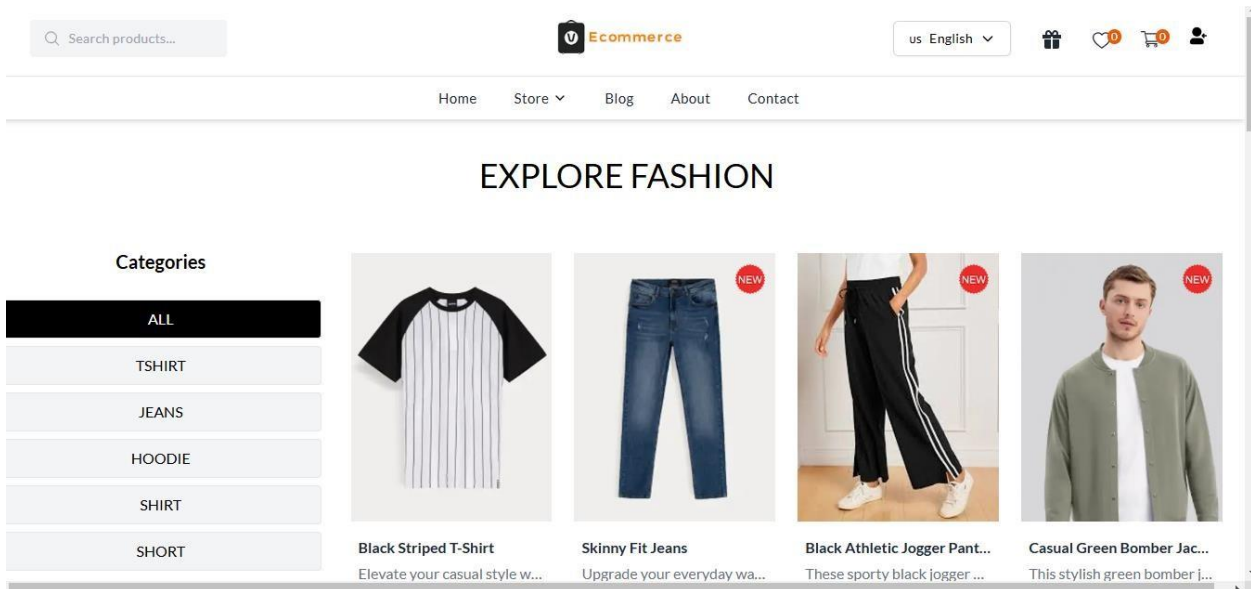


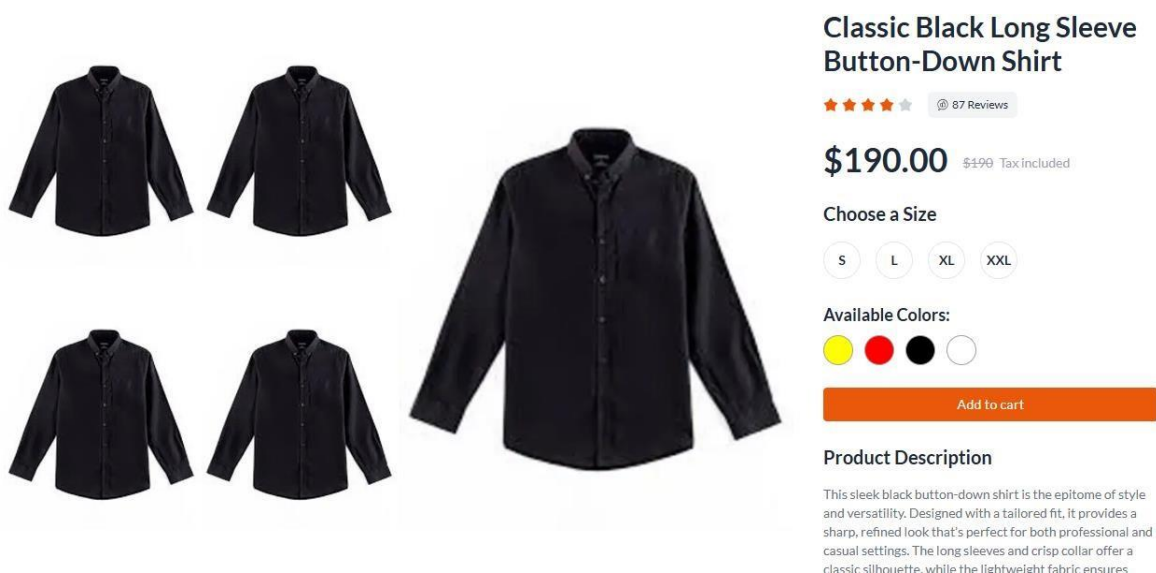
Dynamic Frontend Components

1. Functional Deliverables:

- The product listing page with dynamic data:



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





- Working category filter:


EXPLORE FASHION


Categories

ALL
TSHIRT
JEANS
HOODIE
SHIRT
SHORT


NEW


NEW


NEW


NEW

Skinny Fit Jeans
Upgrade your everyday wa...
\$15 \$240.00
★★★★★
Add to cart

Black Athletic Jogger Pant...
These sporty black jogger ...
\$15 \$180.00
★★★★★
Add to cart

Gray Slim-Fit Jogger Pants
These comfortable gray jog...
\$15 \$170.00
★★★★★
Add to cart

Beige Slim-Fit Jogger Pants
These beige jogger pants c...
\$15 \$269.00
★★★★★
Add to cart

- Working Search bar:

jeans

Ecommerce


us English


Home
Store
Blog
About
Contact

EXPLORE FASHION

Categories

ALL
TSHIRT
JEANS
HOODIE
SHIRT
SHORT


NEW



Skinny Fit Jeans
Upgrade your everyday wa...

Classic Black Straight-Leg ...
These classic black jeans of...

- Working Size filter and Price range slider:

SHORT

Sizes

ALL

MEDIUM

LARGE

SMALL

EXTRA EXTRA LARGE

EXTRA LARGE

Filter by Price

\$112

\$208


Black Striped T-Shirt

Elevate your casual style w...

120.00 \$120.00

★★★★☆

Add to cart




Classic Polo Shirt

Classic Polo Shirt Upgrade...

180.00 \$180.00

★★★★☆

Add to cart




Classic White Pullover Ho...

This white pullover hoodie ...

135.00 \$150.00

★★★★☆

Add to cart




Classic Black Straight-Leg ...

These classic black jeans of...

142.80 \$170.00

★★★★☆

Add to cart




Classic Black Long Sleeve ...

This sleek black button-do...

190.00 \$190.00

★★★★☆

Add to cart



- Checkout Page

Checkout

01

Personal Details

First name

Last name

Email address

Phone number

02

Shopping Address

Street address

City

State

Zip Code

03

Payment method

☐

VISA

AMERICAN EXPRESS

MASTERCARD

☐

PayPal

Card number

EXP.

CVV

Pay later

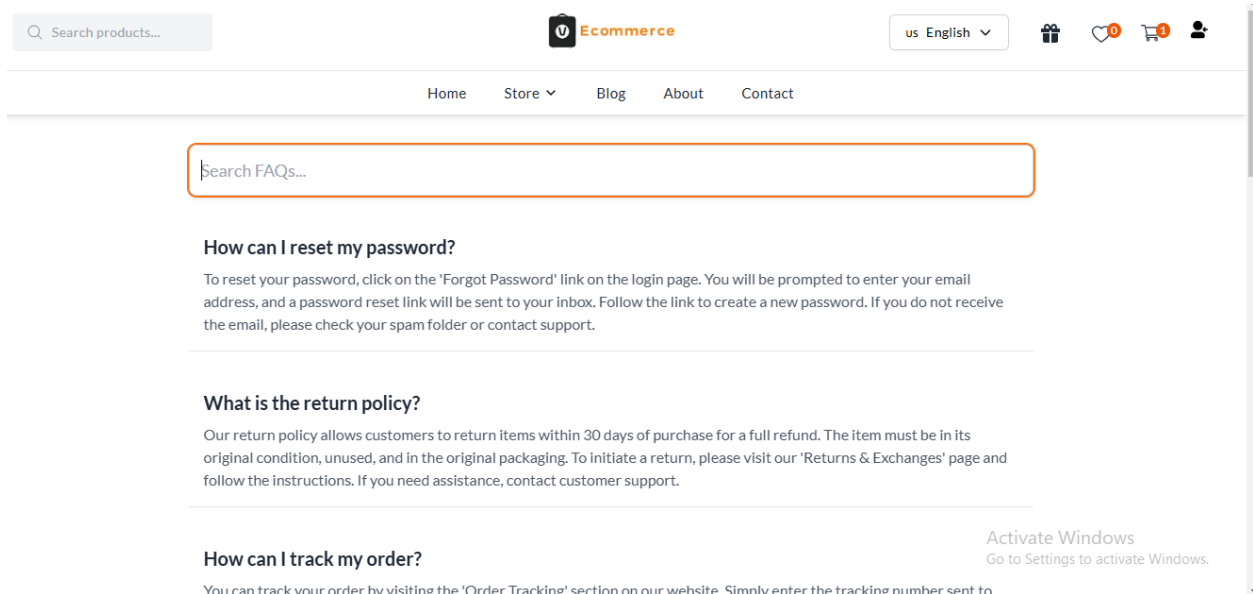
Pay now

Activate Windows

Settings to activate Windows.

Page 3 | 17

- **FAQs page with Search Functionality**



2. Code Deliverables:

Code snippets for ProductCard

```

<div>
  <div className="font-sans p-4 mx-auto lg:max-w-6xl md:max-w-3xl">
    <div className="grid grid-cols-2 sm:grid-cols-3 lg:grid-cols-4 gap-4 sm:gap-6">
      {filteredProducts.length > 0 ? (
        filteredProducts.map((product) => (
          <div className="bg-white flex flex-col overflow-hidden cursor-pointer hover:shadow-md transition-all">
            <div className="w-full">
              {product.image ? (
                <Link
                  href={` /products/${product._id} `}>
                <div>
                  <div>
                    {product.isNew ? (
                      <Image

```

```

        src={tag}
        alt="tag"
        width={40}
        height={40}
        className="absolute

md:m1-
[170px] m1-[80px] m-2 z-[100]"
        />
    ) : null}
</div>
<Image

src={urlFor(product.image).width(4
00).url()}

        alt={product.name}
        width={400}
        height={300}
        className="w-full object-
cover object-top aspect-[230/307] hover:scale-110 transition
duration-3"
        />
    </div>
</Link>
) : (
    <p className="w-[100%] text-center
p-2 py-48">
        No Image Available
    </p>
    )}
</div>

    <div className="p-2 flex-1 flex flex-
col">
        <div className="flex-1">
            <h5 className="mt-2 text-sm sm:text-
base font-bold text-gray-800 truncate">
                {product.name}
            </h5>
            <p className="mt-1 text-gray-500
truncate">
                {product.description}
            </p>
            <div className="flex flex-wrap
justifybetween gap-2 mt-2">
                <div className="flex gap-2">

```

```

    <h6 className="text-sm sm:text-
base font-bold text-gray-800">
    {
      product.price -
      (product.price
product.discountPercent) /
      100
    }.toFixed(2)}
  </h6>
  <h6 className="text-sm sm:text-
base text-gray-500">
    <span className="line-
through">
      ${product.price.toFixed(2)}
    </span>
  </h6>
</div>
<div className="flex items-center
gap-
0.5">
    <svg
      className="w-[14px] h-[14px]
fillorange-600"
      viewBox="0 0 14 13"
      fill="none"
xmlns="http://www.w3.org/2000/svg"
    >
      <path d="M7 0L9.4687
3.60213L13.6574 4.83688L10.9944 8.29787L11.1145 12.6631L7
11.2L2.8855 12.6631L3.00556 8.29787L0.342604 4.83688L4.5313
3.60213L7 0Z" />
    </svg>
    <svg
      className="w-[14px] h-[14px]
fillorange-600"
      viewBox="0 0 14 13"
      fill="none"
xmlns="http://www.w3.org/2000/svg"
    >
      <path d="M7 0L9.4687
3.60213L13.6574 4.83688L10.9944 8.29787L11.1145 12.6631L7
11.2L2.8855 12.6631L3.00556 8.29787L0.342604 4.83688L4.5313

```

```
3.60213L7 0Z" />
```

```

    </svg>
    <svg
      className="w-[14px] h-[14px]
      fillorange-600"
      viewBox="0 0 14 13"
      fill="none"

      xmlns="http://www.w3.org/2000/svg"
    >
      <path d="M7 0L9.4687
3.60213L13.6574 4.83688L10.9944 8.29787L11.1145 12.6631L7
11.2L2.8855 12.6631L3.00556 8.29787L0.342604 4.83688L4.5313
3.60213L7 0Z" />
    </svg>
    <svg
      className="w-[14px] h-[14px]
      fill-
      [#CED5D8]"
      viewBox="0 0 14 13"
      fill="none"

      xmlns="http://www.w3.org/2000/svg"
    >
      <path d="M7 0L9.4687
3.60213L13.6574 4.83688L10.9944 8.29787L11.1145 12.6631L7
11.2L2.8855 12.6631L3.00556 8.29787L0.342604 4.83688L4.5313
3.60213L7 0Z" />
    </svg>
    <svg
      className="w-[14px] h-[14px]
      fill-
      [#CED5D8]"
      viewBox="0 0 14 13"
      fill="none"

      xmlns="http://www.w3.org/2000/svg"
    >
      <path d="M7 0L9.4687
3.60213L13.6574 4.83688L10.9944 8.29787L11.1145 12.6631L7
11.2L2.8855 12.6631L3.00556 8.29787L0.342604 4.83688L4.5313
```

```
3.60213L7 0Z" />
```

```

    </svg>
  </div>
</div>
</div>
<div className="flex items-center
gap-2 mt-4">
  <div
    className="bg-pink-100 hover:bg-
pink200 w-12 h-9 flex items-center justify-center rounded
cursorpointer"
    title="Wishlist"
  >
    {Wishlist[product._id] ? (
      <GoHeartFill
        onClick={() =>
handleWishList(product)}
        className="text-pink-600
fillHeart"
      />
    ) : (
      <GoHeart
        onClick={() =>
handleWishList(product)}
        className="text-pink-600
emptyHeart"
      />
    )}{" "}
  </div>
  <button
    onClick={() => {
      addToCart(product);
      toast.success("Product added to
cart!");
    }}
    type="button"
    className="text-sm px-2 min-h-
[36px] w-full bg-orange-600 hover:bg-orange-700 transition-
transform duration-300 ease-in-out transform hover:scale-105
active:scale-95
    text-white tracking-wide ml-auto
    outline-none border-none rounded"
  >
```



```

    >
    Add to cart
  </button>
</div>
</div>
</div>
))
) : (
  <div className="w-full text-center w-[70vw]
py-
10">
    No product Found
  </div>
  </div>
</div>
</div>

```

Code snippets for Search bar:

```

'use client' import { createContext, useContext, useState, ReactNode }
from "react";

// Define the context type type
SearchContextType = {  searchTerm:
string;  setSearchTerm: (term: string)
=> void;
};

// Create the context const SearchContext = createContext<SearchContextType |
undefined>(undefined);
// Provider component export function SearchProvider({ children }: {
children: ReactNode }) {  const [searchTerm, setSearchTerm] =
useState("");

  return (
    <SearchContext.Provider value={{ searchTerm, setSearchTerm }}>
{children}
    </SearchContext.Provider>
  );
}

```

```
// Hook to use the context export function useSearch() { const
context = useContext(SearchContext); if (!context) { throw new
Error("useSearch must be used within a SearchProvider");
} return
context;
}
```

Scripts for dynamic routing:

```
▪ "use client";
▪ import GetProductData from "@sanity/sanity.query";
▪ import Image from "next/image";
▪ import { urlFor } from "../../utils/sanity";
▪ import { useEffect, useState } from "react";
▪ import { typeOfData } from "@app/utils/types";
▪ import { useCart } from "@context/CartContext";
▪ import ReviewSection from "@app/components/ReviewSection";
▪ import ShareButtons from "@app/ShareButtons/page";
▪
▪ export default function ProductDetails({
▪   params,
▪ }: {
▪   params: { _id: string };
▪ },
▪ ) {
▪   const [data, setData] = useState<typeOfData[]>([]);
▪   const [loading, setLoading] = useState(true);
▪   const { addToCart } = useCart();
▪
▪   useEffect(() => {
▪     const fetchedProducts = async () => {
▪       try {
▪         const products = await GetProductData();
▪         setData(products);
▪       } catch (err) {
▪         console.error("Error fetching data: ", err);
▪       } finally {
▪         setLoading(false);
▪       }
▪     }
▪   })
▪ }
```

```

    };
    fetchedProducts();
  }, []);

  const details = data.find((e) => e._id === params._id);

  if (loading) {
    return (
      <div className="w-[100vw] h-[100vh] flex justify-center itemscenter">
        <div className="loader"></div>
      </div>
    );
  }

  if (!details) {
    return <p>No details found for the product.</p>;
  }

  const discountedPrice = (
    details.price -
    (details.price * details.discountPercent) / 100
  ).toFixed(2);
  return (
    <div className="md:p-10">
      <div className="grid items-start grid-cols-1 lg:grid-cols-3">

```

```

    <div className="col-span-2 grid grid-cols-2 lg:sticky top-0
gap-
0.5">
    <div className="columns-2 gap-0.5 space-y-0.5">
    <div className="overflow-hidden">
    <Image
    src={urlFor(details.image).width(400).url()}
    width={450}
    height={450}
    alt="Product"
    className="w-full aspect-[253/337] object-cover object-
top shadow-md hover:scale-[1.05] transition-all duration-300"
    />
    </div>
    <div className="overflow-hidden">
    <Image
    src={urlFor(details.image).width(400).url()}
    width={450}
    height={450}
    alt="Product"
    className="w-full aspect-[253/337] object-cover object-
top shadow-md hover:scale-[1.05] transition-all duration-300"
    />
    </div>
    <div className="overflow-hidden">
    <Image
    src={urlFor(details.image).width(400).url()}
    width={450}
    height={450}
    alt="Product"
    className="w-full aspect-[253/337] object-cover object-
top shadow-md hover:scale-[1.05] transition-all duration-300"
    />
    </div>
    <div className="overflow-hidden">
    <Image
    src={urlFor(details.image).width(400).url()}
    width={450}
    height={450}
    alt="Product"
    className="w-full aspect-[253/337] object-cover object-
top shadow-md hover:scale-[1.05] transition-all duration-300"
    />
    </div>
    </div>
    </div>

```

```

    <div className="overflow-hidden">
      <Image
        src={urlFor(details.image).width(400).url()}
        width={450}
        height={450}
        alt="Product"
        className="w-full aspect-[3/4] object-cover object-top
shadow-md hover:scale-[1.05] transition-all duration-300"
      />
    </div>
  </div>

  <div className="py-6 px-8 max-lg:max-w-2xl">
    <div>
      <h2 className="text-3xl font-bold text-gray-
800">{details.name}</h2>
    </div>

    <div className="flex items-center space-x-1 mt-4">
      <svg
        className="w-4 h-4 fill-orange-600"
        viewBox="0 0 14 13"
        fill="none"
        xmlns="http://www.w3.org/2000/svg"
      >
        <path d="M7 0L9.4687 3.60213L13.6574 4.83688L10.9944
8.29787L11.1145 12.6631L7 11.2L2.8855 12.6631L3.00556 8.29787L0.342604
4.83688L4.5313 3.60213L7 0Z" />
      </svg>
      <svg
        className="w-4 h-4 fill-orange-600"
        viewBox="0 0 14 13"
        fill="none"
        xmlns="http://www.w3.org/2000/svg"
      >
        <path d="M7 0L9.4687 3.60213L13.6574 4.83688L10.9944
8.29787L11.1145 12.6631L7 11.2L2.8855 12.6631L3.00556 8.29787L0.342604
4.83688L4.5313 3.60213L7 0Z" />
      </svg>
      <svg
        className="w-4 h-4 fill-orange-600"
        viewBox="0 0 14 13"
        fill="none"
        xmlns="http://www.w3.org/2000/svg"

```

```

    >
    <path d="M7 0L9.4687 3.60213L13.6574 4.83688L10.9944
8.29787L11.1145 12.6631L7 11.2L2.8855 12.6631L3.00556 8.29787L0.342604
4.83688L4.5313 3.60213L7 0Z" />
  </svg>
  <svg
    className="w-4 h-4 fill-orange-600"
    viewBox="0 0 14 13"
    fill="none"
    xmlns="http://www.w3.org/2000/svg"
  >
    <path d="M7 0L9.4687 3.60213L13.6574 4.83688L10.9944
8.29787L11.1145 12.6631L7 11.2L2.8855 12.6631L3.00556 8.29787L0.342604
4.83688L4.5313 3.60213L7 0Z" />
  </svg>
  <svg
    className="w-4 h-4 fill-[#CED5D8]"
    viewBox="0 0 14 13"
    fill="none"
    xmlns="http://www.w3.org/2000/svg"
  >
    <path d="M7 0L9.4687 3.60213L13.6574 4.83688L10.9944
8.29787L11.1145 12.6631L7 11.2L2.8855 12.6631L3.00556 8.29787L0.342604
4.83688L4.5313 3.60213L7 0Z" />
  </svg>

  <button
    type="button"
    className="px-2.5 py-1.5 bg-gray-100 text-xs text-gray-
800 rounded-md flex items-center !ml-4"
  >
    <svg
      xmlns="http://www.w3.org/2000/svg"
      className="w-3 mr-1"
      fill="currentColor"
      viewBox="0 0 32 32"
    >
      <path
        d="M14.236 21.954h-3.6c-.91 0-1.65-.74-1.65-1.65v-
7.201c0-.91-.74-1.65-1.65-1.65h3.6a.75.75 0 0 1 .75.75v9.001a.75.75 0 0
1-.
.75.75zm-3.6-9.001a.15.15 0 0 0-.15.15v7.2a.15.15 0 0 0
.15.15h2.85v7.501z"
        data-original="#000000"
      />
    </path>
  </svg>

```

```

    />
    <path
      d="M20.52 21.954h-6.284a.75.75 0 0 1-.75-.75v-
9.001c0-
.257.132-.495.348-.633.017-.011 1.717-1.118 2.037-3.25.18-1.184 1.118-
2.089 2.28-2.201a2.557 2.557 0 0 1 2.17.868c.489.56.71 1.305.609
2.042a9.468 9.468 0 0 1-.678 2.424h.943a2.56 2.56 0 0 1
1.918.862c.483.547.708 1.279.617 2.006l-.675 5.401a2.565 2.565 0 0 1-
2.535 2.232zm-5.534-1.5h5.533a1.06 1.06 0 0 0 1.048-.922l.675-
5.397a1.046 1.046
0 0 0-1.047-1.182h-2.16a.751.751 0 0 1-.648-1.13 8.147 8.147 0 0 0 1.057-
3
1.059 1.059 0 0 0-.254-.852 1.057 1.057 0 0 0-.795-.365c-.577.052-
.964.435-1.04.938-.326 2.163-1.71 3.507-2.369 4.036v7.874z"
      data-original="#000000"
    />
    <path
      d="M4 31.75a.75.75 0 0 1-.612-1.184c1.014-1.428
1.643-
2.999 1.869-4.667.032-.241.055-.485.07-.719A14.701 14.701 0 0 1 1.25
15C1.25 6.867 7.867.25 16 .25S30.75 6.867 30.75 15 24.133 29.75 16
29.75a14.57 14.57 0 0 1-5.594-1.101c-2.179 2.045-4.61 2.81-6.281
3.09A.774.774 0 0 1 4 31.75zm12-30C8.694 1.75 2.75 7.694 2.75 15c0 3.52
1.375 6.845 3.872 9.362a.75.75 0 0 1 .217.55c-.01.373-.042.78-.095
1.186A11.715 11.715 0 0 1 5.58 29.83a10.387 10.387 0 0 0 3.898-
2.37l.231-
.23a.75.75 0 0 1 .84-.153A13.072 13.072 0 0 0 16 28.25c7.306 0 13.25-
5.944
13.25-13.25S23.306 1.75 16 1.75z"
      data-original="#000000"
    />
  </svg>
  87 Reviews
</button>
</div>

<div className="mt-6">
  <div className="flex items-center flex-wrap gap-4">
    <p className="text-gray-800 text-4xl font-bold">
      ${discountedPrice}
    </p>
    <p className="text-gray-400 text-sm mt-2">
      <span className="line-
through">${details.price}</span>{"

```

```

    "}"

    <span className="ml-1">Tax included</span>
  </p>
</div>
</div>

<div className="mt-6">
  <h3 className="text-xl font-bold text-gray-800">Choose a
Size</h3>
  <div className="flex flex-wrap gap-4 mt-4">
    {details.sizes.map((e: string) => (
      <div>
        <button
          type="button"
          className="w-10 h-10 border hover:border-gray-800
font-semibold text-gray-800 text-sm rounded-full flex items-center
justify-center shrink-0"
        >
          {e}
        </button>
      </div>
    ))}
  </div>
</div>

<div className="mt-6">
  <h3 className="text-lg font-semibold text-gray-800">
    Available Colors:
  </h3>
  <div className="flex gap-3 mt-2">
    {details.colors.map((color: string) => (
      <div
        key={color}
        className="w-8 h-8 rounded-full border border-gray-
400"
        style={{ backgroundColor: color.toLowerCase() }}
        title={color}
      ></div>
    ))}
  </div>
</div>

<div className="mt-6 space-y-4">

```



```

    <button
      onClick={() => {addToCart(details);
        alert("Product added to cart!");
      }}
      type="button"
      className=" transition-transform duration-300 ease-in-out
transform hover:scale-105 active:scale-95
      text-sm px-2 min-h-[36px] w-full bg-orange-600
hover:bgorange-700 text-white tracking-wide ml-auto outline-none
border-none rounded"
    >
      Add to cart
    </button>
  </div>

  <div className="mt-6">
    <div>
      <h3 className="text-xl font-bold text-gray-800">
        Product Description
      </h3>
      <p className="text-sm text-gray-500 mt-4">
        {details.description}
      </p>
      <ShareButtons
        url={details.name || ""}
        message={`Check out this product: ${details.name}`}
      />
    </div>
  </div>
</div>
<ReviewSection/>
</div>
);
}

```