



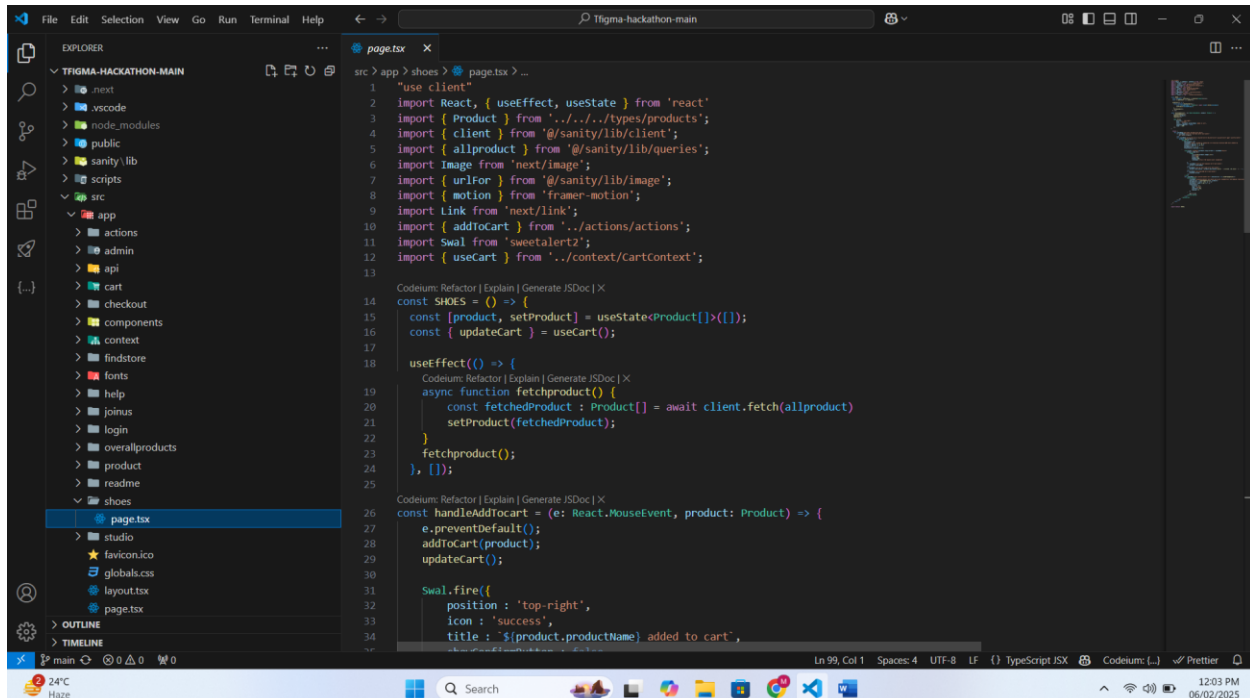
DAY 4- DYNAMIC FRONTEND COMPONENTS- [NIKE ONLINE STORE.]"

Day 4

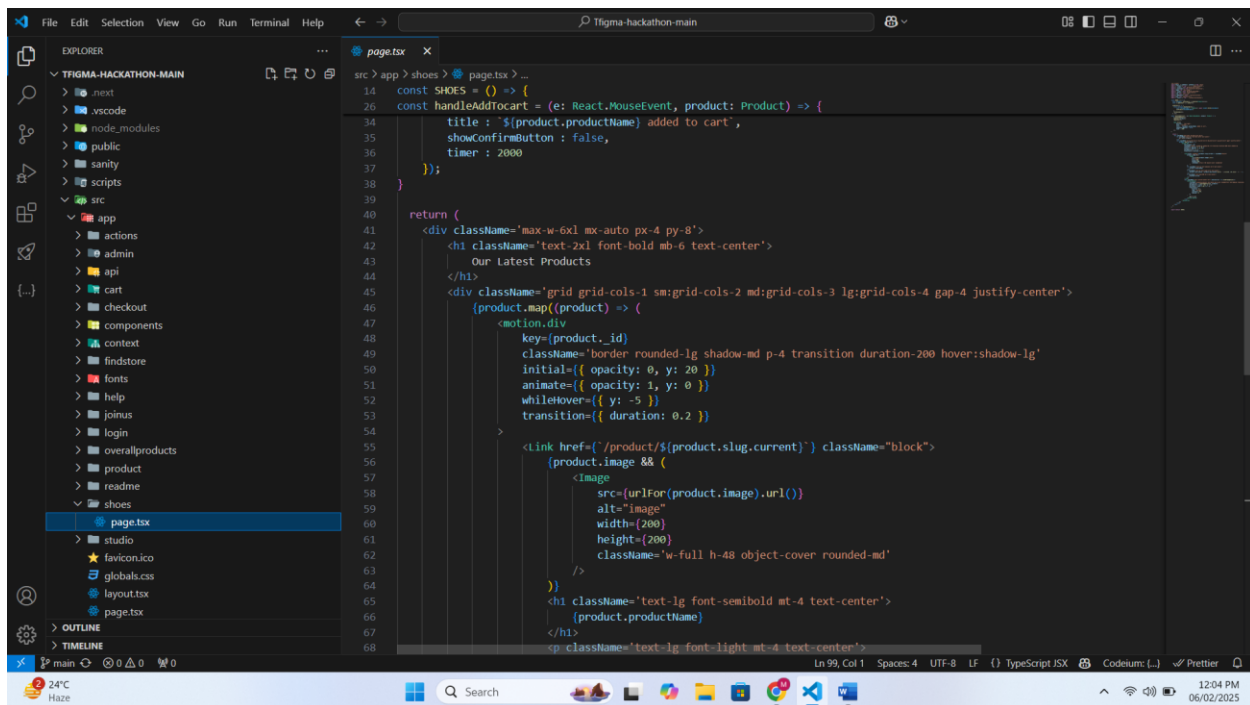


DAY 4- DYNAMIC FRONTEND COMPONENTS- [NIKE ONLINE STORE.]"

- Screenshots, showcasing: The product listing page with dynamic data. **Code Deliverables:**

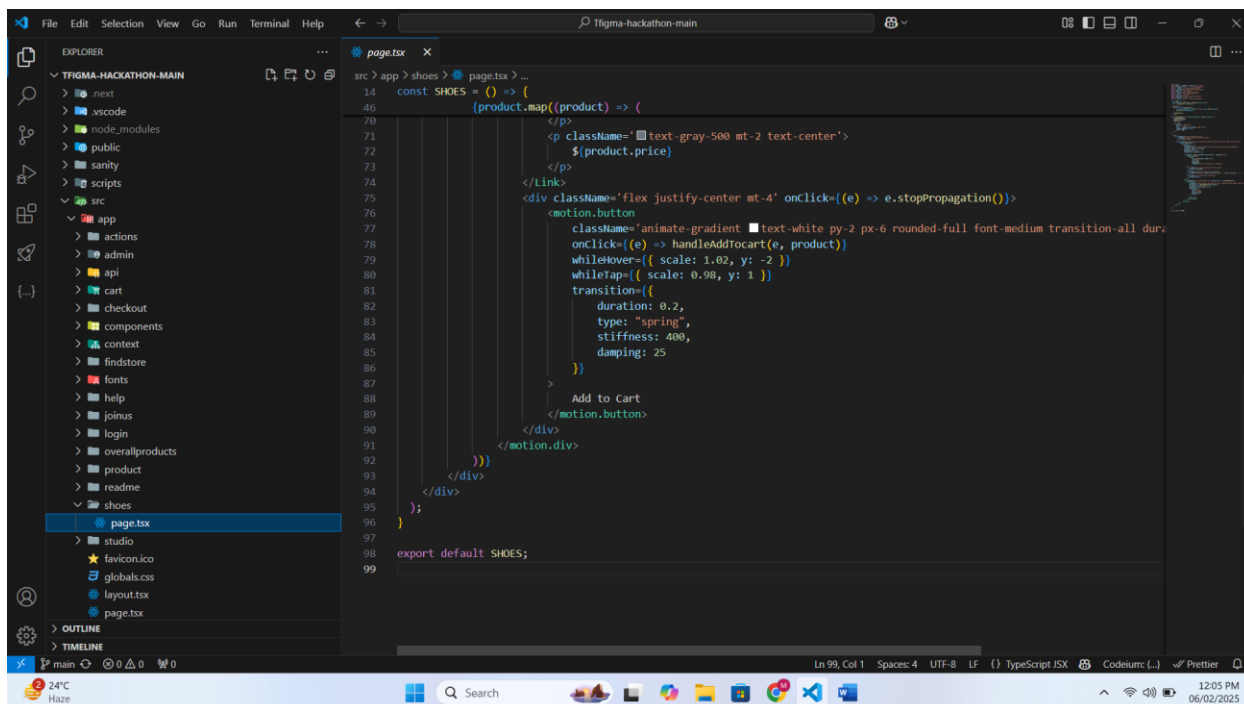
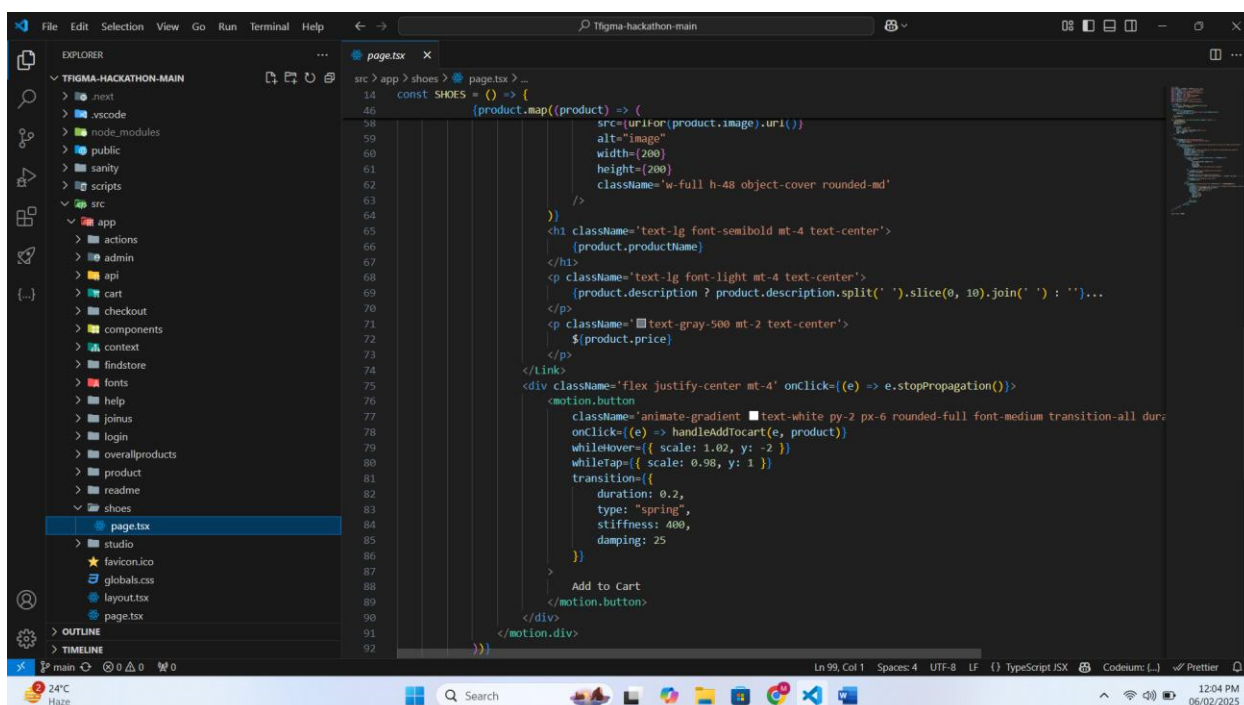


```
1 "use client"
2 import React, { useEffect, useState } from 'react'
3 import { Product } from '../../types/products';
4 import { client } from '@sanity/lib/client';
5 import { allproduct } from '@sanity/lib/queries';
6 import Image from 'next/image';
7 import { urlFor } from '@sanity/lib/image';
8 import { motion } from 'framer-motion';
9 import Link from 'next/link';
10 import { addtoCart } from '../actions/actions';
11 import Swal from 'sweetalert2';
12 import { useCart } from '../context/CartContext';
13
14 const SHOES = () => {
15   const [product, setProduct] = useState<Product[]>([]);
16   const { updateCart } = useCart();
17
18   useEffect(() => {
19     async function fetchProduct() {
20       const fetchedProduct : Product[] = await client.fetch(allproduct)
21       setProduct(fetchedProduct);
22     }
23     fetchProduct();
24   }, []);
25
26   const handleAddtoCart = (e: React.MouseEvent, product: Product) => {
27     e.preventDefault();
28     addtoCart(product);
29     updateCart();
30
31     Swal.fire({
32       position : 'top-right',
33       icon : 'success',
34       title : `${product.productName} added to cart`,
35     });
36   }
37 }
38
39 return (
40   <div className="max-w-6xl mx-auto px-4 py-8">
41     <h1 className="text-2xl font-bold mb-6 text-center">
42       Our Latest Products
43     </h1>
44     <div className="grid grid-cols-1 sm:grid-cols-2 md:grid-cols-3 lg:grid-cols-4 gap-4 justify-center">
45       {product.map((product) => (
46         <motion.div
47           key={product.id}
48           className="border rounded-lg shadow-md p-4 transition duration-200 hover:shadow-lg"
49           initial={{ opacity: 0, y: 20 }}
50           animate={{ opacity: 1, y: 0 }}
51           whileHover={{ y: -5 }}
52           transition={{ duration: 0.2 }}
53         >
54           <Link href={`/product/${product.slug.current}`} className="block">
55             {product.image && (
56               <Image
57                 src={urlFor(product.image).url()}
58                 alt="image"
59                 width={200}
60                 height={200}
61                 className="w-full h-48 object-cover rounded-md"
62               >/>
63             )}
64           <h1 className="text-lg font-semibold mt-4 text-center">
65             {product.productName}
66           </h1>
67           <p className="text-lg font-light mt-4 text-center">
```



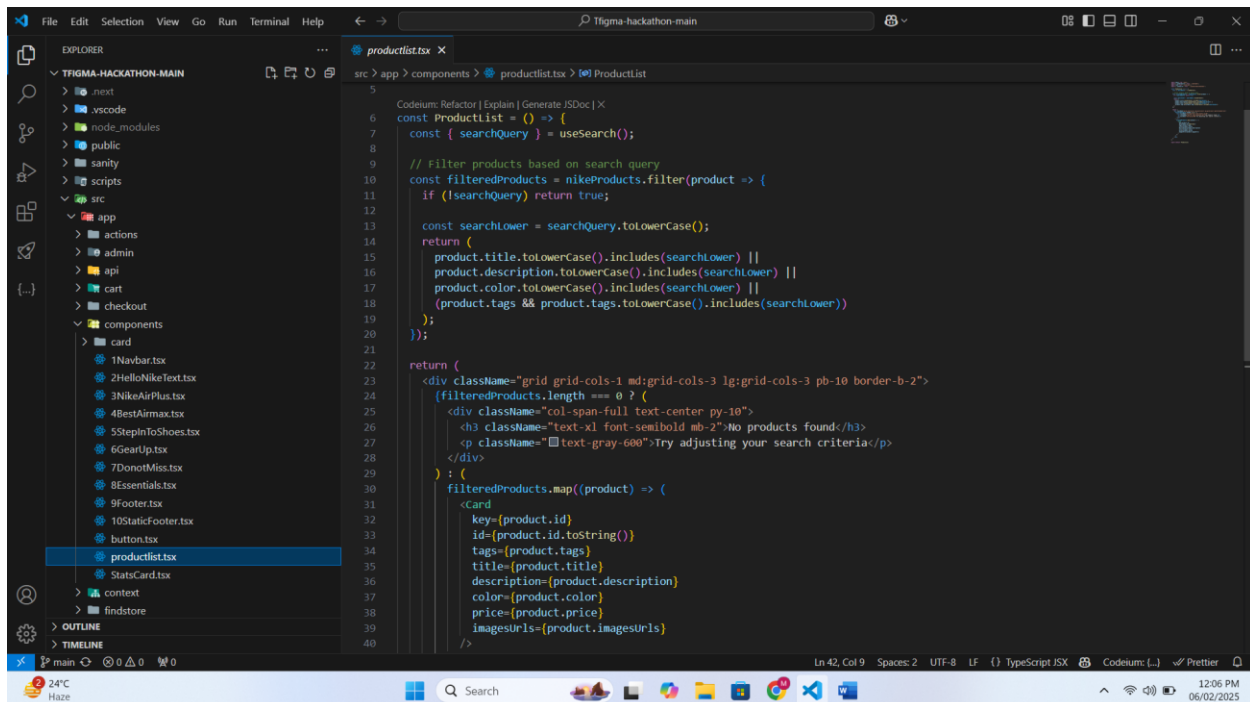
```
14 const SHOES = () => {
15   const [product, setProduct] = useState<Product[]>([]);
16   const { updateCart } = useCart();
17
18   useEffect(() => {
19     async function fetchProduct() {
20       const fetchedProduct : Product[] = await client.fetch(allproduct)
21       setProduct(fetchedProduct);
22     }
23     fetchProduct();
24   }, []);
25
26   const handleAddtoCart = (e: React.MouseEvent, product: Product) => {
27     e.preventDefault();
28     addtoCart(product);
29     updateCart();
30
31     Swal.fire({
32       position : 'top-right',
33       icon : 'success',
34       title : `${product.productName} added to cart`,
35     });
36   }
37 }
38
39 return (
40   <div className="max-w-6xl mx-auto px-4 py-8">
41     <h1 className="text-2xl font-bold mb-6 text-center">
42       Our Latest Products
43     </h1>
44     <div className="grid grid-cols-1 sm:grid-cols-2 md:grid-cols-3 lg:grid-cols-4 gap-4 justify-center">
45       {product.map((product) => (
46         <motion.div
47           key={product.id}
48           className="border rounded-lg shadow-md p-4 transition duration-200 hover:shadow-lg"
49           initial={{ opacity: 0, y: 20 }}
50           animate={{ opacity: 1, y: 0 }}
51           whileHover={{ y: -5 }}
52           transition={{ duration: 0.2 }}
53         >
54           <Link href={`/product/${product.slug.current}`} className="block">
55             {product.image && (
56               <Image
57                 src={urlFor(product.image).url()}
58                 alt="image"
59                 width={200}
60                 height={200}
61                 className="w-full h-48 object-cover rounded-md"
62               >/>
63             )}
64           <h1 className="text-lg font-semibold mt-4 text-center">
65             {product.productName}
66           </h1>
67           <p className="text-lg font-light mt-4 text-center">
```

DAY 4- DYNAMIC FRONTEND COMPONENTS- [NIKE ONLINE STORE.]

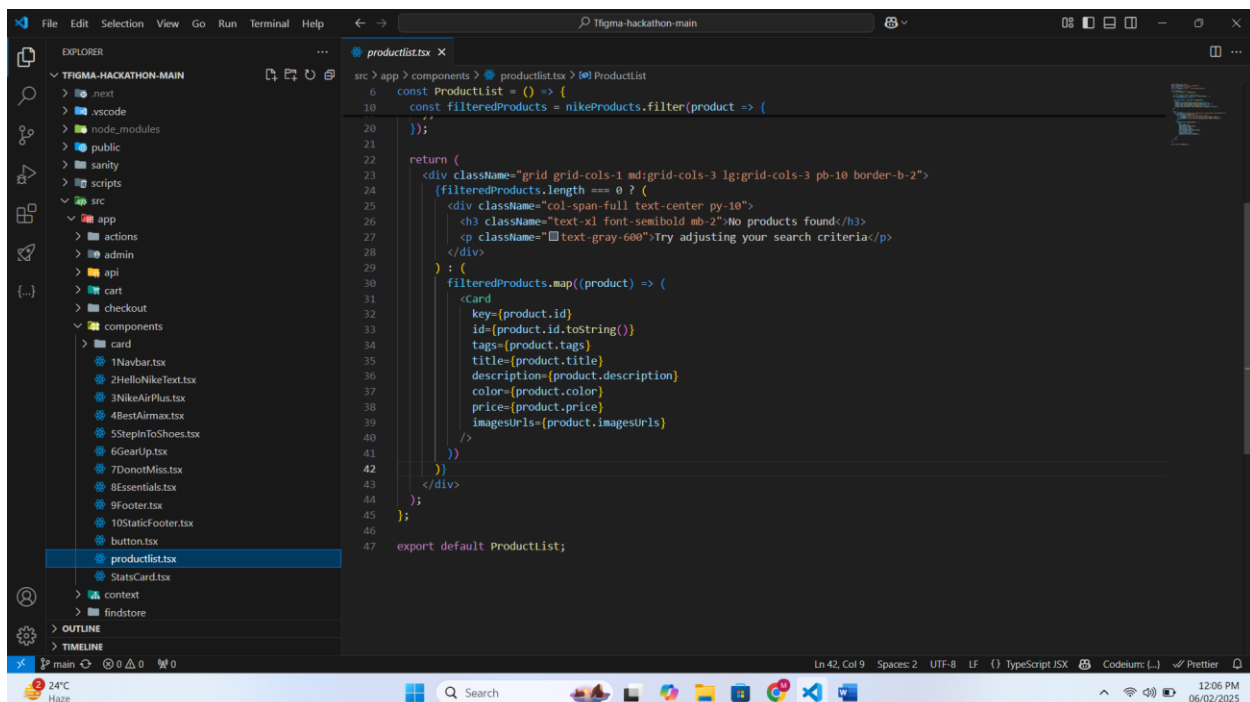


DAY 4- DYNAMIC FRONTEND COMPONENTS- [NIKE ONLINE STORE.]"

Product List :



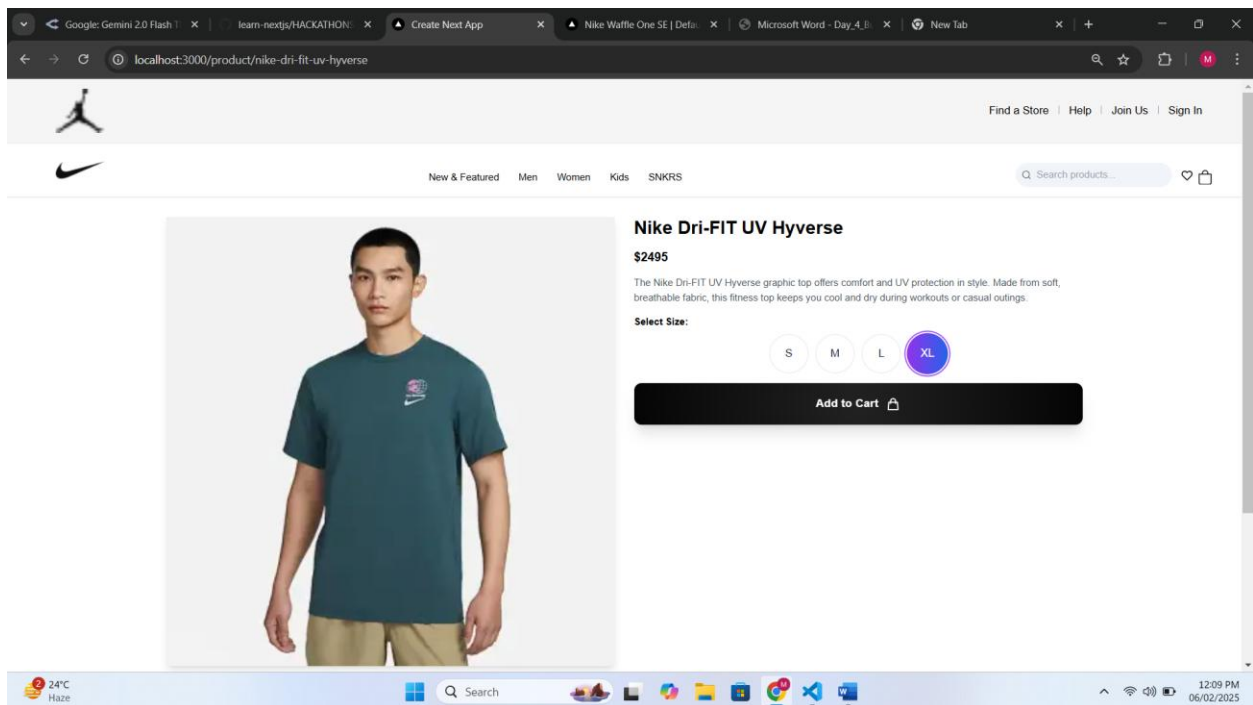
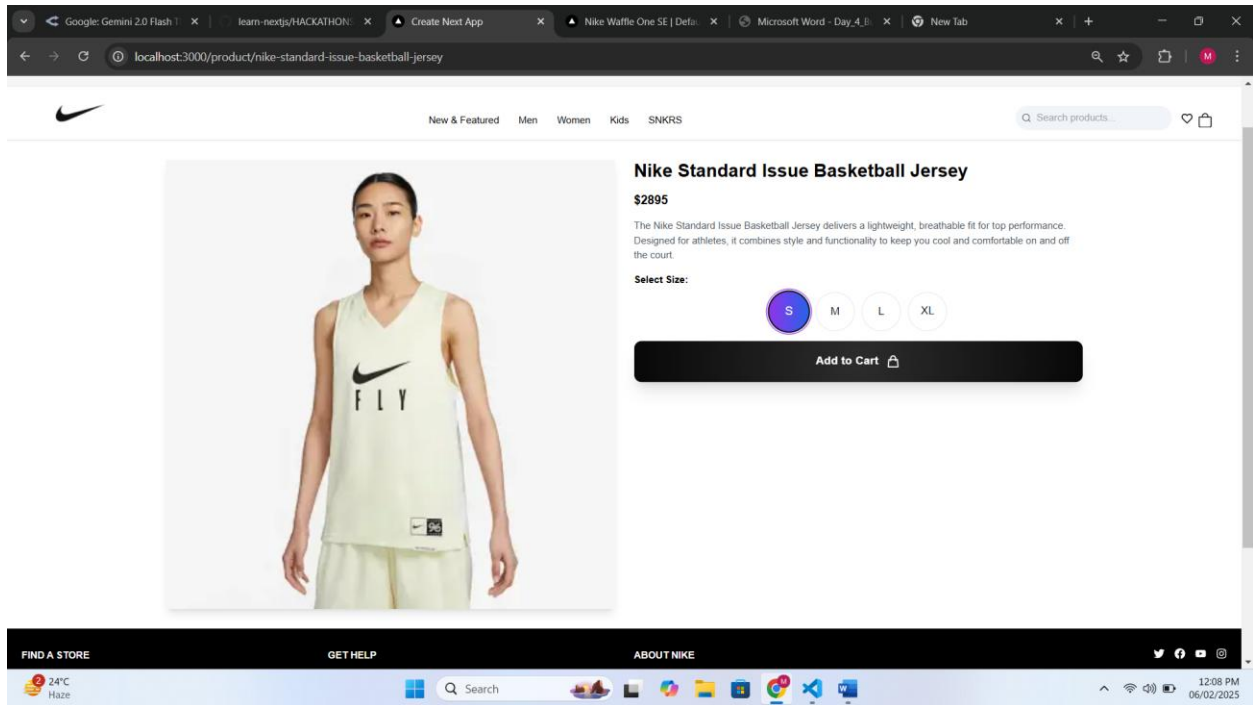
```
5
6 const ProductList = () => {
7   const { searchQuery } = useSearch();
8
9   // Filter products based on search query
10  const filteredProducts = nikeProducts.filter(product => {
11    if (!searchQuery) return true;
12
13    const searchLower = searchQuery.toLowerCase();
14    return (
15      product.title.toLowerCase().includes(searchLower) ||
16      product.description.toLowerCase().includes(searchLower) ||
17      product.color.toLowerCase().includes(searchLower) ||
18      (product.tags && product.tags.toLowerCase().includes(searchLower))
19    );
20  });
21
22  return (
23    <div className="grid grid-cols-1 md:grid-cols-3 lg:grid-cols-3 pb-10 border-b-2">
24      {filteredProducts.length === 0 ? (
25        <div className="col-span-full text-center py-10">
26          <h3 className="text-xl font-sembold mb-2">No products found</h3>
27          <p className="text-gray-600">Try adjusting your search criteria</p>
28        </div>
29      ) : (
30        filteredProducts.map((product) => (
31          <card
32            key={product.id}
33            id={product.id.toString()}
34            tags={product.tags}
35            title={product.title}
36            description={product.description}
37            color={product.color}
38            price={product.price}
39            imageUrl={product.imageUrl}
40          </>
41        ))
42      )}
43    </div>
44  );
45
46 export default ProductList;
```



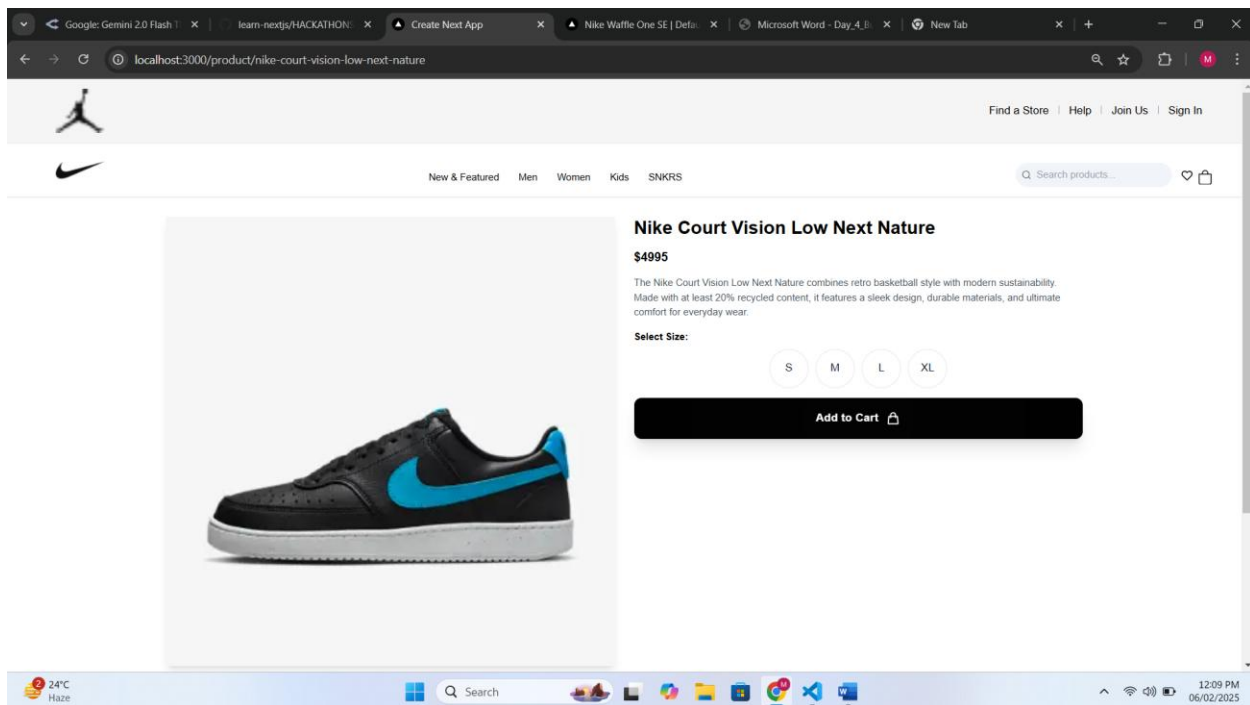
```
6 const ProductList = () => {
10  const filteredProducts = nikeProducts.filter(product => {
20  });
21
22  return (
23    <div className="grid grid-cols-1 md:grid-cols-3 lg:grid-cols-3 pb-10 border-b-2">
24      {filteredProducts.length === 0 ? (
25        <div className="col-span-full text-center py-10">
26          <h3 className="text-xl font-sembold mb-2">No products found</h3>
27          <p className="text-gray-600">Try adjusting your search criteria</p>
28        </div>
29      ) : (
30        filteredProducts.map((product) => (
31          <card
32            key={product.id}
33            id={product.id.toString()}
34            tags={product.tags}
35            title={product.title}
36            description={product.description}
37            color={product.color}
38            price={product.price}
39            imageUrl={product.imageUrl}
40          </>
41        ))
42      )}
43    </div>
44  );
45
46 export default ProductList;
```

DAY 4- DYNAMIC FRONTEND COMPONENTS- [NIKE ONLINE STORE.]"

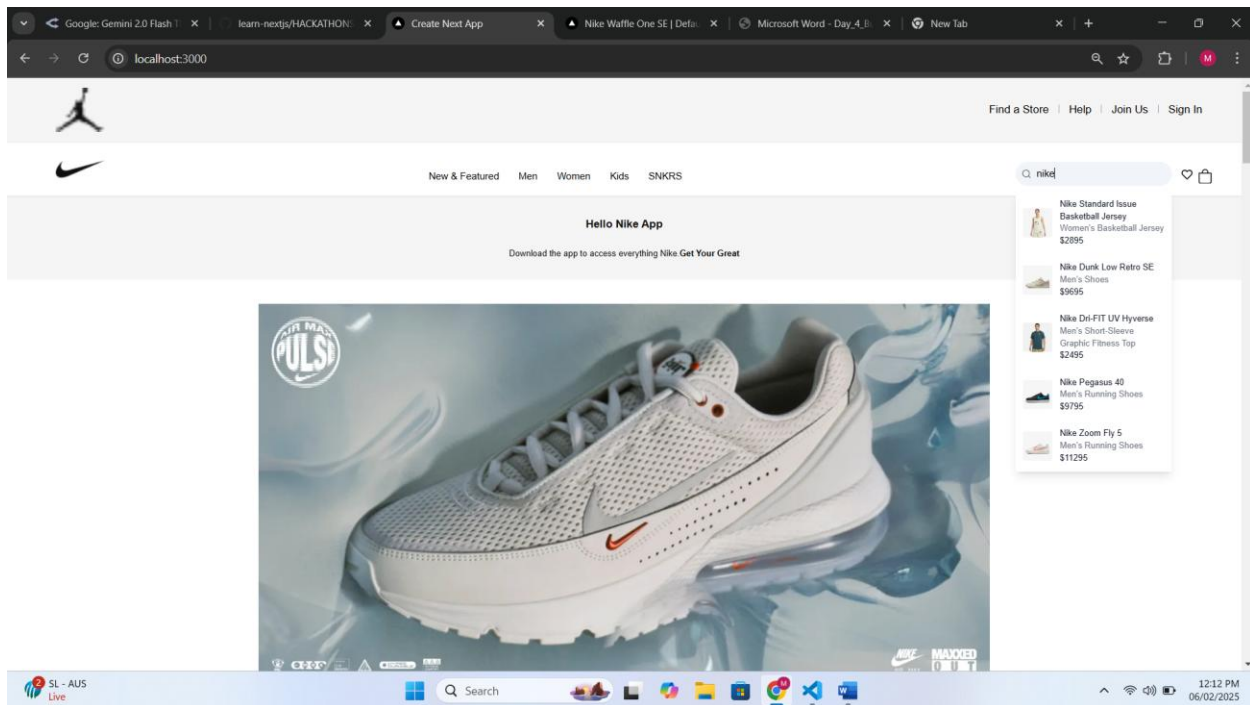
- Individual product detail pages with accurate routing and data rendering.



DAY 4- DYNAMIC FRONTEND COMPONENTS- [NIKE ONLINE STORE.]"



- Working category filters, search bar



DAY 4- DYNAMIC FRONTEND COMPONENTS- [NIKE ONLINE STORE.]"

- User Profile and Registration, Additional Information.

The screenshot shows a web browser window with the URL `localhost:3000/joinus`. The page features a registration form titled "BECOME A NIKE MEMBER" with the Nike logo at the top. The form includes the following fields and elements:

- Text input for "Email Address".
- Text input for "Password".
- Text input for "First Name".
- Text input for "Last Name".
- Text input for "dd/mm/yyyy" with a calendar icon.
- Text: "Get a Nike Member Reward every year on your Birthday."
- Dropdown menu for "India".
- Buttons for "Male" and "Female".
- Checkbox: "Sign up for emails to get updates from Nike on products, offers, and your Member benefits."
- Text: "By creating an account, you agree to Nike's [Privacy Policy](#) and [Terms of Use](#)."
- Orange button labeled "JOIN US".

The browser's taskbar at the bottom shows the system clock as 12:16 PM on 06/02/2025.

This screenshot shows the same registration form as above, but with sample data entered into the input fields. The background is a solid orange color.

- Text input for "Email Address" contains "test@gmail.com".
- Text input for "Password" is empty.
- Text input for "First Name" contains "syed saboor".
- Text input for "Last Name" contains "hasan ali".
- Text input for "dd/mm/yyyy" contains "dd/mm/yyyy".
- Text: "Get a Nike Member Reward every year on your Birthday."
- Dropdown menu for "India".
- Buttons for "Male" and "Female".
- Checkbox: "Sign up for emails to get updates from Nike on products, offers, and your Member benefits." (unchecked).
- Text: "By creating an account, you agree to Nike's [Privacy Policy](#) and [Terms of Use](#)."
- Orange button labeled "JOIN US".

The browser's taskbar at the bottom shows the system clock as 12:18 PM on 06/02/2025.

- User Login Page. Additional Page:

DAY 4- DYNAMIC FRONTEND COMPONENTS- [NIKE ONLINE STORE.]

