Day 4 - Dynamic Frontend Components for Furnniwise Marketplace

Prepared by: Khalid Hussain
RN; 372206
Project Repository: https://github.com/Khalid147-alt/Hackathon

---

1. Introduction

This document presents the implementation of dynamic frontend components for an online marketplace. The project is built using **Next.js**, and integrates features such as authentication, a product listing page, add-to-cart functionality, checkout, and UI components like a navbar and footer. The aim is to create a scalable, modular, and responsive UI that enhances user experience and interaction.

---

2. Project Overview

Technologies Used:

- **Framework:** Next.js
- **State Management:** React Context API / Redux
- **Styling:** Tailwind CSS / Styled Components
- **Backend Integration:** Sanity CMS
- **Authentication:** Firebase Clerk

Key Features Implemented:

✅ Navbar & Footer
✅ Hero Section
✅ Product Listing Page
✅ Add to Cart Functionality
✅ Authentication System
✅ Checkout Process

---

3. Implemented Components

3.1 Navbar & Footer

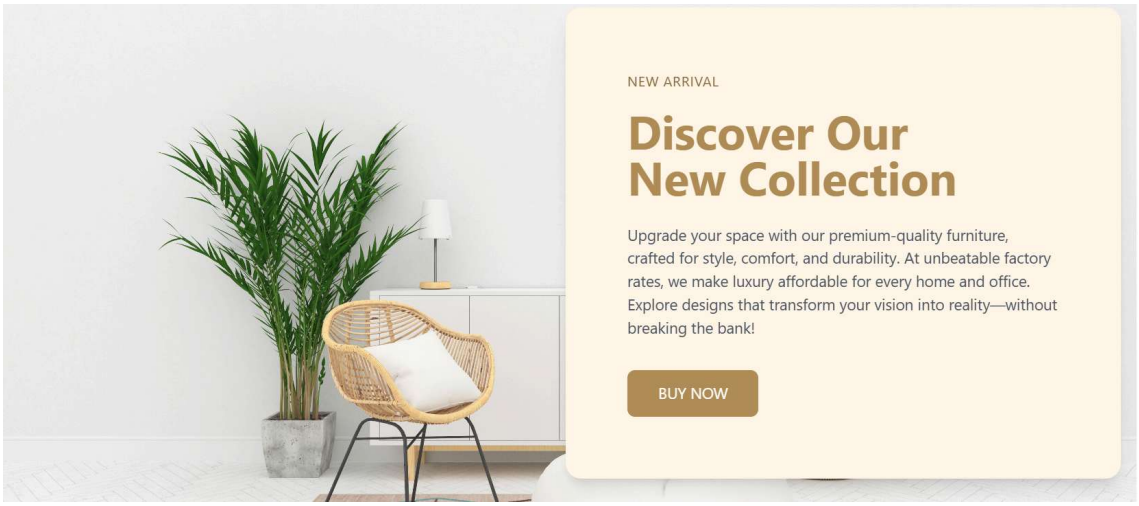
The **Navbar** provides easy navigation between pages, while the **Footer** includes essential links and branding elements.

Screenshot:

3.2 Hero Section

The Hero Section is designed to grab user attention with a visually appealing banner and CTA buttons.

Screenshot

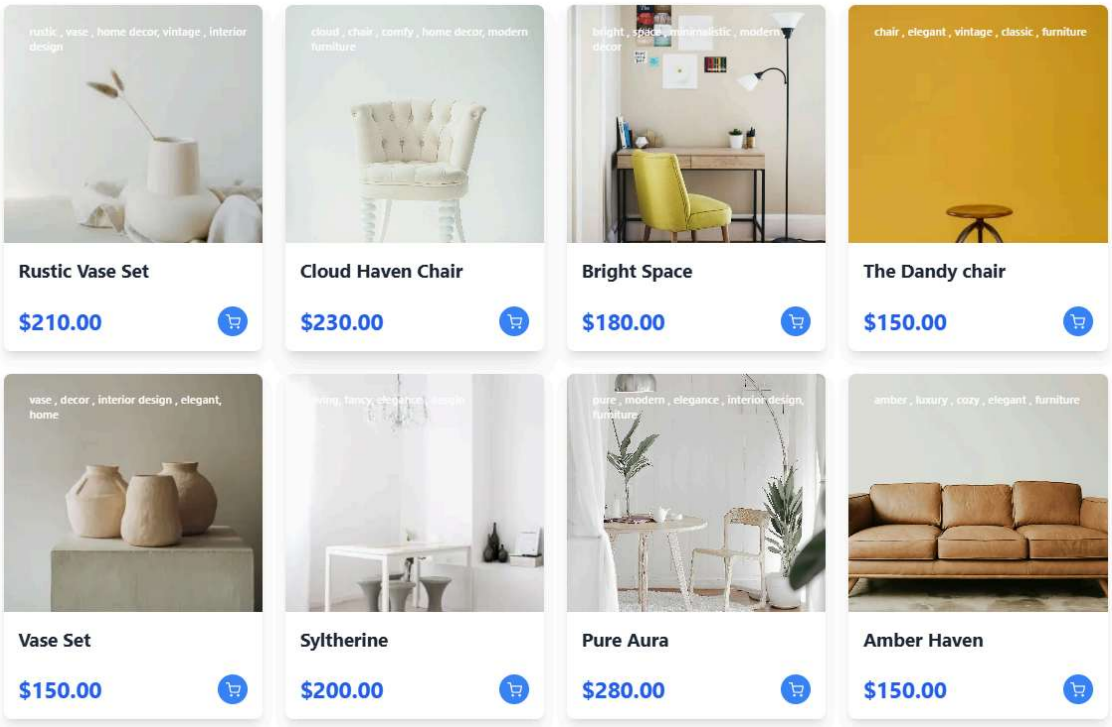


3.3 Product Listing Page

The **Product Listing Component** dynamically fetches and displays products from the backend.

- Features **grid layout** with product images, names, prices, and stock availability.
- Responsive design for both **desktop and mobile**.

Screenshot:

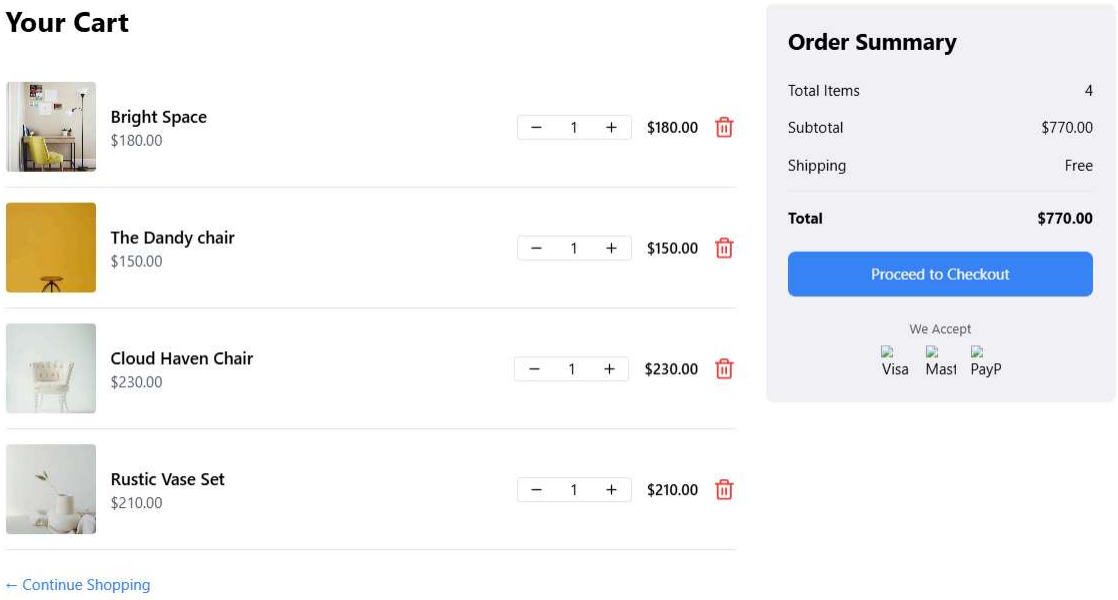


3.4 Add to Cart Functionality

This feature allows users to add products to their cart and manage them before checkout.

- Implements **state management** to persist cart items.
- Shows real-time updates on cart total and quantity.

Screenshot:

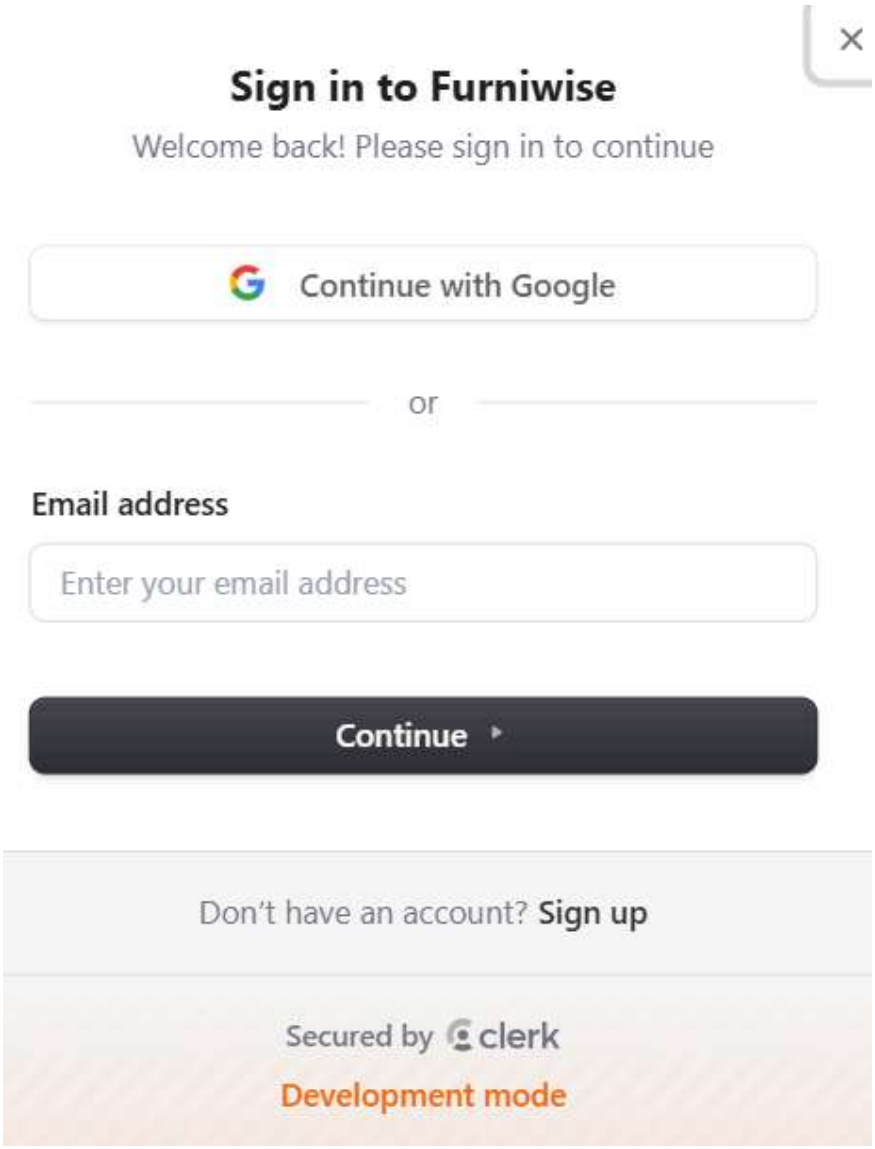


3.5 Authentication System

Secure authentication is implemented using **NextAuth / Firebase / Custom API**.

- Users can **sign up, log in, and log out**.
- **Session management** ensures secure access to user accounts.

Screenshot:

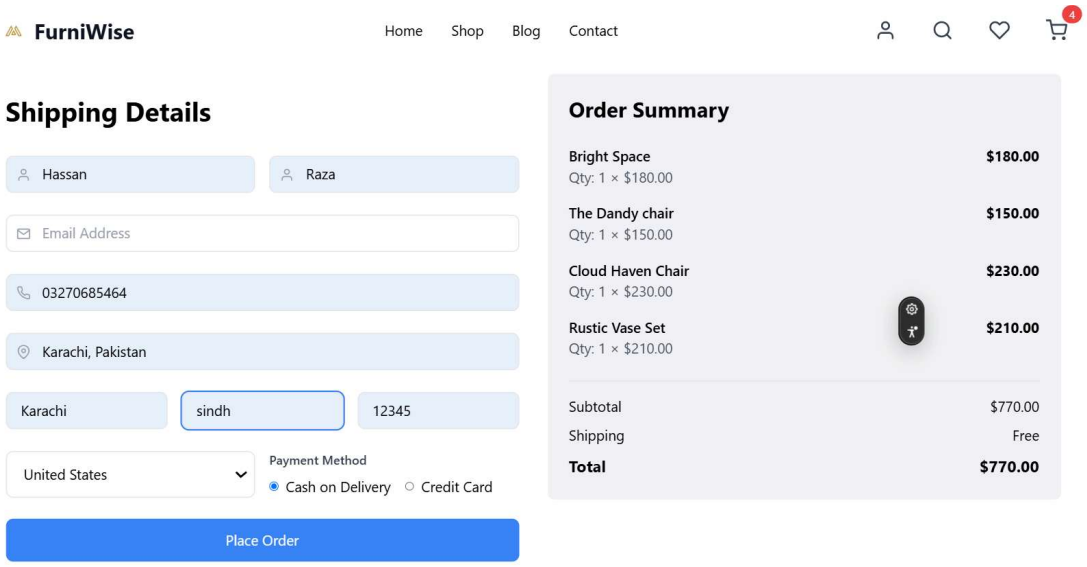


3.6 Checkout Process

The checkout component allows users to complete their purchases through a **multi-step process**.

- **Billing & Shipping Details** input fields.
- **Mock Payment Processing** for a seamless user experience.

Screenshot:



---

4. Best Practices Followed

✓ **Modular Component Design –** Each component is reusable and independent.
✓ **State Management –** Context API/Redux ensures smooth data flow.
✓ **Optimized Performance –** Lazy loading for images and efficient API calls.
✓ **Responsive UI** – Ensures compatibility with mobile and desktop devices.
✓ **Error Handling & Validation –** Proper feedback for users in case of errors.

---

5. Challenges & Solutions

Challenge	Solution
API Fetching Issues	Verified API keys and dataset configuration in Sanity CMS
State Management Complexity	Used Context API for global state handling
Responsive Design	Implemented Tailwind CSS for seamless adaptability

---

6. Conclusion & Next Steps

This project successfully integrates dynamic components essential for a functional marketplace. Future enhancements may include:

- Implementing an **AI-powered recommendation system**.
- Enhancing UI with **animations and interactive elements**.
- Improving **SEO optimization** for better performance.
