

Day 4 - My Progress on Building Dynamic functional Frontend Components

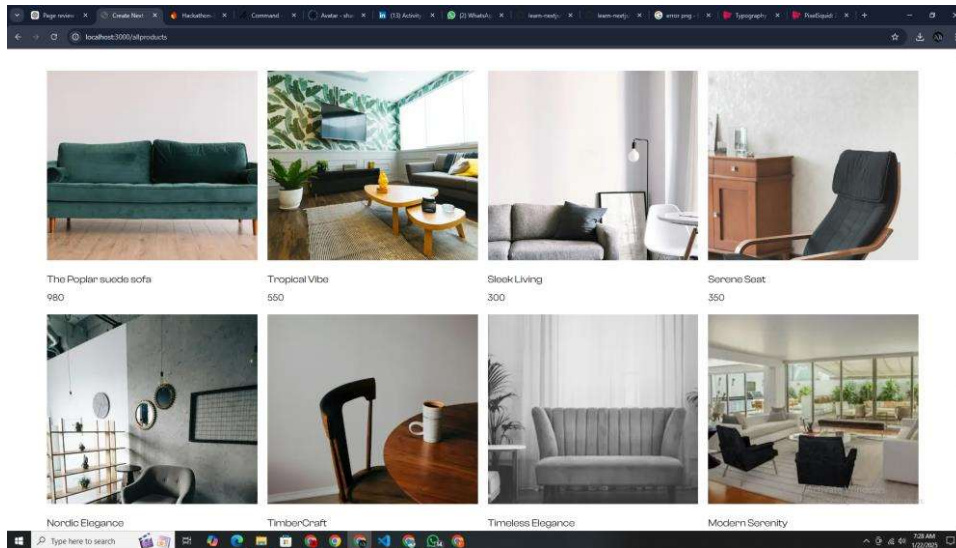
Objective

The focus of today's task was to develop key frontend components for our marketplace application. The goal was to ensure these components are dynamic, reusable, and responsive, fetching data seamlessly from Sanity CMS or APIs.

Product Listing Component:

- Designed a grid layout to display products dynamically, including details such as:
 - **Name**
 - **Price**
 - **Image**
 - **Product dimensions**

- Created a reusable ProductCard component to showcase each product. Data was retrieved using the Sanity CMS API.

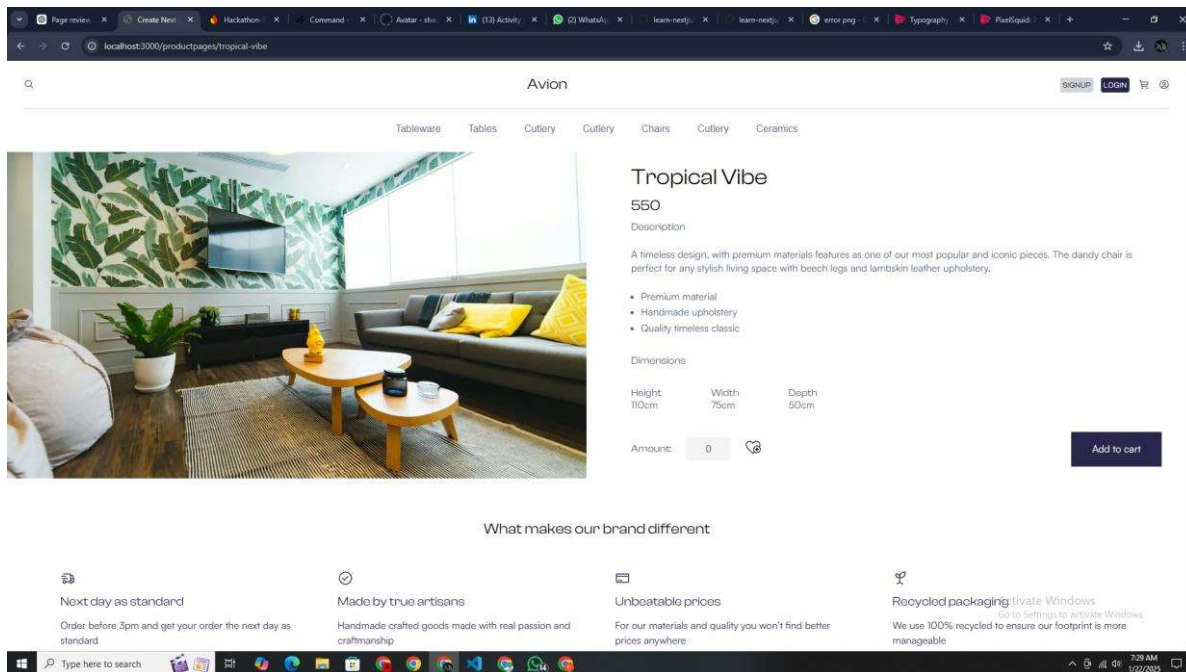


Product Detail Component:

Developed dynamic product pages using Next.js dynamic routing.

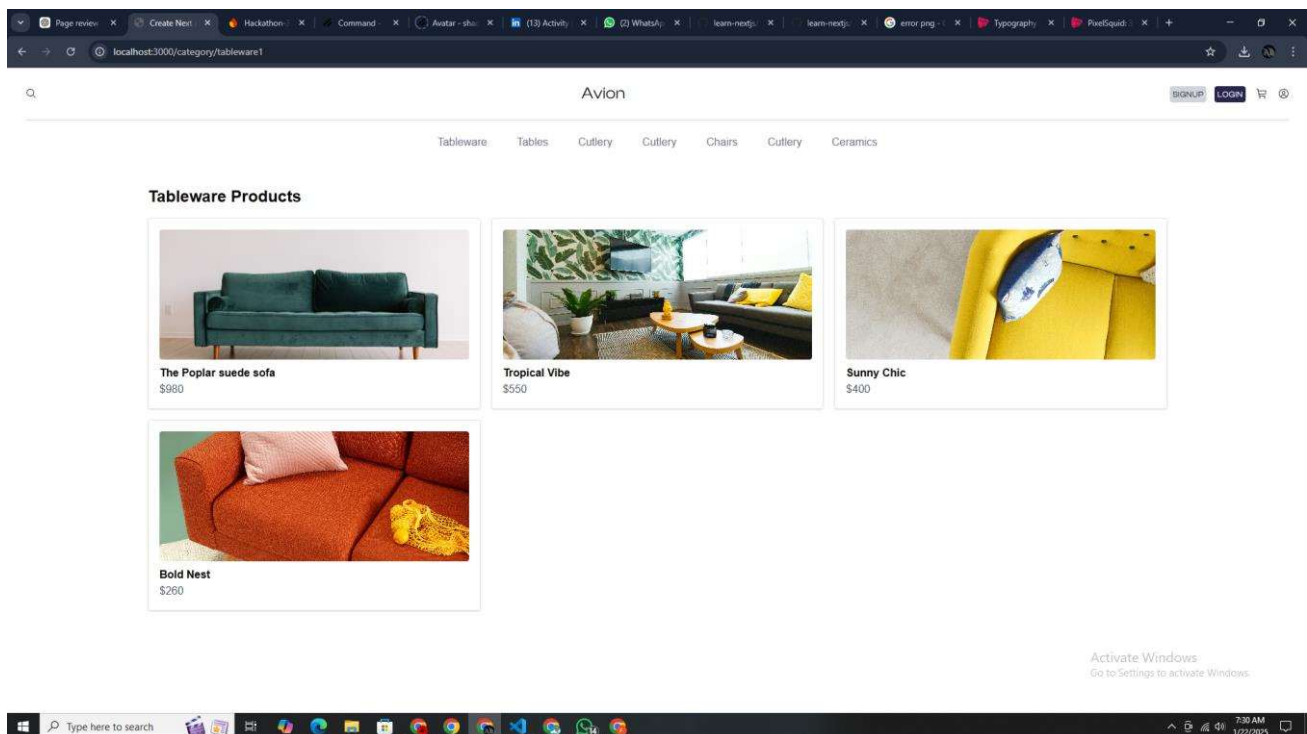
These pages include:

- Product description.
- Pricing details.
- Product name.
- Features.



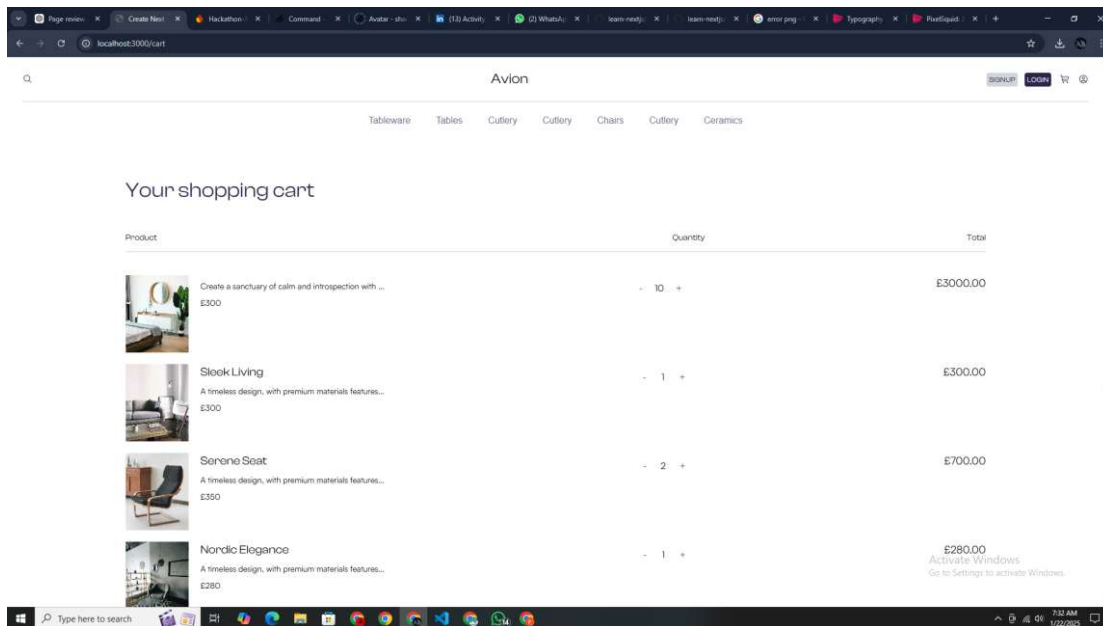
Category Component:

- Implemented a category display to list product categories like "chairs" or "tables".
- Clicking on a category dynamically filters products within that category.



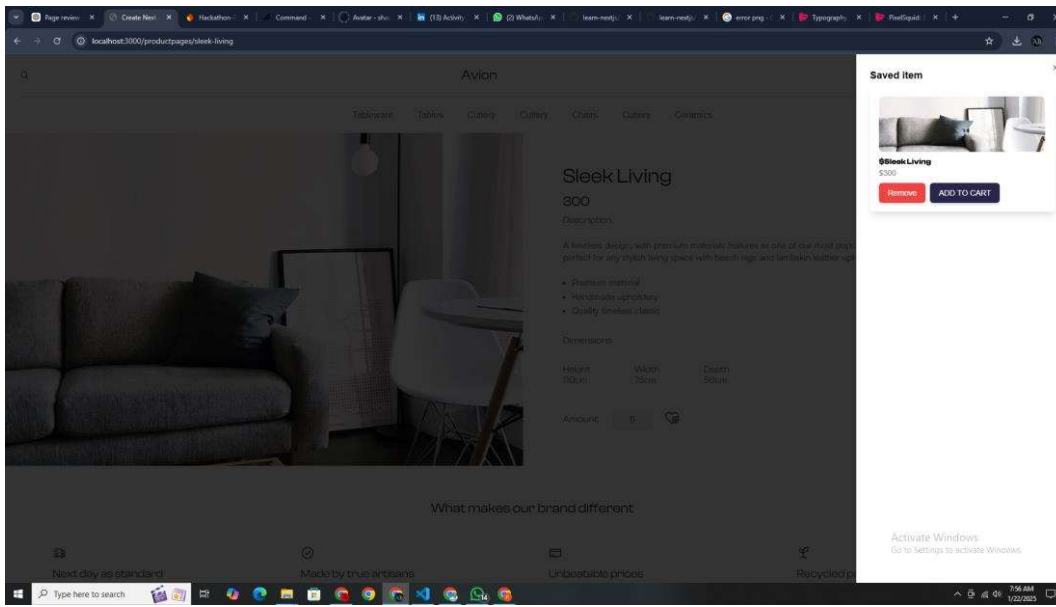
Cart Component:

- Built a functional cart that:
 - Displays added items, their quantities, and the total price.
- Used React state to track and manage cart items.



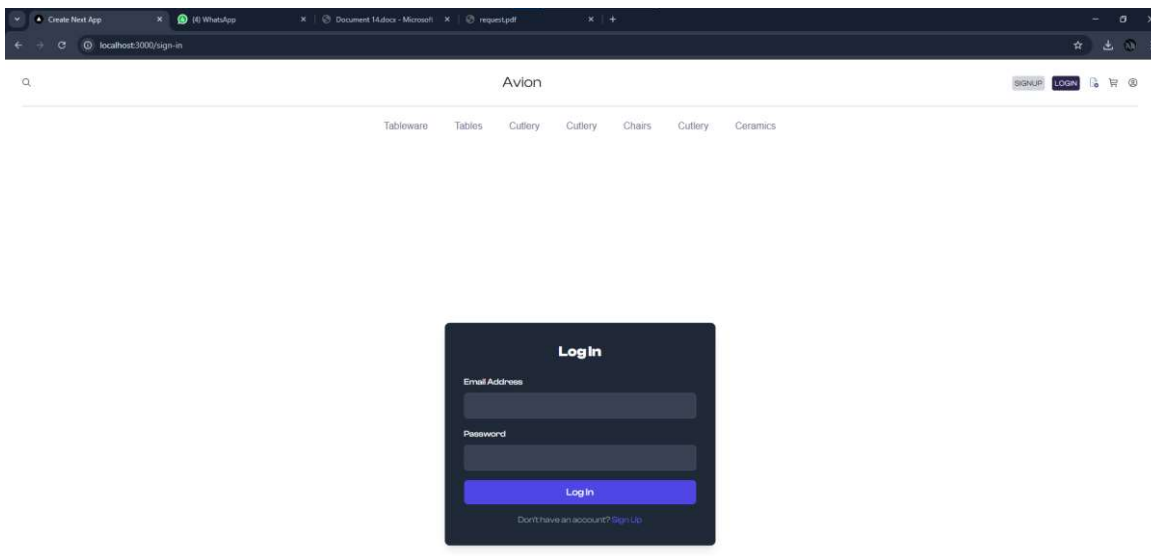
Wishlist Component:

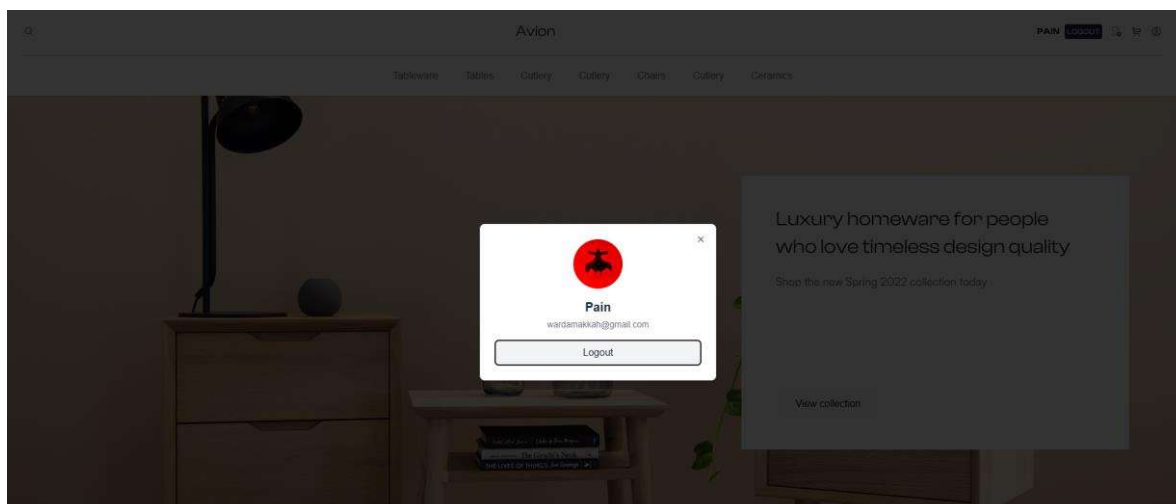
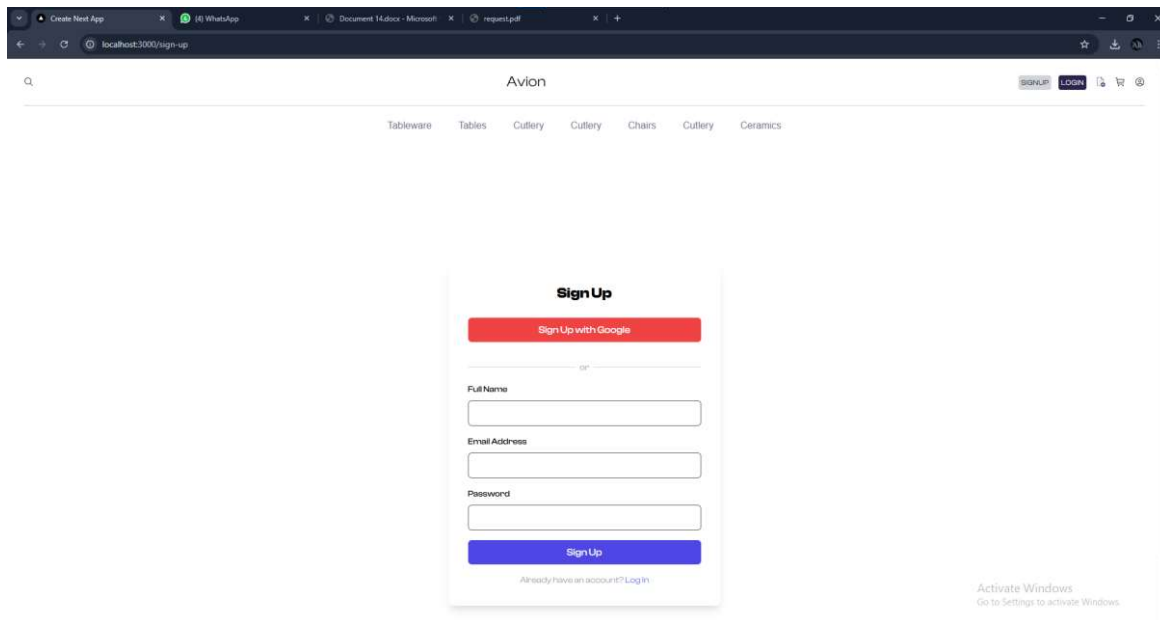
- Added a Wishlist feature where users can save their favourite products for later.
- Stored Wishlist data in local storage to ensure persistence across page reloads.



User Authentication Features:

- Developed a **Sign-Up Page** and a **Login Page** for user registration and login functionality using firebase data base.
- User can sign in through their **google** account.
- Created a **Profile Page** to display user-specific details.





Checkout Flow Component:

I added a checkout flow component to the website to make the purchasing process more efficient.

The component gathers essential user information, including:

- First name
- Last name
- Address
- Cart items

All user data is securely stored on Firebase to ensure reliability and scalability.

I also implemented an order confirmation page that:



- Retrieves the stored user data from Firebase.
- Shows a personalized thank-you message with the user's name.
- Displays the user's entered details and a summary of the ordered products.

This flow streamlines the checkout experience and provides a personal touch to improve the overall user experience.

Check out page:

Q

Avion

PAIN [Logout](#)  

[Tableware](#) [Tables](#) [Cutlery](#) [Cutlery](#) [Chairs](#) [Cutlery](#) [Ceramics](#)

Checkout

Shipping Address

First Name

Last Name

ahmod

saeed

Address

xyz-123-nazimabad

City

Postal Code

karachi

314920

Payment Method

☐ Credit Card (Stripe)

☒ Cash on Delivery

Total

\$2600.00



\$

Confirm Order


Order confirmation page:

Q

Avion

PAIN [Logout](#)  

[Tableware](#) [Tables](#) [Cutlery](#) [Cutlery](#) [Chairs](#) [Cutlery](#) [Ceramics](#)

 Order Confirmed

Thank you for your purchase,ahmed saeed!


Payment Method

cod

Shipping Address

xyz-123-nazimabad,karachi,314920

Order Items

 Sleek Living
Quantity:2

\$600.00

Total Price: \$600.00

Go to Home

Search component code snippet

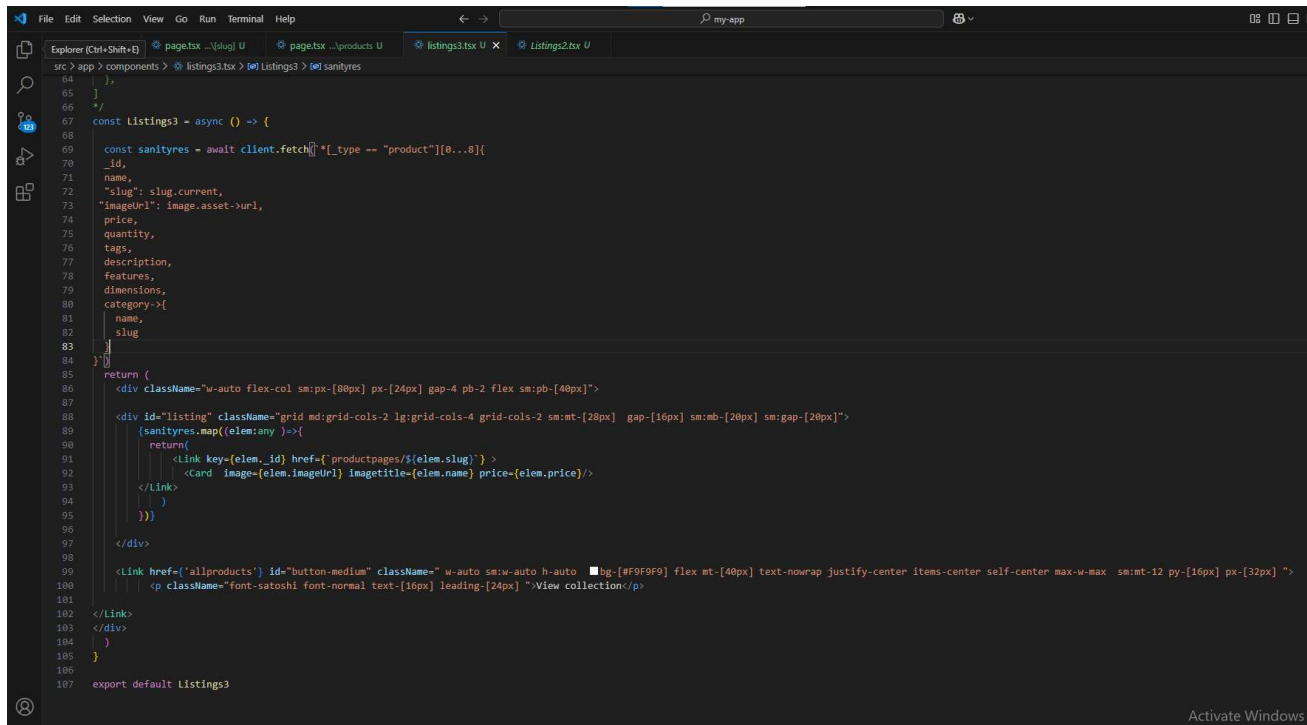
```
File Edit Selection View Go Run Terminal Help

src > app > components > NavBar.tsx > NavBar > wishlist.map() callback
17 SheetHeader,
18 SheetTitle,
19 SheetTrigger,
20 } from "@components/ui/sheet";
21 import {
22   Dialog,
23   DialogContent,
24   DialogDescription,
25   DialogHeader,
26   DialogTitle,
27   DialogTrigger,
28 } from "@components/ui/dialog";
29 import {
30   Popover,
31   PopoverContent,
32   PopoverTrigger,
33 } from "@components/ui/popover";
34 import { auth } from "../firebase/config";
35
36 const NavBar = ({ product }: { product: any[] }) => {
37   const [imageUrl, setImageUrl] = useState<string>("/placeholder.svg");
38   const [isAuthenticated, setIsAuthenticated] = useState<boolean>(false);
39
40   useEffect(() => {
41     const unsubscribe = onAuthStateChanged(auth, (user) => {
42       if (user) {
43         setImageUrl(user.photoURL || user1.src);
44         setIsAuthenticated(true);
45       } else {
46         setImageUrl(user1.src);
47         setIsAuthenticated(false);
48       }
49     });
50
51     return () => unsubscribe();
52   }, []);
53   const { wishlist, RemoveFromWish, AddToCart } = useAppContext();
54   const [searchTerm, setSearchTerm] = useState("");
55   const [filteredProducts, setFilteredProducts] = useState<any[]>([]);
56   const handleSearch = (e: React.ChangeEvent<HTMLInputElement>) => {
57     const query = e.target.value.toLowerCase();
58     setSearchTerm(query);
59
60     if (query.trim() !== "") {
61       const filtered = product.filter((item) =>
62         item.name.toLowerCase().includes(query)
63       );
64       setFilteredProducts(filtered);
65     } else {
66       setFilteredProducts([]);
67     }
68   };
69
70   const [user] = useAuthState(auth);
71
72   return (
73     <div className="h-auto w-auto flex text-nowrap flex-col gap-[20px] px-[28px] pt-[20px] pb-[20px]">
74       <div className="flex justify-between items-center">
75         <Popover>
```

```
File Edit Selection View Go Run Terminal Help
my-app

NavBartsx U X
src > app > components > NavBartsx > [0] Navbar > [0] wishlist.map() callback
36 const Navbar = ({ product }: { product: any[] }) => {
72   return (
73     <div className="h-auto w-auto flex text-nowrap flex-col gap-[20px] px-[28px] pt-[20px] pb-[20px]">
74       <div className="flex justify-between items-center">
75         <Popover>
76           <PopoverTrigger>
77             <Image
78               src={search}
79               alt=""
80               className="sm:w-[16px] sm:h-[16px] sm:block"
81             />
82           </PopoverTrigger>
83           <PopoverContent>
84             <div className="mt-4">
85               <input
86                 type="text"
87                 value={searchTerm}
88                 onChange={handleSearch}
89                 placeholder="Search products..."
90                 className="border rounded px-4 py-2 w-full"
91               />
92             </div>
93             {searchTerm.trim() !== "" && (
94               <div className="mt-4">
95                 {filteredProducts.length > 0 ? (
96                   <ul className="grid grid-cols-1 gap-4">
97                     {filteredProducts.map((item) => (
98                       <Link key={item._id} href={`productpages/${item.slug}`}>
99                         <li
100                           key={item._id}
101                           className="border flex gap-2 p-4 rounded shadow"
102                         >
103                           <Image
104                             src={item.imageUrl}
105                             alt={item.name}
106                             width={900}
107                             height={800}
108                             className="h-[50px] w-[50px] object-fill bg-cover bg-center"
109                           />
110                           <div className="flex flex-col justify-between">
111                             <h2 className="font-bold">{item.name}</h2>
112                             <p className="text-gray-600 font-bold">
113                               ${item.price}
114                             </p>
115                           </div>
116                         </li>
117                       </Link>
118                     )})
119                   </ul>
120                 ) : (
121                   <p>No products found.</p>
122                 )
123               </div>
124             )
125           </PopoverContent>
126         </Popover>
127       </div>
128       <h1 className="font-clash text-[24px]">
129         <Link href="/">Avion</Link>
130       </h1>
131     </div>
132   )
133 }
```

Product list component code snippet



```
64 },
65 }
66 //
67 const Listings3 = async () => {
68
69   const sanityres = await client.fetch`*[_type == "product"]{0...8}{
70     id,
71     name,
72     "slug": slug.current,
73     "imageurl": image.asset->url,
74     price,
75     quantity,
76     tags,
77     description,
78     features,
79     dimensions,
80     category->{
81       name,
82       slug
83     }
84   }
85 }
86
87 return (
88   <div className="w-auto flex-col sm:px-[80px] px-[24px] gap-4 pb-2 flex sm:pb-[40px]">
89     <div id="listing" className="grid md:grid-cols-2 lg:grid-cols-4 grid-cols-2 sm:mt-[28px] gap-[16px] sm:mb-[20px] sm:gap-[20px]">
90       {sanityres.map((elem:any) => {
91         return (
92           <Link key={elem._id} href={`productpages/${elem.slug}`} >
93             <Card image={elem.imageurl} imagetitle={elem.name} price={elem.price}/>
94           </Link>
95         )
96       })}
97     </div>
98     <Link href="/allproducts" id="button-medium" className="w-auto sm:w-auto h-auto bg-[#F9F9F9] flex mt-[40px] text-nowrap justify-center items-center self-center max-w-max sm:mt-12 py-[16px] px-[32px]" >
99       <p className="font-satoshi font-normal text-[16px] leading-[24px]" >View collection</p>
100     </Link>
101   </div>
102 )
103
104 }
105
106 export default Listings3
107
```

(Continue to next page)

Product details component code snippet

```
File Edit Selection View Go Run Terminal Help
NavBar.tsx U page.tsx U X
src > app > productpages > [slug] > page.tsx > [0] Page
7 import React from 'react';
8
9 import ProductDetails2 from '@app/components/ProductDetails2';
10
11 interface Props {
12   params: {
13     slug: string;
14   };
15 }
16
17 const Page = async ({ params }: Props) => {
18   // Sanity data fetch
19   const sanityres = await client.fetch(
20     `*[ _type == "product" && slug.current == $slug ]{
21       _id,
22       name,
23       "slug": slug.current,
24       "imageUrl": image.asset->url,
25       price,
26       quantity,
27       tags,
28       description,
29       features,
30       dimensions {
31         depth,
32         width,
33         height
34       },
35       category->{
36         name,
37         slug
38       }
39     }`,
40     { slug: params.slug }
41   );
42
43   if (!sanityres || sanityres.length === 0) {
44     return <div>Product not found.</div>;
45   }
46
47   return (
48     <>
49       <div>
50         { /* Pass fetched data to ProductDetails */ }
51         <ProductDetails2 product={sanityres[0]} />
52         <Features2 />
53         <Email2 />
54         <Footer2 />
55       </div>
56     </>
57   );
58
59 };
60
61 export default Page;
62
```

```
File Edit Selection View Go Run Terminal Help
src > app > components > ProductDetails2.tsx > ProductDetails2
1 'use client';
2
3 import React, { useState } from 'react';
4 import Image from 'next/image';
5 import { useAppContext } from '@context';
6 import wish from "../assets/add-to-wishlist-icon.png"
7 import { toast } from "sonner"
8 import { Divide } from 'lucide-react';
9
10
11 const ProductDetails2 = ({ product }: { product: any }) => {
12   const { AddToCart, AddToWish } = useAppContext();
13   const [first, setfirst] = useState(0)
14
15   return (
16     <div className="w-auto h-auto">
17       <div>
18         {/* Main Product Section */}
19         <div className="flex flex-col md:flex-row gap-8 items-center">
20           <div className="w-full overflow-hidden max-h-[600px] flex items-center object-cover md:w-1/2">
21             <Image
22               src={product.imageUrl}
23               alt={product.name}
24               width={900}
25               height={800}
26               className="mix-h-[600px] w-[2000px] object-fill bg-cover bg-center"
27             />
28           </div>
29           <div className="w-full md:w-1/2 px-4 sm:px-10 py-6 flex flex-col justify-center">
30             <div>
31               <p className="text-[36px] leading-[44.28px] font-clash font-normal">
32                 {product.name}
33               </p>
34               <p className="py-2 text-[24px] leading-[32.4px] font-clash font-normal">
35                 {product.price}
36               </p>
37             </div>
38             <div className="text-[#505977] text-[16px] leading-[21.6px]">
39               <h1 className="font-clash font-normal leading-[19.68px]">
40                 Description
41               </h1>
42               <div className="my-4 sm:my-6">
43                 <p className="font-satoshi font-normal">{product.description}</p>
44               </div>
45               <div className="m1-4 font-satoshi font-normal">
46                 <ul className="list-disc space-y-1">
47                   {product.features.map((feature: string, index: number) => {
48                     <li key={index}>{feature}</li>
49                   })}
50                 </ul>
51               </div>
52             </div>
53             <div>
54               <div className="my-8">
55                 <h1 className="font-clash font-normal">Dimensions</h1>
56               </div>
57               <div className="flex gap-12 sm:gap-20">
58                 <div>
59                   <h1 className="font-clash font-normal leading-[17.22px]">
60                     Height
61                   </h1>
62                 </div>
63               </div>
64             </div>
65           </div>
66         </div>
67       </div>
68     </div>
69   )
70 }
```