Prepared by Muhammad Nabeel

DAY 4 - DYNAMIC FRONTEND COMPONENTS

Published: January 19, 2025

Name of the Marketplace : www.clothsite.com (not confirmed)

Objective:

On Day 4, we will concentrate on designing and creating dynamic frontend components to showcase marketplace data retrieved from Sanity CMS or APIs. This step highlights the importance of modular and reusable component design, as well as real-world techniques for developing scalable and responsive web applications.

Key Learning Outcomes:

1. Build dynamic frontend components to display data from Sanity CMS or APIs.
2. Implement reusable and modular components.
3. Understand and apply state management techniques.
4. Learn the importance of responsive design and UX/UI best practices.
5. Prepare for real-world client projects by replicating professional workflows.

Overview of System Functions:

The project integrates the following key functionalities:

1. Product Listing Page
2. Individual Product Detail Page
3. Cart Page
4. Search Bar

5. Checkout Page
6. Header
7. Footer
8. Wishlist Component
9. Filter Panel Component
10. Related Products Component
11. Pagination Component
12. User Profile Component
13. Reviews and Ratings Component
14. Notifications Component
15. Analytics Dashboard Component
16. Product Comparison Component
17. Multi-Language Support Component
18. Order Tracking Component
19. FAQ and Help Center Component
20. Subscription Management Component
21. Admin Dashboard Component
22. Discount and Promotion Component
23. Social Media Sharing Component
24. Bulk Upload Component
25. AI Recommendations Component
26. Gift Card and Voucher Component
27. Customer Feedback Component
28. Advanced Search Component

Each feature is integral to constructing a responsive and scalable marketplace platform.

Key Components to Build:

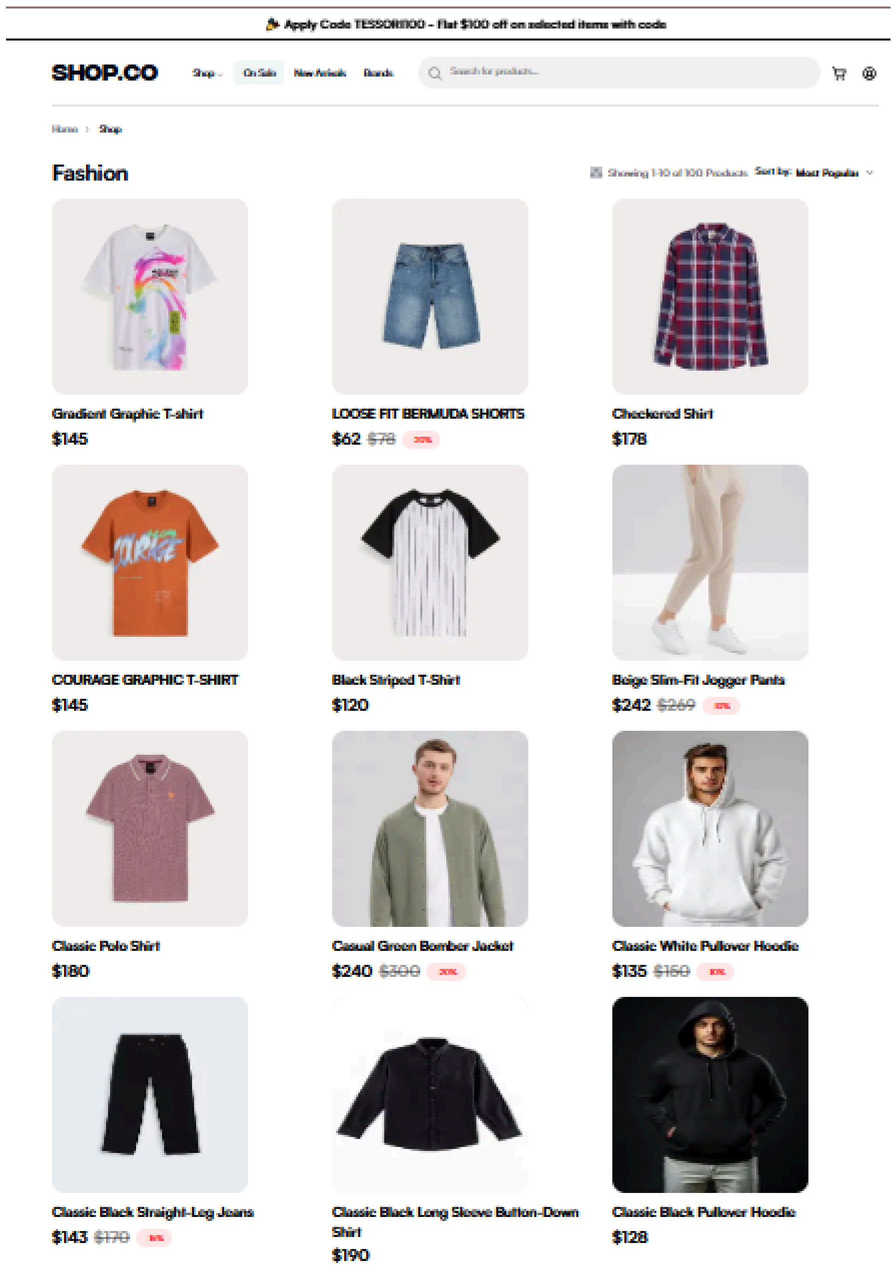
1. Product Listing Component

Code:

```
const { allProducts, addToProducts } = useCart();
  const [isLoading, setIsLoading] = useState(true); // Track loading state
  useEffect(() => {
    fetchProducts()
      .then((fetchedProducts) => {
        addToProducts(fetchedProducts);
        setIsLoading(false); // Data has been loaded
      })
      .catch((error) => {
        console.error("Failed to fetch products:", error);
        setIsLoading(false); // Stop loading even if there's an error
      });
  }, []);


  const productData = allProducts.find(
    (product) => product._id === (params.slug[0])
  );
  if (isLoading) {
    return <div>Loading...</div>; // Show a loading state
  }


  if (!productData?.name) {
    notFound();
  }


const ProductListSec = ({ title, data, viewAllLink }: ProductListSecProps)
=> {
  return (
    <section className="max-w-frame mx-auto text-center">
      <motion.h2
        className={cn([
          integralCF.className,
          "text-[32px] md:text-5xl mb-8 md:mb-14 capitalize",
        ])}
      >
        {title}
      </motion.h2>
      <motion.div
      >
        <Carousel
          opts={{
            align: "start",
          }}
          className="w-full mb-6 md:mb-9"
        >
          <CarouselContent className="mx-4 xl:mx-0 space-x-4
sm:space-x-5">
            {data.map((product) => (
              <CarouselItem
                key={product._id}
                className="w-full max-w-[198px] sm:max-w-[295px] pl-0"
              >
```

```
                <ProductCard data={product} />
              </CarouselItem>
            ))}
          </CarouselContent>
        </Carousel>
        {viewAllLink && (
          <div className="w-full px-4 sm:px-0 text-center">
            <Link
              href={viewAllLink}
              className="w-full inline-block sm:w-[218px] px-[54px] py-4
border rounded-full hover:bg- hover:text-black text- transition-all
font-medium text-sm sm:text-base border-black/10"
            >
              View All
            </Link>
          </div>
        )}
      </motion.div>
    </section>
  );
};
```

Output:


Apply Code TESSORI100 – Flat $100 off on selected items with code
SHOP.CO
Shop
On Sale
New Arrivals
Brands
Search for products...
Home › Shop
Fashion
Showing 1-10 of 100 Products Sort by: Most Popular
Gradient Graphic T-shirt
$145
LOOSE FIT BERMUDA SHORTS
$62 ~~$78~~
Checkered Shirt
$178
COURAGE GRAPHIC T-SHIRT
$145
Black Striped T-Shirt
$120
Beige Slim-Fit Jogger Pants
$242 ~~$269~~
Classic Polo Shirt
$180
Casual Green Bomber Jacket
$240 ~~$300~~
Classic White Pullover Hoodie
$135 ~~$150~~
Classic Black Straight-Leg Jeans
$143 ~~$170~~
Classic Black Long Sleeve Button-Down Shirt
$190
Classic Black Pullover Hoodie
$128

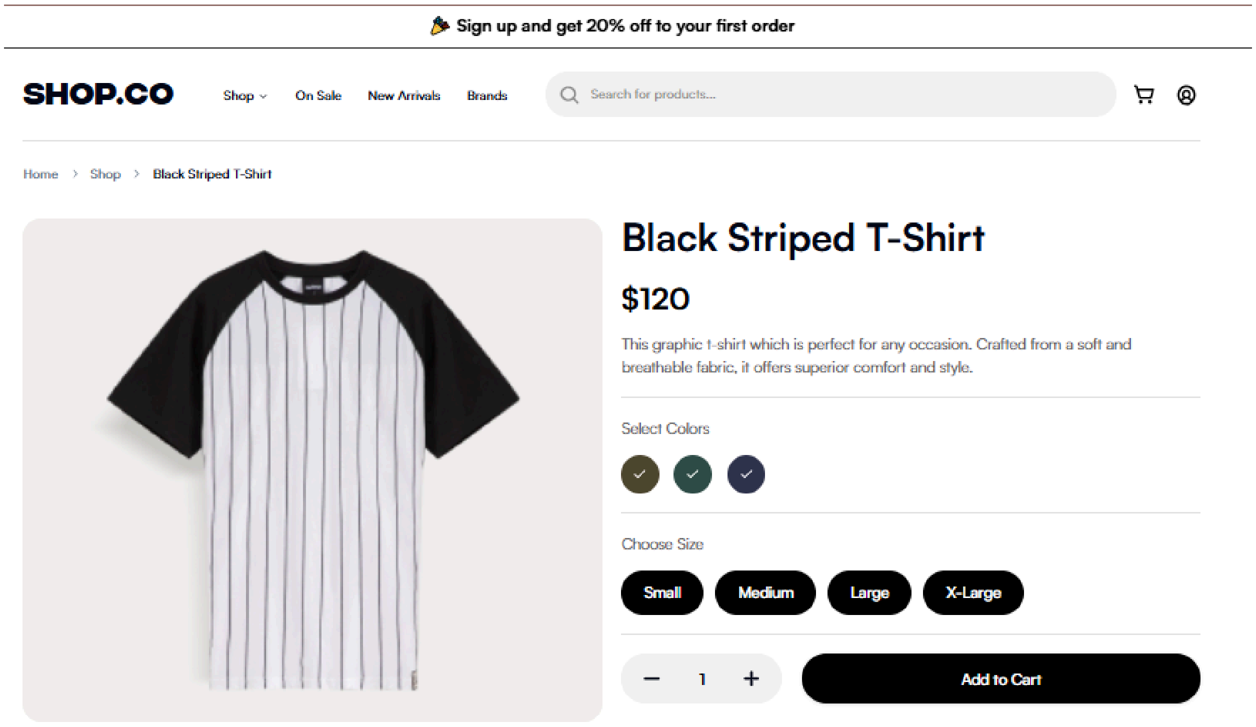
---

2. Individual Product Detail Component

Code:

```
const ProductCard = ({ data }: ProductCardProps) => {
  return (
    <Link
      href={`/shop/product/${data._id}/`}
      className="flex flex-col items-start aspect-auto"
    >
      <div className="bg-[#F0EEED] rounded-[13px] lg:rounded-[20px] w-full
lg:max-w-[295px] aspect-square mb-2.5 xl:mb-4 overflow-hidden">
        <Image
          src={data.imageUrl}
          width={295}
          height={298}
          className="rounded-md w-full h-full object-fill hover:scale-10
transition-all duration-500"
          alt={data.imageUrl}
          priority
        />
      </div>
      <strong className="text-black xl:text-xl">{data.name}</strong>
      <div className="flex items-end mb-1 xl:mb-2">

      </div>
      <div className="flex items-center space-x-[5px] xl:space-x-2.5">
        {data.discountPercent > 0 ? (
          <span className="font-bold text-black text-xl xl:text-2xl">
            {`$${Math.round(
              calculateDiscountedPrice(data.price, data.discountPercent)
            )}`}
          </span>
        ) : calculateDiscountedPrice(data.price, data.discountPercent) > 0
? (
          <span className="font-bold text-black text-xl xl:text-2xl">
            {`$${calculateDiscountedPrice(data.price,
data.discountPercent)}`}
          </span>
        ) : (
          <span className="font-bold text-black text-xl xl:text-2xl">
            ${data.price}
          </span>
        )}

        {data.discountPercent > 0 && (
          <span className="font-bold text-black/40 line-through text-xl
xl:text-2xl">
            ${data.price}
          </span>
        )}
        {data.discountPercent > 0 ? (
          <span className="font-medium text-[10px] xl:text-xs py-1.5
px-3.5 rounded-full bg-[#FF3333]/10 text-[#FF3333]">
            {`-${data.discountPercent}%`}
          </span>
        ) : (
          data.discountPercent > 0 && (
            <span className="font-medium text-[10px] xl:text-xs py-1.5
px-3.5 rounded-full bg-[#FF3333]/10 text-[#FF3333]">
              {`-$${data.price}`}
            </span>
          )
        )}
      </div>
    </Link>
  );
};

export default ProductCard;
```

Output:



🎉 Sign up and get 20% off to your first order
SHOP.CO
Shop
On Sale
New Arrivals
Brands
Search for products...
Home › Shop › Black Striped T-Shirt
Black Striped T-Shirt
$120
This graphic t-shirt which is perfect for any occasion. Crafted from a soft and breathable fabric, it offers superior comfort and style.
Select Colors
Choose Size
Small
Medium
Large
X-Large
1
Add to Cart

3. Cart Component

Code:

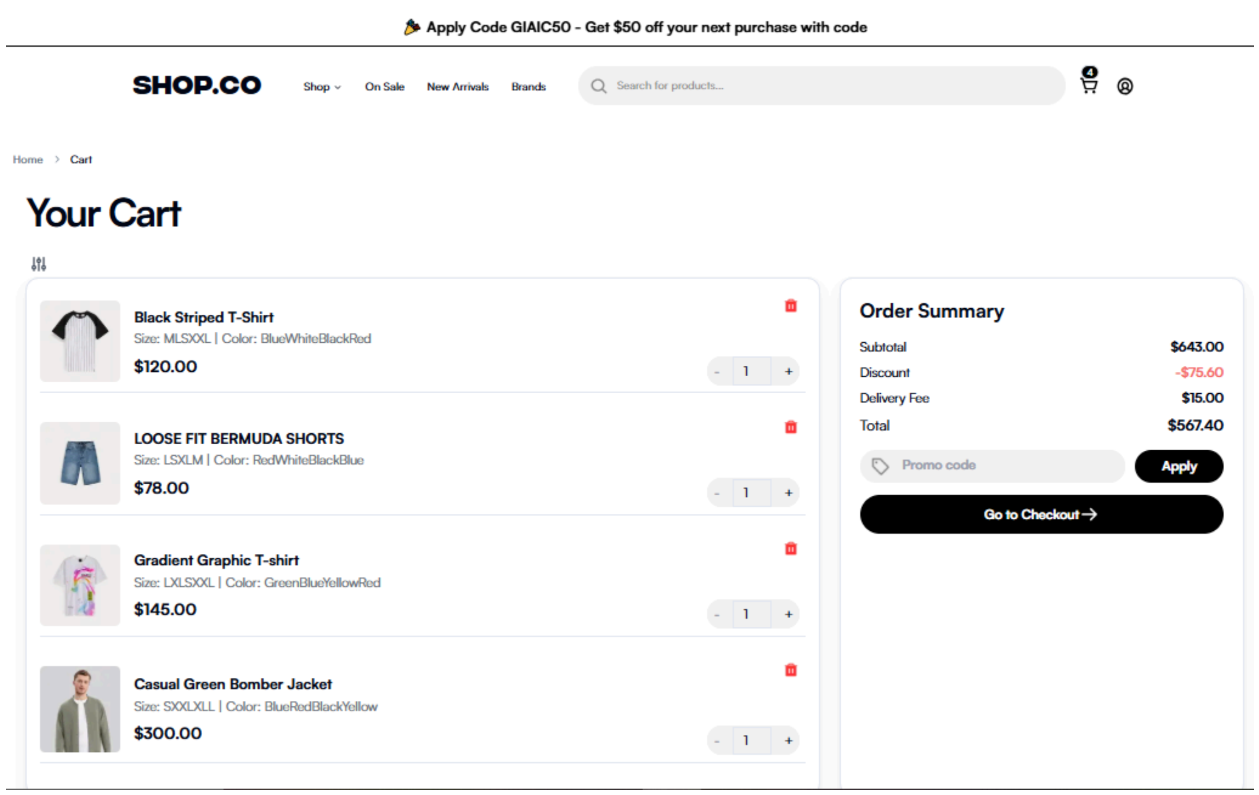
```
 useEffect(() => {

       const validatePromoCode = (promoCode: string) => {
           const promoMatching = validPromoCodes.find(
               (singlePromo) => singlePromo.code === promoCode.trim()
           );

           if (!promoMatching) {
               setPromoError("Invalid promo code!");
               return 0;
```

```
            }
            const currentDate = new Date();
            const expiryDate = new Date(promoMatching.expiry);

            if (expiryDate < currentDate) {
                setPromoError("Promo code has expired!");
                return 0;
            }
            setPromoError("");
            return promoMatching.discount; // Return valid promo discount
        };

        // Get the promo discount (if valid)
        pricing.promoDiscount = promoCode ? validatePromoCode(promoCode) :
0;

        const discountedTotal = cart.reduce((total, item) => {
            if (item.discountPercent > 0) {
                return total + calculateDiscountedPrice(item.price,
item.discountPercent) * item.quantity;
            }
            return total;
        }, 0);

        const nonDiscountedTotal = cart.reduce((total, item) => {
            if (item.discountPercent === 0) {
                return total + item.price * item.quantity;
            }
            return total;
        }, 0);

        // Calculate the subtotal (sum of all products at original price)
        const subtotal = cart.reduce((total, item) => total + item.price *
item.quantity, 0);

        // Calculate the total amount after discounts
        const total = discountedTotal + nonDiscountedTotal;

        // Amount of discount applied
        const amountOfDiscount = subtotal - total;

        // Update the pricing state
        setPricing({
            ...pricing,
            total: total,
            discountedAmount: discountedTotal,
            nonDiscountedAmount: nonDiscountedTotal,
            subtotal: subtotal,
            amountOfDiscount: amountOfDiscount,
            amountAfterDiscount: total,
        });
    }, [cart, promoCode, finalizedTotal]);
```

```
<div className="container mx-auto px-4 my-8">
            <h1 className="text-5xl mb-6 font-integralCF tracking-tighter
font-extrabold">Your Cart</h1>
            <div>
                <FilterMenu />
            </div>
            <div className="flex flex-col lg:flex-row gap-6">
                <div className="flex-1 bg-white rounded-2xl shadow-lg
border p-4 ">
                    {cart.map((item) => (
                        <div key={item._id} className="relative border-b-2
flex items-center gap-4 mb-6 px-0 py-3">
                            <button
                                onClick={() => removeFromCart(item._id)}
                                className="absolute top-2 right-2
text-red-500 hover:text-red-700"
                            >
                                <Image src={DeleteIcon} alt="Delete"
width={20} height={20} />
                            </button>

                            <Image src={item.imageUrl} alt={item.name}
width={100} height={100} className="rounded-md" />

                            <div className="flex-1">
                                <h3 className="text-lg
font-semibold">{item.name}</h3>
                                <p className="text-gray-500">Size:
{item.sizes} | Color: {item.colors}</p>
                                <p className="font-bold text-xl
mt-2">${item.price.toFixed(2)}</p>
                            </div>

                            <div className="absolute bg-[#F0F0F0]
rounded-full bottom-2 right-2 flex items-center gap-2">
                                <button
                                    onClick={() =>
updateQuantity(item._id, item.quantity - 1)}
                                    className="px-2 py-1.5 bg-[#F0F0F0]
rounded-l-full hover:bg-[#F0F0F0]"
                                >
                                    -
                                </button>
                                <input
                                    type="number"
                                    value={item.quantity}
                                    className="w-12 text-center
bg-[#F0F0F0] p-1 border rounded-none outline-none"
                                    min="1"
                                />
                                <button
                                    onClick={() =>
updateQuantity(item._id, item.quantity + 1)}
```

```
                                    className="px-2 py-1.5 bg-[#F0F0F0]
rounded-r-full hover:bg-[#F0F0F0]"
                                >
                                    +
                                </button>
                            </div>
                        </div>
                    ))}
                </div>
                <div className="w-full lg:max-w-[500px] bg-white
font-satoshi font-bold shadow-lg border rounded-2xl p-6">
                    <h2 className="text-2xl font-semibold mb-4">Order
Summary</h2>
                    <div className="flex justify-between mb-2">
                        <span className='font-normal
font-satoshi'>Subtotal</span>
                        <span>${pricing.subtotal.toFixed(2)}</span>
                    </div>
                    <div className="flex justify-between mb-2">
                        <span className='font-normal
font-satoshi'>Discount </span>
                        <span
className='text-red-400'>-${pricing.amountOfDiscount.toFixed(2)}</span>
                    </div>
                    <div className="flex justify-between mb-2">
                        <span className='font-normal
font-satoshi'>Delivery Fee</span>
                        <span>${deliveryFee.toFixed(2)}</span>
                    </div>
                    {pricing.promoDiscount > 0 && promoCode.length > 0 &&
                        <div className="flex justify-between mb-2">
                            <span className='font-normal
font-satoshi'>Promo Code Discount</span>
                            <span
className='text-red-400'>-${pricing.promoDiscount.toFixed(2)}</span>
                        </div>
                    }
                    <div className="flex justify-between font-bold
text-lg">
                        <span className='font-normal
font-satoshi'>Total</span>
                        <span>${(finalizedTotal).toFixed(2)}</span>
                    </div>
                    <div className="mt-4 w-full flex flex-row">
                        {/* Promo Code Input */}
                        <div className="flex w-full flex-row bg-[#F0F0F0]
rounded-full items-center">
                            <Image
                                src={PriceTag}
                                alt="price-tag"
                                width={25}
                                height={25}
                                className="ml-3"
                            />
                            <input
                                type="text"
                                onChange={(e) =>
setPromo((e.target.value).toUpperCase())}
                                value={promo}
                                placeholder="Promo code"
                                className="w-full pl-4 py-2 bg-[#F0F0F0]
rounded-full focus:outline-none"
                            />
                        </div>
                        {/* Apply Button */}
                        <button onClick={() => setPromoCode(promo)}
className="w-full md:w-1/3 bg-black text-white py-2 ml-3 rounded-full
hover:bg-gray-800">
                            Apply
                        </button>
                    </div>

                    <Link href="/cart/checkout">
                        <button
                            onClick={() => totalFromCart(finalizedTotal)}
                            className="w-full flex justify-center
items-center bg-black text-white py-3 rounded-full mt-4 hover:bg-black"
                        >
                            Go to Checkout
                            <span>
                                <Image src={Arrow} alt='arrow' width={25}
height={25} />
                            </span>
                        </button>
                    </Link>
                </div>
            </div>
        </div>
```

Output:



Apply Code GIAIC50 - Get $50 off your next purchase with code

SHOP.CO Shop On Sale New Arrivals Brands Search for products...

Home > Cart

Your Cart

Black Striped T-Shirt
Size: MLSXXL | Color: BlueWhiteBlackRed
$120.00
- 1 +

LOOSE FIT BERMUDA SHORTS
Size: LSXLM | Color: RedWhiteBlackBlue
$78.00
- 1 +

Gradient Graphic T-shirt
Size: LXLSXXL | Color: GreenBlueYellowRed
$145.00
- 1 +

Casual Green Bomber Jacket
Size: SXXLXLL | Color: BlueRedBlackYellow
$300.00
- 1 +

Order Summary

Subtotal	$643.00
Discount	-$75.60
Delivery Fee	$15.00
Total	$567.40

Promo code Apply

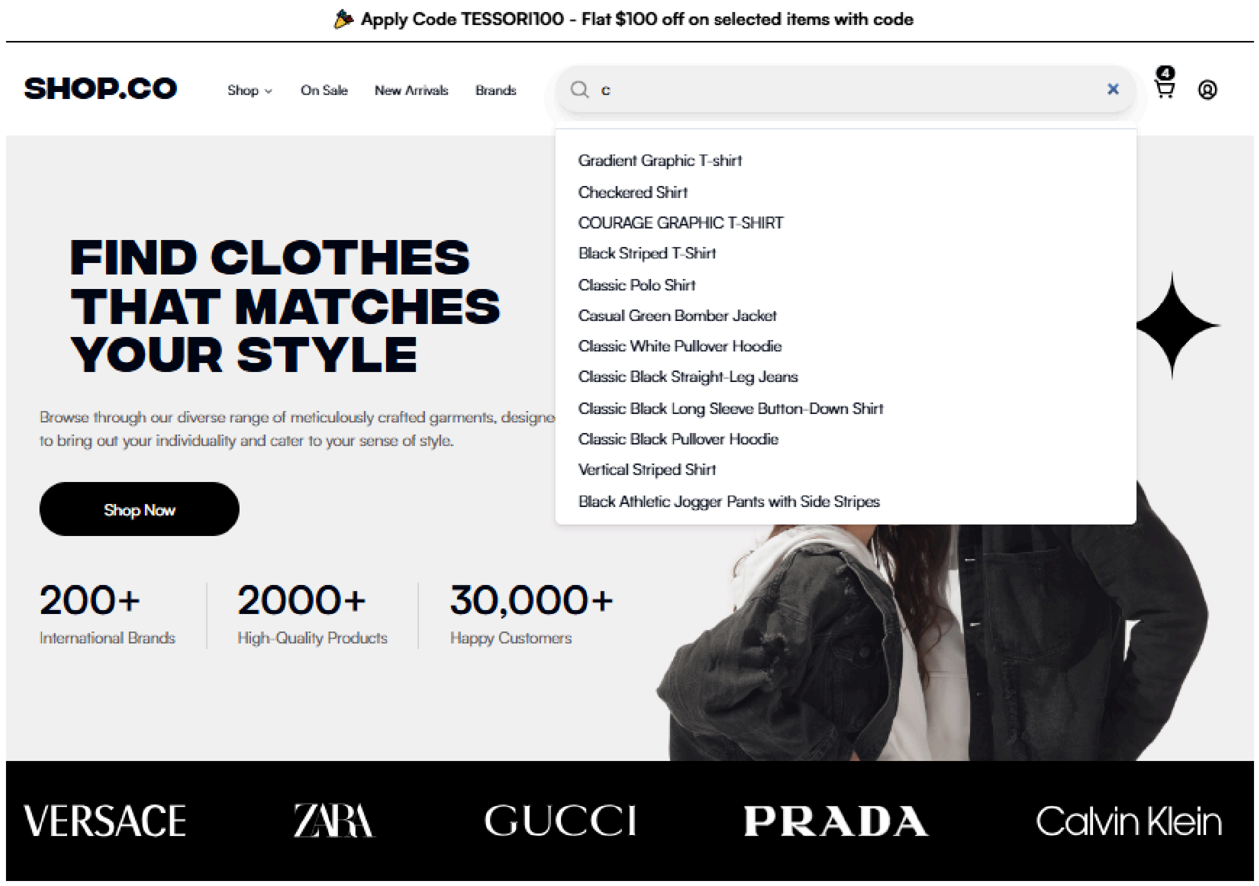
Go to Checkout →

4. Search Bar Component

Code:

```
const SearchBar: React.FC<SearchBarProps> = ({ products }) => {
  const [isDropdownOpen, setIsDropdownOpen] = useState(false);
  const [searchQuery, setSearchQuery] = useState("");
  const [isInteractingWithDropdown, setIsInteractingWithDropdown] =
    useState(false);

  const handleFocus = () => {
    setIsDropdownOpen(true);
  };

  const handleBlur = () => {
    // Close only if the user is not interacting with the dropdown
    if (!isInteractingWithDropdown) {
      setIsDropdownOpen(false);
    }
  };

  const handleDropdownMouseEnter = () => {
    setIsInteractingWithDropdown(true);
  };
  const handleDropdownMouseLeave = () => {
    setIsInteractingWithDropdown(false);
  };

  const handleCloseDropdown = () => {
    setIsDropdownOpen(false);
  };

  // Filter products based on search query
  const filteredProducts = products.filter((product) =>
    product.name.toLowerCase().includes(searchQuery.toLowerCase())
  );

  return (
    <div className="relative w-full">
      {/* Search Input */}
      <InputGroup className="hidden w-full md:flex bg-[#F0F0F0] mr-3
lg:mr-10 shadow-none">
        <InputGroup.Text>
          <Image
            priority
            src="/icons/search.svg"
            height={20}
            width={20}
            alt="search"
            className="min-w-5 min-h-5 shadow-none"
          />
        </InputGroup.Text>
        <InputGroup.Input
          type="search"
          name="search"
          value={searchQuery}
          onChange={(e) => {
            setSearchQuery(e.target.value)
            handleFocus()
          }}
          placeholder="Search for products..."
          className="bg-transparent placeholder:text-black/40 shadow-none"
          // onFocus={handleFocus} // Open dropdown on focus
          onBlur={handleBlur} // Close dropdown on blur (only if not
interacting)
        />
      </InputGroup>

      {/* Dropdown */}
      {isDropdownOpen && (
        <div
          className="absolute top-full mt-2 w-full bg-white shadow-md
rounded-md z-10"
          onMouseEnter={handleDropdownMouseEnter}
          onMouseLeave={handleDropdownMouseLeave}
        >

          {/* Separator */}
          <div className="border-t my-2" />

          {/* Dynamic Product Results */}
          <div className="px-4 py-2">
            {filteredProducts.length > 0 ? (
              filteredProducts.map((product) => (
                <Link
                  key={product._id} // Use the unique product ID as the
key
                  href={`/shop/product/${product._id}`}
                  onClick={handleCloseDropdown} // Close dropdown on click
                  className="block px-2 py-1 hover:bg-gray-100 rounded
cursor-pointer"
                >
                  {product.name}
                </Link>
              ))
            ) : (
              <div className="text-gray-500 px-2 py-1">No products
found</div>
            )}
          </div>
        </div>
      )}
    </div>
```

Output:



🎉 Apply Code TESSORI100 - Flat $100 off on selected items with code

SHOP.CO Shop On Sale New Arrivals Brands

c

Gradient Graphic T-shirt
Checkered Shirt
COURAGE GRAPHIC T-SHIRT
Black Striped T-Shirt
Classic Polo Shirt
Casual Green Bomber Jacket
Classic White Pullover Hoodie
Classic Black Straight-Leg Jeans
Classic Black Long Sleeve Button-Down Shirt
Classic Black Pullover Hoodie
Vertical Striped Shirt
Black Athletic Jogger Pants with Side Stripes

FIND CLOTHES THAT MATCHES YOUR STYLE

Browse through our diverse range of meticulously crafted garments, designe to bring out your individuality and cater to your sense of style.

Shop Now

200+ International Brands | 2000+ High-Quality Products | 30,000+ Happy Customers

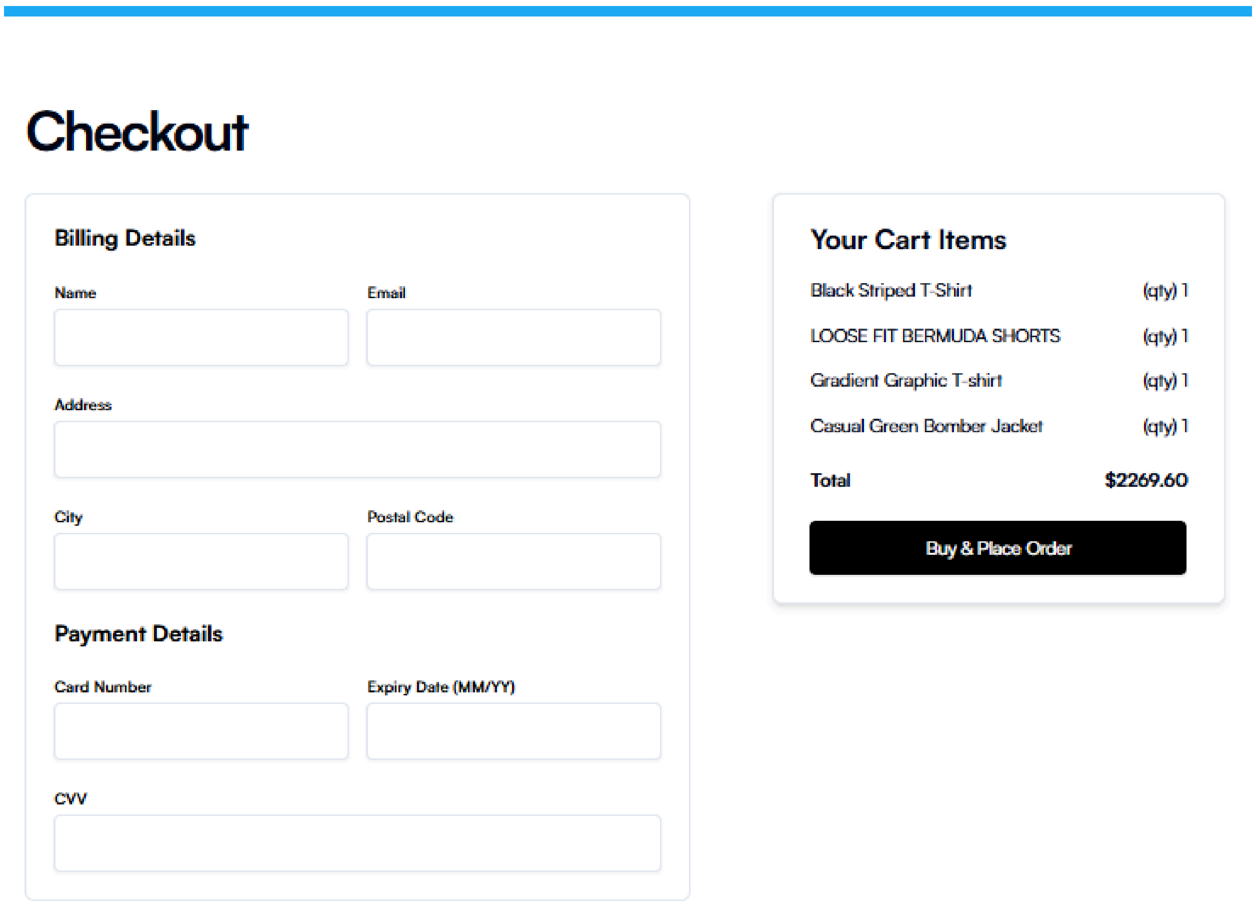
VERSACE ZARA GUCCI PRADA Calvin Klein

5. Checkout Component

Code:

```
const checkoutSchema = z.object({
    name: z.string().min(2, "Name must be at least 2 characters long"),
    email: z.string().email("Invalid email address"),
    address: z.string().min(5, "Address must be at least 5 characters
long"),
    city: z.string().min(2, "City must be at least 2 characters long"),
    postalCode: z
        .string()
        .regex(/^\d{5}$/, "Postal Code must be exactly 5 digits"),
    cardNumber: z
        .string()
        .regex(/^\d{16}$/, "Card Number must be exactly 16 digits"),
    expiryDate: z
        .string()
        .regex(/^(0[1-9]|1[0-2])\/\d{2}$/, "Expiry Date must be in MM/YY
format"),
    cvv: z.string().regex(/^\d{3}$/, "CVV must be exactly 3 digits"),
});

type CheckoutFormData = z.infer<typeof checkoutSchema>;

export default function CheckoutMain() {
    const [dialogOpen, setDialogOpen] = useState(false)
    const {
        register,
        handleSubmit,
        formState: { errors },
    } = useForm<CheckoutFormData>({
        resolver: zodResolver(checkoutSchema),
    });

    const onSubmit = (data: CheckoutFormData) => {
        console.log("Form Data:", data);
        setDialogOpen(true);
        alert("Checkout Successful!");
    };
```

Output:



Checkout

Billing Details

Name

Email

Address

City

Postal Code

Payment Details

Card Number

Expiry Date (MM/YY)

CVV

Your Cart Items

Black Striped T-Shirt	(qty) 1
LOOSE FIT BERMUDA SHORTS	(qty) 1
Gradient Graphic T-shirt	(qty) 1
Casual Green Bomber Jacket	(qty) 1
Total	$2269.60

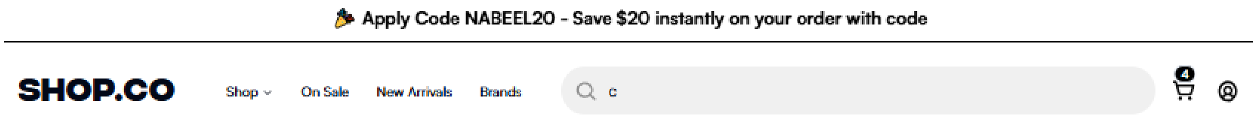
6. Header Component

Code:

```
TopNavbar = () => {
  const { allProducts, addToProducts } = useCart();

  useEffect(() => {
    fetchProducts()
      .then((fetchedProducts) => addToProducts(fetchedProducts))
      .catch((error) => console.error('Failed to fetch products:',
error));
  }, []);

  return (
    <nav className="sticky top-0 bg-white border-t-2 border-black z-20">
      <div className="flex relative max-w-frame mx-auto items-center
justify-between md:justify-start py-5 md:py-6 px-4 xl:px-0">
        <div className="flex items-center">
          <div className="block md:hidden mr-4">
            <ResTopNavbar data={data} />
          </div>
          <Link
```

```
            href="/"
            className={cn([
              integralCF.className,
              "text-2xl lg:text-[32px] mb-2 mr-3 lg:mr-10",
            ])}
          >
            SHOP.CO
          </Link>
        </div>
        <NavigationMenu className="hidden md:flex mr-2 lg:mr-7">
          <NavigationMenuList>
            {data.map((item) => (
              <React.Fragment key={item.id}>
                {item.type === "MenuItem" && (
                  <MenuItem label={item.label} url={item.url} />
                )}
                {item.type === "MenuList" && (
                  <MenuList data={item.children} label={item.label} />
                )}
              </React.Fragment>
            ))}
          </NavigationMenuList>
        </NavigationMenu>
        <SearchBar products={allProducts} />
        <div className="flex items-center">
          <Link href="/search" className="block md:hidden mr-[14px] p-1">
            <Image
              priority
              src="/icons/search-black.svg"
              height={100}
              width={100}
              alt="search"
              className="max-w-[22px] max-h-[22px]"
            />
          </Link>
          <CartBtn />
          <Link href="/#signin" className="p-1">
            <Image
              priority
              src="/icons/user.svg"
              height={100}
              width={100}
              alt="user"
              className="max-w-[22px] max-h-[22px]"
            />
          </Link>
        </div>
      </div>
    </nav>
  );
};
```

Output:


🎉 Apply Code NABEEL20 - Save $20 instantly on your order with code
SHOP.CO
Shop
On Sale
New Arrivals
Brands
c
4

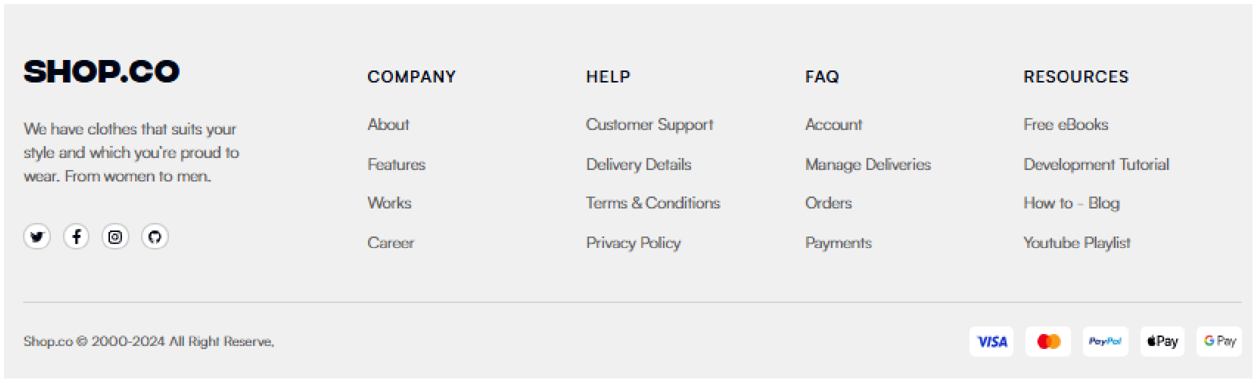
7. Footer Component

Code:

```
const Footer = () => {
  return (
    <footer className="mt-10">
      <div className="relative">
        <div className="absolute bottom-0 w-full h-1/2
bg-[#F0F0F0]"></div>
        <div className="px-4">
          <NewsLetterSection />
        </div>
      </div>
      <div className="pt-8 md:pt-[50px] bg-[#F0F0F0] px-4 pb-4">
        <div className="max-w-frame mx-auto">
          <nav className="lg:grid lg:grid-cols-12 mb-8">
            <div className="flex flex-col lg:col-span-3 lg:max-w-[248px]">
              <h1
                className={cn([
                  integralCF.className,
                  "text-[28px] lg:text-[32px] mb-6",
                ])}
              >
                SHOP.CO
              </h1>
              <p className="text-black/60 text-md mb-9">
                We have clothes that suits your style and which you're
proud to
                wear. From women to men.
              </p>
              <div className="flex items-center">
                {socialsData.map((social) => (
                  <Link
                    href={social.url}
                    key={social.id}
                    className="bg-white hover:bg-black hover:text-white
transition-all mr-3 w-7 h-7 rounded-full border border-black/20 flex
items-center justify-center p-1.5"
                  >
                    {social.icon}
                  </Link>
                ))}
              </div>
            </div>
            <div className="hidden lg:grid col-span-9 lg:grid-cols-4
lg:pl-10">
              <LinksSection />
            </div>
            <div className="grid lg:hidden grid-cols-2 sm:grid-cols-4">
              <LinksSection />
            </div>
          </nav>

          <hr className="h-[1px] border-t-black/10 mb-6" />
          <div className="flex flex-col sm:flex-row justify-center
sm:justify-between items-center mb-2">
            <p className="text-sm text-center sm:text-left text-black/60
mb-4 sm:mb-0 sm:mr-1">
              Shop.co © 2000-2024 All Right Reserve
              <Link
                href=""
                className="text-black font-medium"
              >

              </Link>
              {", "}

            </p>
            <div className="flex items-center">
              {paymentBadgesData.map((badge, _, arr) => (
                <span
                  key={badge.id}
                  className={cn([
                    arr.length !== badge.id && "mr-3",
                    "w-[46px] h-[30px] rounded-[5px] border-[#D6DCE5]
bg-white flex items-center justify-center",
                  ])}
                >
                  <Image
                    priority
                    src={badge.srcUrl}
                    width={33}
                    height={100}
                    alt="user"
                    className="max-h-[15px]"
                  />
                </span>
              ))}
            </div>
          </div>
        </div>
        <LayoutSpacing />
      </div>
    </footer>
  );
```

```
};
```

Output:



SHOP.CO

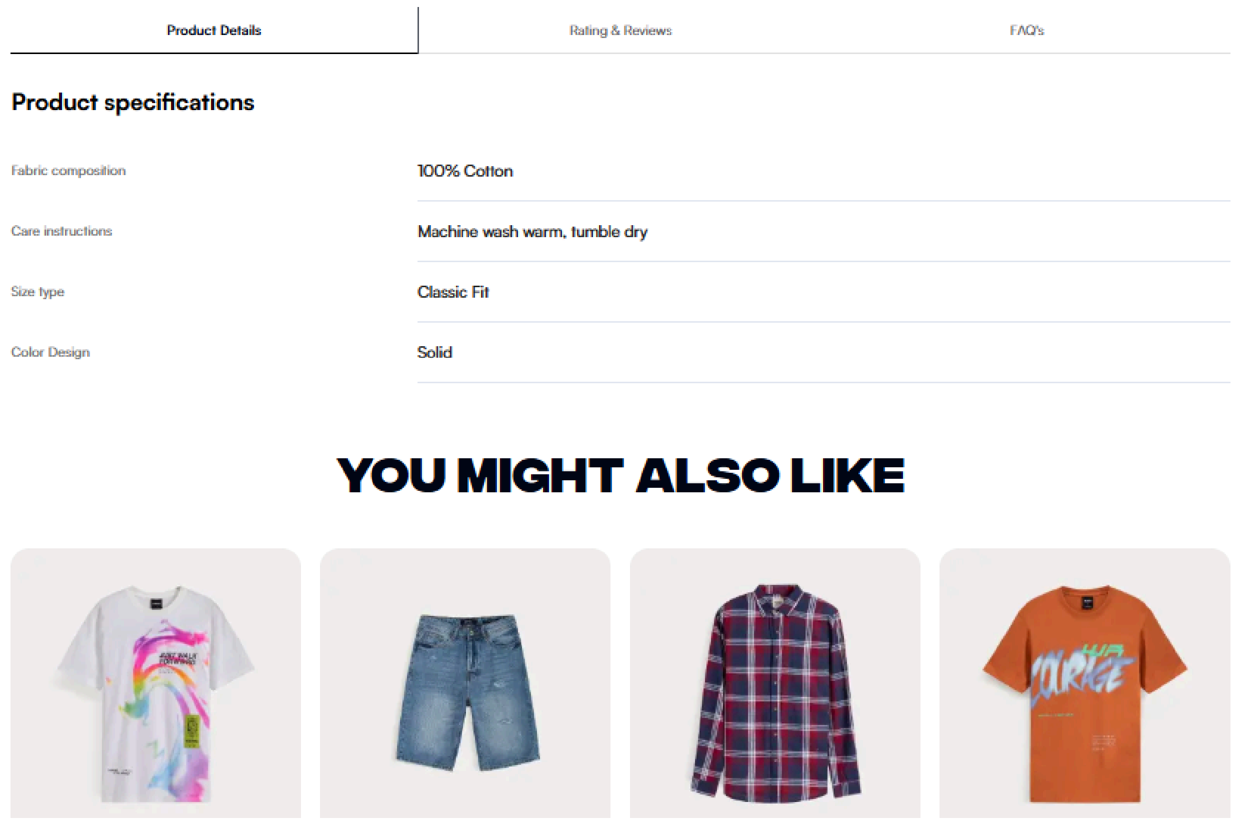
We have clothes that suits your style and which you're proud to wear. From women to men.

COMPANY	HELP	FAQ	RESOURCES
About	Customer Support	Account	Free eBooks
Features	Delivery Details	Manage Deliveries	Development Tutorial
Works	Terms & Conditions	Orders	How to - Blog
Career	Privacy Policy	Payments	Youtube Playlist

Shop.co © 2000-2024 All Right Reserve.

8. Related Products Component

Output:



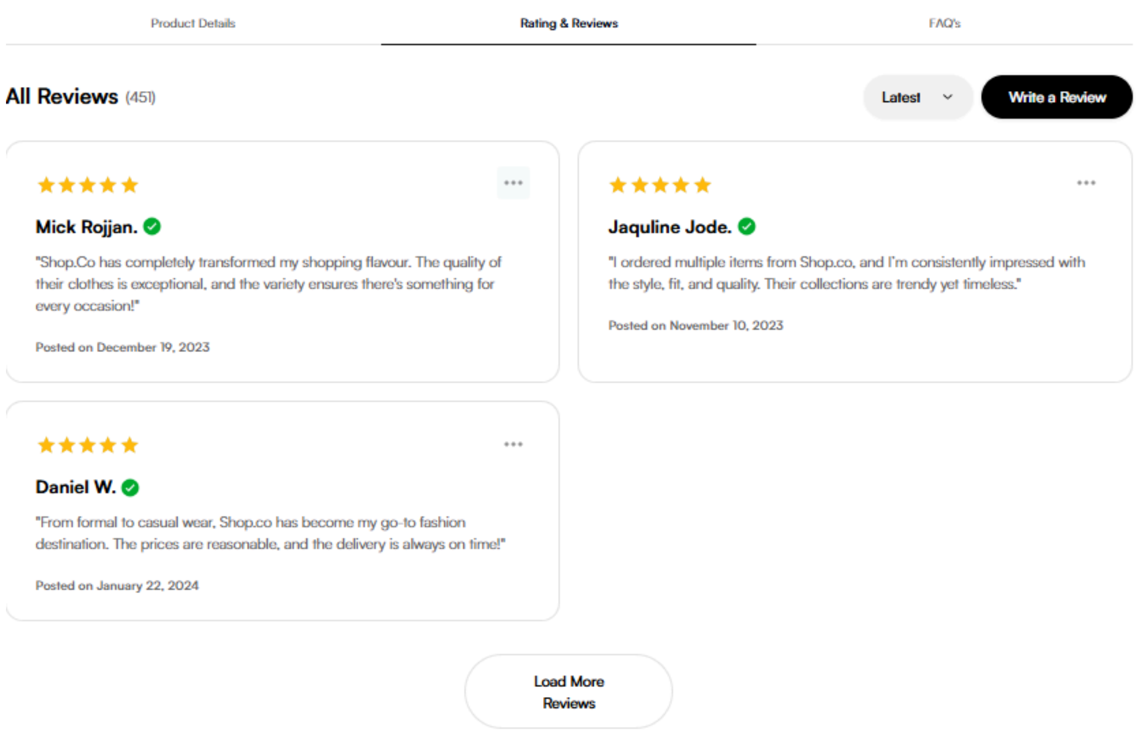
Product specifications

Fabric composition	100% Cotton
Care instructions	Machine wash warm, tumble dry
Size type	Classic Fit
Color Design	Solid

YOU MIGHT ALSO LIKE

9. Reviews and Ratings Component

Output:



All Reviews (451)

★★★★★

Mick Rojjan. ✓

"Shop.Co has completely transformed my shopping flavour. The quality of their clothes is exceptional, and the variety ensures there's something for every occasion!"

Posted on December 19, 2023

★★★★★

Jaquline Jode. ✓

"I ordered multiple items from Shop.co, and I'm consistently impressed with the style, fit, and quality. Their collections are trendy yet timeless."

Posted on November 10, 2023

★★★★★

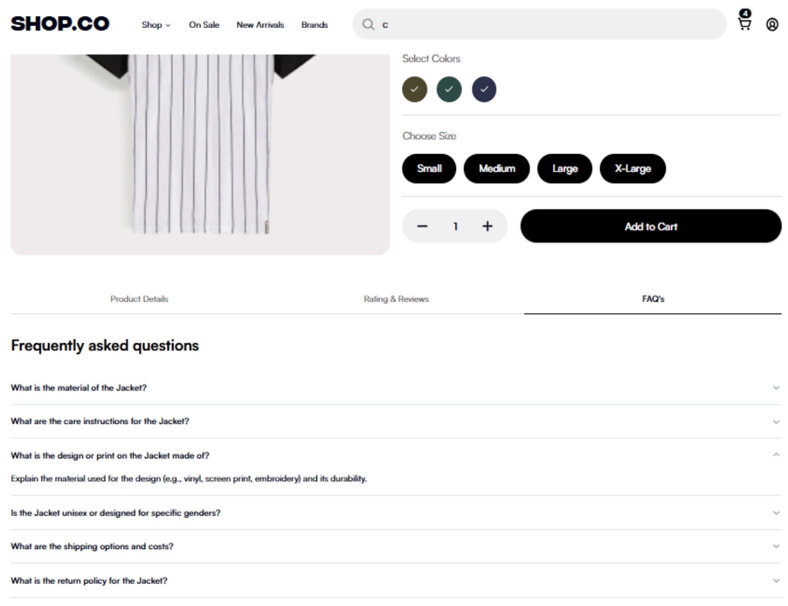
Daniel W. ✓

"From formal to casual wear, Shop.co has become my go-to fashion destination. The prices are reasonable, and the delivery is always on time!"

Posted on January 22, 2024

10. FAQ and Help Center Component

Output:



Frequently asked questions

What is the material of the Jacket?

What are the care instructions for the Jacket?

What is the design or print on the Jacket made of?

Explain the material used for the design (e.g., vinyl, screen print, embroidery) and its durability.

Is the Jacket unisex or designed for specific genders?

What are the shipping options and costs?

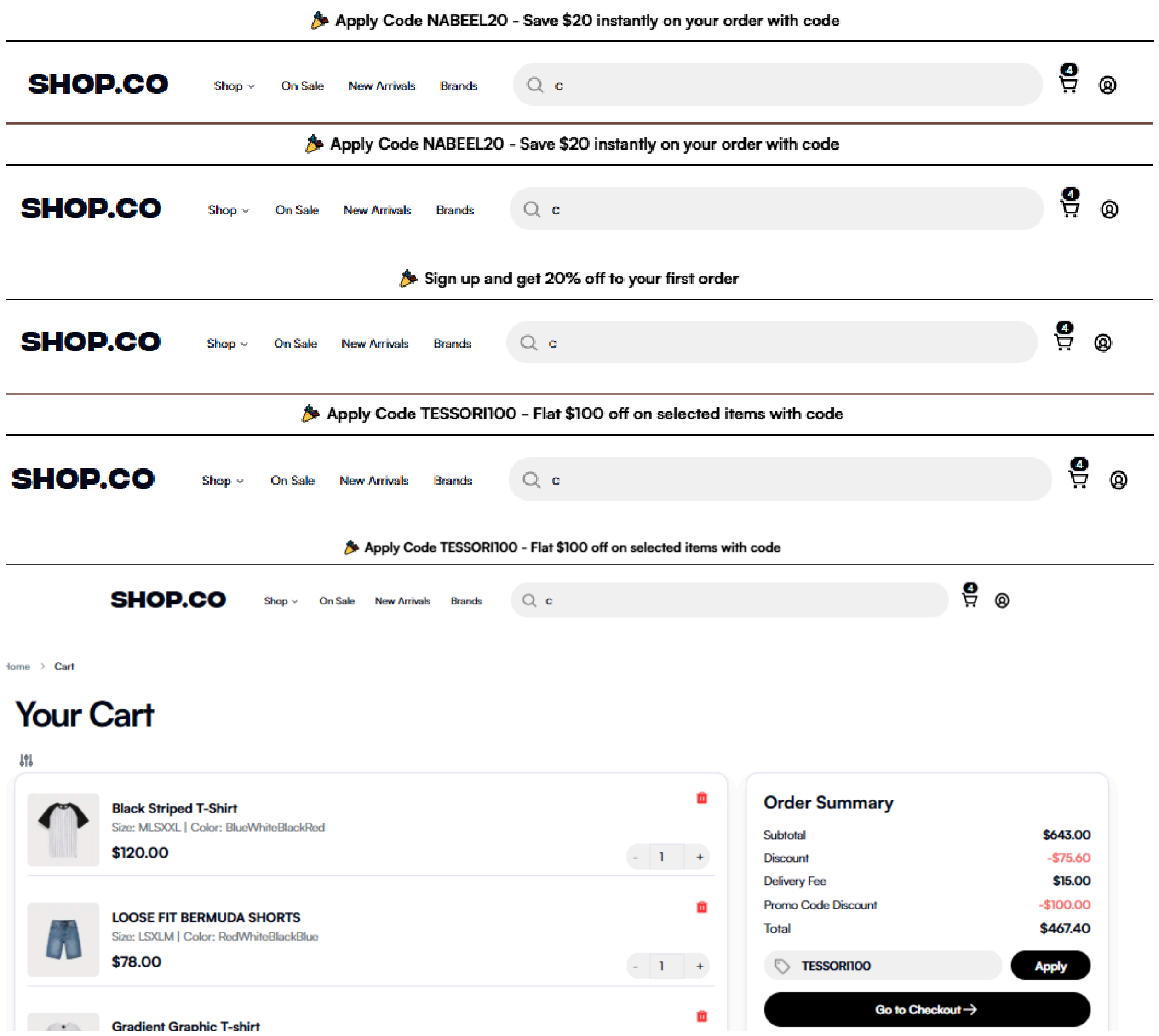
What is the return policy for the Jacket?

11. Discount and Promotion Component

Code:

// Code snippet for Discount and Promotion Component

Output:



Apply Code NABEEL20 - Save $20 instantly on your order with code

SHOP.CO Shop On Sale New Arrivals Brands c 4

Apply Code NABEEL20 - Save $20 instantly on your order with code

SHOP.CO Shop On Sale New Arrivals Brands c 4

Sign up and get 20% off to your first order

SHOP.CO Shop On Sale New Arrivals Brands c 4

Apply Code TESSORI100 - Flat $100 off on selected items with code

SHOP.CO Shop On Sale New Arrivals Brands c 4

Apply Code TESSORI100 - Flat $100 off on selected items with code

SHOP.CO Shop On Sale New Arrivals Brands c 4

Home > Cart

Your Cart

Black Striped T-Shirt
Size: MLSXXL | Color: BlueWhiteBlackRed
$120.00
- 1 +

LOOSE FIT BERMUDA SHORTS
Size: LSXLM | Color: RedWhiteBlackBlue
$78.00
- 1 +

Gradient Graphic T-shirt

Order Summary

Subtotal	$643.00
Discount	-$75.60
Delivery Fee	$15.00
Promo Code Discount	-$100.00
Total	$467.40

TESSORI100 Apply

Go to Checkout →

Prepared by: ***Muhammad Nabeel***