

# HACKATHON: 03

## DAY: 04

QAHIR ISHAQ

00460710

## Dynamic Fronted Components

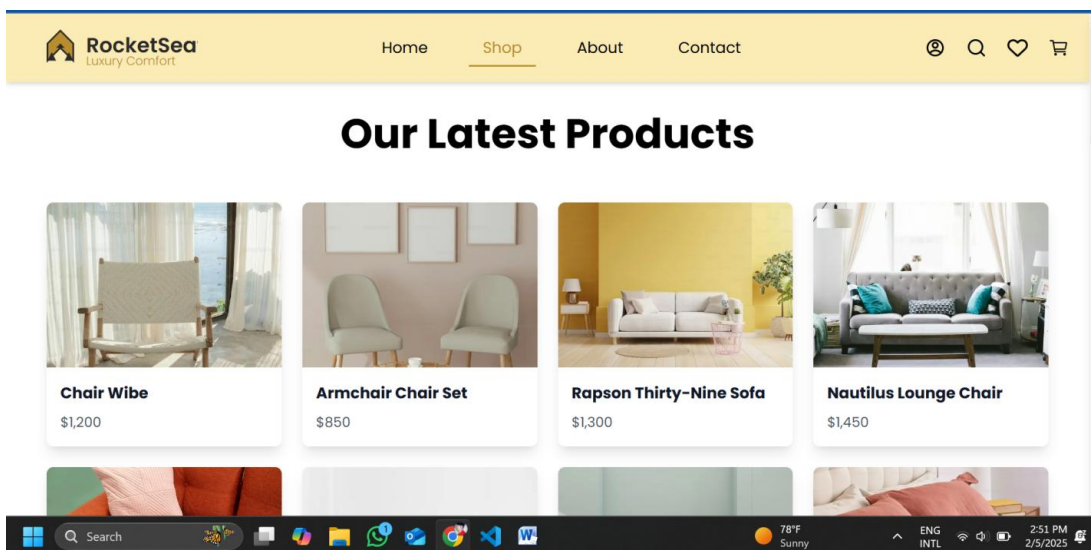
### 1. Introduction

This document gives a simple overview of the interactive features created for a furniture marketplace. The goal is to make it easy for users to browse, select, and buy products. Important features include a product list, individual product pages, user signup/login, a shopping cart, checkout, wishlist, search bar, and more.

### 2. Functional Components

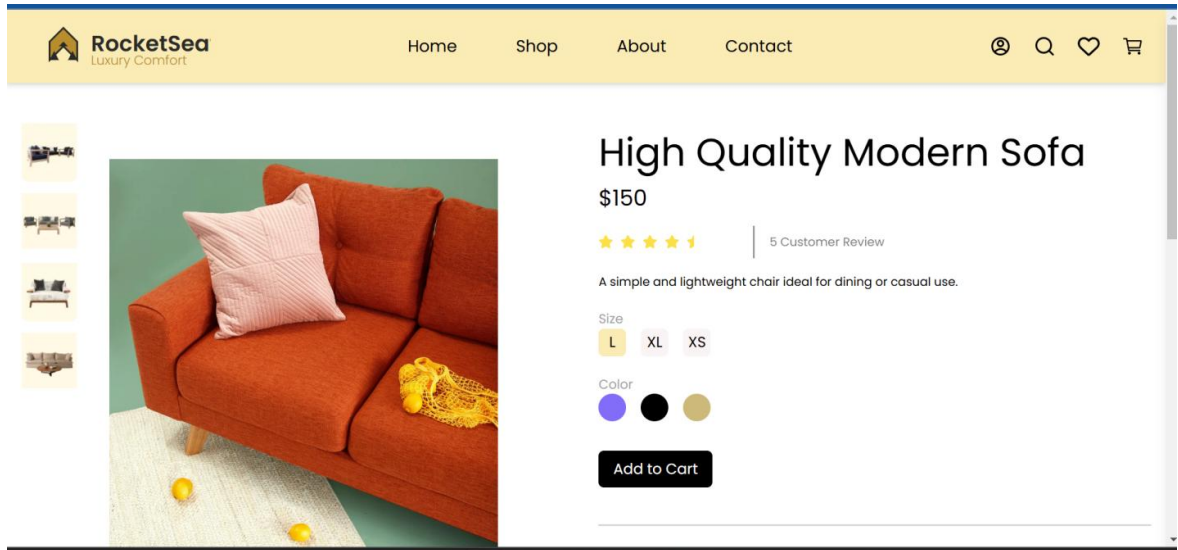
#### a. Product Catalog Page

- Showcases all available footwear dynamically, featuring images, titles, costs, and stock details.
- Merchandise is retrieved via an API, ensuring the latest inventory data is displayed.
- Users can refine their search by categories such as Chair wibe, Sofa etc.



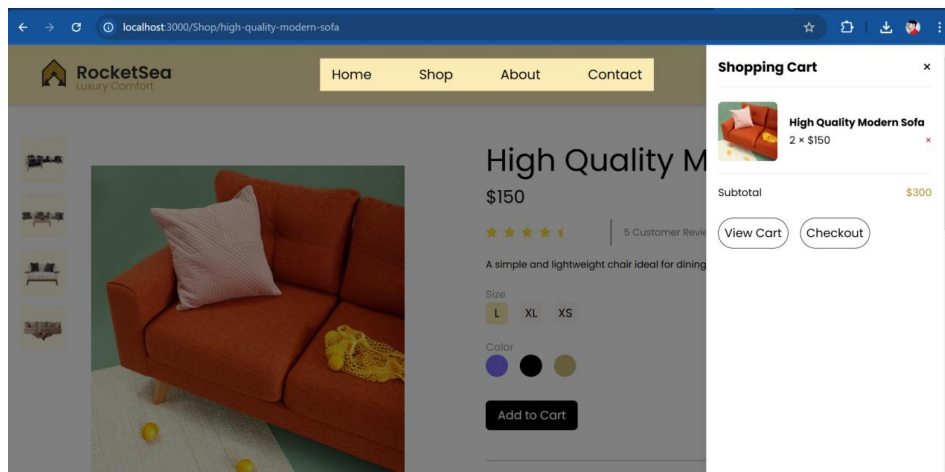
## b. Individual Product Information Pages

- Every item has a dedicated page displaying specifications, visuals, available sizes, and pricing.
- Next.js dynamic routing (`/product/[id]`) facilitates the automatic rendering of product details.
- Live data retrieval ensures customers always access the most current information.



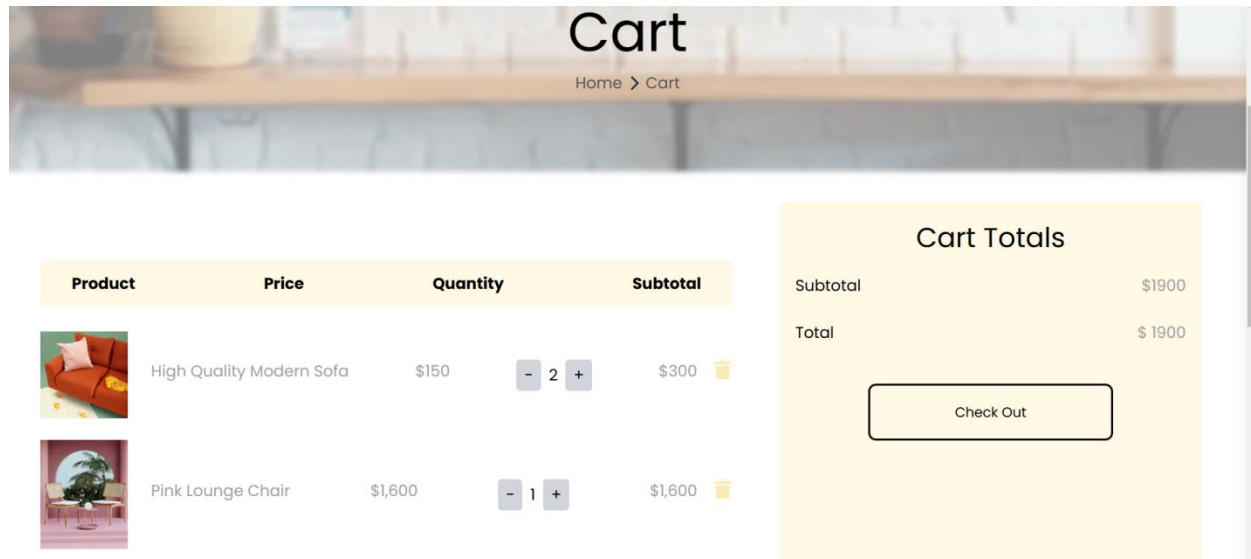
## c. Shopping Cart Page

- Lists products added by the user, offering options to adjust quantities or remove selections.
- Incorporates centralized state management using React Context API or similar tools.
- The overall cost updates in real-time based on cart modifications.



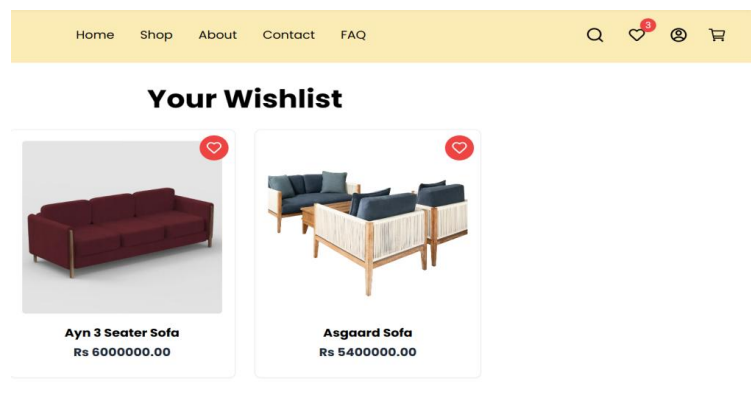
#### d. Order Completion Page

- Enables users to input delivery details, select a payment option, and review their purchase summary.
- Includes a structured form to gather user information and dynamically displays order specifics.
- Payment gateway integration ensures safe and seamless transactions.



#### e. Favorites Page

- Allows customers to bookmark products for future consideration.
- Supports adding and removing items while preserving data using local storage or a backend system.

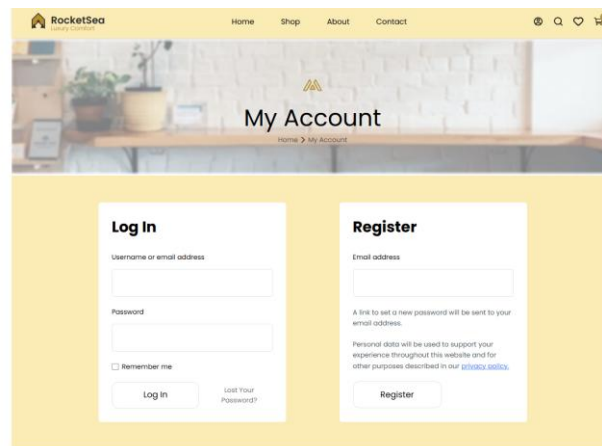


## f. Registration Page

- New users can create an account by submitting credentials.
- Features form validation and redirects users upon successful sign-up.

## g. Membership Invitation Page

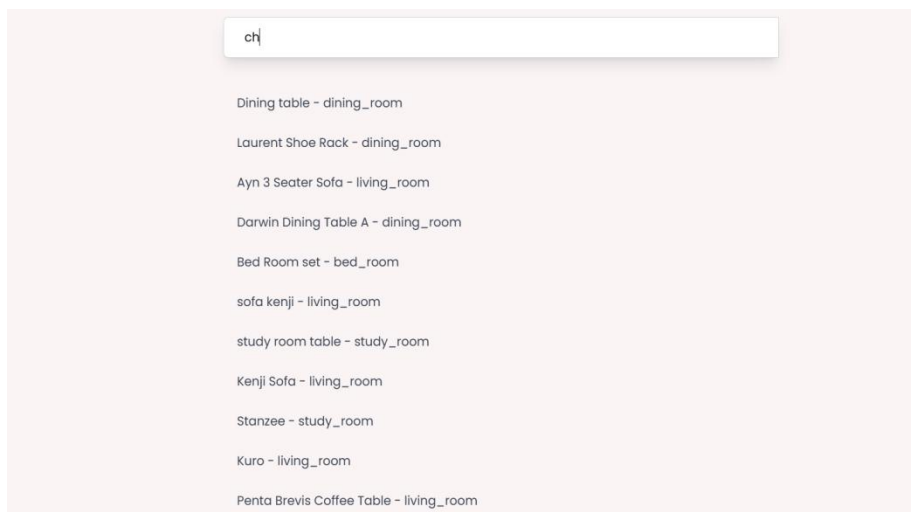
- Encourages visitors to join by showcasing platform advantages.
- Incorporates a registration form with live input validation.



The screenshot shows the 'My Account' page of the RocketSea website. The page has a yellow header with the RocketSea logo and navigation links: Home, Shop, About, and Contact. Below the header is a banner image of a desk with a plant and a laptop. The main content area is divided into two columns: 'Log In' and 'Register'. The 'Log In' form has fields for 'Username or email address' and 'Password', a 'Remember me' checkbox, and a 'Log In' button. The 'Register' form has an 'Email address' field, a 'Register' button, and a note about a password reset link. A link to the 'Privacy Policy' is also present.

## h. Search Functionality

- Enables users to look up furniture based on attributes such as name or category.
- Offers instant search results and predictive suggestions.



The screenshot shows the search functionality of the website. A search bar at the top contains the text 'ch'. Below the search bar, a list of search results is displayed, each consisting of a furniture item name followed by its category in parentheses. The results are: Dining table - dining\_room, Laurent Shoe Rack - dining\_room, Ayn 3 Seater Sofa - living\_room, Darwin Dining Table A - dining\_room, Bed Room set - bed\_room, sofa kenji - living\_room, study room table - study\_room, Kenji Sofa - living\_room, Stanzee - study\_room, Kuro - living\_room, and Penta Brevis Coffee Table - living\_room.

### 3. Challenges & Resolutions

- **Handling Product Pages:** It was tricky to load product pages correctly with the right information.  
**Solution:** Used Next.js to create dynamic product pages and fetch data from an API.
- **Managing Cart & Wishlist:** Keeping track of the cart and wishlist across different pages was difficult.  
**Solution:** Used React Context API and local storage to store cart and wishlist data.
- **Improving Speed:** The website needed to load faster.  
**Solution:** Used lazy loading and code splitting to make pages load quickly.

### 4. Conclusion

This initiative effectively incorporated interactive frontend modules for an engaging footwear marketplace. Key functionalities such as dynamic product showcasing, a streamlined checkout process, a wishlist, and a responsive search feature contributed to a high-performance and expandable eCommerce solution. Obstacles related to managing global state and retrieving data dynamically were mitigated through industry best practices, ensuring a smooth and intuitive user journey.