DAY 4 - BUILDING DYNAMIC FRONTEND COMPONENTS - METROMART.

Objective:

Day 4 focuses on designing and developing dynamic frontend components to display marketplace data fetched from Sanity CMS or APIs. This step emphasizes modular, reusable component design and real-world practices for building scalable and responsive web applications.

Key Learning Outcomes:

- Build dynamic frontend components to display data from Sanity CMS or APIs.
- Implement reusable and modular components.
- Understand and apply state management techniques.
- Learn the importance of responsive design and UX/UI best practices.
- Prepare for real-world client projects by replicating professional workflows.

Key Components to Build:

1. **Product Listing Component:** Displays products in a grid layout with name, price, image, and stock status.
2. **Product Detail Component:** Dynamic routing for individual product details with descriptions and variations.
3. **Category Component:** Dynamically fetch and display product categories for filtering.
4. **Search Bar:** Enables users to search for products by name or tags.
5. **Cart Component:** Manages selected products, quantities, and total price.
6. **Wishlist Component:** Allows users to save products for later.
7. **Checkout Flow Component:** Implements multi-step checkout including shipping and payment details.

8. **User Profile Component:** Displays user information and order history.
9. **Reviews and Ratings Component:** Allows users to submit and view product reviews.
10. **Pagination Component:** Divides large product listings into multiple pages.
11. **Filter Panel Component:** Enables advanced filtering by price, brand, and availability.
12. **Related Products Component:** Displays suggestions based on user behavior.
13. **Footer and Header Components:** Provides consistent navigation and branding.
14. **Notifications Component:** Displays real-time alerts for actions like adding to cart.
15. **Analytics Dashboard Component:** Showcases sales and user activity insights.
16. **Product Comparison Component:** Allows users to compare product features side by side.
17. **Multi-Language Support Component:** Adds language switcher for multilingual support.
18. **Order Tracking Component:** Provides users with real-time updates on their order status.
19. **FAQ and Help Center Component:** Includes a searchable FAQ section and support contact.
20. **Subscription Management Component:** Enables product or service subscriptions.
21. **Admin Dashboard Component:** Allows managing products, orders, users, and reports.
22. **Discount and Promotion Component:** Displays and manages discounts dynamically.
23. **Social Media Sharing Component:** Allows sharing products via social media.

24. **Bulk Upload Component:** Enables admins to upload product data via CSV files.
25. **AI Recommendations Component:** Suggests products based on user behavior.
26. **Gift Card and Voucher Component:** Manages gift card purchases and redemptions.
27. **Customer Feedback Component:** Collects feedback and displays aggregated insights.
28. **Advanced Search Component:** Implements faceted search with multiple filters.

Frontend Best Practices:

- **Reusable Components:** Modular components for better maintainability.
- **State Management:** Use React state, Context API, or Redux for data handling.
- **Styling:** Implement Tailwind CSS or styled-components for responsive UI.
- **Performance Optimization:** Utilize lazy loading, pagination, and caching.

Steps for Implementation:

1. Set up a Next.js project and connect it to Sanity CMS or APIs.
2. Build and test dynamic components ensuring responsiveness.
3. Integrate API data and implement proper state management.
4. Validate data handling and component reusability.
5. Deploy and document the completed components.

Expected Output:

- Fully functional product listing and detail pages.
- Implemented search, category filtering, and pagination.
- Responsive and dynamic UI components.
- Modular and reusable frontend components.

Submission Requirements:

- GitHub repository with:
  - All developed components.
  - README.md summarizing project structure.
  - Screenshots or screen recordings of components.
- A professional technical report in PDF or Markdown format.

Checklist for Day 4:

☑ ~~All required frontend components implemented.~~

☑ ~~Responsive design and state management applied.~~

☑ ~~Code structured in a modular and reusable format.~~

☑ ~~Documentation and submission complete.~~

This document ensures a structured and professional approach to developing dynamic frontend components for a marketplace application.