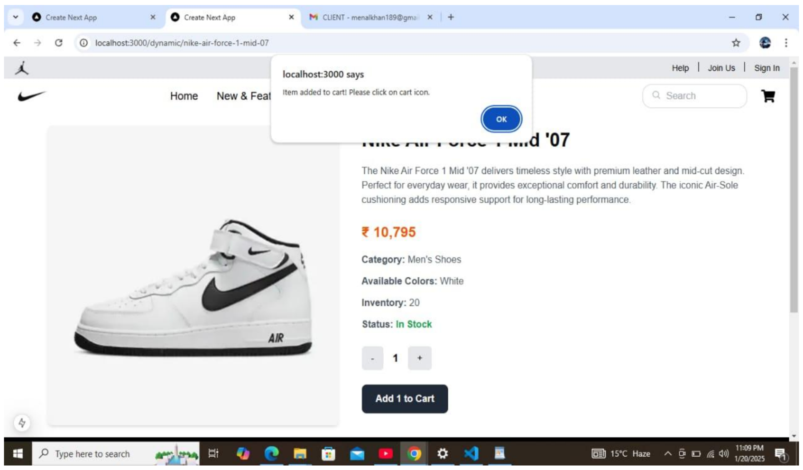
Available Colors: White

Inventory: 20

Status: In Stock

ALERT MESSAGE IN PRODUCT DTAILS PAGE:

By clicking on add to cart you will get the notification



localhost:3000 says

Item added to cart! Please click on cart icon.

Nike Air Force 1 Mid '07

The Nike Air Force 1 Mid '07 delivers timeless style with premium leather and mid-cut design. Perfect for everyday wear, it provides exceptional comfort and durability. The iconic Air-Sole cushioning adds responsive support for long-lasting performance.

₹ 10,795

Category: Men's Shoes

Available Colors: White

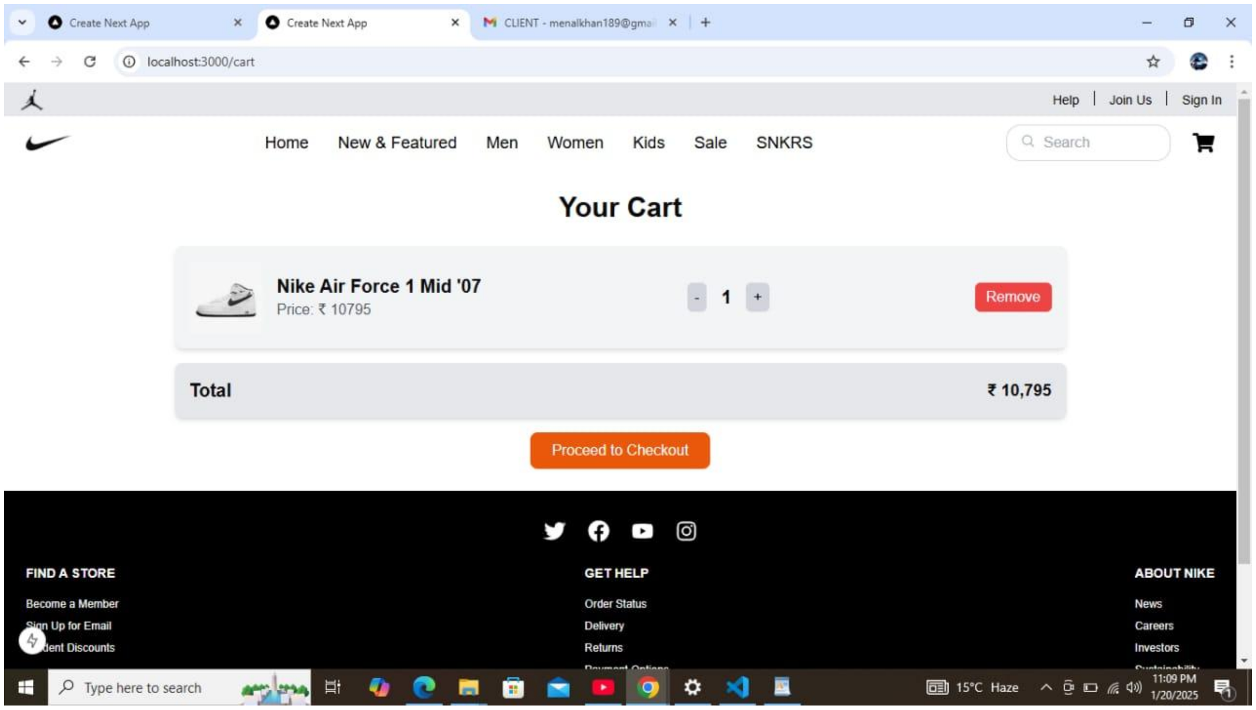
Inventory: 20

Status: In Stock

CART PAGE:

Here is my beautifully designed Cart page in which the item will be visible which user has bought in product detail page.

In cart page user can also increase or decrease the quantity.



Your Cart

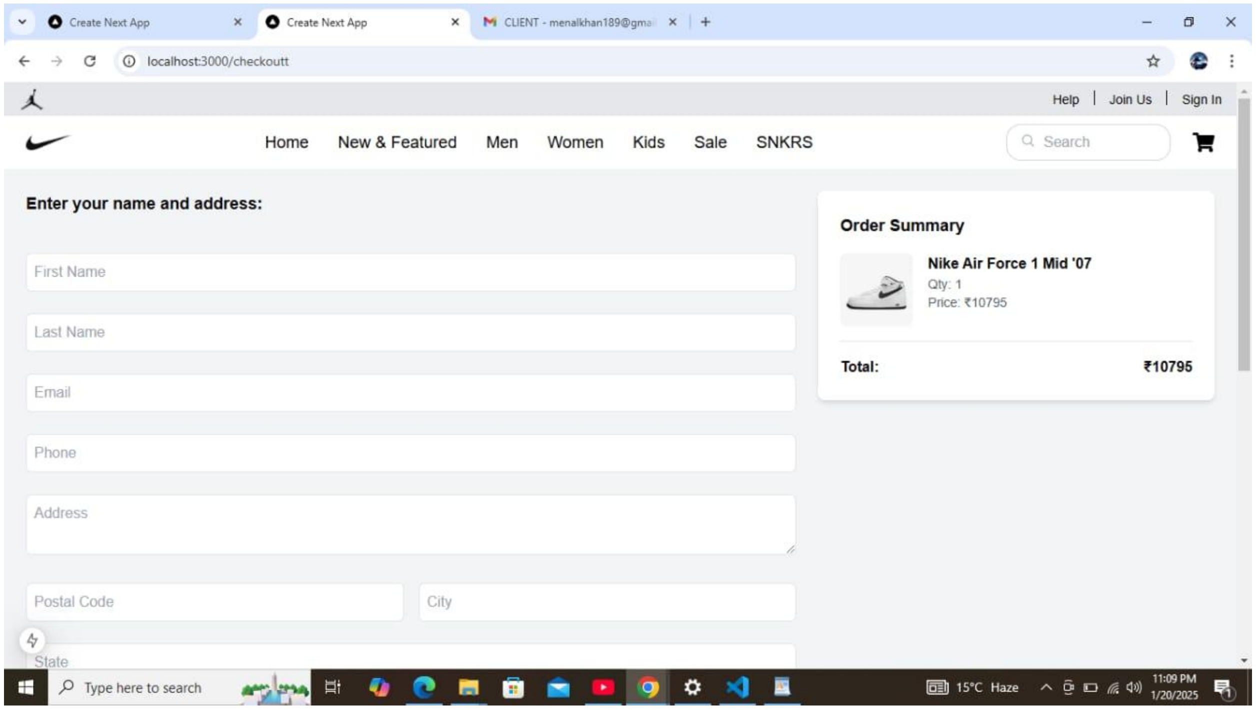
Nike Air Force 1 Mid '07
Price: ₹ 10795
- 1 +
Remove

Total ₹ 10,795

Proceed to Checkout

PROCEED TO CHECKOUT PAGE:

I have also made Proceed to Checkout page when user click on proceed to checkout button user will be forwarded to Checkout page.



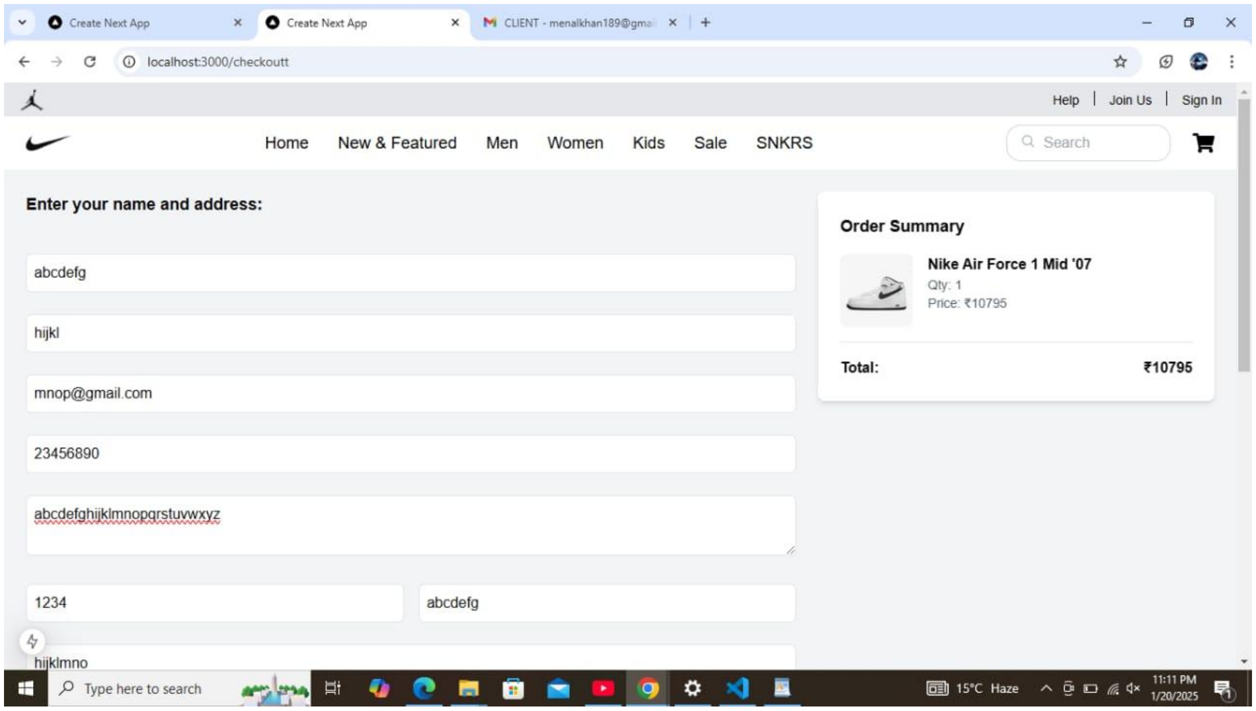
Enter your name and address:

Order Summary

Nike Air Force 1 Mid '07
Qty: 1
Price: ₹10795

Total: ₹10795

After this user will have to fill all these input fields:



Enter your name and address:

abcdefg

hijkl

mnop@gmail.com

23456890

abcdefghijklmnopqrstuvwxyz

1234 | abcdefg

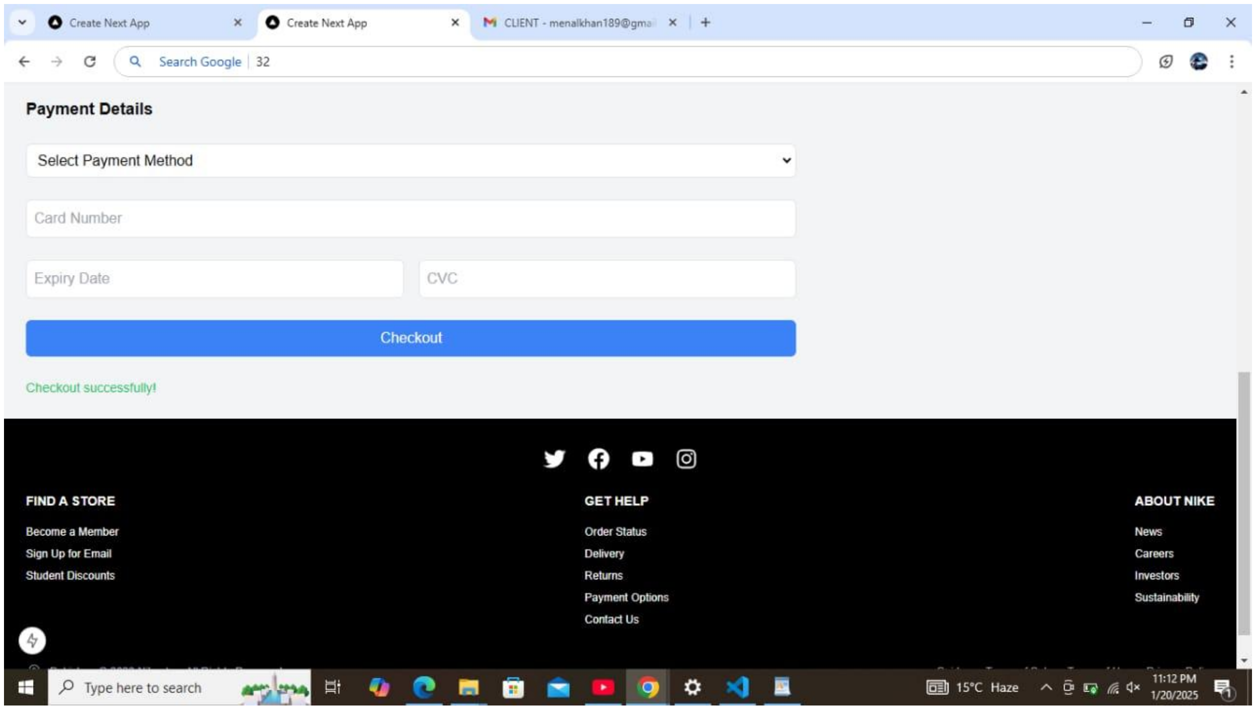
hijklmno

Order Summary

Nike Air Force 1 Mid '07
Qty: 1
Price: ₹10795

Total: **₹10795**

After filling all the input fields when user click on Checkout Button so then user will get a message that the checkout is successfully.



Payment Details

Select Payment Method

Card Number

Expiry Date | CVC

Checkout successfully!

DATA STORING:

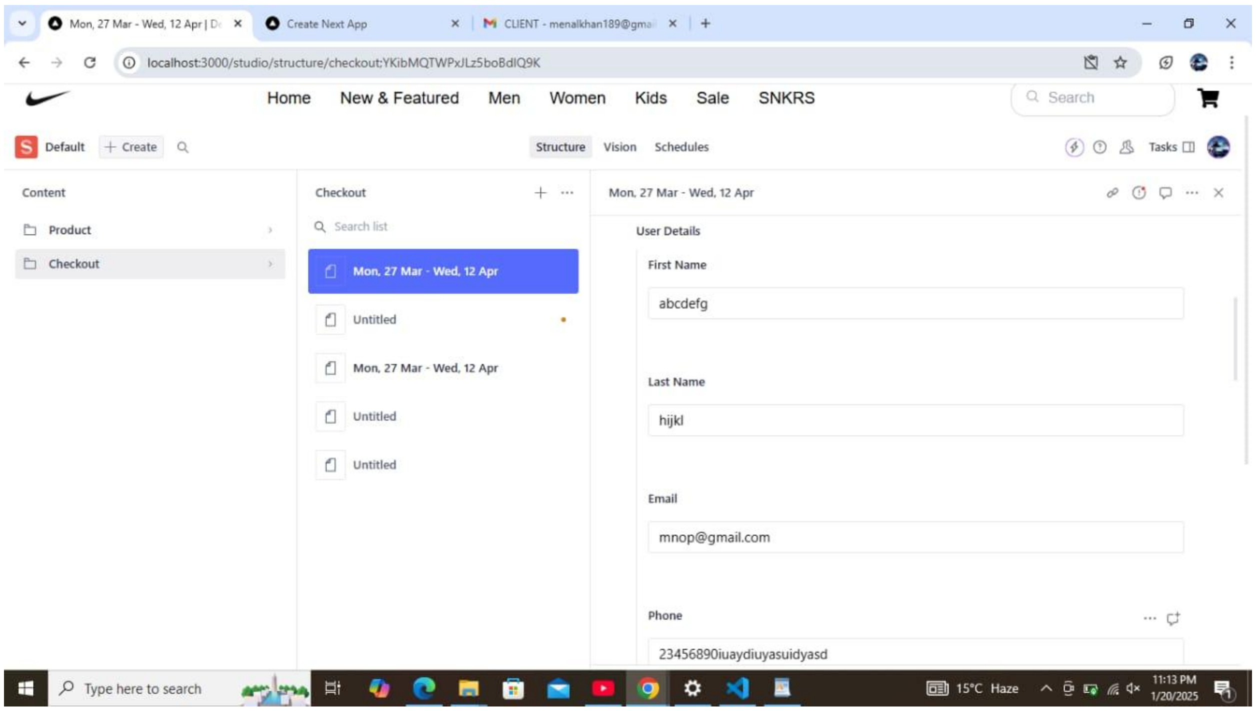
I used Shadcn Library to import input fields. And to transfer user details of checkout page.

As well as I have made sanity schema to handle the user details.

SANITY (CMS):

I have get all the details provided by the user.

Details of a user.



User Details

First Name

abcdefg

Last Name

hijkl

Email

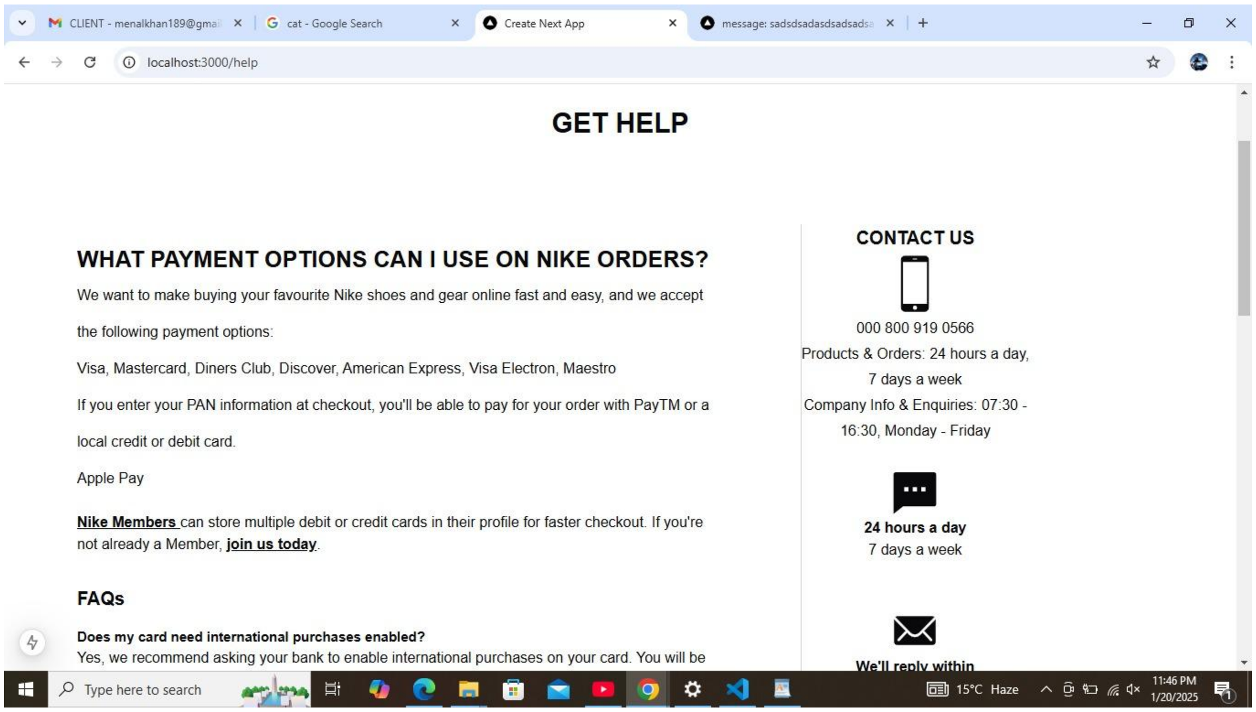
mnop@gmail.com

Phone

23456890iuaydiuyasuidyasd

HELP & FAQs CENTER:

I also work on Help & FAQs Center. When user faced any issue related to website , buying items , services and anything. So they can Contact Us as well as they can write a report message after clicking on submit button user will get alert message and the report message send by user will send to my sanity CMS.



GET HELP

WHAT PAYMENT OPTIONS CAN I USE ON NIKE ORDERS?

We want to make buying your favourite Nike shoes and gear online fast and easy, and we accept the following payment options:

Visa, Mastercard, Diners Club, Discover, American Express, Visa Electron, Maestro

If you enter your PAN information at checkout, you'll be able to pay for your order with PayTM or a local credit or debit card.

Apple Pay

Nike Members can store multiple debit or credit cards in their profile for faster checkout. If you're not already a Member, **join us today**.

FAQs

Does my card need international purchases enabled?

Yes, we recommend asking your bank to enable international purchases on your card. You will be

CONTACT US

000 800 919 0566

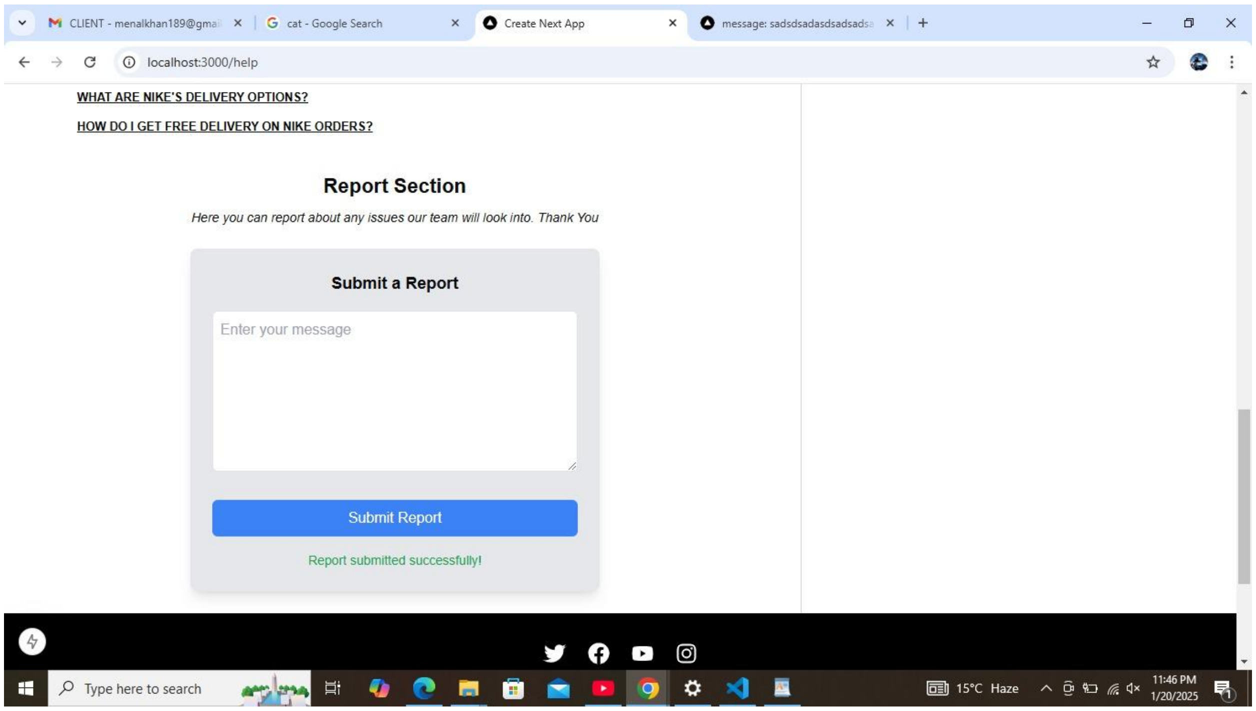
Products & Orders: 24 hours a day, 7 days a week

Company Info & Enquiries: 07:30 - 16:30, Monday - Friday

24 hours a day

7 days a week

We'll reply within


localhost:3000/help

WHAT ARE NIKE'S DELIVERY OPTIONS?

HOW DO I GET FREE DELIVERY ON NIKE ORDERS?

Report Section

Here you can report about any issues our team will look into. Thank You

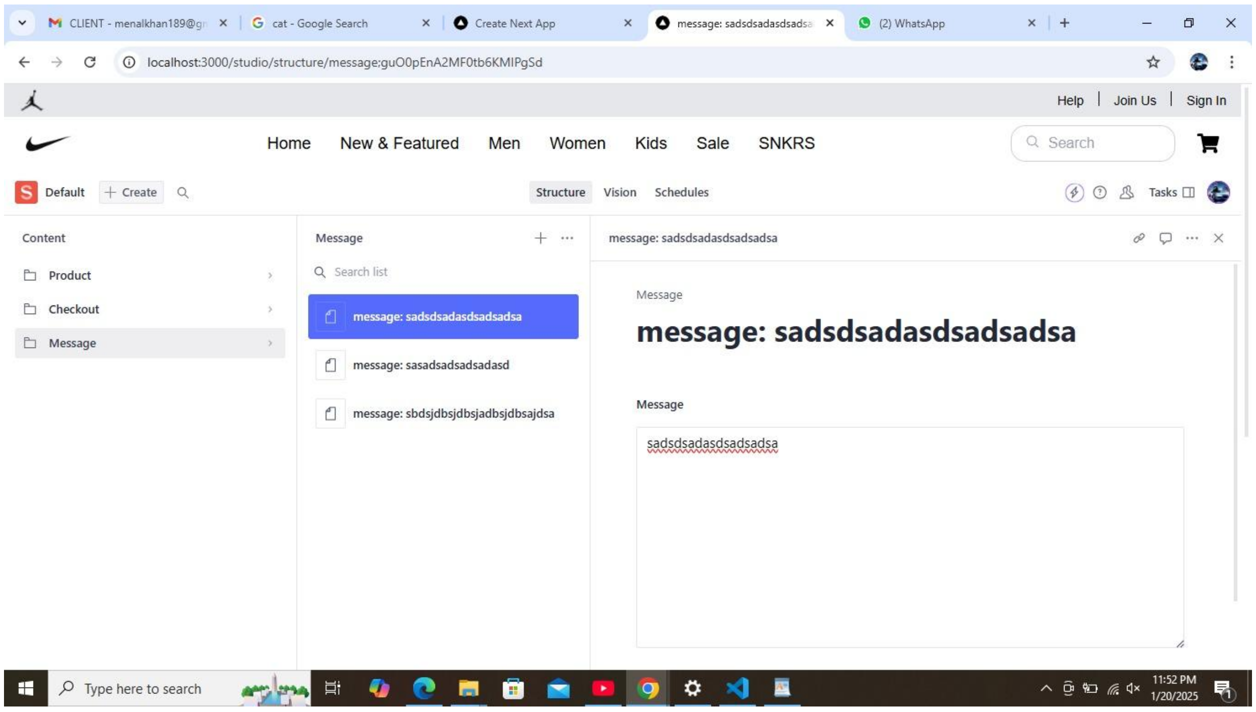
Submit a Report

Enter your message

Submit Report

Report submitted successfully!

Here is the message which was send by the user.


localhost:3000/studio/structure/message;guO0pEnA2MF0tb6KMIPgSd

Content: Product, Checkout, Message

Message:
- message: sadsdsadasdsadsadsa
- message: sasadsadsadsadasd
- message: sbdsjdbsjdbsjadbsjdbsajdsa

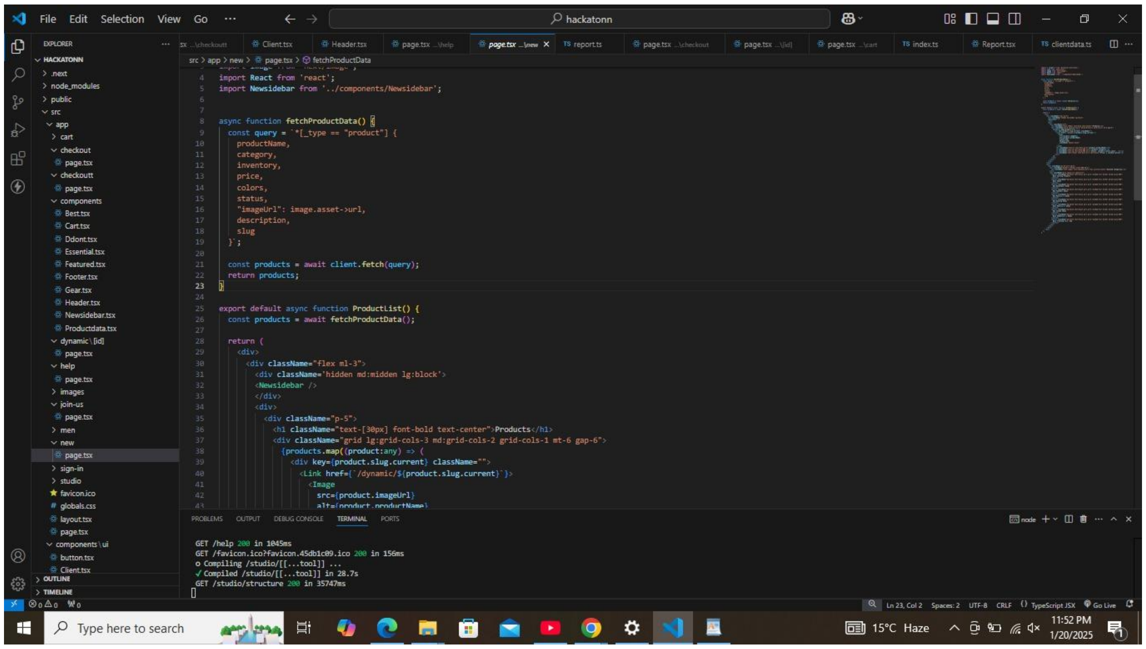
message: sadsdsadasdsadsadsa

Message

sadsdsadasdsadsadsa

The Codes of the Frontend Design and Functionality:

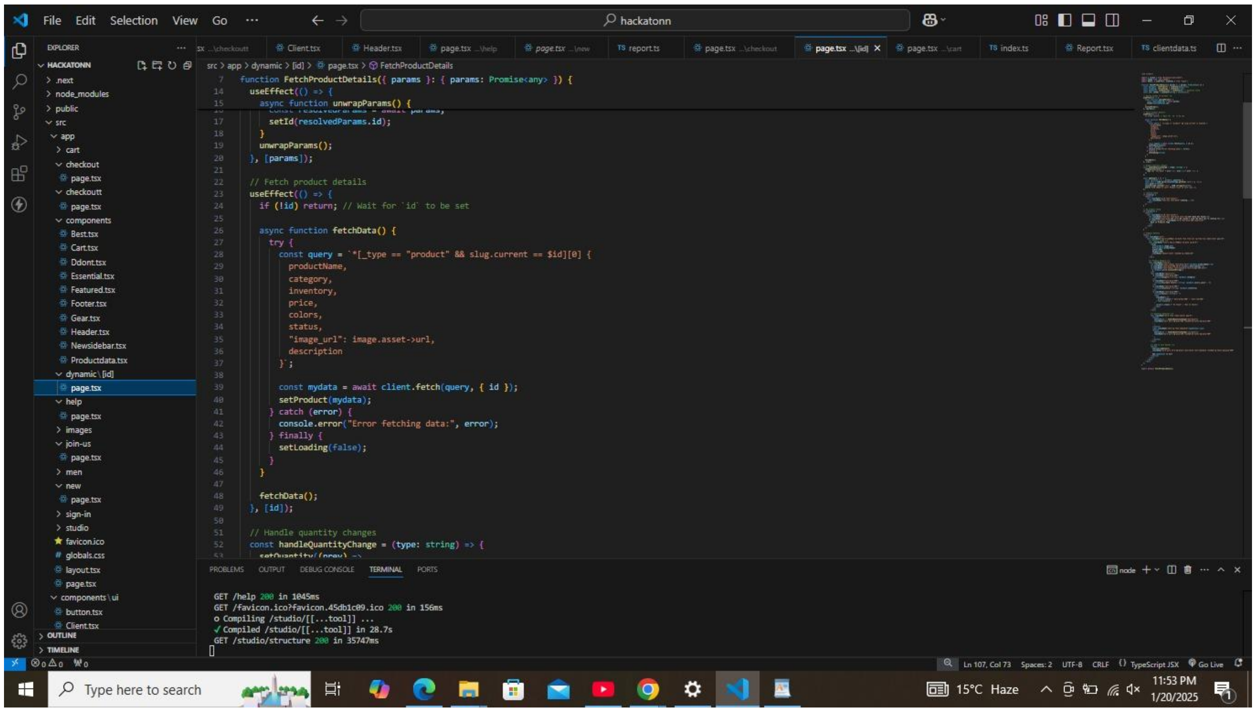
Shop Page Code:



```tsx
import React from 'react';
import Newsidebar from '../components/Newsidebar';


async function fetchProductData() {
  const query = `*[_type == "product"] {
    productName,
    category,
    inventory,
    price,
    colors,
    status,
    "imageUrl": image.asset->url,
    description,
    slug
  }`;

  const products = await client.fetch(query);
  return products;
}

export default async function ProductList() {
  const products = await fetchProductData();

  return (
    <div>
      <div className="flex ml-3">
        <div className='hidden md:midden lg:block'>
        <Newsidebar />
        </div>
        <div>
          <div className="p-5">
            <h1 className="text-[30px] font-bold text-center">Products</h1>
            <div className="grid lg:grid-cols-3 md:grid-cols-2 grid-cols-1 mt-6 gap-6">
              {products.map((product:any) => (
                <div key={product.slug.current} className="">
                  <Link href={`/dynamic/${product.slug.current}`}>
                    <Image
                      src={product.imageUrl}
                      alt={product.productName}
```

```
GET /help 200 in 1045ms
GET /favicon.ico?favicon.45db1c09.ico 200 in 156ms
o Compiling /studio/[[...tool]] ...
✓ Compiled /studio/[[...tool]] in 28.7s
GET /studio/structure 200 in 35747ms
```

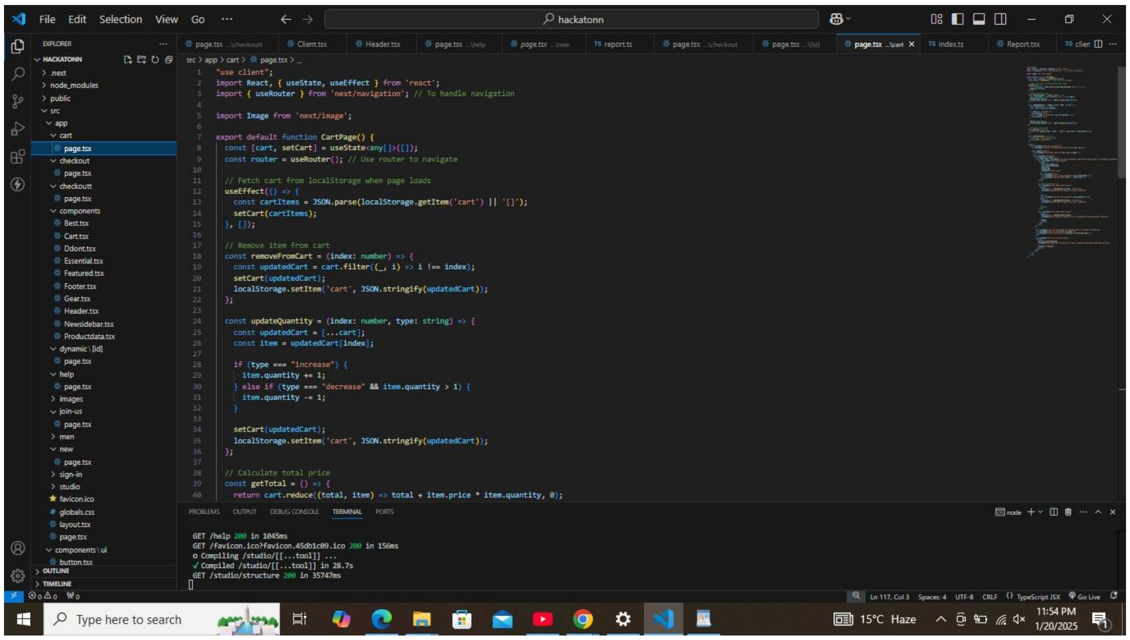
Shop Details Page Code:



```
function FetchProductDetails({ params }: { params: Promise<any> }) {
  useEffect(() => {
    async function unwrapParams() {
      setId(resolvedParams.id);
    }
    unwrapParams();
  }, [params]);

  // Fetch product details
  useEffect(() => {
    if (!id) return; // Wait for `id` to be set

    async function fetchData() {
      try {
        const query = `*[_type == "product" && slug.current == $id][0] {
          productName,
          category,
          inventory,
          price,
          colors,
          status,
          "image_url": image.asset->url,
          description
        }`;

        const mydata = await client.fetch(query, { id });
        setProduct(mydata);
      } catch (error) {
        console.error("Error fetching data:", error);
      } finally {
        setLoading(false);
      }
    }

    fetchData();
  }, [id]);

  // Handle quantity changes
  const handleQuantityChange = (type: string) => {
```

```
GET /help 200 in 1045ms
GET /favicon.ico?favicon.45db1c09.ico 200 in 156ms
o Compiling /studio/[[...tool]] ...
✓ Compiled /studio/[[...tool]] in 28.7s
GET /studio/structure 200 in 35747ms
```

Cart Page Code:

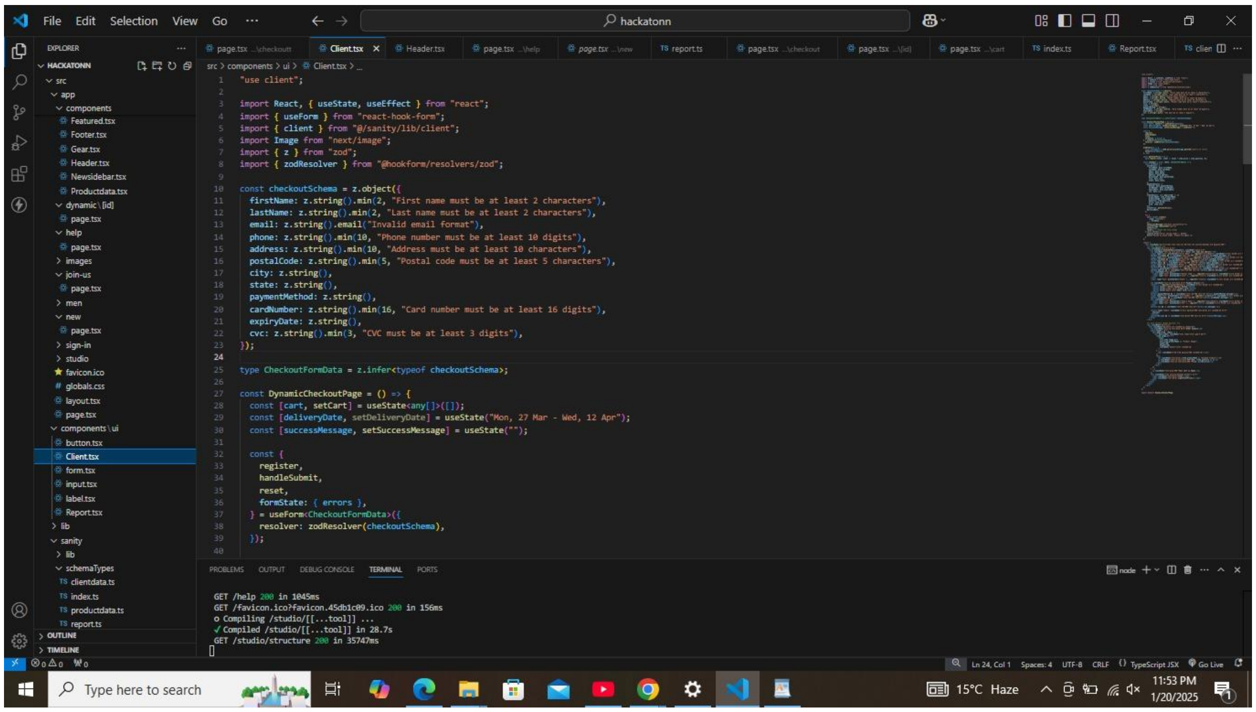


```
"use client";
import React, { useState, useEffect } from 'react';
import { useRouter } from 'next/navigation'; // To handle navigation

import Image from 'next/image';

export default function CartPage() {
  const [cart, setCart] = useState<any[]>([]);
  const router = useRouter(); // Use router to navigate

  // Fetch cart from localStorage when page loads
  useEffect(() => {
    const cartItems = JSON.parse(localStorage.getItem('cart') || '[]');
    setCart(cartItems);
  }, []);

  // Remove item from cart
  const removeFromCart = (index: number) => {
    const updatedCart = cart.filter((_, i) => i !== index);
    setCart(updatedCart);
    localStorage.setItem('cart', JSON.stringify(updatedCart));
  };

  const updateQuantity = (index: number, type: string) => {
    const updatedCart = [...cart];
    const item = updatedCart[index];

    if (type === "increase") {
      item.quantity += 1;
    } else if (type === "decrease" && item.quantity > 1) {
      item.quantity -= 1;
    }

    setCart(updatedCart);
    localStorage.setItem('cart', JSON.stringify(updatedCart));
  };

  // Calculate total price
  const getTotal = () => {
    return cart.reduce((total, item) => total + item.price * item.quantity, 0);
```

```
GET /help 200 in 1045ms
GET /favicon.ico?favicon.45db1c09.ico 200 in 156ms
o Compiling /studio/[[...tool]] ...
✓ Compiled /studio/[[...tool]] in 28.7s
GET /studio/structure 200 in 35747ms
```

Checkout Page Code:

Checkout page is created By Shadcn and connected it to Sanity.



```
"use client";

import React, { useState, useEffect } from "react";
import { useForm } from "react-hook-form";
import { client } from "@/sanity/lib/client";
import Image from "next/image";
import { z } from "zod";
import { zodResolver } from "@hookform/resolvers/zod";

const checkoutSchema = z.object({
  firstName: z.string().min(2, "First name must be at least 2 characters"),
  lastName: z.string().min(2, "Last name must be at least 2 characters"),
  email: z.string().email("Invalid email format"),
  phone: z.string().min(10, "Phone number must be at least 10 digits"),
  address: z.string().min(10, "Address must be at least 10 characters"),
  postalCode: z.string().min(5, "Postal code must be at least 5 characters"),
  city: z.string(),
  state: z.string(),
  paymentMethod: z.string(),
  cardNumber: z.string().min(16, "Card number must be at least 16 digits"),
  expiryDate: z.string(),
  cvc: z.string().min(3, "CVC must be at least 3 digits"),
});

type CheckoutFormData = z.infer<typeof checkoutSchema>;

const DynamicCheckoutPage = () => {
  const [cart, setCart] = useState<any[]>([]);
  const [deliveryDate, setDeliveryDate] = useState("Mon, 27 Mar - Wed, 12 Apr");
  const [successMessage, setSuccessMessage] = useState("");

  const {
    register,
    handleSubmit,
    reset,
    formState: { errors },
  } = useForm<CheckoutFormData>({
    resolver: zodResolver(checkoutSchema),
  });

```

```
GET /help 200 in 1045ms
GET /favicon.ico?favicon.45db1c09.ico 200 in 156ms
o Compiling /studio/[[...tool]] ...
✓ Compiled /studio/[[...tool]] in 28.7s
GET /studio/structure 200 in 35747ms
```

Schema of Sanity For Checkout:

ASSINGMENT OF DAY-04 Ended Here.