

Marketplace Hackathon

E-Commerce Store

REPORT

Introduction:

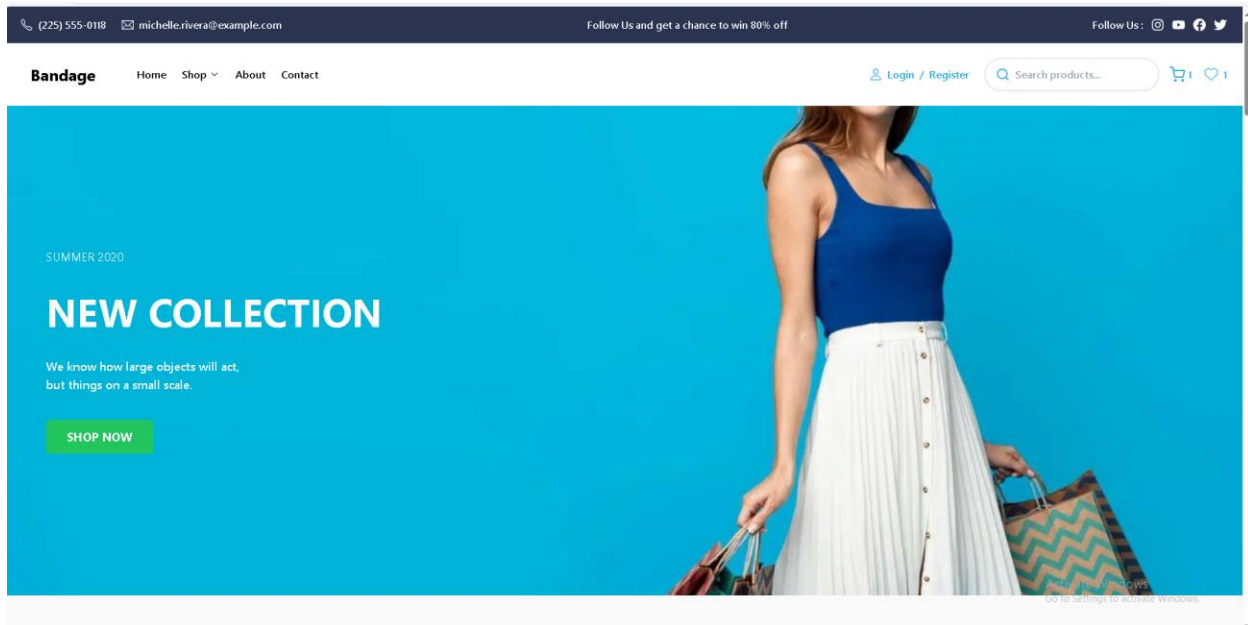
- An E-Commerce Store , responsive design , dynamic routing.

Frontend :

Nextjs , tailwind css , shadcn.

Backend :

Sanity.



Sanity:

The screenshot shows the Sanity.io dashboard for a project named "day3 api". The URL in the browser is https://www.sanity.io/manage/personal/project/qge7xfcl/api. The user is logged in as "Abdullah Arif". The dashboard has a sidebar with a "DA" logo and a "day3 api" header. Below the header, there are tabs for "PLAN" (Growth Trial), "STATUS" (Active), and "PROJECT ID" (qge7xfcl). The main navigation bar includes links for "Getting started", "Overview", "Members", "Studios", "Datasets", "Access", "Activity", "Usage", "Plan", "API", and "Settings". The "API" tab is selected, showing a "Webhooks" section with a "GROQ-powered webhooks" subsection. It explains that HTTP callbacks are triggered by changes in the content lake and shows a "0 of 2 webhooks" status. Below this is a "CORS origins" section with a table of hosts that can connect to the project API. A "What's new" box on the left mentions "Sanity Create Content Mapping, Visual Editing, and Content Releases". An "Activate Windows" watermark is visible on the right side.

What's new
Sanity Create Content Mapping, Visual Editing, and Content Releases

ORIGIN	CREDENTIALS	CREATED
https://hackathon3-snowy-alpha.vercel.app	Allowed	52 minutes
http://localhost3000	Allowed	2 weeks

Activate Windows
Go to Settings to activate Windows

```
Run Terminal Help
Hackathon3
shop.tsx page.tsx ...[id] CartContext.tsx page.tsx ...Login M SearchBar.tsx .env.local SearchResults.tsx shop-cart.tsx page.tsx ...search header.tsx header2.tsx
$ .env.local
1 NEXT_PUBLIC_SANITY_PROJECT_ID="qje7xfz7"
2 NEXT_PUBLIC_SANITY_DATASET="production"
3 NEXT_PUBLIC_SANITY_PROJECT_TOKEN="sk57fzTT0372xW9eYA7N3SbrZnFDM9uN73ZmLd5wcW04r1Y1lQu1L6ZY8BVFNSKFG9a7NFg1uHRWWTswBuWBYmkLLIAYqLVtq3X09aCqEmJ2a3aeXTAe3BkRfowuIw05aEkMLjser3FgfegY4"
```

```
sanity > schemaTypes > TS products > ...
1 import { defineType, defineField } from "sanity";
2
3 export const product = defineType({
4   name: "product",
5   title: "Product",
6   type: "document",
7   fields: [
8     defineField({
9       name: "title",
10      title: "Title",
11      type: "string",
12      validation: (rule) => rule.required(),
13      description: "The name of the product. This will be used for search and display purposes.",
14    }),
15    defineField({
16      name: "slug",
17      title: "Slug",
18      type: "slug",
19      options: {
20        source: "title",
21        maxLength: 200, // Adjust as needed
22      },
23      validation: (rule) => rule.required(),
24      description: "A unique identifier for the product used in URLs.",
25    }),
26    defineField({
27      name: "description",
28      type: "text",
29      title: "Description",
30      validation: (rule) => rule.required(),
31      description: "A detailed description of the product.",
32    }),
33    defineField({
34      name: "productImage",
35      type: "image",
36      title: "Product Image",
37      validation: (rule) => rule.required(),
38      options: {
39        hotspot: true, // Enables image cropping and hotspot functionality
40      },
41      description: "The main image of the product.",
42    }),
43    defineField({
44      name: "price",
45      type: "number",
46      title: "Price",
47      validation: (rule) => rule.required().min(0)
```

Product Listing :



Rustic Vase Set

Bring the charm of nature into your home with the Rustic Vase Set. Perfect for those who appreciate timeless beauty and a warm, inviting atmosphere. [T...](#)

\$210.00



[\\$rustic.vase.home.decor.vintage.interior.design](#)



Amber Haven

Step into a world of warmth and tranquility with Amber Haven—a collection inspired by the golden glow and timeless beauty of amber. This sophisticated...

\$150.00



[\\$amber.luxury.cozy.elegant.furniture](#)



Cloud Haven Chair

Sink into comfort with the Cloud Haven Chair—where softness meets support in a beautifully designed piece that feels like a personal retreat. Inspired...

\$230.00



[\\$cloud.chair.comfy.home.decor.modern.furniture](#)



Marble Ease

Introducing MarbleEase—a luxurious collection that brings the timeless elegance of marble into your home with ease and sophistication. Designed for th...

\$419.00



[\\$marble.luxury.furniture.modern.elegance](#)



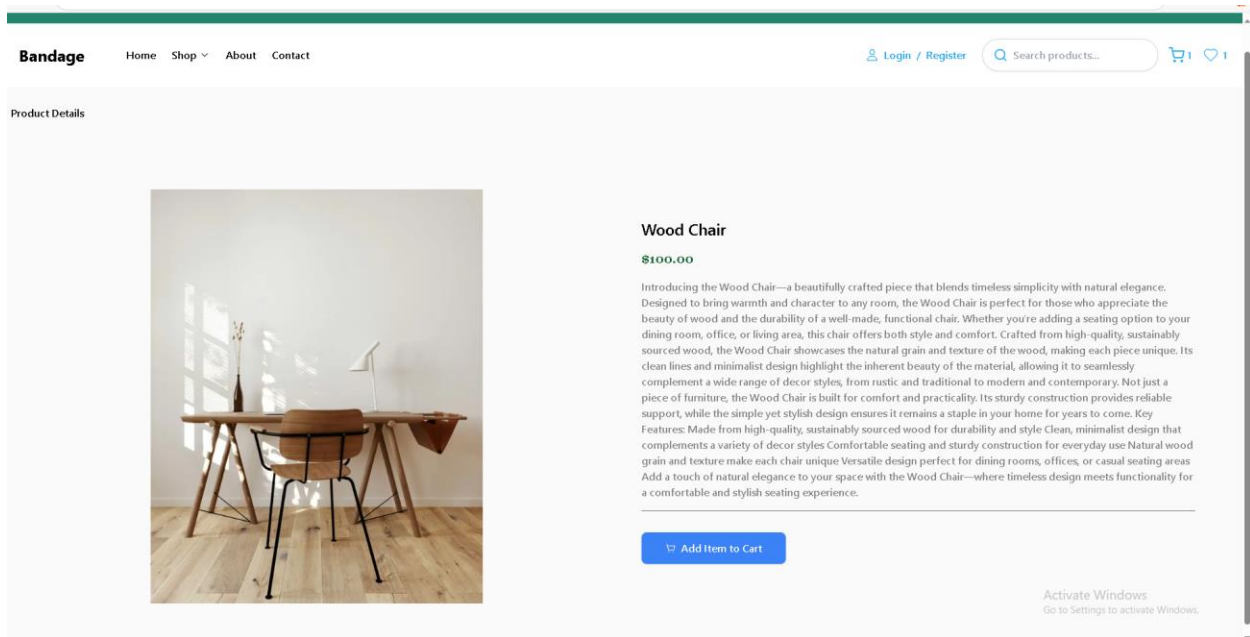
Dynamic Route :

```
File Edit Selection View Go Run Terminal Help
shop.txt page.txt...[id] page.txt...products CartContext.txt SearchBar.txt SearchResults.txt shop-cart.txt page.txt...search header.txt header2.txt

EXPLORER
HACKATHON
  routes
    cart
      page.txt
    contact
    login
    pricing
    products
      [id]
        page.txt
      page.txt
    register
    search
    page.txt
    fonts
    multi-components
      aboutsection
      contactsection
      headers
        green-header.txt
        header.txt
        header2.txt
        top-header.txt
      heart
        heart-card.txt
        heart-product.txt
    home
      classic-product.txt
      editor.txt
      future-post.txt
      hero-section.txt
      neural.txt
      our-product.txt
    pricingsection
    productlistpage
    shop-cart.txt
    shop.txt
    search
      SearchBar.txt
      SearchResults.txt

OUTLINE
  TABLE

1 "use client";
2
3 import { useEffect, useState } from "react";
4 import { useParams } from "next/navigation"; // Use useParams instead of destructuring directly
5 import { client } from "@sanity/lib/client";
6 import { Product } from "@app/multi-components/productlistpage/shop-cart";
7 import Image from "next/image";
8 import { Button } from "@components/ui/button";
9 import { PShoppingCartSimpleThin } from "react-icons/pl";
10 import Green_Header from "@app/multi-components/headers/green-header";
11 import Header from "@app/multi-components/headers/header";
12 import { useCart } from ".../cart/CartContext";
13
14 const getProduct = async (id: string): Promise<Product | null> => {
15   if (!id) return null; // Ensure ID is valid before querying
16
17   const query = `*[_type == "product" && _id == ${id}][0] {
18     _id,
19     title,
20     description,
21     "imageUrl": productImage.asset->url,
22     price,
23     tags,
24     discountPercentage,
25     isNew
26   }`;
27
28   try {
29     return await client.fetch(query, { id });
30   } catch (error) {
31     console.error("Error fetching product:", error);
32     return null;
33   }
34 }
35
36 export default function ProductDetail() {
37   const params = useParams(); // Correct way to get dynamic route params in Next.js 13+
38   const id = params?.id as string; // Ensure id is treated as a string
39   const [product, setProduct] = useState<Product | null>(null);
40   const [loading, setLoading] = useState(true);
41   const [error, setError] = useState<string | null>(null);
42   const [addToCart] = useCart();
43
44   useEffect(() => {
45     if (!id) return;
46
47     async function fetchData() {
48       try {
49         const product = await getProduct(id);
50         setProduct(product);
51         setLoading(false);
52       } catch (error) {
53         setError("Error fetching product details");
54       }
55     }
56     fetchData();
57   }, [id]);
58
59   if (loading) return <div>Loading...</div>;
60   if (error) return <div>{error}</div>;
61   if (!product) return <div>Product not found</div>;
62
63   return (
64     <div>
65       <Header />
66       <div>
67         <h2>{product.title}</h2>
68         <img alt={product.imageUrl} />
69         <p>{product.description}</p>
70         <p>Price: {product.price}</p>
71         <p>Tags: {product.tags}</p>
72         <p>Discount: {product.discountPercentage}</p>
73         <p>Is New: {product.isNew}</p>
74       </div>
75       <div>
76         <button>Add to Cart</button>
77       </div>
78     </div>
79   );
80 }
```



Cart :

example.com

hackathon3-snowy-alpha.vercel.app says

Added to cart successfully!

OK

off

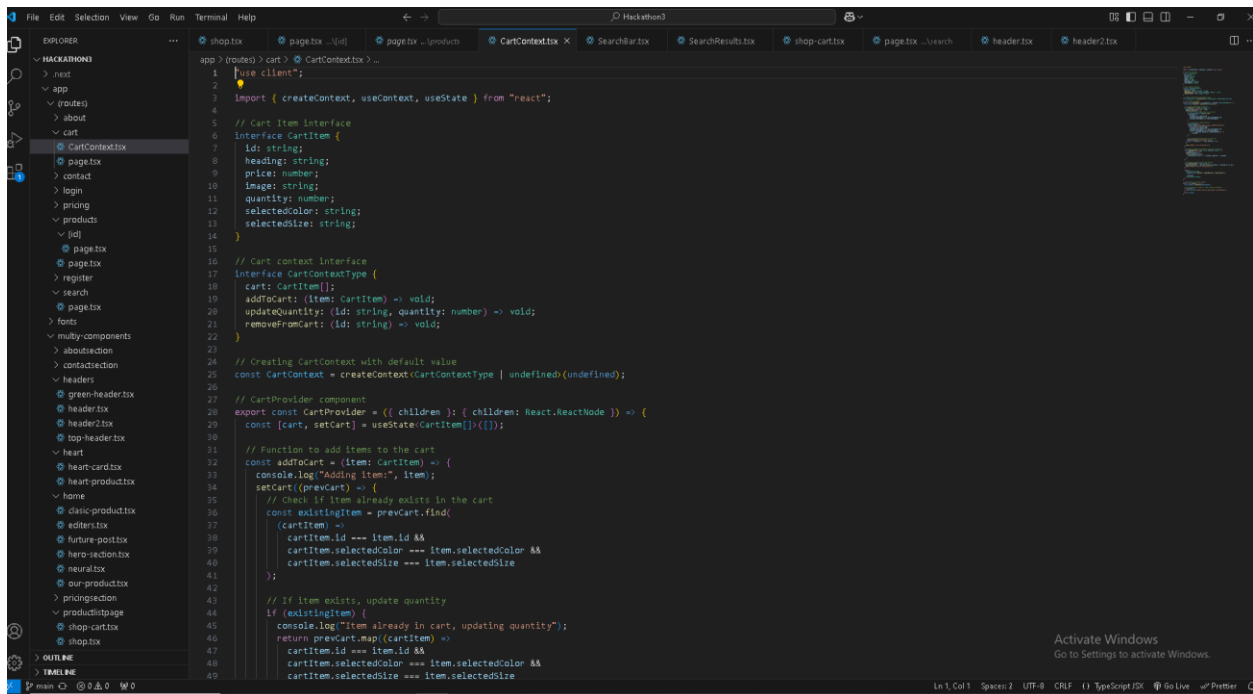
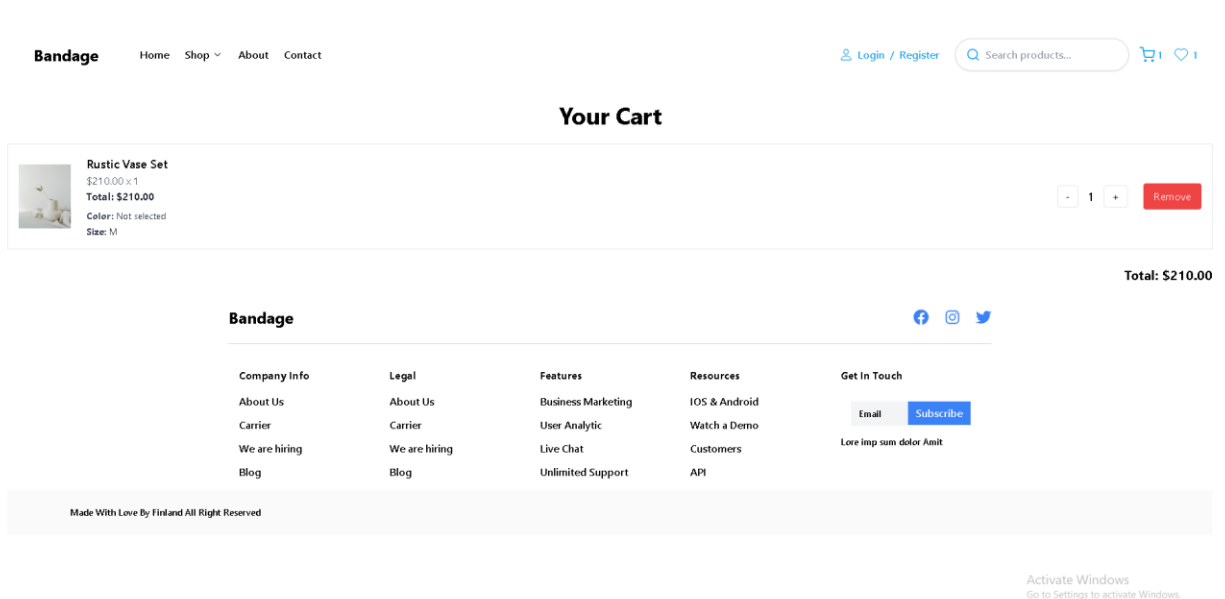
[Login](#) / [Register](#)

Amber Haven

\$150.00

Step into a world of warmth and tranquility with Amber Haven—a timeless beauty of amber. This sophisticated line combines the soft design to create a space that feels both welcoming and luxurious. Haven infuses your home with a sense of calm, warmth, and understated materials and a keen eye for detail, each piece in the Amber Haven collection offers comfort and relaxation. The soft, amber-toned accents, paired with a touch of nature's beauty into your home. Whether you're furnishing a new space or updating an existing one, Amber Haven creates an inviting atmosphere that radiates peace. Its versatility allows it to seamlessly integrate into various décor styles, making it a choice for those who love both timeless elegance and contemporary comfort and luxury, where every piece is designed to enhance your home. The warm amber tones and elegant design create a cozy, inviting ambiance. Its durability and long-lasting appeal. Versatile style complements a variety of interior designs. Perfect for creating a serene and luxurious space in any room. Bring the golden glow of Amber Haven and timeless design come together to create a peaceful and elegant home.

[Add Item to Cart](#)



Login / Register :

Welcome Back!

Enter your credentials to access your account.

Email

Password

[Log In](#)

[Don't have an account? Register here](#)

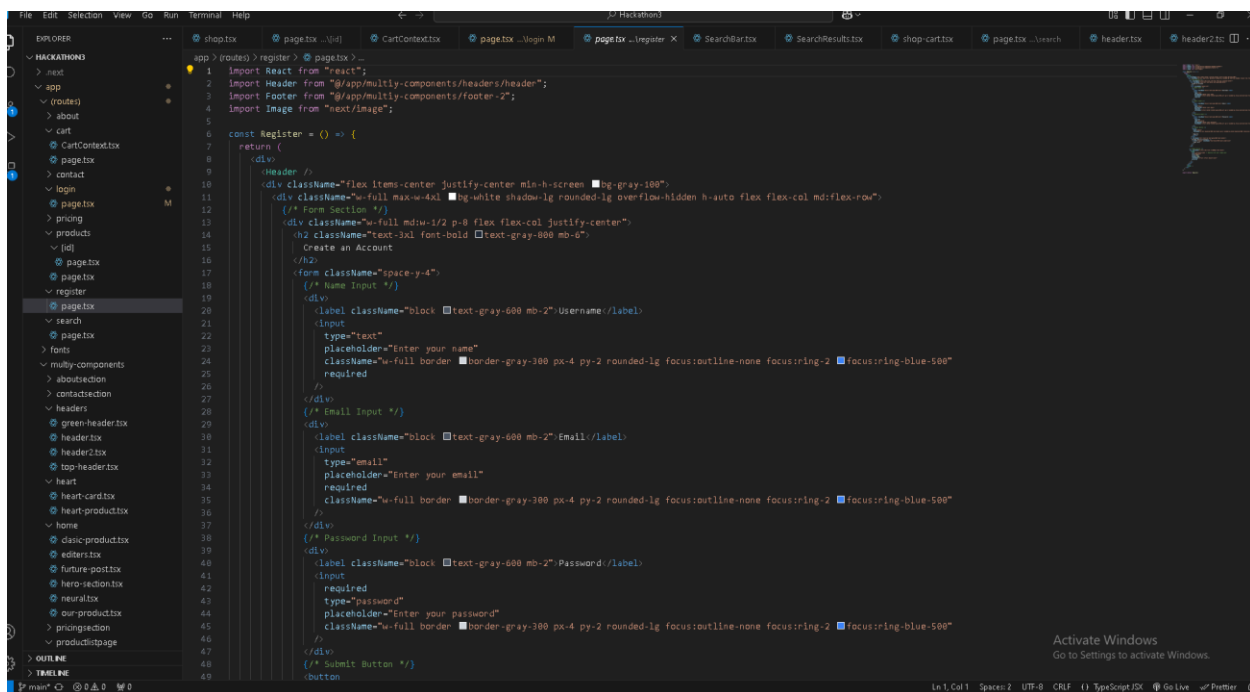
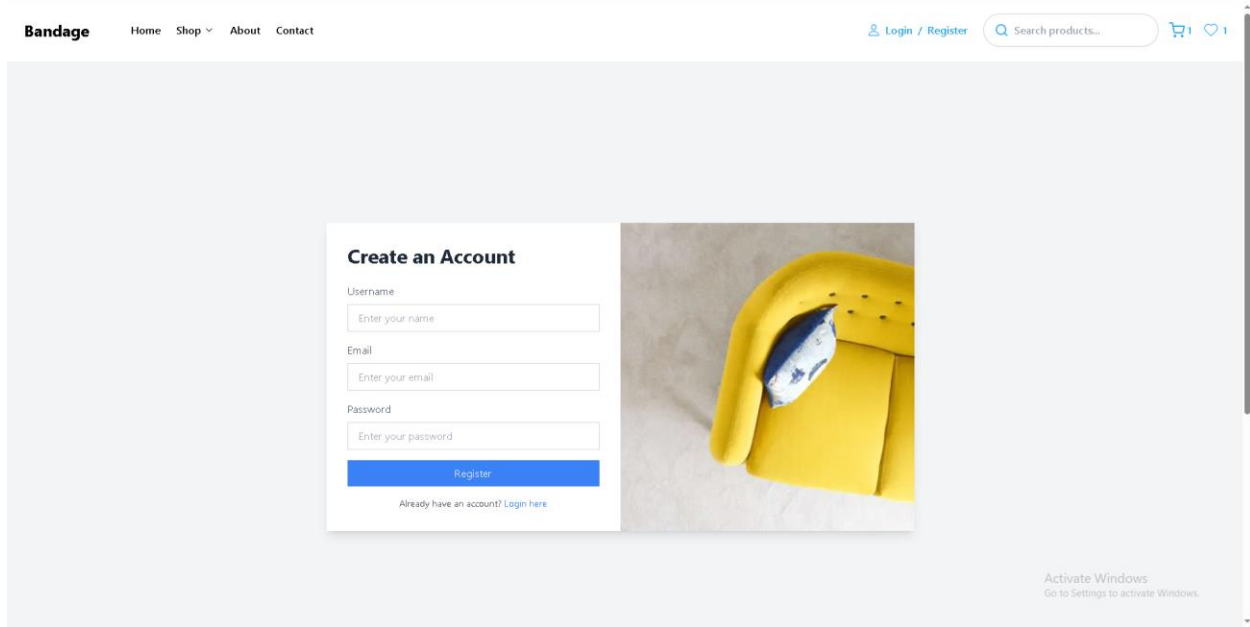


Activate Windows

```
shop.tsx  page.tsx ...[id]  CartContext.tsx  page.tsx ...login M  SearchBar.tsx  SearchResults.tsx  shop-cart.tsx  page.tsx ...search  header.tsx

CKATHONS
next
app
  (routes)
  > about
  > cart
  > CartContext.tsx
  > page.tsx
  > contact
  > login
  > page.tsx M
  > pricing
  > products
  > [id]
  > page.tsx
  > register
  > search
  > page.tsx
  fonts
  multi-components
  > aboutsection
  > contactsection
  > headers
  > green-header.tsx
  > header.tsx
  > header2.tsx
  > top-header.tsx
  > heart
  > heart-card.tsx
  > heart-product.tsx
  > home
  > dasic-product.tsx
  > editors.tsx
  > furture-post.tsx
  > hero-section.tsx
  > neural.tsx
  > our-product.tsx
  > pricingsection
  > productlistpage
  > shop-cart.tsx

app > (routes) > login > page.tsx > login
1  import React from "react";
2  import Image from "next/image";
3  import Header from "@app/multi-components/headers/header";
4  import Footer from "@app/multi-components/footer-2";
5
6  const Login = () => {
7    return (
8      <div>
9        <Header />
10       <div className="flex items-center justify-center min-h-screen bg-gray-100">
11         <div className="w-full max-w-4xl bg-white shadow-lg rounded-lg overflow-hidden h-auto flex flex-col md:flex-row">
12           { /* Form Section */ }
13           <div className="w-full md:w-1/2 p-8 flex flex-col justify-center">
14             <h2 className="text-2xl font-bold text-gray-800 mb-4">Welcome Back!</h2>
15             <p className="text-sm text-gray-600 mb-8">Enter your credentials to access your account.
16             </p>
17             <form className="space-y-4">
18               { /* Email Input */ }
19               <div>
20                 <label
21                   htmlFor="email"
22                   className="text-sm font-semibold text-gray-600"
23                 >
24                   Email
25                 </label>
26                 <input
27                   type="email"
28                   id="email"
29                   placeholder="Enter your email"
30                   className="w-full px-4 py-2 mt-2 border rounded-lg text-gray-700 focus:outline-none focus:ring-2 focus:ring-blue-500"
31                   required
32                 />
33               </div>
34             </div>
35             { /* Password Input */ }
36             <div>
37               <label
38                 htmlFor="password"
39                 className="text-sm font-semibold text-gray-600"
40               >
41                 Password
42               </label>
43               <input
44                 type="password"
45                 id="password"
46               />
47             </div>
48           </div>
49         </div>
50       </div>
51     );
52   };
53   export default Login;
```

Search Bar :

