

# **DAY 4 - BUILDING DYNAMIC FRONTEND COMPONENTS: Nike**

This documentation provides a comprehensive overview of the essential functionalities for a dynamic marketplace. The focus is on modularity, reusability, and seamless integration with Sanity CMS. Each feature is explained in detail, followed by a conclusion summarizing the overall approach.

## **Step 1: Overview of Functionalities**

The project incorporates the following key features:

- 1. Dynamic Routing**
- 2. Cart Functionality**
- 3. Wishlist Component**
- 4. Product Detail Component**
- 5. Checkout Process**
- 6. FAQ and Help Center Component**
- 7. Footer and Header Components**
- 8. Related Products Component**
- 9. Product Listing page**

## **Step 2: Functionalities in Detail**


# 1. Dynamic Routing

Dynamic routing allows for the creation of individual product detail pages, enabling users to view detailed information about each product.

## ***Detailed Description:***

- Every product has a unique identifier (ID or slug) used to dynamically generate its URL (e.g., /product/[id]).
- These pages are server-rendered to improve SEO and provide faster initial load times.
- Dynamic routes display details like product descriptions, images, price, stock status, and reviews.
- This approach ensures scalability, allowing new products to automatically generate corresponding pages without manual intervention.

```
const ProductPage = ({ params }: { params: { productId: string } }) => {  
  <div className="grid grid-cols-1 lg:grid-cols-2 gap-10">  
    <div className="flex justify-center items-center">  
      {imageUrl && (  
        <Image  
          src={imageUrl}  
          alt={productName}  
          width={1300}  
          height={700}  
          className="w-full h-full object-cover rounded-xl shadow-xl"  
        />  
      )}  
    </div>  
  
    <div className="flex flex-col space-y-6">  
      <h1 className="text-4xl font-semibold text-gray-900">{productName}</h1>  
      <p className="text-lg text-gray-600">Category: {category}</p>  
      <p className="text-3xl font-semibold text-black">${price}</p>  
      <p  
        className={`text-sm font-medium ${status === 'In Stock' ? 'text-green-600' : 'text-red-600'}`}  
      >  
        {status}  
      </p>  
      <p className="text-sm text-gray-600">Stock: {inventory}</p>  
      <p className="text-sm text-gray-700 mt-4">{description}</p>  
  
      <div className="space-y-4 mt-6">  
        <button  
          onClick={handleAddToCart}  
          className="bg-black text-white w-full px-6 py-3 rounded-full hover:bg-gray-800 transition-all"  
        >  
          Add to Cart  
        </button>  
      </div>  
    </div>  
  </div>  
}
```



## Air Jordan 1 Elevate Low

Category: Women's Shoes

**\$11895**

Promo Exclusion

Stock: 8

The Air Jordan 1 Elevate Low features a clean, minimal design with premium materials. Its platform sole adds a touch of height and modern edge, while ensuring comfort and durability for all-day wear.

Add to Bag

♥ Favorites

Activate Windows  
Go to Settings to activate Windows

## 2. Cart Functionality

The Cart Functionality tracks user-selected items, ensuring a smooth shopping experience by summarizing costs and updating in real time.

### ***Detailed Description:***

- Users can add products to the cart from product listings or detail pages.
- The cart dynamically updates quantities and calculates the total cost.
- A mini-cart provides a quick summary, while a detailed cart page allows item editing or removal.
- React Context or Redux manages the cart state across the application.
- Local or session storage ensures the cart data persists even after a page refresh.

```
type CartItem = {
  id: string;
  title: string;
  price: number;
  quantity: number;
  image: string;
  category: string;
};

type CartContextType = {
  cart: CartItem[];
  addToCart: (item: CartItem) => void;
  removeFromCart: (id: string) => void;
};

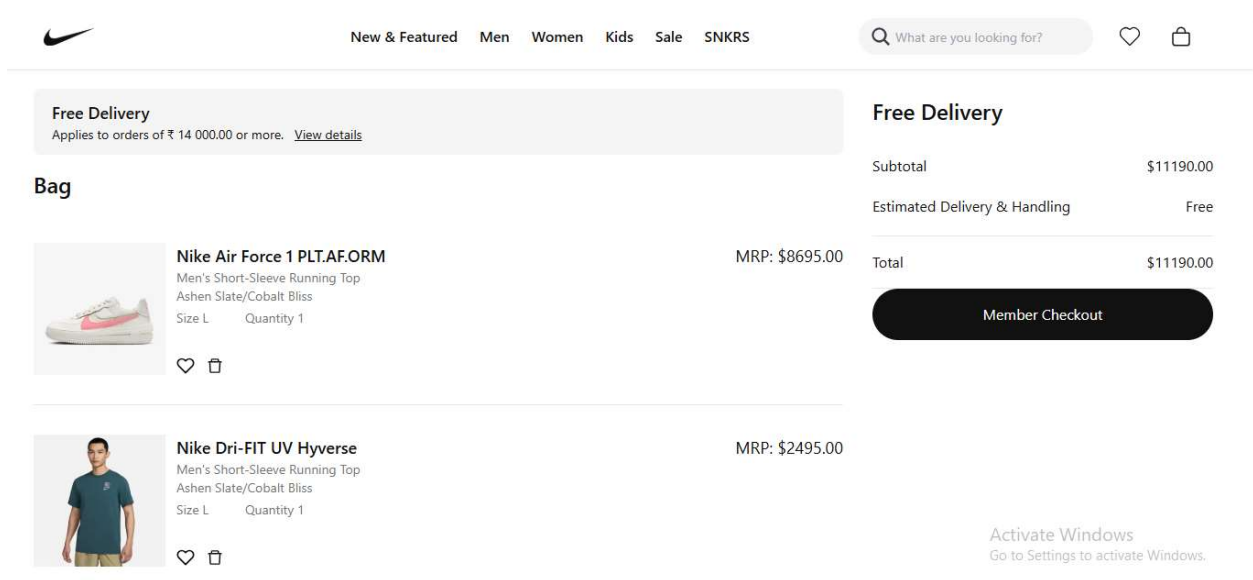
const CartContext = createContext<CartContextType | undefined>(undefined);

export const CartProvider = ({ children }: { children: ReactNode }) => {
  const [cart, setCart] = useState<CartItem[]>([]);

  // Load cart from localStorage on component mount
  useEffect(() => {
    const savedCart = localStorage.getItem('cart');
    if (savedCart) {
      setCart(JSON.parse(savedCart));
    }
  }, []);

  // Save cart to localStorage whenever it changes
  useEffect(() => {
    if (cart.length > 0) {
      localStorage.setItem('cart', JSON.stringify(cart));
    }
  }, [cart]);
};
```

Activate Window  
Go to Settings to activate



### 3. Wishlist Component

The Wishlist Component enhances user engagement by allowing them to save and manage their favorite products for future reference.

#### ***Detailed Description:***

- Users can add or remove products from their wishlist on both the product listing and detail pages.
- The wishlist dynamically updates to reflect changes in real time.
- Persistent data storage on local storage, ensures the wishlist remains intact across sessions.
- React Context or Redux manages the wishlist state consistently across the application.

```

type WishlistItem = {
  id: string;
  title: string;
  price: number;
  image: string;
};

type WishlistContextType = {
  wishlist: WishlistItem[];
  addToWishlist: (item: WishlistItem) => void;
  removeFromWishlist: (id: string) => void;
  isInWishlist: (id: string) => boolean;
};

const WishlistContext = createContext<WishlistContextType | undefined>(undefined);

export const WishlistProvider = ({ children }: { children: ReactNode }) => {
  const [wishlist, setWishlist] = useState<WishlistItem[]>([]);





  useEffect(() => {
    const savedWishlist = localStorage.getItem('wishlist');
    if (savedWishlist) {
      setWishlist(JSON.parse(savedWishlist)); // Load wishlist from localStorage
    }
  }, []);

  useEffect(() => {
    if (wishlist.length > 0) {
      localStorage.setItem('wishlist', JSON.stringify(wishlist)); // Save wishlist to localStorage
    }
  }, [wishlist]);

```

Activate Wind  
 Go to Settings to a

## Favorites

	Air Jordan 1 Elevate Low	MRP: \$11895.00
		
<hr/>		
	Nike Standard Issue Basketball Jersey	MRP: \$2895.00
		
<hr/>		

## 4. Product Detail Component

The Product Detail Component provides users with comprehensive information about a selected product.

### ***Detailed Description:***

- Each product has a unique detail page, accessible via dynamic routes using its ID.
- The component displays detailed product information, including images, descriptions, prices, stock and status.
- Users can add the product directly to their cart or wishlist from this page.
- Data is fetched dynamically from Sanity CMS to keep the information accurate and up-to-date.



```


    | className= w-full h-full object-cover rounded-xl shadow-xl
    | />
  | })
  | </div>

  <div className="flex flex-col space-y-6">
    <h1 className="text-4xl font-semibold text-gray-900">{productName}</h1>
    <p className="text-lg text-gray-600">Category: {category}</p>
    <p className="text-3xl font-semibold text-black">${price}</p>
    <p
      | className={`text-sm font-medium ${status === 'In Stock' ? 'text-green-600' : 'text-red-600'}}`
    >
      | {status}
    </p>
    <p className="text-sm text-gray-600">Stock: {inventory}</p>
    <p className="text-sm text-gray-700 mt-4">{description}</p>

    <div className="space-y-4 mt-6">
      <button
        | onClick={handleAddToCart}
        | className="bg-black text-white w-full px-6 py-3 rounded-full hover:bg-gray-800 transition-all"
      >
        | {isAddedToBag ? 'Added to Bag' : 'Add to Bag'}
      </button>
      <button
        | onClick={handleWishlistToggle}
        | className="border border-gray-400 text-black w-full px-6 py-3 rounded-full hover:bg-gray-200 tra
      >
        | {isInWishlist(_id) ? (
        |   <FaHeart className="text-red-600 mr-2" />
        | ) : (
        |   <FaRegHeart className="text-gray-500 mr-2" />

```

Activate Windows  
Go to Settings to activate



## Nike Air Force 1 Mid '07

Category: Men's Shoes

**\$10795**

Just In

Stock: 20

The Nike Air Force 1 Mid '07 delivers timeless style with premium leather and mid-cut design. Perfect for everyday wear, it provides exceptional comfort and durability. The iconic Air-Sole cushioning adds responsive support for long-lasting performance.

Add to Bag

♥ Favorites

Activate Windows  
Go to Settings to activate Windows.

## 5. Checkout Process

The Checkout functionality streamlines the purchase process, collecting and validating user information to finalize the order.

### ***Detailed Description:***

- The checkout process is divided into multiple steps: billing details, shipping address, and payment information.
- A dynamic progress tracker indicates the current step, enhancing the user experience.
- Input validation ensures that all required fields are filled correctly, reducing errors during order submission.
- Order summaries are displayed at the end, allowing users to confirm their details before finalizing the purchase.

```

import { HiOutlineInboxArrowDown } from "react-icons/hi2"

const Formsection = () => {
  return (
    <div className="flex-1 md:p-2 lg:px-20">
      <h1 className="text-2xl font-medium py-4">How would you like to get your order?</h1>
      <p className="text-zinc-500 text-lg">
        Customs regulation for India require a copy of the recipient's KYC. The address on the KYC
      </p>
      <div className="w-full border-2 border-[#111] rounded-lg py-5 px-6 flex items-center gap-4 text-lg m">
        <HiOutlineInboxArrowDown className="text-2xl" />
        <span className="font-medium">Deliver it</span>
      </div>
      <h1 className="text-2xl font-medium py-4">Enter your name and address:</h1>

      <form action="" className="space-y-5 border-b pb-12">
        <input type="text" placeholder="First Name" className="w-full p-4 rounded-lg border-2 placeholde">
        <input type="text" placeholder="Last Name" className="w-full p-4 rounded-lg border-2 placeholder">
        <input type="text" placeholder="Address Line 1" className="w-full p-4 rounded-lg border-2 placeh">
        <input type="text" placeholder="Address Line 2" className="w-full p-4 rounded-lg border-2 placeh">
        <input type="text" placeholder="Address Line 3" className="w-full p-4 rounded-lg border-2 placeh">
        <div className="grid grid-cols-2 gap-5">
          <input type="text" placeholder="Postal Code" className="grid-span-1 w-full p-4 rounded-lg bo">
          <input type="text" placeholder="Locality" className="grid-span-1 w-full p-4 rounded-lg borde">
          <input type="text" placeholder="State/Territory" className="grid-span-1 w-full p-4 rounded-l">
          <input type="text" placeholder="Postal Code" className="grid-span-1 w-full p-4 rounded-lg bo">
        </div>
        <div className="space-y-5 py-6">
          <div className="flex items-center gap-4">
            <input type="checkbox" className="w-4 h-4" />
            <span>Save this address to my profile</span>
          </div>
        </div>
      </form>
    </div>
  )
}

```

Activate Windows  
 Go to Settings to activate

### How would you like to get your order?

Customs regulation for India require a copy of the recipient's KYC. The address on the KYC needs to match the shipping address. Our courier will contact you via SMS/email to obtain a copy of your KYC. The KYC will be stored securely and used solely for the purpose of clearing customs (including sharing it with customs officials) for all orders and returns. If your KYC does not match your shipping address, please click the link for more information. [Learn More](#)

 Deliver it

### Enter your name and address:

First Name

Last Name

Address Line 1

Address Line 2

Address Line 3

Postal Code

Locality

State/Territory

Postal Code

☐ Save this address to my profile

☐ Make this my preferred address

### Order Summary

Subtotal	\$0.00
Estimated Delivery & Handling	Free
<b>Total</b>	<b>\$0.00</b>

(The total reflects the price of your order, including all duties and taxes)

## 6. FAQ and Help Center Component

The FAQ and Help Center Component enhances user experience by providing quick access to common queries and support.

### ***Detailed Description:***

- Features a well-structured, searchable FAQ section with collapsible answers for easy navigation.
- Includes a contact form or support options for unresolved queries, directing users to customer support.
- Designed with a responsive layout, ensuring accessibility across devices.
- Improves user engagement and reduces support workload by addressing common concerns efficiently.

```
'use client';
import React, { useState } from "react";
import Link from "next/link";
import { AiFillLike, AiFillDislike } from "react-icons/ai";

const faqs = [
  {
    ques: "Does my card need international purchases enabled?",
    ans: "Yes, we recommend asking your bank to enable international purchases on your card. You will be notified.",
  },
  {
    ques: "Can I pay for my order with multiple methods?",
    ans: "No, payment for Nike orders can't be split between multiple payment methods."
  },
  {
    ques: "What payment method is accepted for SNKRS orders?",
    ans: "You can use any accepted credit card to pay for your SNKRS order."
  },
  {
    ques: "Why don't I see Apple Pay as an option?",
    ans: "To see Apple Pay as an option in the Nike App or on Nike.com, you'll need to use a compatible Apple device."
  },
  {
    ques: "Can I use a prepaid card to pay for my order?",
    ans: "Yes, prepaid cards are accepted as long as they meet the standard payment requirements."
  },
  {
    ques: "How do I know if my payment was successful?",
    ans: "You will receive a confirmation email once your payment has been successfully processed."
  },
  {
    ques: "Are my payment details secure?"
  }
];
```

Activate Windows  
Go to Settings to activate

## FAQs

Search FAQs...

### Does my card need international purchases enabled?

Yes, we recommend asking your bank to enable international purchases on your card. You will be notified at checkout if international purchases need to be enabled.

### Can I pay for my order with multiple methods?

No, payment for Nike orders can't be split between multiple payment methods.

### What payment method is accepted for SNKRS orders?

You can use any accepted credit card to pay for your SNKRS order.

### Why don't I see Apple Pay as an option?

To see Apple Pay as an option in the Nike App or on Nike.com, you'll need to use a compatible Apple device running the latest OS, be signed in to your iCloud account and have a supported card in your Wallet. Additionally, you'll need to use Safari to use Apple Pay on Nike.com.

### Can I use a prepaid card to pay for my order?

Yes, prepaid cards are accepted as long as they meet the standard payment requirements.

### How do I know if my payment was successful?

You will receive a confirmation email once your payment has been successfully processed.

### Are my payment details secure?

Yes, Nike ensures that your payment details are secure and encrypted during the transaction process.

### Can I change my payment method after placing an order?

No, payment methods cannot be changed after the order has been placed.

### Why was my payment declined?

Your payment may be declined for several reasons, including insufficient funds, incorrect card details, or restrictions set by your bank. Please contact your bank for more information.

### Can I use my Nike gift card online?

Yes, Nike gift cards can be used for online purchases. Enter the gift card details during checkout.

Was this answer helpful?



## Contact Us

Name

Enter your name

First Name



24 hours a day  
7 days a week



We'll reply within  
five business days



## STORE LOCATION

Find Nike retail stores near you

## 7. Footer and Header Components

The Footer and Header Components provide consistent navigation and essential information across the website.

## Header Component

- **Navigation Bar:** Contains links to major sections like Home, Shop, About, and Contact.
- **Search Bar:** Allows users to search for products or content seamlessly.
- **Dynamic Elements:** Displays user-specific options such as Login/Logout, Cart, and Wishlist count.
- **Responsive Design:** Ensures accessibility on all devices, including mobile and tablet screens.

```
import Image from "next/image";
import Link from "next/link";

const Headertop = () => {
  return (
    <div className="w-full sm:px-10 px-2 py-2 bg-[#F5F5F5] flex justify-between items-center">
      <div className="w-6 h-6">
        <Image
          src={"/logo2.png"}
          alt=""
          width={700}
          height={700}
          className="w-full h-full object-contain"
        />
      </div>
      <div className="flex justify-center items-center gap-4 text-xs sm:text-sm font-medium">
        <Link href={"/"}>Find a Store</Link> |
        <Link href={"/contact"}>Help</Link> |
        <Link href={"/joinus"}>Join Us</Link> |
        <Link href={"/signin"}>Sign In</Link>
      </div>
    </div>
  );
};

export default Headertop;
```

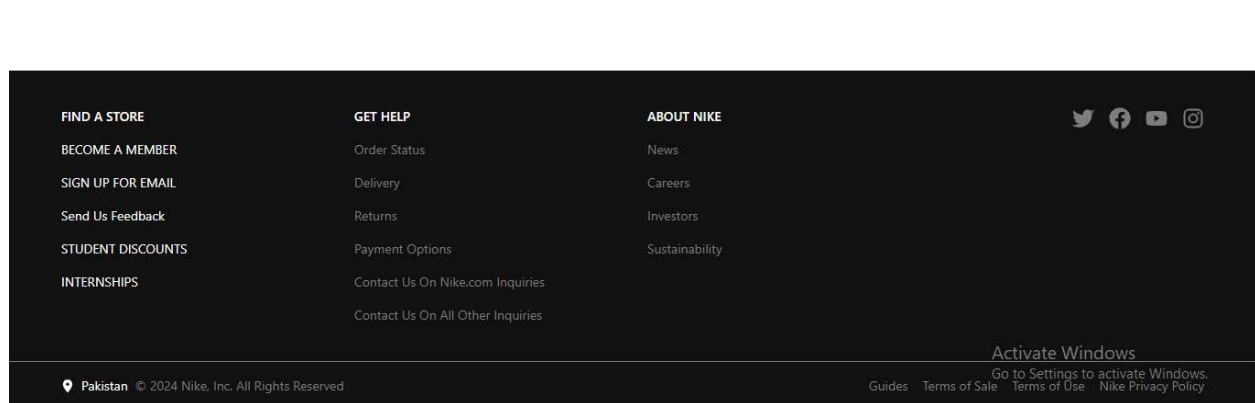






```
Footer = () => {
  turn (
    <footer className="bg-[#111] text-[#7E7E7E] text-sm pt-10">
      <div className="max-w-7xl mx-auto px-4 grid grid-cols-2 md:grid-cols-4 gap-8">
        {/* Column 1 */}
        <div>
          <h3 className="font-semibold mb-4 text-[#FFFFFF]">FIND A STORE</h3>
          <ul className="space-y-4 text-[#FFFFFF]">
            <li><Link href="/membership" className="hover:underline">BECOME A MEMBER</Link></li>
            <li><Link href="/" className="hover:underline">SIGN UP FOR EMAIL</Link></li>
            <li><Link href="/contact" className="hover:underline">Send Us Feedback</Link></li>
            <li><Link href="/" className="hover:underline">STUDENT DISCOUNTS</Link></li>
            <li><Link href="/internship" className="hover:underline">INTERNSHIPS</Link></li>
          </ul>
        </div>

        {/* Column 2 */}
        <div>
          <h3 className="font-semibold mb-4 text-[#FFFFFF]">GET HELP</h3>
          <ul className="space-y-4">
            <li><Link href="/" className="hover:underline">Order Status</Link></li>
            <li><Link href="/" className="hover:underline">Delivery</Link></li>
            <li><Link href="/" className="hover:underline">Returns</Link></li>
            <li><Link href="/" className="hover:underline">Payment Options</Link></li>
            <li><Link href="/" className="hover:underline">Contact Us On Nike.com Inquiries</Link></li>
            <li><Link href="/" className="hover:underline">Contact Us On All Other Inquiries</Link></li>
          </ul>
        </div>
      </div>
    </footer>
  )
}
```

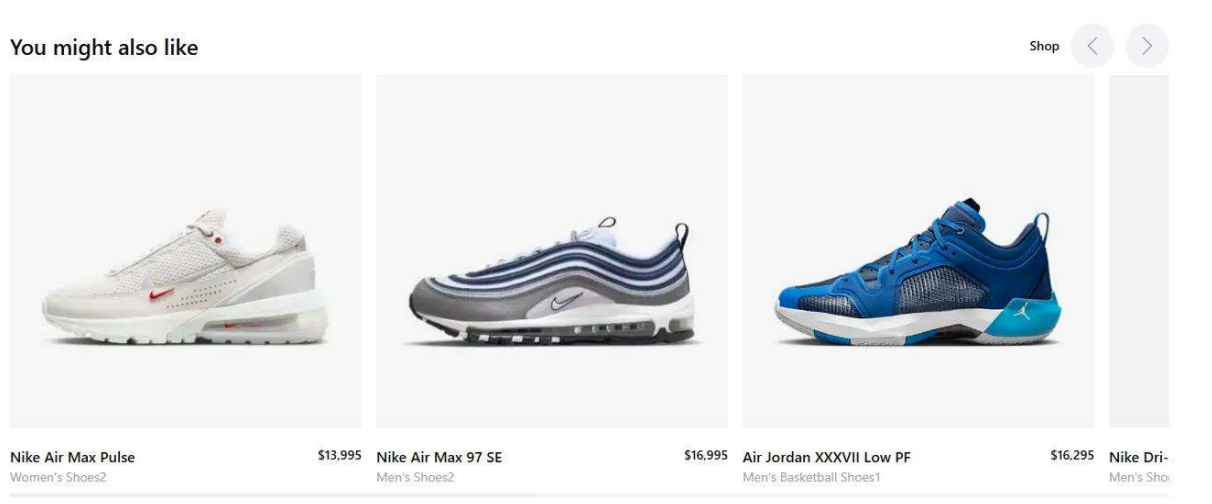


## 8. Related Products Component

The Related Products Component displays items similar to the product currently being viewed to enhance user engagement.

### ***Detailed Description:***

- Fetches related products based on categories, tags, or user browsing behavior.
- Ensures accurate and real-time suggestions.
- Displays products in a carousel or grid format.
- Allows users to preview, add to cart, or wishlist directly.
- Optimized for various screen sizes.

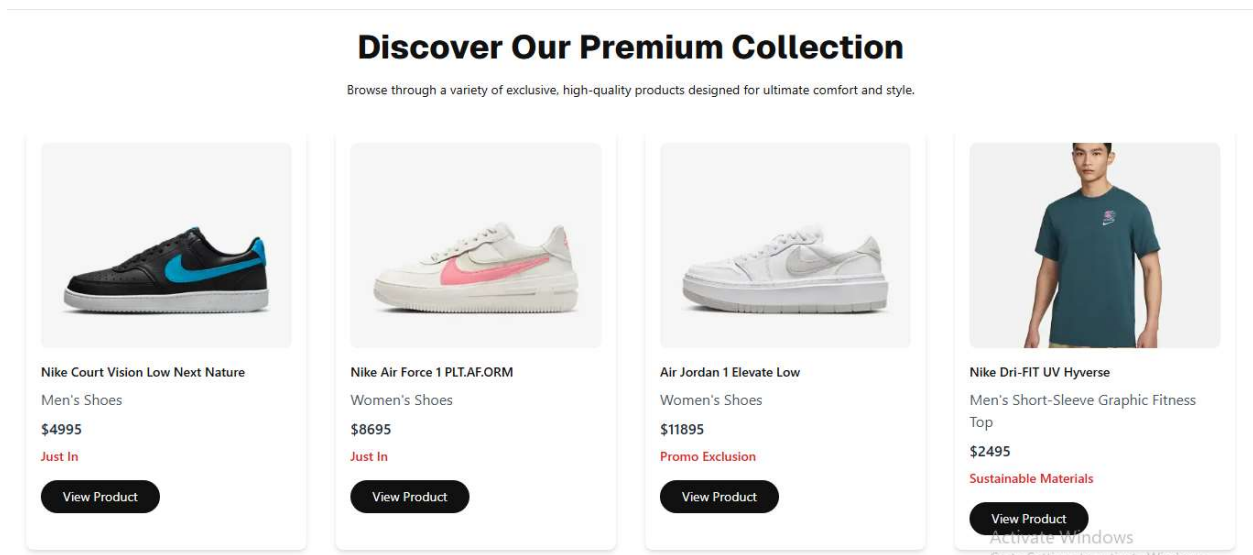


## **9. Product Listing Page**

The Product Listing Page provides a catalog view of all available products, allowing users to browse and filter items effortlessly.

### **Detailed Description:**

- Retrieves product data from Sanity CMS, ensuring updated content.
- Enables sorting by price, category, ratings, or availability.
- **Interactive Cards:** Displays product images, prices and categories.
- Ensures a seamless experience across desktop and mobile devices.



## **Conclusion**

Each feature is designed to create a responsive, user-centric, and scalable marketplace. The integration with Sanity CMS allows dynamic content management while maintaining modular and reusable components for streamlined development.