

# Day 4 - Dynamic Frontend Development for Marketplace

## Objective:

The goal of **Day 4** is to design and implement **dynamic frontend components** for displaying marketplace data retrieved from **Sanity CMS** or **external APIs**. The focus is on **modular, reusable, and scalable** components, ensuring an optimized and responsive **user experience**.

---

## Task Overview:

✓ **Develop a Product Listing Component** for dynamic product display.

✓ **Integrate key features**, including:

- 🔍 **Search Bar** for real-time filtering.
  - 🏷️ **Filter Panel** for category-based sorting.
  - 📄 **Product Detail Pages** with comprehensive product insights.
    - ✓ Ensure **modular & reusable component architecture**.
    - ✓ Leverage **Tailwind CSS** for a **responsive design**.
- 

## 1. Product Listing Component

A **dynamic, responsive grid layout** efficiently displays products. Each **product card** includes:

✓ **Product Name**

✓ **Price**

✓ **Image**

✓ **Stock Availability**

★ **Built-in Responsiveness**: The layout seamlessly adapts to **all screen sizes** using **Tailwind CSS**.

---

## 2. Search Bar & Filter Panel

🔍 **Search Bar:** Enables **real-time filtering** by **product name or tags**.

🔍 **Filter Panel:** Provides filtering based on **product categories** (e.g., **Chairs, Sofas, Tables**).

💡 **User Experience Focus:** Simplifies **product discovery** and enhances **navigation efficiency**.

---

### 3. Product Detail Component

★ **Dynamic Routing Implementation:** Each product gets an individual **detail page** displaying:

✓ **Detailed Description**

✓ **Pricing & Category Info**

✓ **Customer Reviews & Ratings**

✓ 🛒 **"Add to Cart" & "Buy Now" Buttons**

✓ **Optimized UI/UX Design** for a seamless and engaging **shopping experience**.

---

### 4. Wishlist & Cart Components

🔖 **Wishlist Feature:** Allows users to **save products** for future purchases.

🛒 **Cart Management Features:**

- **Modify product quantity**
  - **View subtotal and total price calculations**
  - 📁 **Local Storage Integration** to maintain data persistence even after page refresh.
- 

### 5. Checkout & Payment System

🛒 **Multi-Step Checkout Process:** Guides users through a **structured purchasing process**.

💳 **Mock Payment Gateway:** Simulates **real transactions** for testing purposes.

---

## 🔧 Technical Challenges & Solutions

● **1. Deployment Issues on Vercel**

✓ **Solution:** Adjusted **environment variables** and resolved **API configuration errors**.

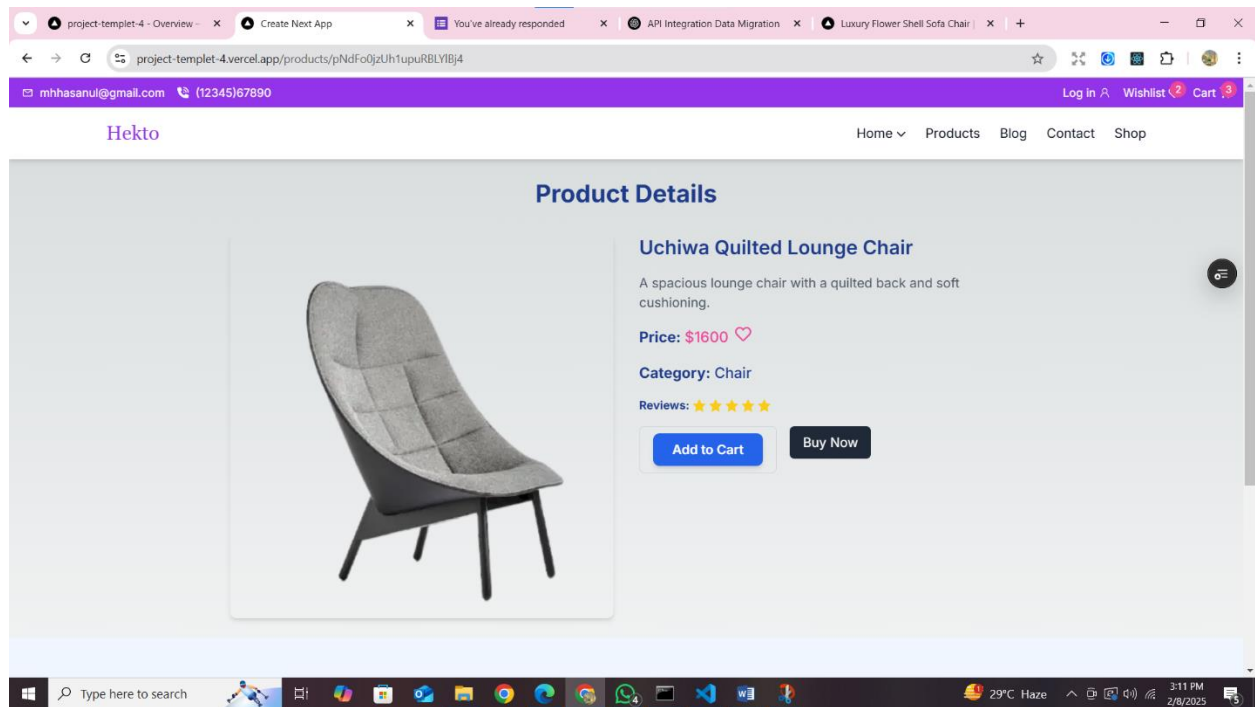
## ● 2. Wishlist & Cart Data Loss After Refresh

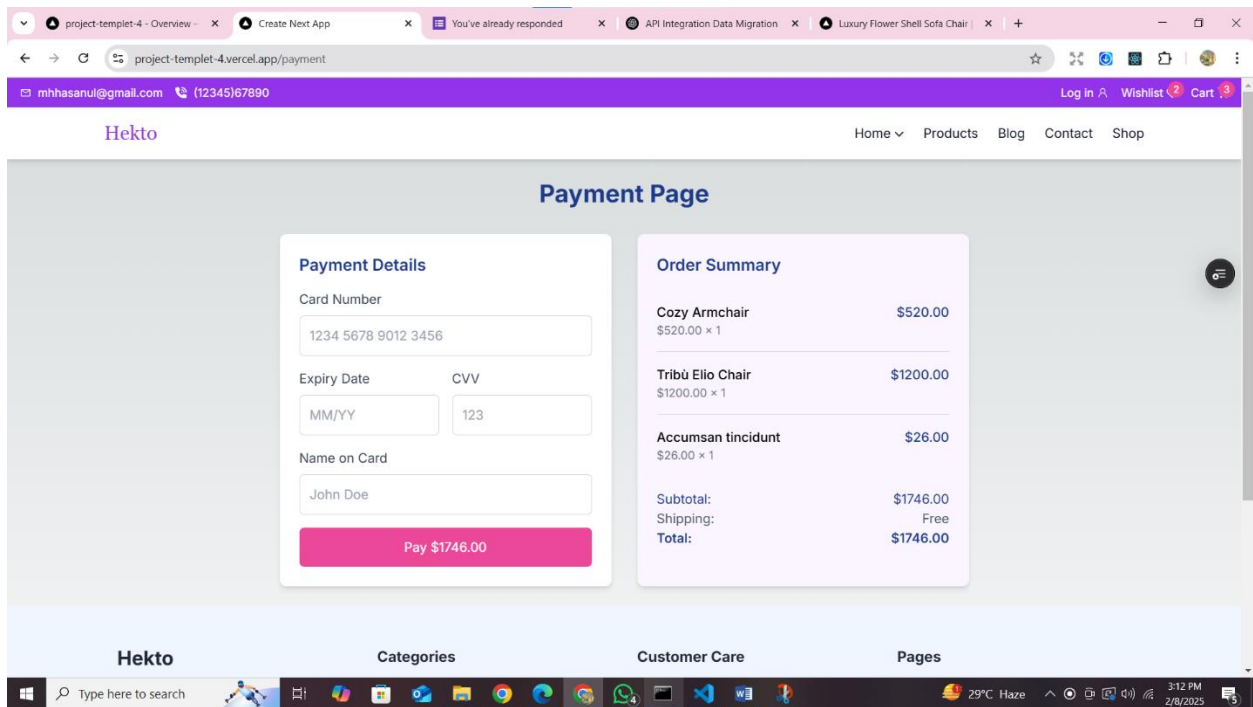
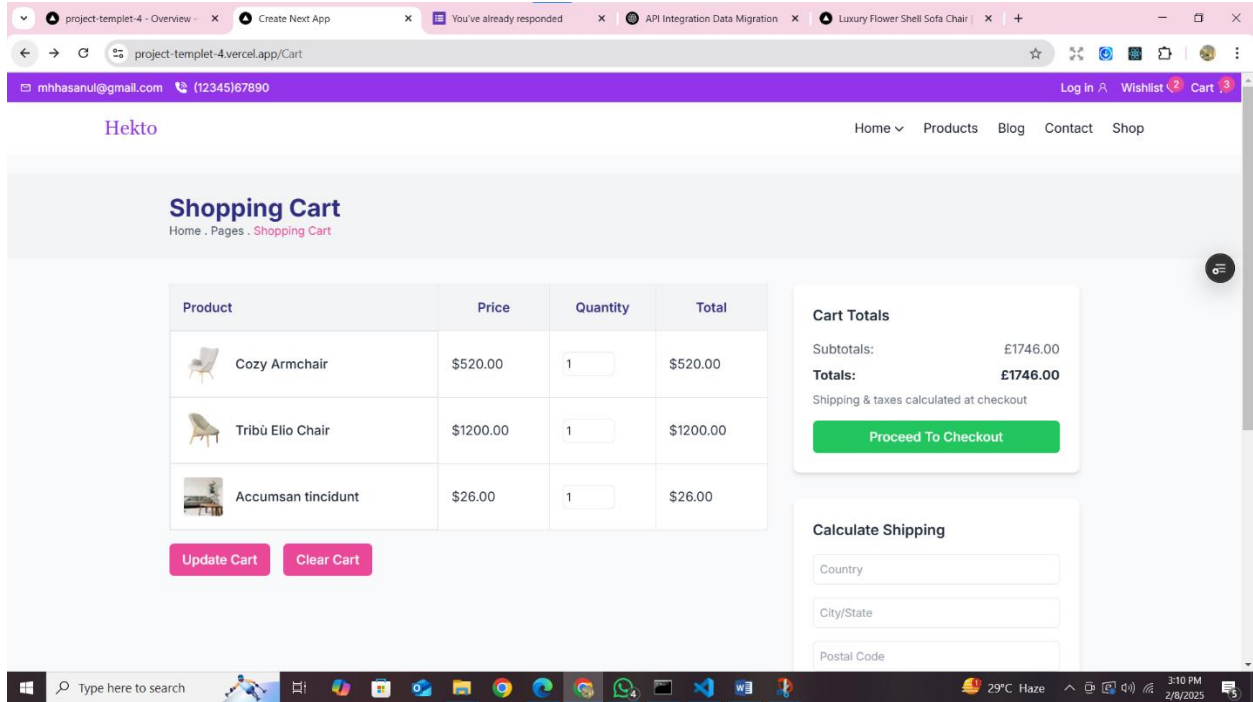
✓ **Solution:** Integrated **Local Storage** to ensure persistent user data.

## ● 3. Dynamic Routing Not Functioning Correctly

✓ **Solution:** Verified **route structures** and tested with **placeholder IDs**.

## 📸 Screenshots of Work Done:





🔗 By: Nuzhat Saba