

Day:4 Dynamic Components and Functionalities

This documentation provides an in depth analysis of the key functionalities for a dynamic marketplace, emphasizing modularity, reusability, and integration with Sanity CMS. Each feature is described comprehensively, followed by a conclusion summarizing the approach.

Step 1: Functionalities Overview

The project implements the following core functionalities:

1. Product Listing Page
2. Dynamic Route
3. Cart Functionality
4. Checkout
5. Filter
6. Search
7. Price Calculation

Each functionality contributes to building a responsive and scalable marketplace.

Step 2: Functionalities in Detail

ProductListing Page

The Product Listing Page is the primary interface where users can view all the available products in a structured and visually appealing format. Products are displayed dynamically, fetched from Sanity CMS, and rendered in a grid or list layout.

The screenshot shows a website for 'Bandage' with a dark blue header. The header includes a logo, navigation links for Home, Shop, About, Pricing, Contact, Team, a search bar, and a shopping cart icon with 0 items.

The main content area displays a grid of products:

- Top Row:**
 - Classic Black Long Sleeve Button-Down Shirt**: SHIRT, \$199.00
 - Gray Slim-Fit Jogger Pants**: JEANS, \$179.25
 - Vertical Striped Shirt**: SHIRT, \$229.114.50
 - Black Striped T-Shirt**: TSHIRT, \$129.00
- Second Row:**
 - Skinny Fit Jeans**
 - Black Athletic Jogger Pants with Side Stripes**: PANTS
 - Classic White Pullover Hoodie**: HOODIE
 - Classic Black Pullover Hoodie**: HOODIE

A URL in the address bar indicates the current page: <localhost:3000/shop/hoodie/classic-black-pullover-hoodie>.

Detailed Description:

The page offers sorting and filtering options to enhance usability, allowing users to organize products based on price, categories, or popularity.

Pagination ensures the seamless handling of large datasets, improving performance and user experience.

Responsive design ensures compatibility across devices, from desktops to mobile phones.

- Integration with Sanity CMS ensures real-time updates, so any product changes in the backend are instantly reflected.

2. Dynamic Route

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

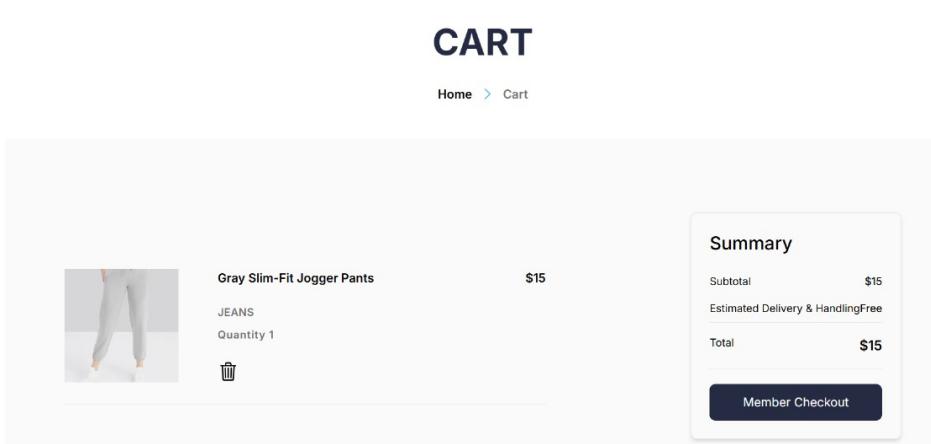


Detailed Description:

- 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.

3. Cart Functionality

The Cart Functionality manages the user's selected items, providing a seamless shopping experience by tracking their choices and summarizing costs.



Detailed Description:

- Users can add products to their cart directly from the product listing or detail page.
- The cart dynamically updates quantities and calculates the total cost, ensuring a real-time experience.
- A mini-cart displays a quick summary of selected items, while a detailed cart page offers options to edit or remove items.
- State management tools, such as React Context or Redux, are used to maintain the cart state across the application.
- Cart data persistence is achieved using local storage or session storage, ensuring the cart remains intact even if the page is refreshed.

4. Checkout

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

The screenshot shows a checkout process for a website called "Bandage". The top navigation bar includes links for Home, Shop, About, Pricing, Contact, Team, a search bar, and icons for a shopping cart and heart.

Shipping Method

Standard Delivery Rs 200.00

Payment Options

All transactions are secure and encrypted.

Cash on Delivery (COD)

Bank Details

Bank Name: HBL
Account Title: Bandage
Account Number: 12345678910
IBAN Number: PK12345678910

Billing Address

Same as shipping address

Complete 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.
- Although payment integration can be mocked initially, it is designed to be extendable with payment gateways like Stripe or PayPal.
- Order summaries are displayed at the end, allowing users to confirm their details before finalizing the purchase.

5. Price Calculation

Price Calculation dynamically computes the total cost of items in the cart, factoring in taxes, discounts, and other adjustments.

```

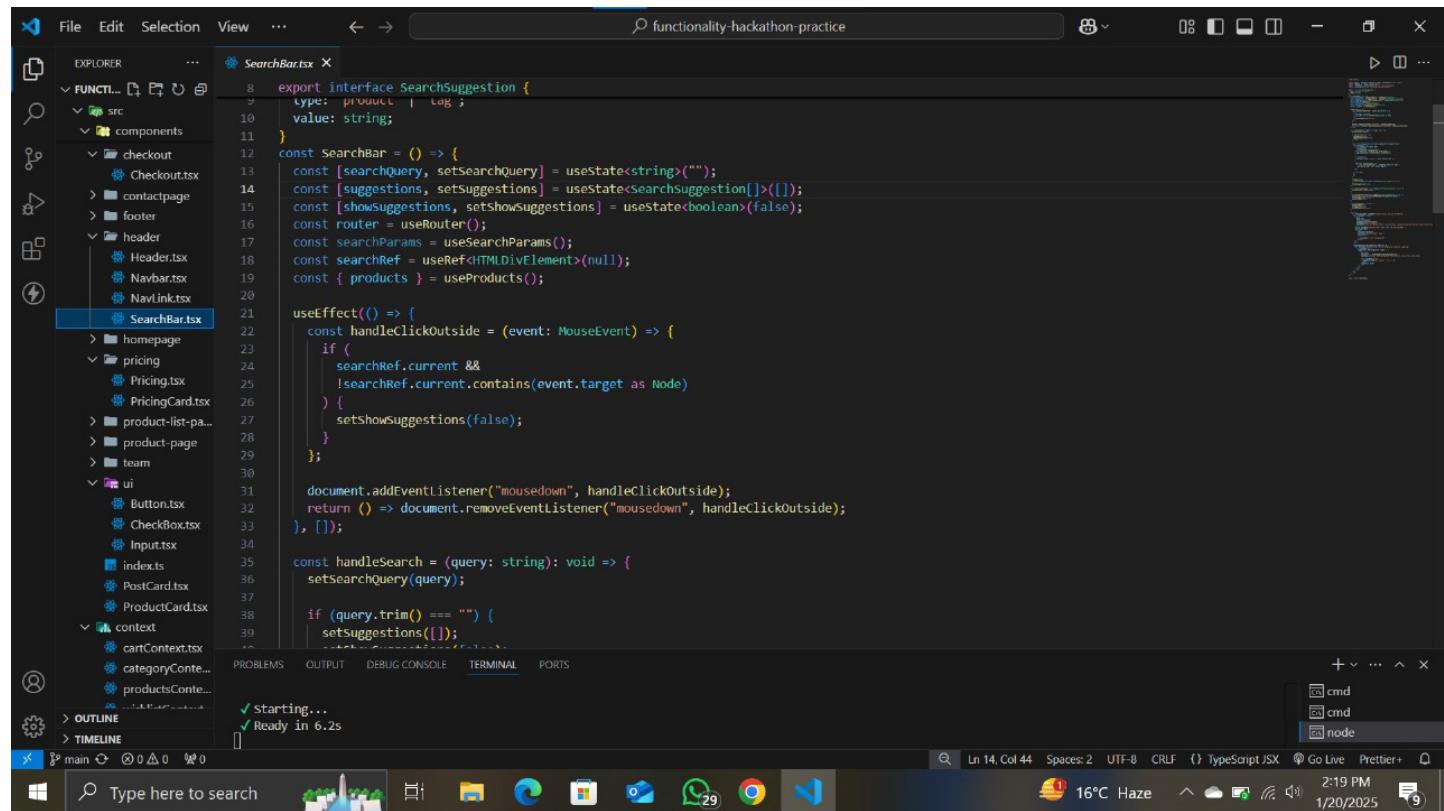
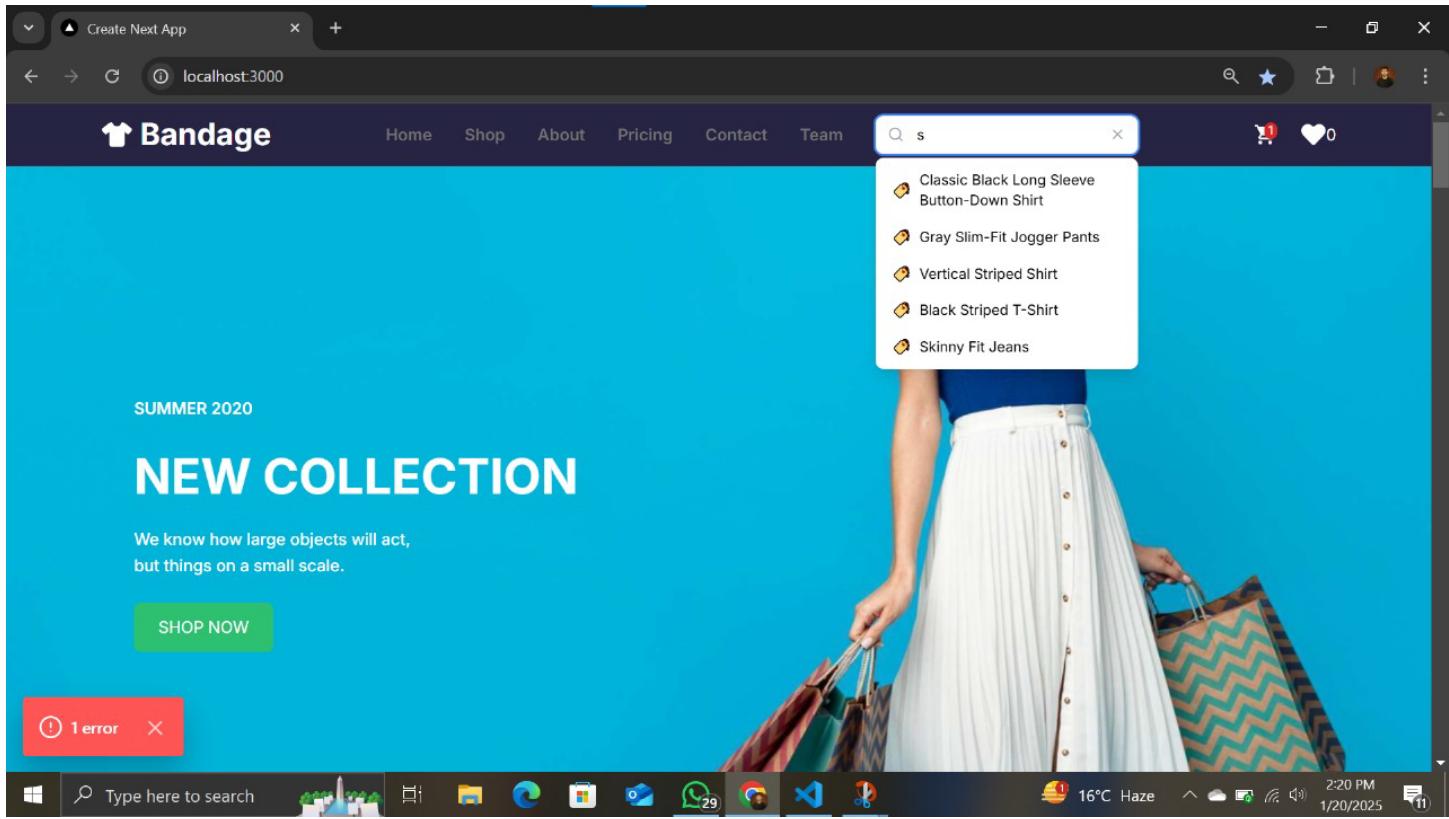
 7 const Page = () => {
 8
 9   if (cartItems.length > 0) {
10     <div className="bg-[#fafafa] py-24 w-[90%] mx-auto flex flex-col md:flex-row justify-center items-center gap-12 md:gap-12 lg:gap-36 xl:gap-48">
11       /* cart items */
12       <div className="flex justify-center items-center flex-col gap-6">
13         {cartItems.map(item: CartItem) => (
14           <CartCard key={item.id} item={item} />
15         )));
16       </div>
17       /* summary */
18       {cartItems && (
19         <div className="border rounded-lg p-6 shadow-md">
20           <h4 className="text-2xl font-medium mb-4">Summary</h4>
21           <div className="mt-3 text-sm">
22             <div className="flex justify-between py-2">
23               <span>Subtotal</span>
24               <span className="font-medium">${cartTotal}</span>
25             </div>
26             <div className="flex justify-between py-2">
27               <span>Estimated Delivery & Handling:</span>
28               <span>Free</span>
29             </div>
30             <div className="flex justify-between py-4 border-t border-b border-gray-200">
31               <span>Total:</span>
32               <span className="text-lg font-semibold">${cartTotal}</span>
33             </div>
34           </div>
35           <a href="/checkout" className="block w-full mt-5 px-4 py-3 text-center text-white bg-myHeading rounded-lg hover:bg-myBlue-dark transition-all">
36             Check Out
37           </a>
38         </div>
39       )};
40     );
41   );
42   </div>
43 }
44 
```

Detailed Description:

The subtotal updates in real adjusted.

- Taxes and discounts are calculated dynamically based on predefined rules, offering flexibility to apply promotional codes.
- The calculation logic is optimized to handle multiple scenarios, such as bulk discounts or tiered pricing.
- This functionality improves transparency by breaking down costs, giving users a clear understanding of the final price.

6. Search



Parepared by Muzammil Bukhari

7. Filter

The screenshot shows a web browser window with the URL `localhost:3000/shop`. The page is titled "Bandage" and features a navigation bar with links for Home, Shop, About, Pricing, Contact, Team, and a search bar. A shopping cart icon shows 0 items, and a heart icon shows 0 likes.

The main content area displays five categories of products: SHORT (4 items), TSHIRT (6 items), JEANS (9 items), SHIRT (6 items), and HOODIE (6 items). Below these categories, a message says "Showing all 17 results". To the right, there is a "Popularity" dropdown menu with options: "Popularity", "Price: Low to High", "Price: High to Low", and "Newest First". A blue "Filter" button is located next to the popularity dropdown.

Product thumbnails are shown below the categories:

- A dark long-sleeved shirt.
- A pair of white pants.
- A green striped button-down shirt.
- A dark long-sleeved shirt.
- A dark hoodie.

Conclusion

This documentation outlines a comprehensive approach to building dynamic and responsive marketplace components. By leveraging Sanity CMS for backend management and modular frontend development techniques, the application achieves scalability, efficiency, and a superior user experience.

Each functionality—from product listing to inventory management—plays a vital role in delivering a professional marketplace that meets real-world needs. Future enhancements, such as integrating advanced analytics or AI-based recommendations, can further elevate the platform.

For any additional details, enhancements, or implementation support, feel free to reach out!