Technical Documentation: Dynamic Frontend Marketplace

Project Overview

This project is a dynamic marketplace frontend built using Next.js and Sanity CMS.
It allows users to browse products, filter them by categories, search for specific items,
add them to their cart, and proceed to checkout.

Features

1. Home Page:
   - Displays featured products and categories.
   - Includes a search bar for quick product lookup.

2. Product Listing Page:
   - Dynamically fetches products from Sanity CMS.
   - Shows product images, names, and prices.

3. Category Filtering:
   - Users can filter products by categories like Chairs, Tables, Cutlery, etc.
   - Clicking a category redirects users to a category-specific product listing.

4. Search Functionality:
   - Users can search for products using keywords.
   - Displays matching results in real-time.

5. Product Detail Page:
   - Shows full product details including price and description.
   - Option to add products to the cart.

6. Shopping Cart:
   - Displays all added products with quantity controls.
   - Shows total items and total price.
   - Users can proceed to checkout.

7. Responsive Design:
   - Optimized for mobile, tablet, and desktop screens.

Components Breakdown

- Header Component: Contains the brand logo, search bar, and category links.
- ProductCard Component: Displays product image, title, price, and "View Details" button.
- SearchBar Component: Allows users to search for products in real-time.
- CategoryFilter Component: Enables users to filter products by selecting a category.
- ShoppingCart Component: Manages cart items and checkout process.

Technology Stack

- Next.js: Used for server-side rendering and routing.
- Sanity CMS: Headless CMS for managing products and categories.
- Tailwind CSS: Used for responsive styling.
- Zustand: State management for shopping cart.
- React Icons: Used for adding icons to the UI.

Project Structure

```
/src
  /components
    - Header.jsx
    - ProductCard.jsx
    - SearchBar.jsx
    - CategoryFilter.jsx
    - ShoppingCart.jsx
  /pages
    - index.jsx (Homepage)
    - category/[slug].jsx (Category pages)
    - product/[id].jsx (Product detail pages)
    - cart.jsx (Shopping cart page)
  /styles
  /sanity
  /public (images, assets)
```

Challenges & Solutions

1. Dynamic Routing for Products and Categories

   Challenge: Implementing dynamic routes for categories and product details.

   Solution: Used Next.js dynamic routing ([slug].js & [id].js) to fetch and render product data dynamically.

2. State Management for Shopping Cart

   Challenge: Maintaining cart state across pages.

Solution: Implemented Zustand to manage global cart state efficiently.

3. Optimizing API Calls

Challenge: Reducing redundant API calls while fetching product data.

Solution: Used caching and optimized Sanity GROQ queries to minimize data requests.

Future Improvements

- Implement User Authentication (Login/Signup)
- Add Wishlist Feature
- Integrate Payment Gateway for Checkout
- Improve UI/UX for better user engagement

Contact Information

For any queries, reach out at your.email@example.com or visit our GitHub Repository.