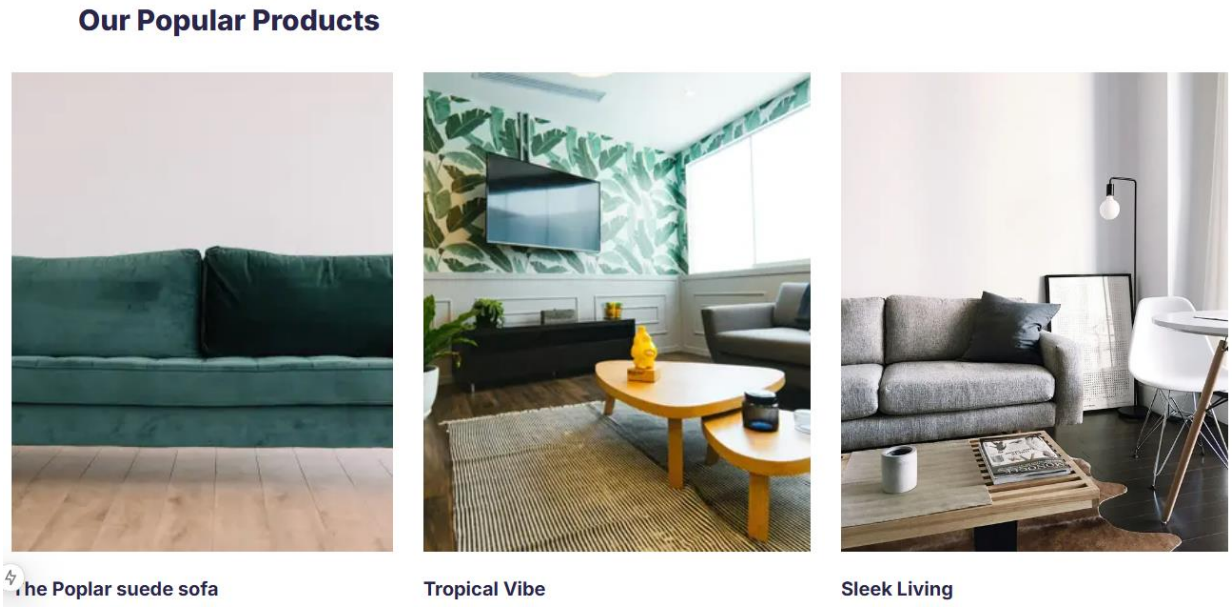
Day 4 - Dynamic Frontend Components | Avion

On **Day 4** of the **Hackathon**, I focused on enhancing the frontend by implementing dynamic components to improve the user experience.

Key Components Developed:

✅ **Product Listing Component**
I built a dynamic **Product Listing Component** that efficiently renders product data in a structured grid layout. This component displays:

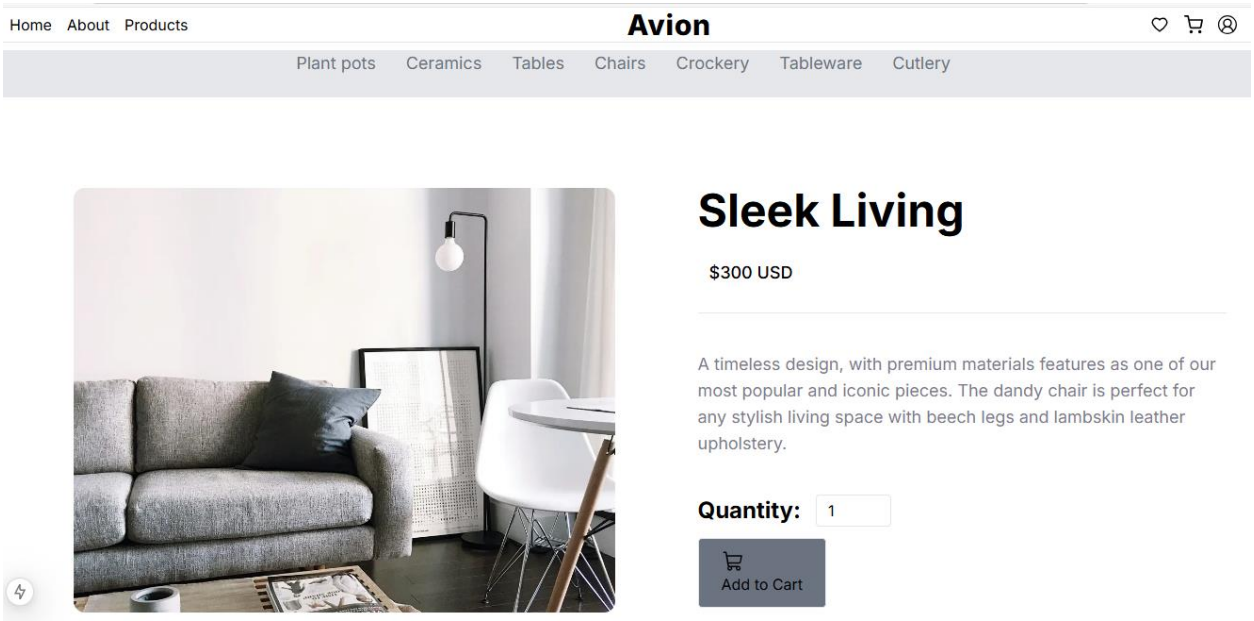
- **Product Name:** Clearly presents the name of each product.
- **Price:** Showcases the cost of the product.
- **Image:** Provides a visual representation to help users easily identify products.



✅ **Product Detail Component**

Each product page provides comprehensive details about a specific product. The component is designed to display:

- **Product Description:** A detailed overview highlighting the features and specifications of the product.
- **Price:** Clearly states the cost of the product to assist with purchasing decisions.
- **Dynamic Data Fetching:** Implemented using **Sanity CMS API**, ensuring real-time updates.
- **Unique Slug Identifier:** Each product has a unique slug, dynamically fetching and displaying accurate product details on the corresponding page.



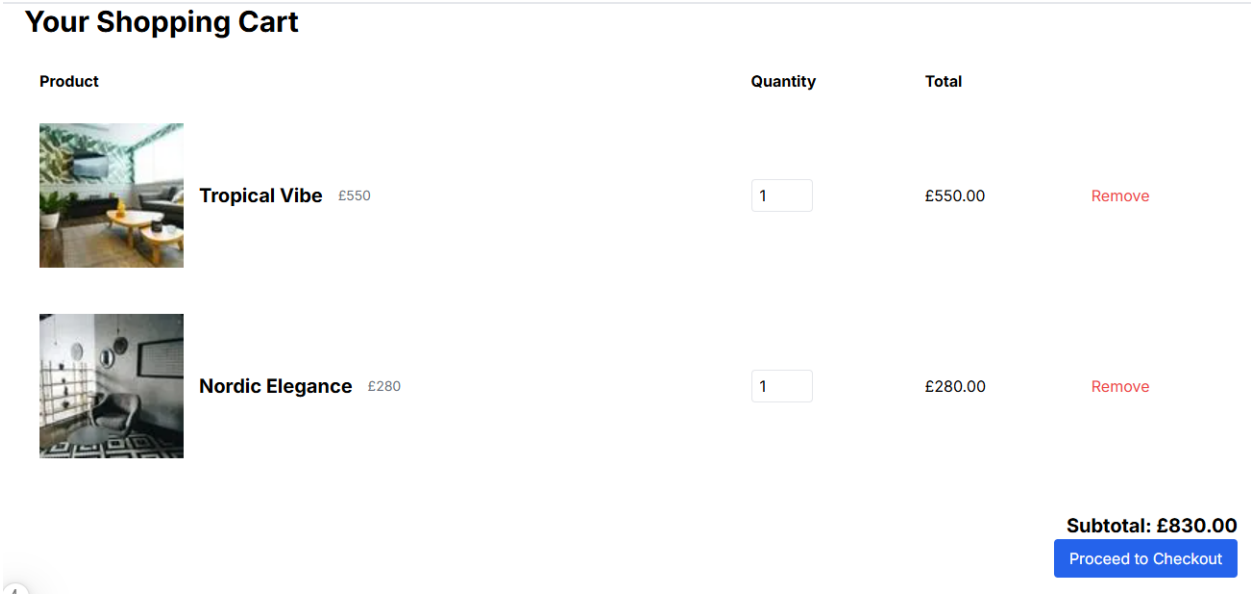
✅ Cart Component The Cart Component provides users with a seamless shopping experience by dynamically managing their selected products. It includes:

Items Added: Displays the products that users have added to their cart.

Quantity: Shows the number of units of each product.

Total Price: Automatically updates the total cost based on quantity changes.

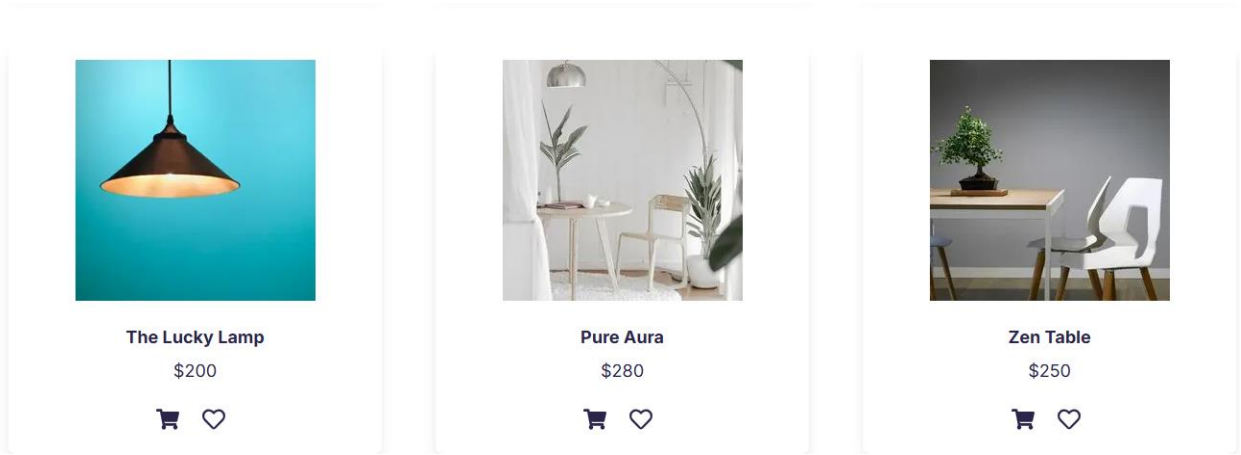
State Management: Utilizes dynamic state tracking to ensure real-time updates, making the shopping process smooth and user-friendly.



✅ Pagination Component

To enhance usability and navigation, I developed a **Pagination Component** that efficiently manages large product listings by breaking them into smaller, more digestible pages. Key features include:

- **Previous & Next Buttons:** Enable smooth navigation between pages.
- **Numbered Pagination:** Allows users to jump directly to a specific page for quick access.
- **Improved User Experience:** Ensures users are not overwhelmed by an extensive product catalog on a single page.



✅ Header & Footer Components

To ensure a consistent user experience across all pages, I developed **Header and Footer Components** with seamless navigation and accessibility.

- **Header Component:** Features essential navigation links, allowing users to quickly access key pages such as **Home, About, and Products**.
- **Footer Component:** Includes additional information, relevant links, and ensures a unified browsing experience.
- **Fully Responsive Design:** Adapts seamlessly to various screen sizes and devices.
- **Accessibility-Focused:** Follows web design best practices to ensure usability for all users.

✅ Related Products Component

To improve product discovery, I integrated a **Related Products Component** that displays similar products below the **Product Details Page**.

- **Enhanced Navigation:** Users can quickly find and explore related products.
- **Quick Access:** Clicking on any related product directs users to its detailed page.
- **Personalized Experience:** Helps users discover items based on their interests.

✅ Toast Notification Integration

To provide instant feedback and improve user interaction, I implemented a **Toast Notification System** that triggers alerts when:

- A user **adds a product to the cart**.
- The cart is **updated** with new items.
- A successful action is performed, displaying confirmation messages like "Added to Cart."

✅ Wishlist Component

I developed a **Wishlist Component** that enables users to save their favorite products for future reference.

- **Items Saved:** Displays products added to the wishlist.
- **Product Details:** Shows key information such as price and description.
- **Add to Cart:** Allows users to move wishlist items to the cart with a single click.
- **State Management:** Dynamically tracks saved items, enhancing the shopping experience.

✅ Checkout Flow Component

I designed a seamless **Checkout Flow Component** to simplify the purchasing process.

- **Multi-Step Form:** Collects customer details, billing/shipping addresses, and payment information.
- **Conditional Steps:** Guides users through a structured, user-friendly checkout experience.
- **Order Summary Page:** Displays customer details, selected products, and payment method before order submission.
- **Cart Validation:** Ensures users have items in their cart before proceeding to checkout.

Technical Report Summary

📌 **Steps Taken to Build and Integrate Components:**

- Designed components based on project requirements.
- Integrated components while maintaining a structured layout.
- Ensured functionality and responsiveness before proceeding to the next phase.
- Fetched real-time data from **Sanity CMS** using its API for dynamic updates.

📌 **Challenges & Solutions:**

- **Responsiveness Across Devices:** Used **Tailwind CSS media queries** to ensure a mobile-friendly design.
- **State Management Across Components:** Implemented **useState** and **useContext** for efficient data tracking.
- **Dynamic Product Data Fetching:** Used **Sanity CMS & GROQ queries** for optimal performance.
- **Unique Product Slug Handling:** Implemented **Next.js dynamic routing** to display correct product details.

📌 **Best Practices Followed:**

- Ensured **modular and reusable** code structure.
- Added **clear and concise comments** for better maintainability.
- Adhered to **accessibility standards** for an inclusive web experience.

✅ **Self-Validation Checklist for Day 4:**

✓ Frontend Component Development
✓ Styling and Responsiveness
✓ Code Quality
✓ Documentation & Submission