

Day 4: Dynamic Frontend Components

Technical Report: Dynamic Product Display with Add to Cart and Checkout Functionality

1. Overview: This report outlines the implementation of a dynamic product display system integrated with an "Add to Cart" and "Checkout" functionality. The system is designed to showcase products dynamically, allow users to view detailed product information, and facilitate a seamless shopping experience.

2. Features:

2.1 Dynamic Product Display:

Products are fetched dynamically from a backend (e.g., Sanity, or a custom API).

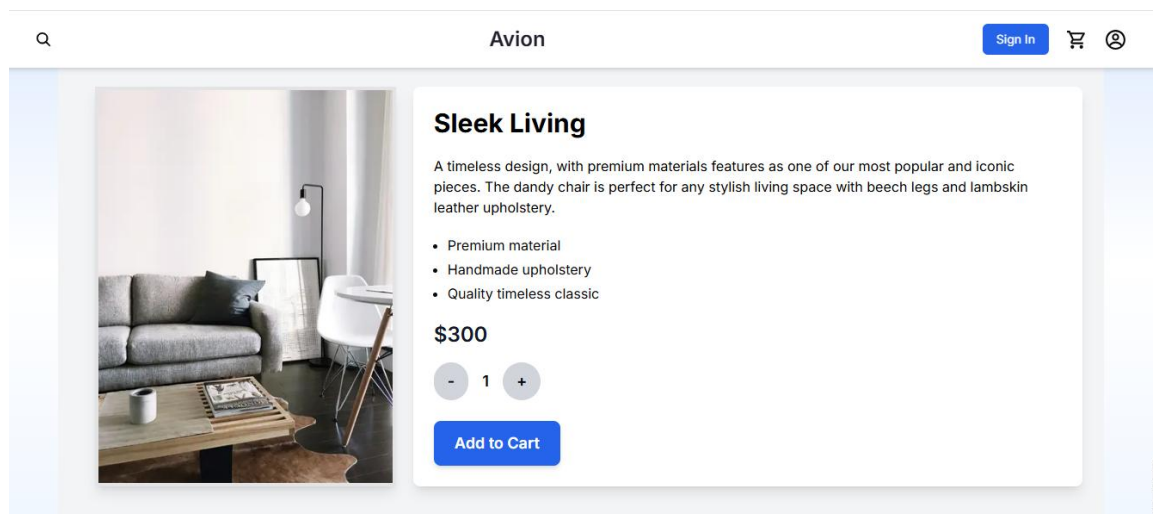
The product grid includes details such as:

- Product Image
- Name
- Price
- Add to Cart (Button)
- Stock
- Rating

2.2 Product Details Page:

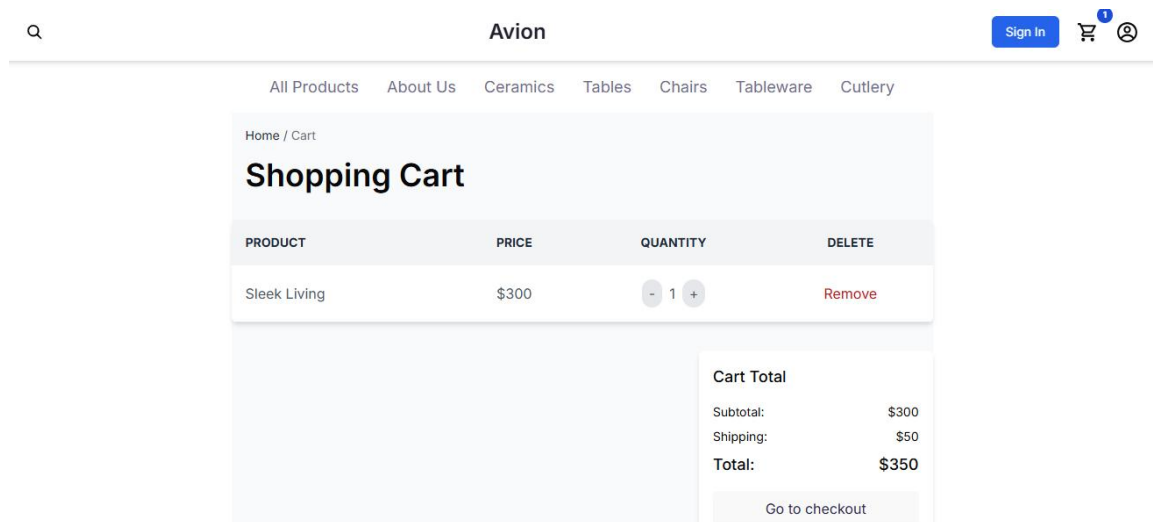
The detailed view includes:

- High-resolution product image(s)
- Description
- Price
- Add to Cart button



2.3 Add to Cart Functionality: (component)

Users can add products to the cart by clicking the "Add to Cart" button on the product details page.



2.4 Checkout Component:

The checkout process includes:

- A form for user details (e.g., name, address, payment information)

- Summary of selected products with their prices

- A "Place Order" button

The screenshot shows the 'Checkout' form. It is divided into two main sections: 'Billing Information' and 'Order Summary'. The 'Billing Information' section contains fields for 'Full Name', 'Email Address', 'Shipping Address', 'City', 'Postal Code', and 'Country' (a dropdown menu with 'Pakistan' selected). The 'Order Summary' section shows 'Sleek Living' with a quantity of '1' and a 'Total: \$ 350'. A blue 'Place Order' button is located at the bottom right of the form.

3. Implementation Details

3.1 Technology Stack:

- Frontend: React (Next.js) with TypeScript

- Backend: Sanity for data management

- Data Management: Sanity

- Data Fetching: Mock Api

3.2 Folder Structure:

src/app/

product/: Displays the product grid dynamically

[id]/: Handles individual product details

cart/: Manages the cart functionality and integrates the *Checkout* component

Checkout: Handles the checkout form and order placement

products: all products data display import from sanity through fetching using groc query

components/

ProductDetails: Handles detailed product view

productListing: *displays a product card with an image, name, and price.*

3.3 Dynamic Dropdown:

Dropdown menus are implemented using React state and hooks.

Each dropdown dynamically populates options such as product categories and brands based on the fetched data.

4. Current Work and Future Enhancements

4.1 Ongoing Tasks:

Integrating and testing the dropdown functionality.

Adding functionality for filtering and sorting by category and price.

Ensuring proper validation of the checkout form.

Adding Authentication using clerkjs login/signup for user.

Adding payment integration (stripe).

. Ship-engine for tracking orders.

4.2 Planned Features:

Search functionality to filter products by name or category.

Recommendations based on user preferences.

Multi-language support.

Enhanced alert modals for a better user experience.

5. Conclusion: This system ensures a dynamic and user-friendly shopping experience.