Hasan Rafay : 0029297

MARKETPLACE FURNITURE ECOMMERCE WEBSITE

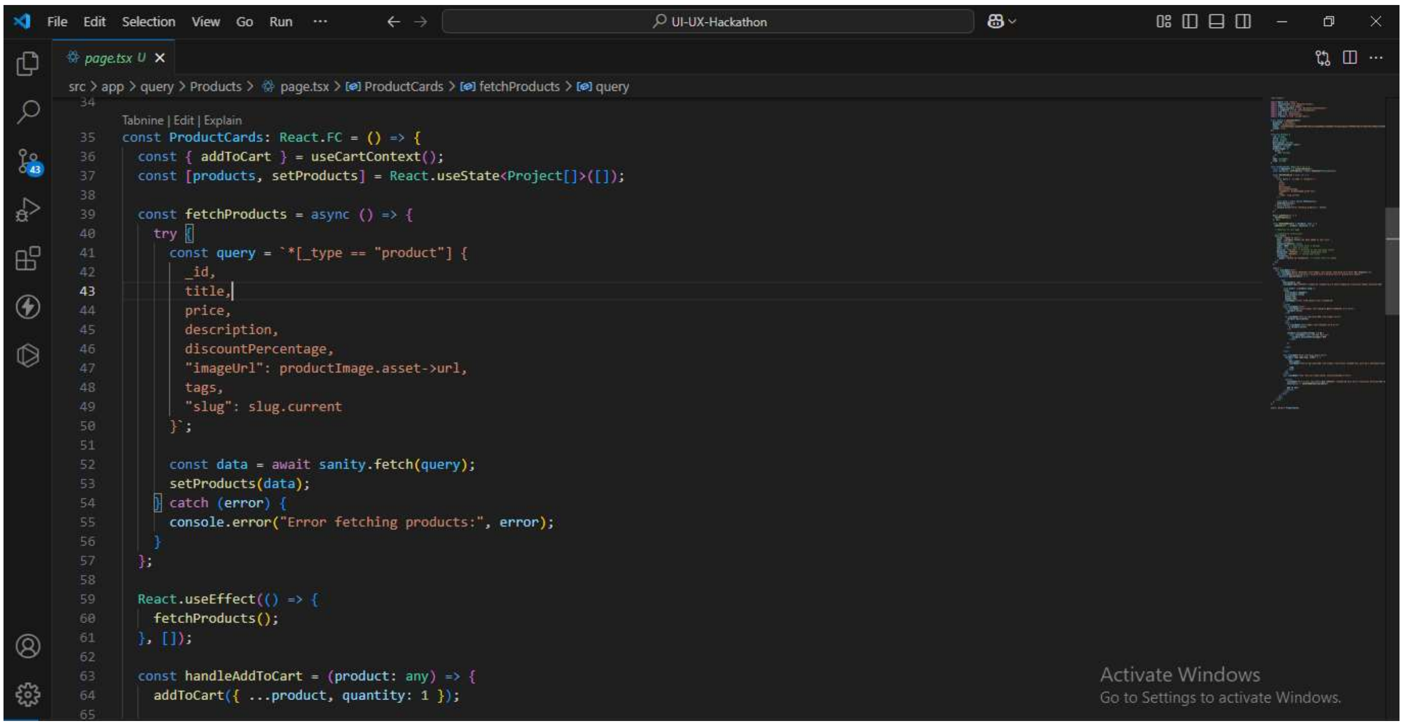
DOCUMENTATION:-

Continuing from Day 03, which focused on fetching APIs and displaying products, I accomplished the following:

- i) Created dynamic routes for product descriptions.
- ii) Implemented "Add to Cart" functionality using a Content Management System.
- iii) Displayed notifications with a toaster when the "Add to Cart" button is clicked.
- iv) Integrated features such as a search bar, product filters, sorting by newest products, and pagination.

CHALLENGES AND ISSUES

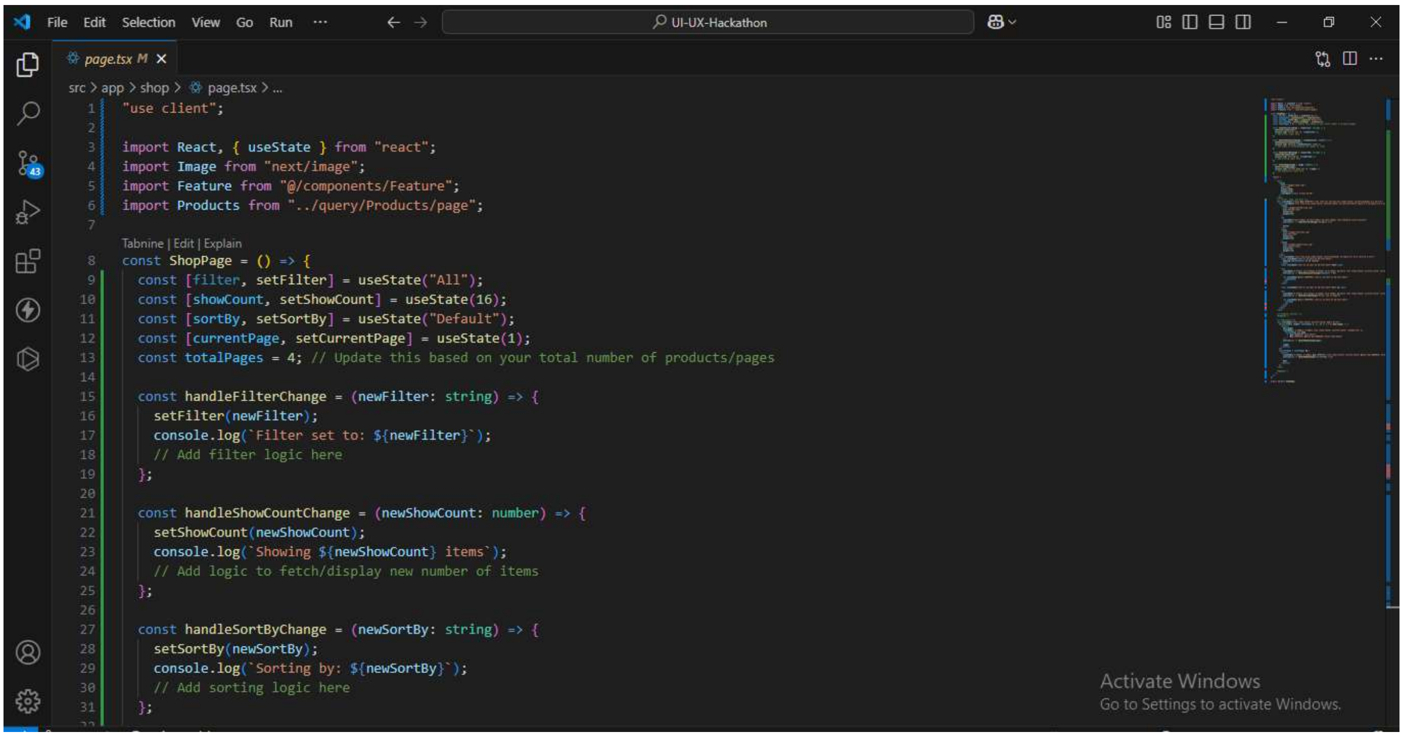
- i) Encountered difficulties in building dynamic pages.
- ii) Faced challenges in managing the Content Management System.
- iii) Dealt with problems on both the server-side and client-side components.


File Edit Selection View Go Run ···
UI-UX-Hackathon
page.tsx U
src > app > query > Products > page.tsx > ProductCards > fetchProducts > query
Tabnine | Edit | Explain
Activate Windows
Go to Settings to activate Windows.

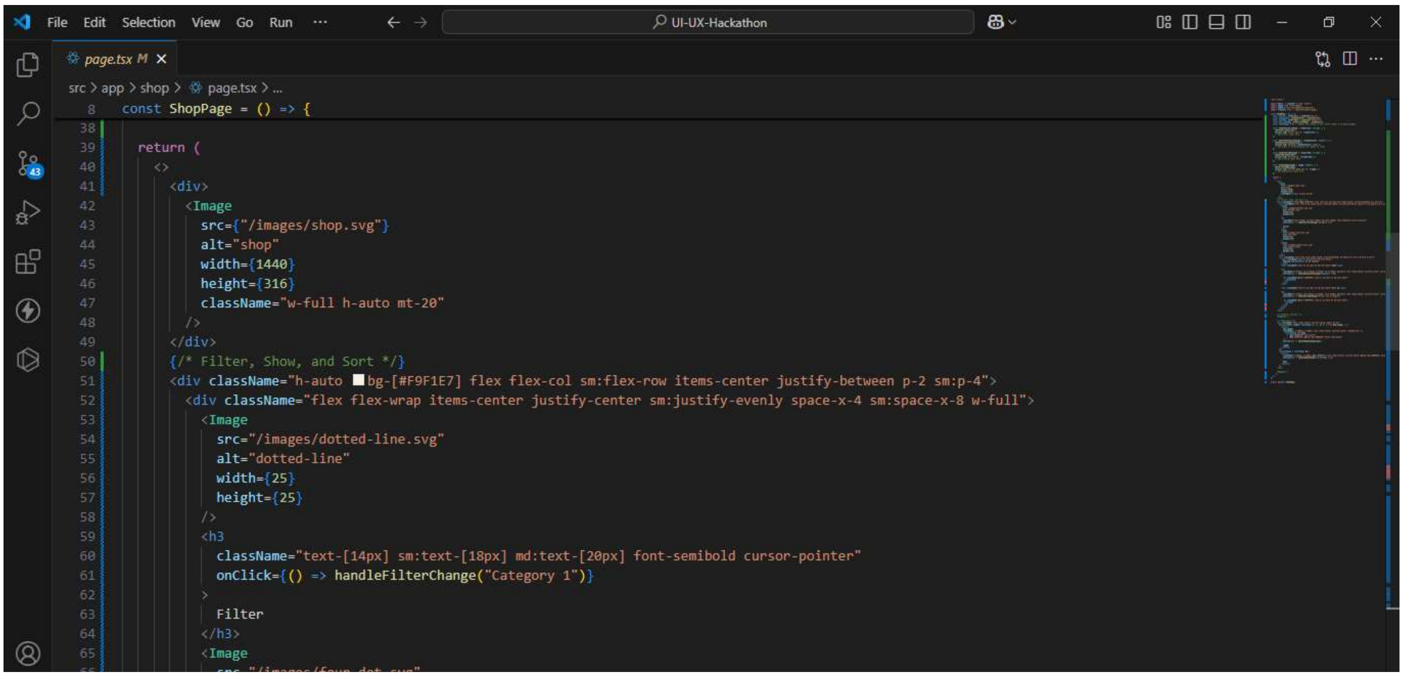
```
const ProductCards: React.FC = () => {
  const { addToCart } = useCartContext();
  const [products, setProducts] = React.useState<Project[]>([]);

  const fetchProducts = async () => {
    try {
      const query = `*[_type == "product"] {
        _id,
        title,
        price,
        description,
        discountPercentage,
        "imageUrl": productImage.asset->url,
        tags,
        "slug": slug.current
      }`;

      const data = await sanity.fetch(query);
      setProducts(data);
    } catch (error) {
      console.error("Error fetching products:", error);
    }
  };

  React.useEffect(() => {
    fetchProducts();
  }, []);

  const handleAddToCart = (product: any) => {
    addToCart({ ...product, quantity: 1 });
```

Hasan Rafay : 0029297

❖ The shop page will display all the products, allowing the client to browse and explore them.


File Edit Selection View Go Run ···
UI-UX-Hackathon
page.tsx M
src > app > shop > page.tsx > ...
Tabnine | Edit | Explain
Activate Windows
Go to Settings to activate Windows.

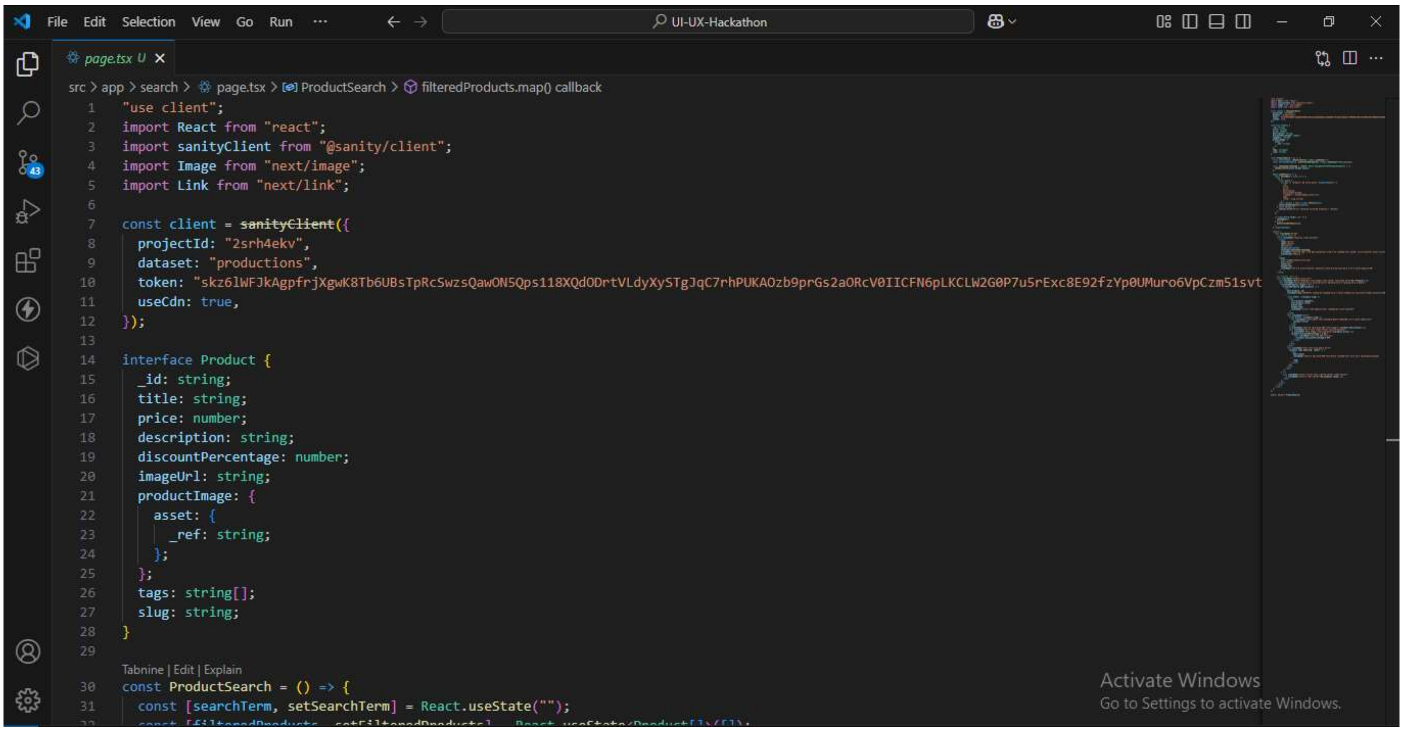
```tsx
"use client";

import React, { useState } from "react";
import Image from "next/image";
import Feature from "@/components/Feature";
import Products from "../query/Products/page";

const ShopPage = () => {
  const [filter, setFilter] = useState("All");
  const [showCount, setShowCount] = useState(16);
  const [sortBy, setSortBy] = useState("Default");
  const [currentPage, setCurrentPage] = useState(1);
  const totalPages = 4; // Update this based on your total number of products/pages

  const handleFilterChange = (newFilter: string) => {
    setFilter(newFilter);
    console.log(`Filter set to: ${newFilter}`);
    // Add filter logic here
  };

  const handleShowCountChange = (newShowCount: number) => {
    setShowCount(newShowCount);
    console.log(`Showing ${newShowCount} items`);
    // Add logic to fetch/display new number of items
  };

  const handleSortByChange = (newSortBy: string) => {
    setSortBy(newSortBy);
    console.log(`Sorting by: ${newSortBy}`);
    // Add sorting logic here
  };
```


File Edit Selection View Go Run ···
UI-UX-Hackathon
page.tsx M
src > app > shop > page.tsx > ...

```tsx
const ShopPage = () => {

  return (
    <>
      <div>
        <Image
          src={"/images/shop.svg"}
          alt="shop"
          width={1440}
          height={316}
          className="w-full h-auto mt-20"
        />
      </div>
      {/* Filter, Show, and Sort */}
      <div className="h-auto bg-[#F9F1E7] flex flex-col sm:flex-row items-center justify-between p-2 sm:p-4">
        <div className="flex flex-wrap items-center justify-center sm:justify-evenly space-x-4 sm:space-x-8 w-full">
          <Image
            src="/images/dotted-line.svg"
            alt="dotted-line"
            width={25}
            height={25}
          />
          <h3
            className="text-[14px] sm:text-[18px] md:text-[20px] font-semibold cursor-pointer"
            onClick={() => handleFilterChange("Category 1")}
          >
            Filter
          </h3>
          <Image
```

Hasan Rafay : 0029297

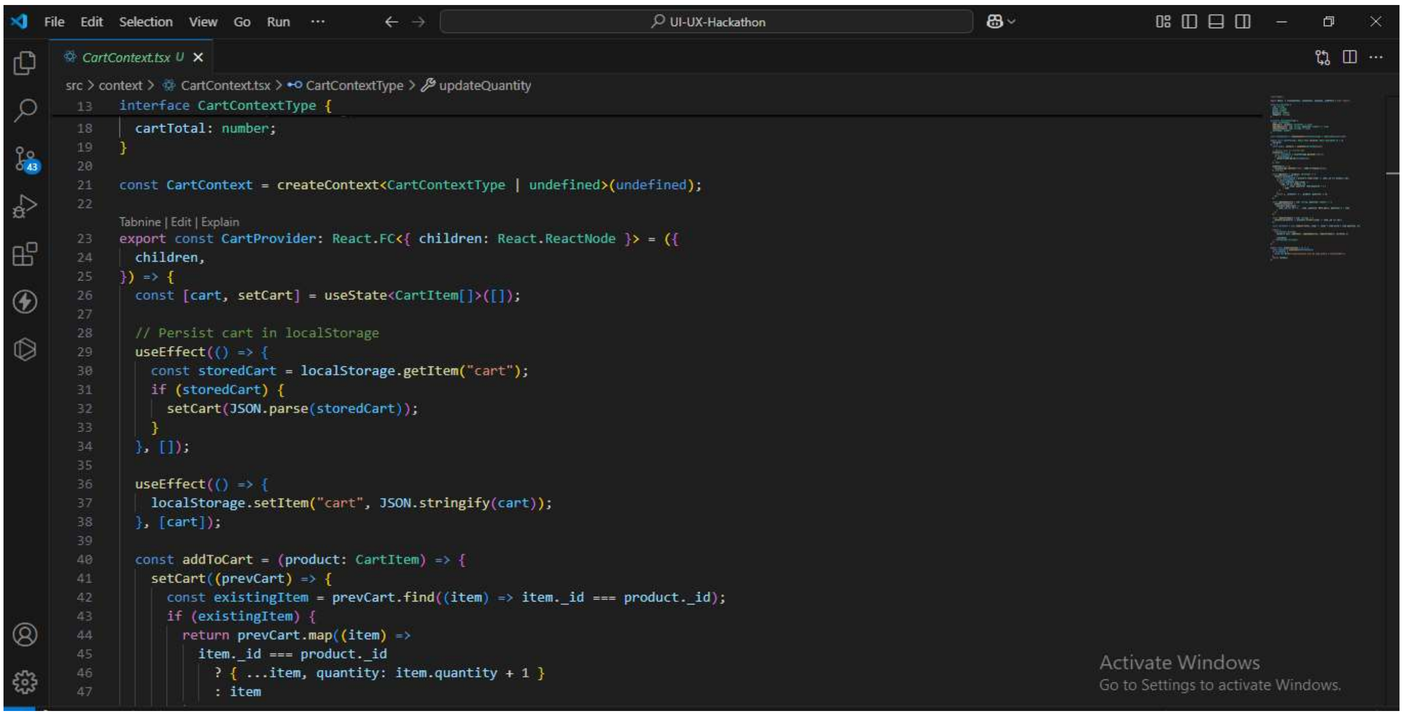
❖ The product search page will make it easier for clients to find the products they are looking for.



src > app > search > page.tsx > ProductSearch > filteredProducts.map() callback

```tsx
"use client";
import React from "react";
import sanityClient from "@sanity/client";
import Image from "next/image";
import Link from "next/link";

const client = ~~sanityClient~~({
  projectId: "2srh4ekv",
  dataset: "productions",
  token: "skz6lWFJkAgpfrjXgwK8Tb6UBsTpRcSwzsQawON5Qps118XQdODrtVLdyXySTgJqC7rhPUKAOzb9prGs2aORcV0IICFN6pLKCLW2G0P7u5rExc8E92fzYp0UMuro6VpCzm51svt
  useCdn: true,
});

interface Product {
  _id: string;
  title: string;
  price: number;
  description: string;
  discountPercentage: number;
  imageUrl: string;
  productImage: {
    asset: {
      _ref: string;
    };
  };
  tags: string[];
  slug: string;
}

Tabnine | Edit | Explain
const ProductSearch = () => {
  const [searchTerm, setSearchTerm] = React.useState("");
```

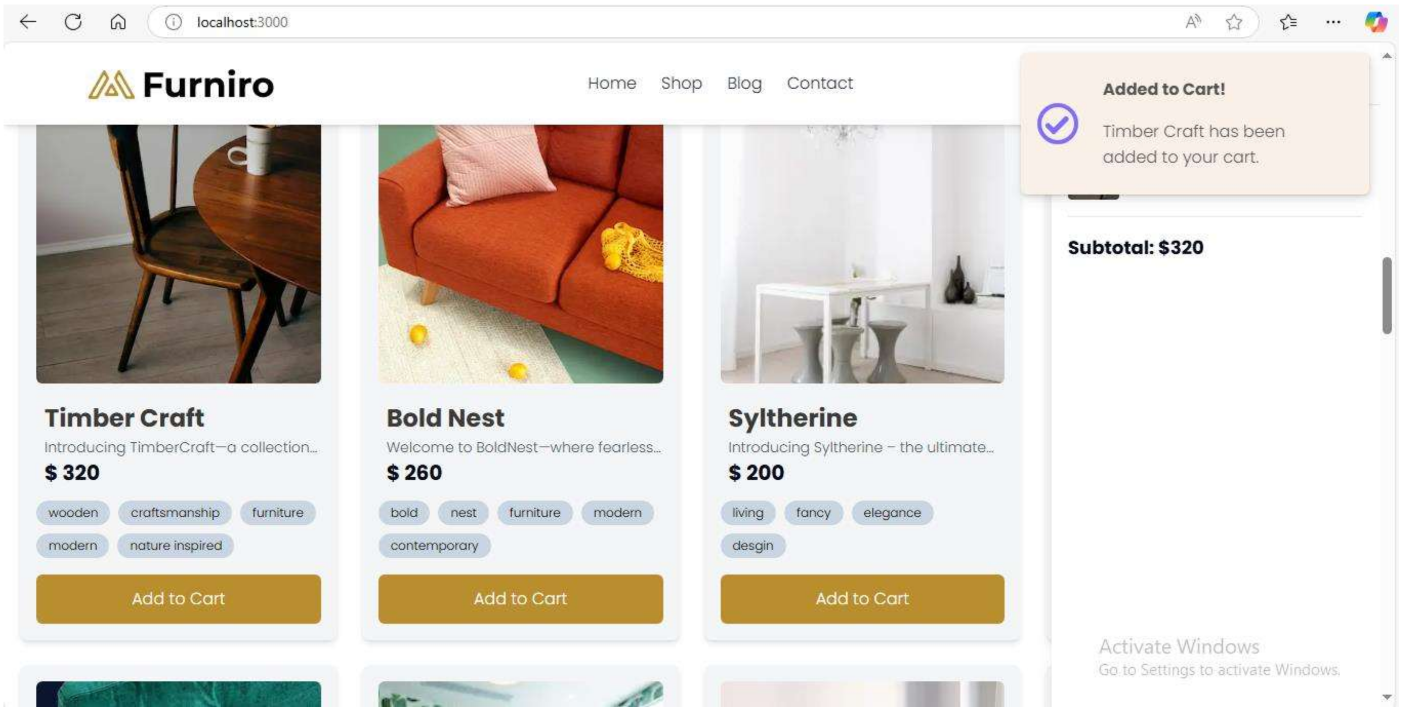
❖ The cart will store the items selected by the client for purchase.



src > context > CartContext.tsx > CartContextType > updateQuantity

```tsx
interface CartContextType {
  cartTotal: number;
}

const CartContext = createContext<CartContextType | undefined>(undefined);

Tabnine | Edit | Explain
export const CartProvider: React.FC<{ children: React.ReactNode }> = ({
  children,
}) => {
  const [cart, setCart] = useState<CartItem[]>([]);

  // Persist cart in localStorage
  useEffect(() => {
    const storedCart = localStorage.getItem("cart");
    if (storedCart) {
      setCart(JSON.parse(storedCart));
    }
  }, []);

  useEffect(() => {
    localStorage.setItem("cart", JSON.stringify(cart));
  }, [cart]);

  const addToCart = (product: CartItem) => {
    setCart((prevCart) => {
      const existingItem = prevCart.find((item) => item._id === product._id);
      if (existingItem) {
        return prevCart.map((item) =>
          item._id === product._id
            ? { ...item, quantity: item.quantity + 1 }
            : item
```

Hasan Rafay : 0029297


localhost:3000

Furniro

Home Shop Blog Contact

Added to Cart!

Timber Craft has been added to your cart.

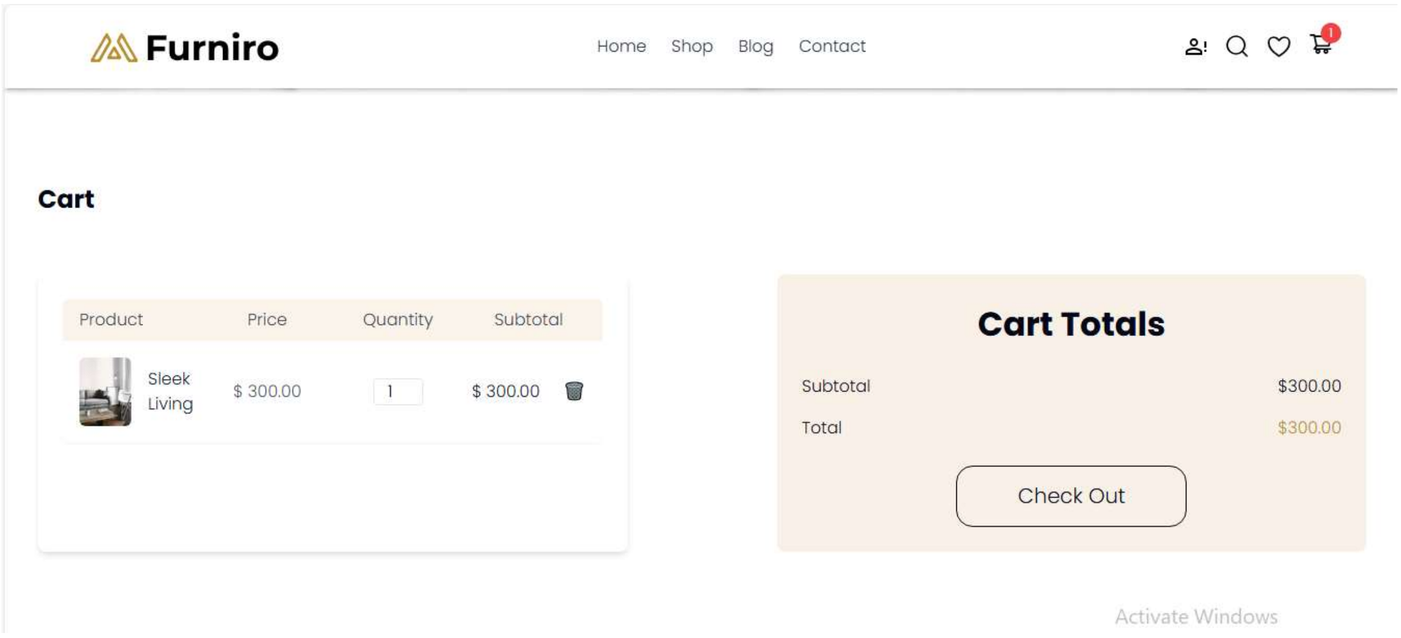
Subtotal: $320

Timber Craft
Introducing TimberCraft—a collection...
$ 320
wooden craftsmanship furniture modern nature inspired
Add to Cart

Bold Nest
Welcome to BoldNest—where fearless...
$ 260
bold nest furniture modern contemporary
Add to Cart

Syltherine
Introducing Syltherine – the ultimate...
$ 200
living fancy elegance desgin
Add to Cart

Activate Windows
Go to Settings to activate Windows.


Furniro

Home Shop Blog Contact

Cart

Product	Price	Quantity	Subtotal
Sleek Living	$ 300.00	1	$ 300.00

Cart Totals

Subtotal	$300.00
Total	$300.00

Check Out

Activate Windows

Day 4 completed