

Day 4 - Dynamic Frontend Components - [Avion]

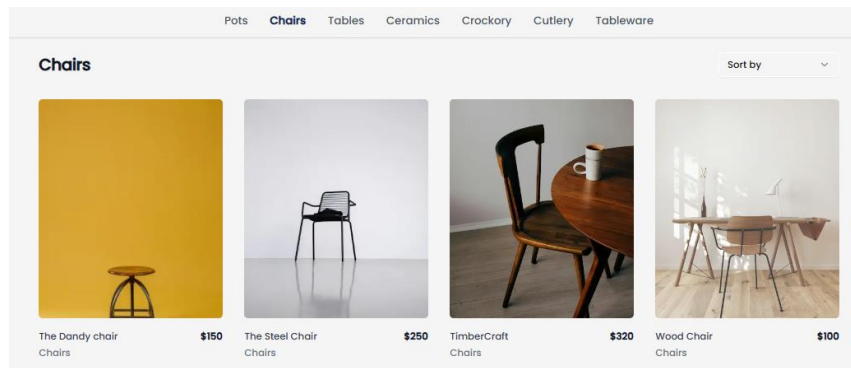
Prepared by: Hashir Hussain

1. Functional Deliverables

Screenshots/Screen Recordings

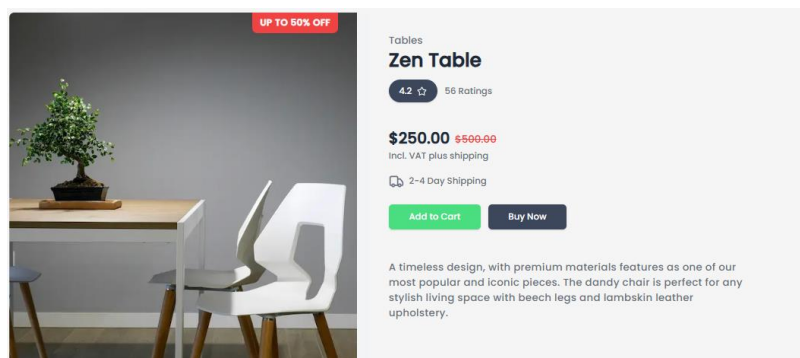
1.1 Product Listing Page with Dynamic Data

The product listing page displays products fetched dynamically from the backend, showing details such as name, price, and image. The page updates based on selected filters and the search bar.



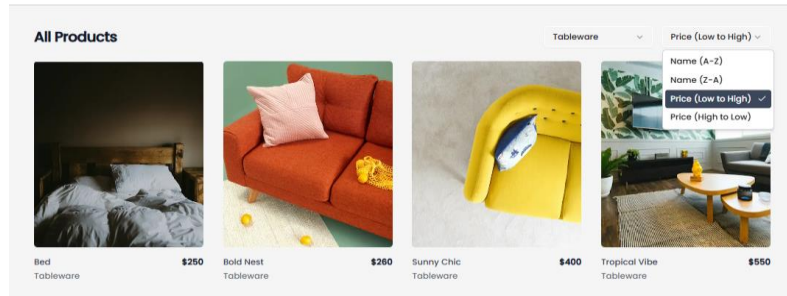
1.2 Individual Product Detail Pages with Accurate Routing and Data Rendering

The individual product detail page is displayed when a user clicks on a product card. This page shows detailed information, including the product's name, price, description, and image. It also uses dynamic routing based on the `slug` or unique identifier of the product.

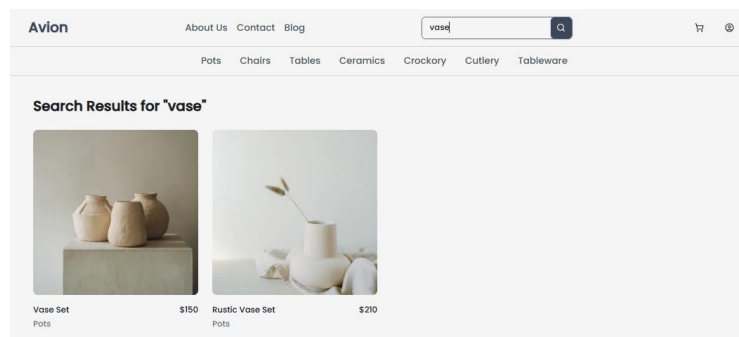


1.3 Working Category Filters, Search Bar:

- **Category Filters:** Users can filter products based on their categories (e.g., Chairs, Pots). and price ranges

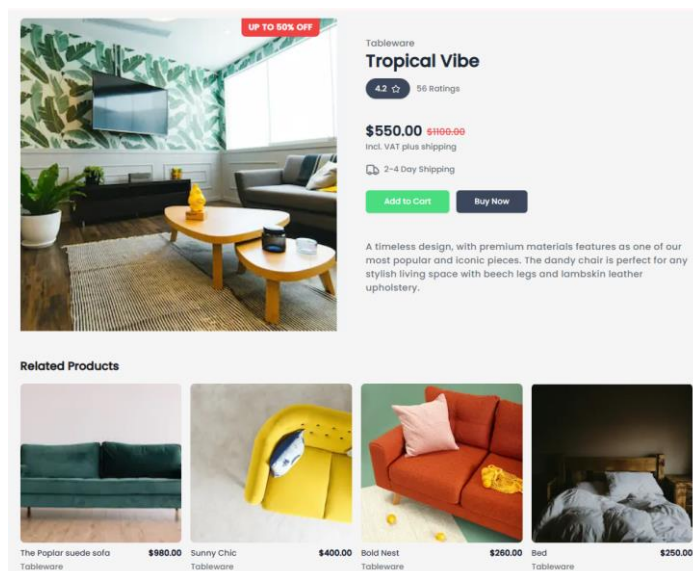


- **Search Bar:** The search bar allows users to search for products by name or category.



1.4 Related Products Section

Under each product detail page, a "Related Products" section dynamically displays other products from the same category, fetched based on the category of the current product.



3. Documentation

Steps Taken to Build and Integrate Components

1. **Dynamic Product Pages:** Implemented dynamic product detail pages using Next.js dynamic routing. Each product page is fetched using its unique `slug`, with related product information based on category.
2. **Product Category:** Created a category page that fetches products from the Sanity CMS. Displaying products in a grid layout, making use of dynamic data fetching for each page.
3. **Search and Filter:** Added a search bar to the product listing page that allows users to search for products by name. Filters were integrated for categories, helping users narrow down their options.
4. **Related Products:** Created a "Related Products" section on the product detail page that displays other products in the same category. This section fetches data dynamically using the category information from the current product.

Challenges Faced and Solutions Implemented

1. **Dynamic Data Fetching:** Initially faced issues with fetching and displaying dynamic data from Sanity. This was resolved by ensuring the correct API queries were used, with proper filtering.
2. **Routing and Slug Management:** Encountered routing challenges when linking to product detail pages. The solution was to correctly implement dynamic routes in Next.js and ensure that slugs were unique.

Self-Validation Checklist: E-Commerce Marketplace Development

1. Code Quality

- ✓ Clean Code Practices
 - ✓ Reusability and Modularity
 - ✓ Code Comments and Documentation
 - ✓ Avoid Code Duplication
 - ✓ Error Handling
 - ✓ Consistent Styling
-

2. Frontend Components

- ✓ Product Listing Page
 - ✓ Product Detail Page
 - ✓ Search Functionality
 - ✓ Category Filters
 - ✓ Pagination
 - ✓ Responsive Design
-

3. Dynamic Data Fetching

- ✓ API Integration
 - ✓ State Management for Dynamic Data
 - ✓ Dynamic Routing
 - ✓ Performance Optimization
-
-

4. Documentation

- ✓ README File
- ✓ Code Documentation
- ✓ Technical Report