

# DAY 4 IMPLEMENTATION REPORT

## Introduction:

This website is designed as an **online store** aimed at providing a seamless platform for users to browse, explore, and purchase products. It features a user-friendly interface that showcases a wide range of products, complete with detailed descriptions, pricing, and images. The primary purpose of the website is to facilitate **product selling** by allowing customers to add items to their cart, view a summary of their selections, and proceed to checkout, making the shopping experience efficient and enjoyable.

## Steps Taken to Build and Integrate Components:

### Frontend Development:

1. **Website Design Creation:**
  - Designed the core pages of the website, including the **Homepage**, and **Product Page**, ensuring a user-friendly layout.
  - Focused on creating a visually appealing interface that enhances the shopping experience for users.
2. **Technology Used:**
  - **Next.js**
  - **React**
3. **Responsive Design:**
  - Used **Tailwind CSS** to develop a mobile-first, fully responsive design, ensuring the website performs well across different screen sizes and devices.
  - Incorporated Tailwind's utility classes to create a clean and consistent layout while saving development time.
4. **Integration of Components:**
  - Implemented functionality for adding products to the cart and displaying cart details dynamically.
  - Developed interactive buttons and hover effects for a better user experience.

## Backend Development:

### 1. User Authentication:

- We will learn user authentication after the hackathon.

### 2. Payment Gateway Integration:

- We plan to explore and integrate a **payment gateway**, after completing the hackathon to enable secure and seamless transactions for our customers. Potential payment solutions include **Stripe, Expay, JazzCash Merchant, Easypaisa Merchant, and Bank Transfer**. These gateways will provide multiple payment options, ensuring a convenient checkout process. Before implementation, we aim to gain a thorough understanding of **user authentication** and payment processing to ensure a secure and efficient system.

### 3. Component Integration:

- The integration of components was carried out to ensure a seamless user experience across the website. Each component, such as the **Homepage, Product Listing Page, and Cart Page**, was designed and connected dynamically to enable smooth navigation and data sharing.  
**Key aspects of integration included:**  
Fetching and displaying product data from the backend.  
Implementing the **Add to Cart** functionality to dynamically update the cart.  
Ensuring consistent styling and responsive behavior across all components using **Tailwind CSS**.  
The modular approach made it easier to maintain and extend the website's functionality.

### 4. Future Enhancements:

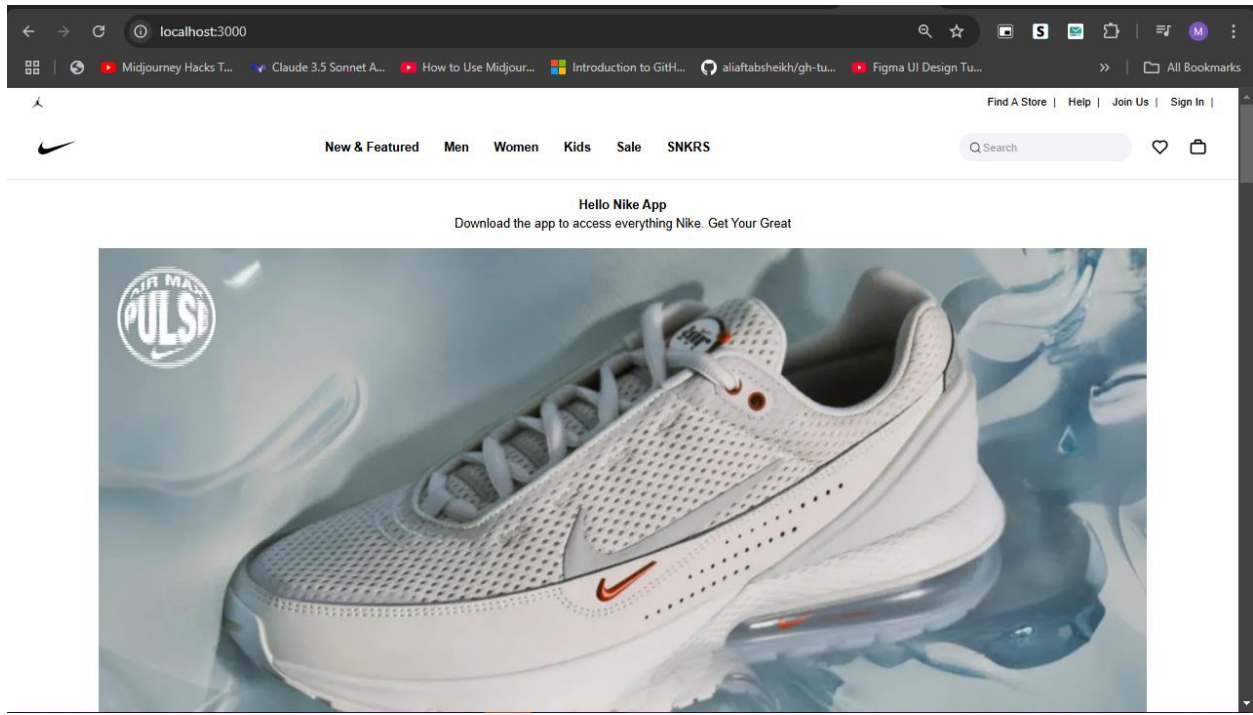
- Payment Gateway Integration
- Advanced User Authentication
- AI Recommendations
- Analytics Dashboard
- Social Media Sharing

**Making the website ready for scalability and extended features.**

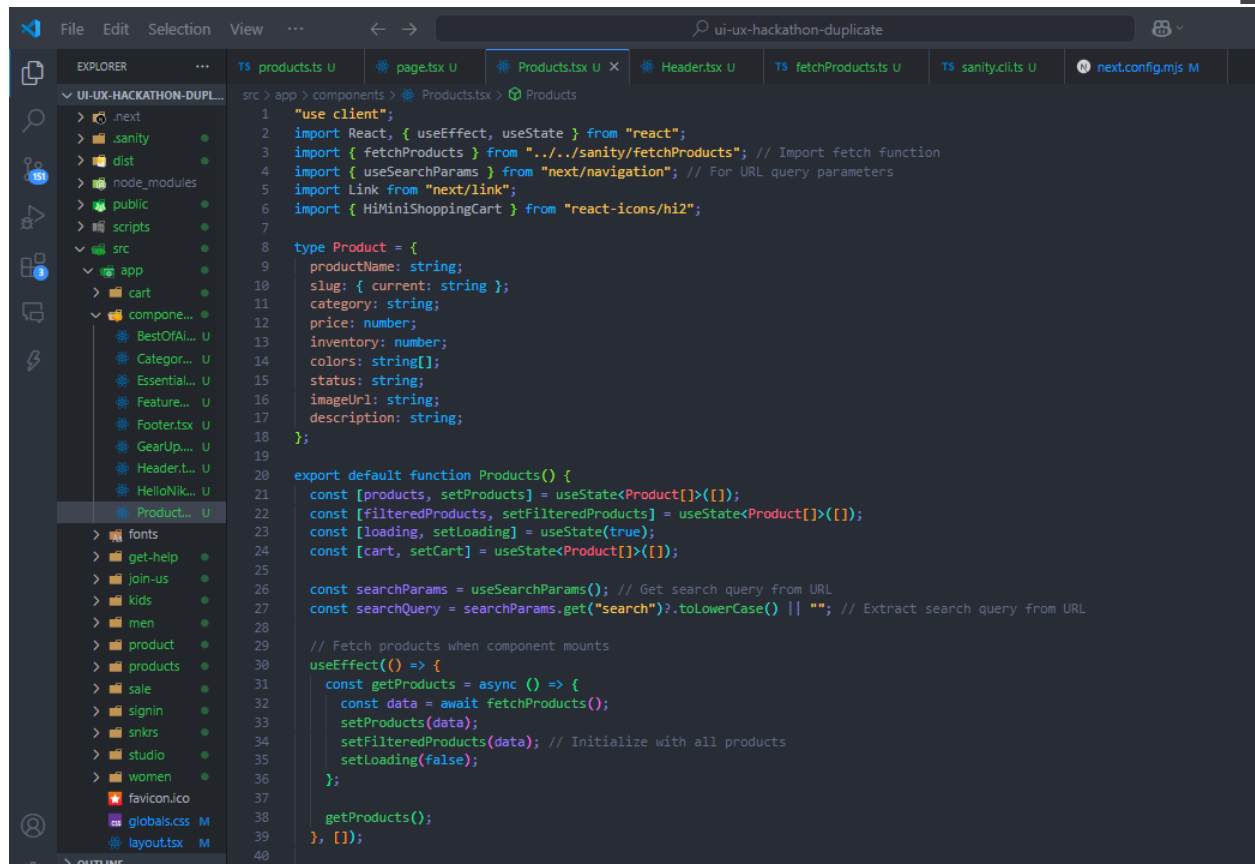
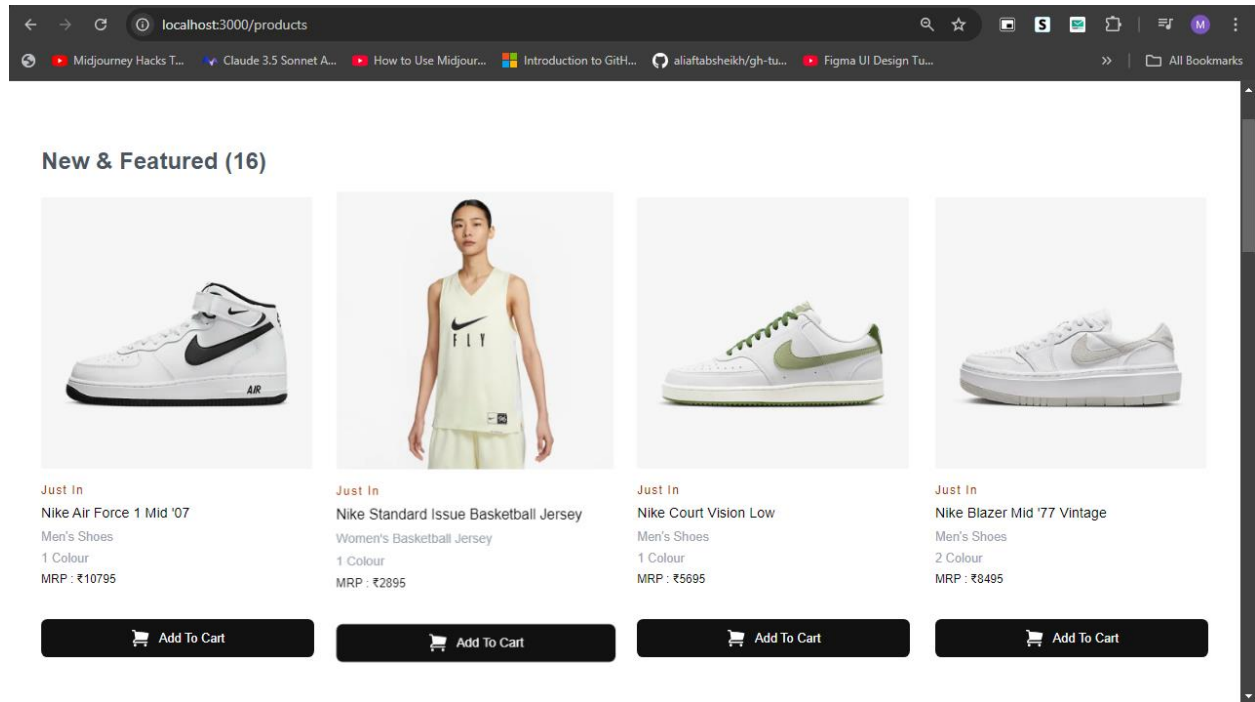
## Conclusion:

The final result is a **user-friendly and scalable e-commerce platform** designed to provide customers with a seamless shopping experience. By utilizing **Next.js** and **Tailwind CSS**, the website ensures fast performance and adaptability across all devices. With well-integrated components such as the **Product Page** and **Cart Page**, the platform offers a smooth navigation experience, dynamic data handling, and an efficient checkout process.

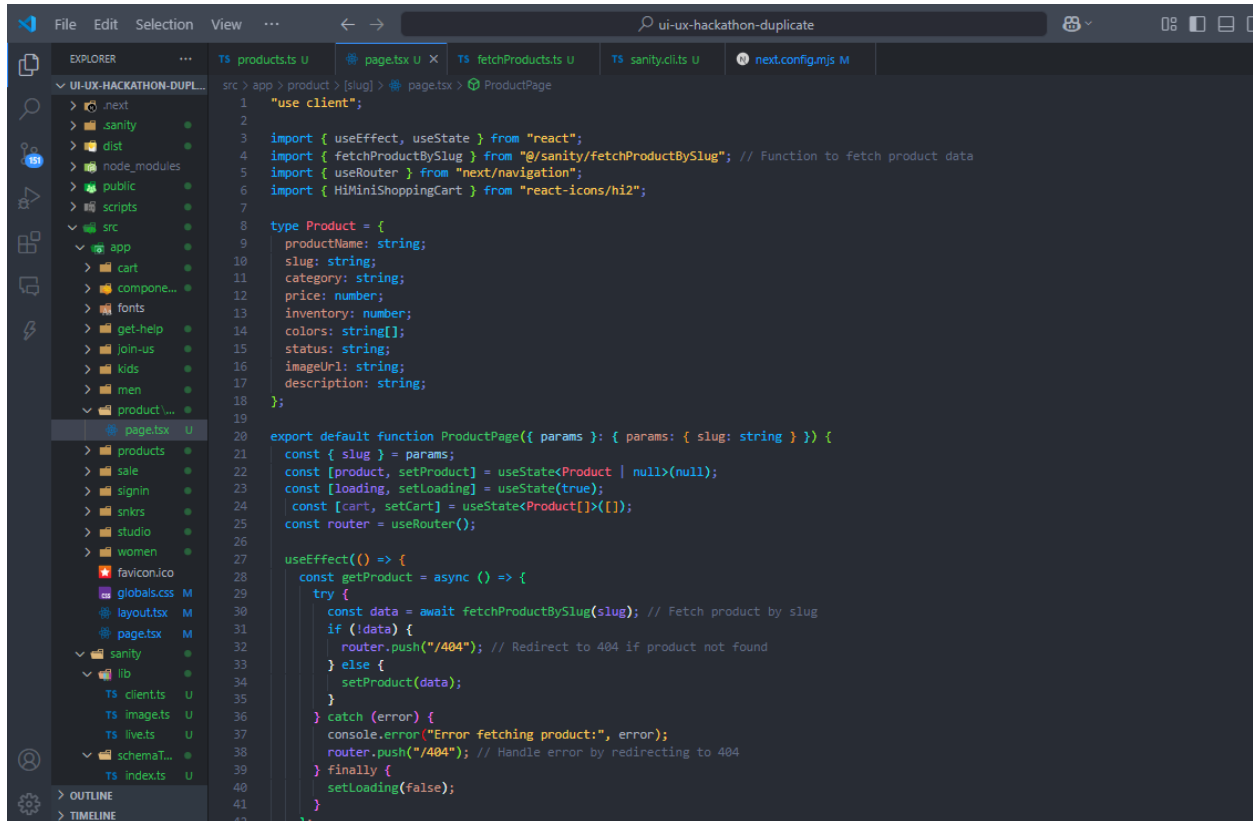
# HOME PAGE



# PRODUCTS LISTING PAGE



# DYNAMIC ROUTING (Product Detail Page)



```
1 "use client";
2
3 import { useEffect, useState } from "react";
4 import { fetchProductBySlug } from "@sanity/fetchProductBySlug"; // Function to fetch product data
5 import { useRouter } from "next/navigation";
6 import { HiMiniShoppingCart } from "react-icons/hi2";
7
8 type Product = {
9   productName: string;
10  slug: string;
11  category: string;
12  price: number;
13  inventory: number;
14  colors: string[];
15  status: string;
16  imageUrl: string;
17  description: string;
18 };
19
20 export default function ProductPage({ params }: { params: { slug: string } }) {
21   const { slug } = params;
22   const [product, setProduct] = useState<Product | null>(null);
23   const [loading, setLoading] = useState(true);
24   const [cart, setCart] = useState<Product[]>([]);
25   const router = useRouter();
26
27   useEffect(() => {
28     const getProduct = async () => {
29       try {
30         const data = await fetchProductBySlug(slug); // Fetch product by slug
31         if (!data) {
32           router.push("/404"); // Redirect to 404 if product not found
33         } else {
34           setProduct(data);
35         }
36       } catch (error) {
37         console.error("Error fetching product:", error);
38         router.push("/404"); // Handle error by redirecting to 404
39       } finally {
40         setLoading(false);
41       }
42     }
43   });
44 }
```

File Edit Selection View ... ui-ux-hackathon-duplicate

EXPLORER

- UI-UX-HACKATHON-DUPL...
- next
- sanity
- dist
- node\_modules
- public
- scripts
- src
  - app
  - cart
  - compon...
  - fonts
  - get-help
  - join-us
  - kids
  - men
  - product...
  - page.tsx U
  - products
  - sale
  - signin
  - snkrs
  - studio
  - women
- favicon.ico
- globals.css M
- layout.tsx M
- page.tsx M
- sanity
- lib
  - clients U
  - images U
  - lives U
  - schemaT... U
  - index U

OUTLINE

TIMELINE

```
src > app > product > [slug] > page.tsx > ProductPage
20 export default function ProductPage({ params }: { params: { slug: string } }) {
27   useEffect(() => {
44     getProduct();
45   }, [slug, router]);
46   const addToCart = (products: Product) => {
47     setCart((prevCart) => [...prevCart, products]);
48     alert(`${products.productName} added to cart`);
49   };
50   if (loading) {
51     return (
52       <div className="text-center text-4xl h-full py-14 my-28 text-black">
53         Loading...
54       </div>
55     );
56   }
57   if (!product) {
58     return <div className="text-center text-4xl my-28">Product not found</div>;
59   }
60
61   return (
62     <section className="text-gray-400 body-font overflow-hidden">
63       <div className="container px-5 py-24 mx-auto">
64         <div className="lg:w-4/5 mx-auto flex flex-wrap">
65           <img
66             src={product.imageUrl}
67             alt={product.productName}
68             className="lg:w-1/2 w-full lg:h-auto h-full object-cover object-center rounded"
69           />
70           <div className="lg:w-1/2 w-full lg:pl-10 lg:py-6 mt-6 lg:mt-0">
71             <h3 className="text-[#9E3500] text-[20px] tracking-widest title-font mb-1">
72               {product.status}
73             </h3>
74             <h1 className="text-black text-5xl title-font font-medium mb-1">
75               {product.productName}
76             </h1>
77             <p className="mt-3 text-3xl text-gray-600 mb-3">
78               {product.category}
79             </p>
80             <p className="title-font font-medium text-2xl text-black my-5">
81               Price: ₹{product.price}
82             </p>

```

File Edit Selection View ... ui-ux-hackathon-duplicate

EXPLORER

- UI-UX-HACKATHON-DUPL...
- next
- sanity
- dist
- node\_modules
- public
- scripts
- src
  - app
  - cart
  - compon...
  - fonts
  - get-help
  - join-us
  - kids
  - men
  - product...
  - page.tsx U
  - products
  - sale
  - signin
  - snkrs
  - studio
  - women
- favicon.ico
- globals.css M
- layout.tsx M
- page.tsx M
- sanity
- lib
  - clients U
  - images U
  - lives U
  - schemaT... U
  - index U

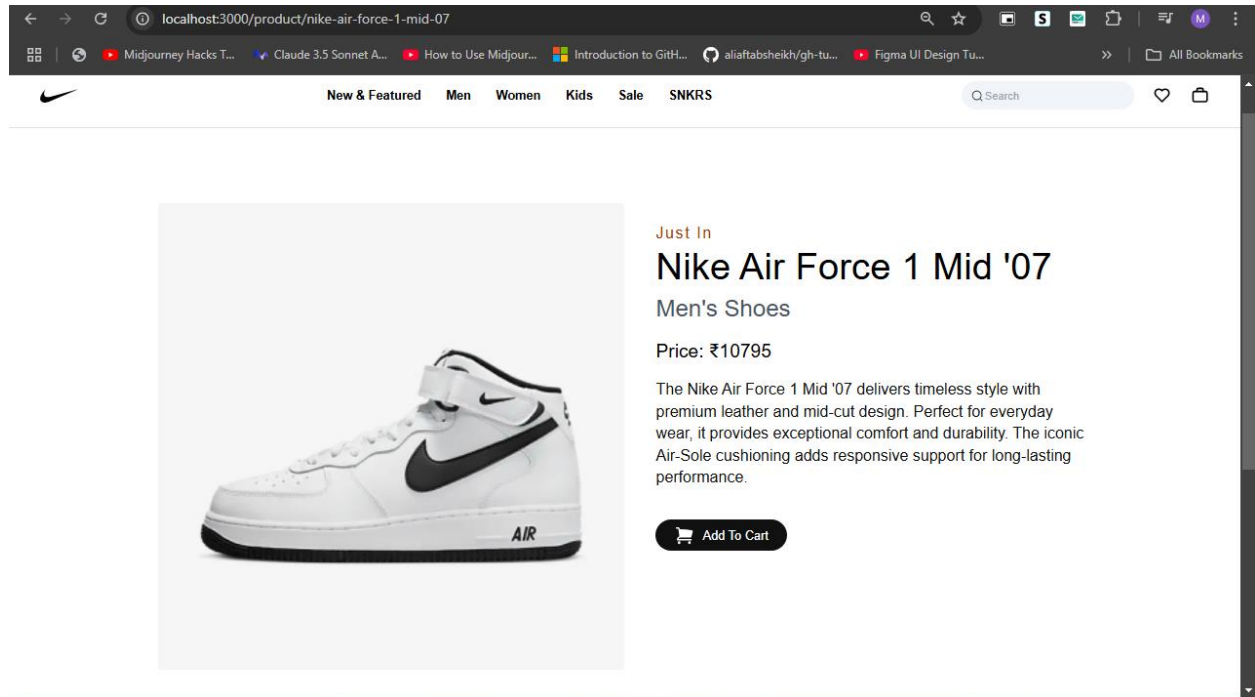
OUTLINE

TIMELINE

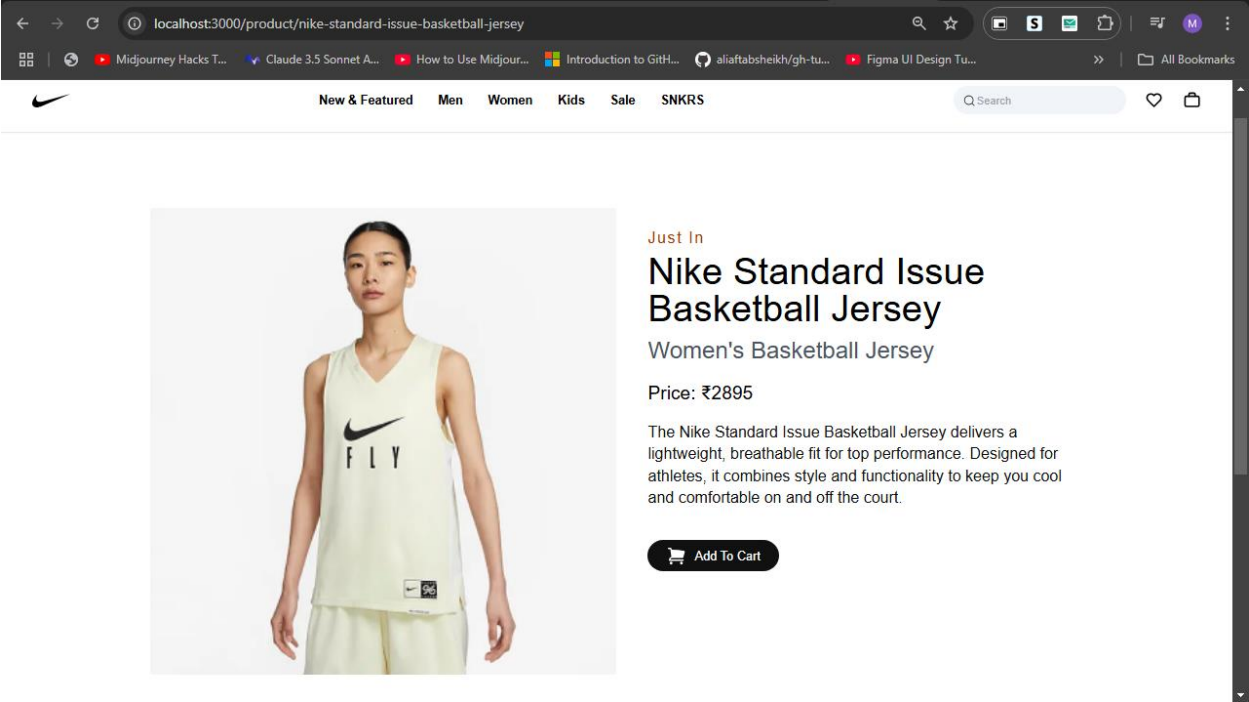
```
src > app > product > [slug] > page.tsx > ProductPage
20 export default function ProductPage({ params }: { params: { slug: string } }) {
21   <p>
22     <p className="title-font font-medium text-2xl text-black my-5">
23       Price: ₹{product.price}
24     </p>
25     <p className="leading-relaxed text-[20px] text-[#111111] line-height">
26       {product.description}
27     </p>
28     <div className="my-5">
29       <button onClick={() => addToCart(product)} className="flex flex-row gap-3 text-white my-10 bg-[#111111] py-2 px-6
30         focus:outline-none hover:bg-[#414141] duration-700 b rounded">
31         <HiMiniShoppingCart className="text-white text-2xl" />
32         Add To Cart
33       </button>
34     </div>
35   </div>
36   </div>
37   </div>
38   </div>
39   </div>
40   </div>
41   </div>
42   </div>
43   </div>
44   </div>
45   </div>
46   </div>
47   </div>
48   </div>
49   </div>
50   </div>
51   </div>
52   </div>
53   </div>
54   </div>
55   </div>
56   </div>
57   </div>
58   </div>
59   </div>
60   </div>
61   </div>
62   </div>
63   </div>
64   </div>
65   </div>
66   </div>
67   </div>
68   </div>
69   </div>
70   </div>
71   </div>
72   </div>
73   </div>
74   </div>
75   </div>
76   </div>
77   </div>
78   </div>
79   </div>
80   </div>
81   </div>
82   </div>
83   </div>
84   </div>
85   </div>
86   </div>
87   </div>
88   </div>
89   </div>
90   </div>
91   </div>
92   </div>
93   </div>
94   </div>
95   </div>
96   </div>
97   </div>
98   </div>
99   </div>
100   </div>
101   </div>

```

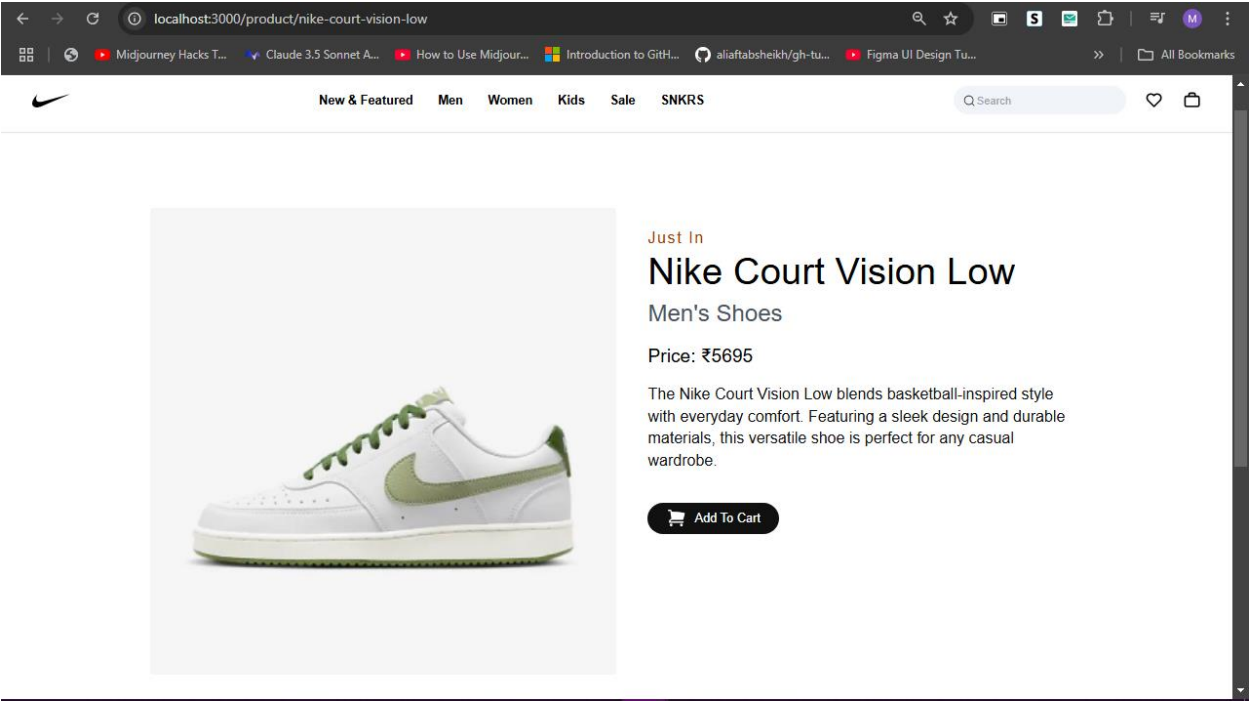
# Product 1



# Product 2

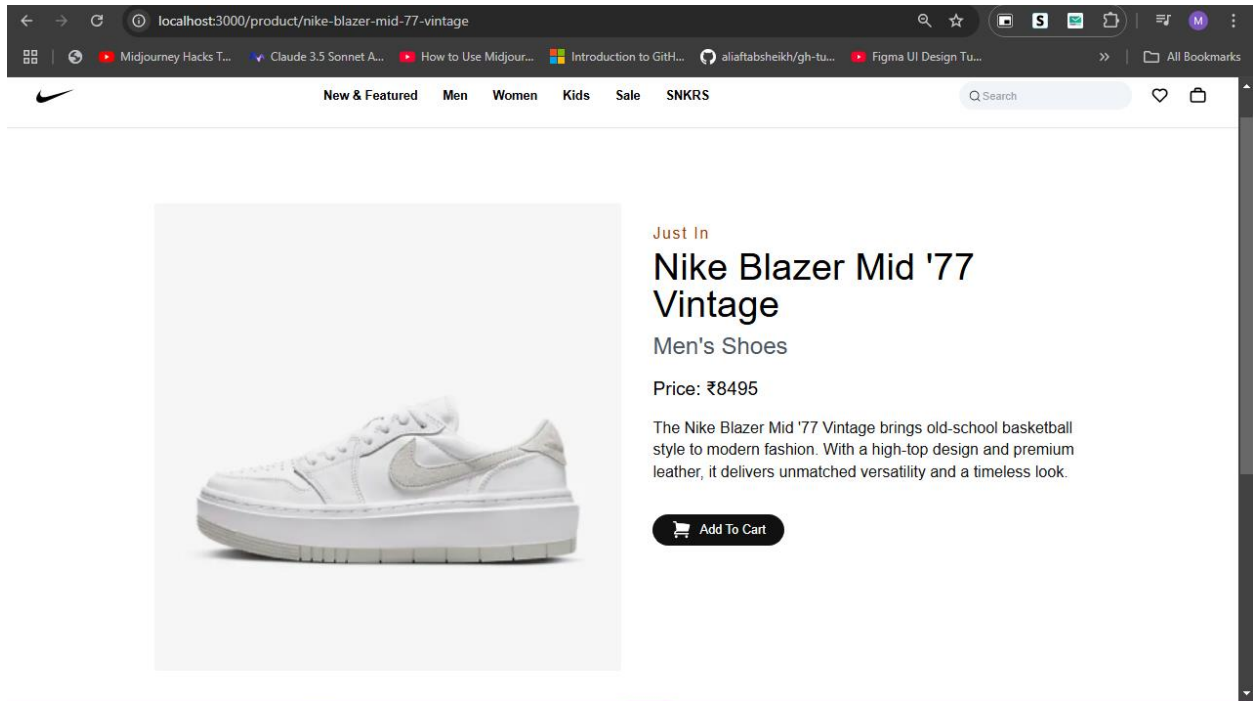


# Product 3

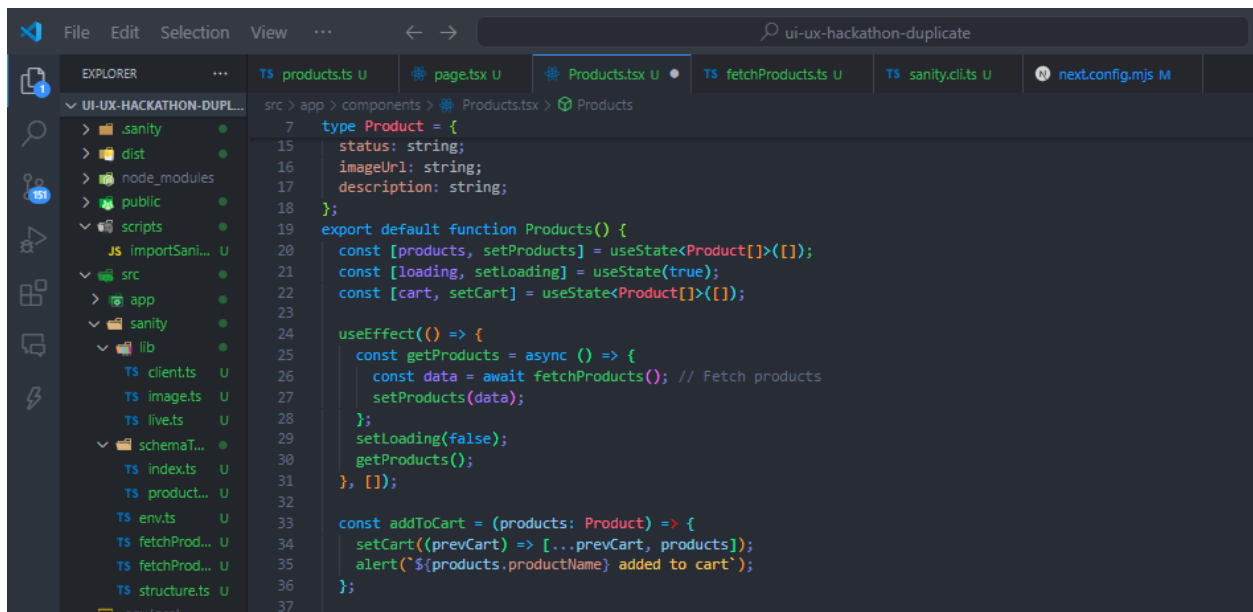




# Product 4

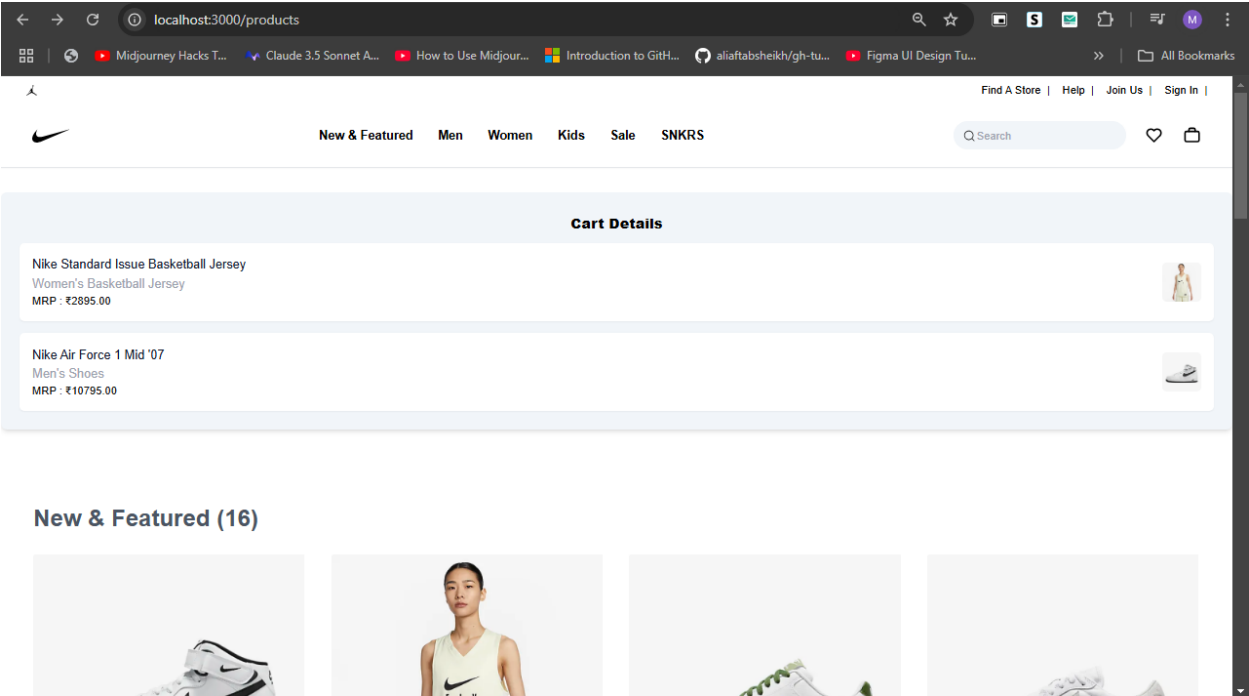
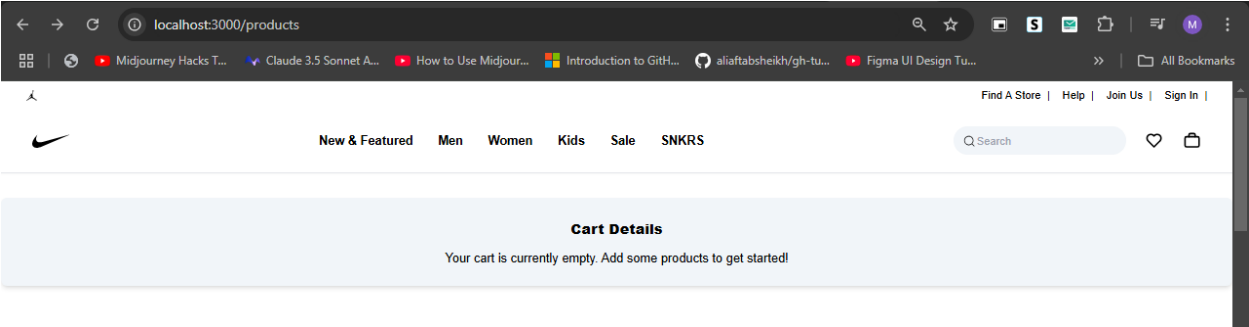


## CART

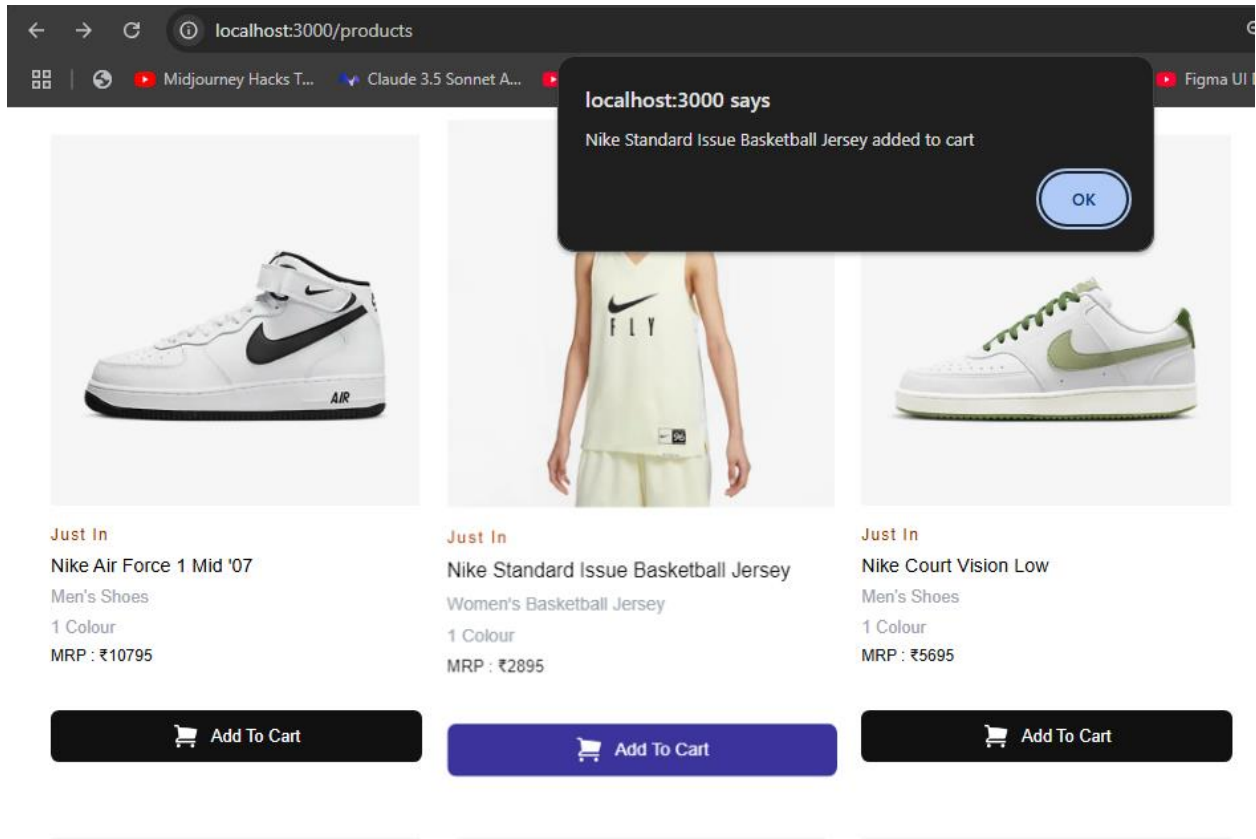


```
File Edit Selection View ... ui-ux-hackathon-duplicate
EXPLORER
  UI-UX-HACKATHON-DUPL...
    .sanity
    dist
    node_modules
    public
    scripts
    JS importSani... U
    src
    app
    sanity
    lib
      client.ts U
      image.ts U
      live.ts U
    schemaT...
      index.ts U
      product... U
      env.ts U
      fetchProd... U
      fetchProd... U
      structure.ts U
    .env.local
    .eslintrc.json
    .gitignore
    next-env.d.ts
    next.config.mjs M
    package-loc... M
    package.json M
    postcss.config.mjs
    README.mmd
    sanity.cli.ts U
    sanity.config... U
    tailwind.config.ts
    tsconfig.json
  OUTLINE
  TIMELINE
src > app > components > Products.tsx > Products > cart.map() callback
19 export default function Products() {
20
21   /* cart details */
22
23   <div className="mt-8 bg-slate-100 p-6 rounded-lg shadow-md">
24     <h2 className="text-lg font-black text-black text-center mb-3">
25       Cart Details
26     </h2>
27     {cart.length > 0 ? (
28       <ul className="space-y-4">
29         {cart.map((item, index) => (
30           <li
31             key={index}
32             className="flex justify-between items-center bg-white shadow-sm p-4 rounded-md"
33           >
34             <div>
35               <p className="font-medium text-slate-900">
36                 {item.productName}
37               </p>
38               <p>{item.category}</p>
39               <p className="text-sm text-[#111111]">
40                 { " " }
41                 MRP : ₹{item.price.toFixed(2)}
42               </p>
43             </div>
44             <img
45               src={item.imageUrl}
46               alt={item.productName}
47               width={50}
48               height={50}
49               className="rounded-md"
50             </img>
51           </li>
52         )
53       )
54     ) : (
55       <p className="text-black text-center">
56         Your cart is currently empty. Add some products to get started!
57       </p>
58     )
59   }
60 </div>
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
```

```
File Edit Selection View ... ui-ux-hackathon-duplicate
EXPLORER
  UI-UX-HACKATHON-DUPL...
    .sanity
    dist
    node_modules
    public
    scripts
    JS importSani... U
    src
    app
    sanity
    lib
      client.ts U
      image.ts U
      live.ts U
    schemaT...
      index.ts U
      product... U
      env.ts U
      fetchProd... U
    fetchProd... U
  OUTLINE
  TIMELINE
src > app > components > Products.tsx > Products > cart.map() callback
19 export default function Products() {
20
21   {products.map((product, i) => (
22     <Link>
23
24   /* add to cart functionality */
25
26   <div className="my-5">
27     <button
28       onClick={() => addToCart(product)}
29       className="flex flex-row justify-center gap-3 text-white my-10 bg-[#111111] py-3 px-6 focus:outline-none
30       hover:bg-indigo-600 rounded-lg w-full duration-700"
31     >
32       <HiMiniShoppingCart className="text-white text-2xl" />
33       Add To Cart
34     </button>
35   </div>
36 </div>
37
38   )
39 )
40 </div>
41
42 </section>
43
```



# NOTIFICATION



```
const addToCart = (products: Product) => {  
  setCart((prevCart) => [...prevCart, products]);  
  alert(`${products.productName} added to cart`);  
};
```

# SEARCH BAR

