Hackathon Day 4

Bandage Marketplace Template: Building Dynamic Frontend Components

Objective

On Day 4, students focused on designing and developing dynamic frontend components to display marketplace data fetched from Sanity CMS or APIs. This task emphasized creating modular, reusable components while learning real-world practices for building scalable and responsive web applications.

Key Learning Outcomes

1. Build dynamic frontend components that fetch and display data from Sanity CMS or APIs.
2. Implement reusable and modular components for easier maintenance and scalability.
3. Apply state management techniques to manage data flow across components.
4. Focus on responsive design and implement UX/UI best practices.
5. Prepare for real-world client projects by replicating professional workflows.

Key Components to Build

1. **Product Listing Component**:
   a. Displays a list of products dynamically, including details such as image, availability, description, color, size, rate, reviews, and category.
   b. Highly reusable and flexible, allowing data to be passed as props.
2. **Product Detail Component**:
   a. Shows detailed information about a single product, including product image, heading, department, pricing (both original and discounted), and a link to the product's page.
3. **Category Component**:
   a. Filters products based on different categories such as Electronics, Clothing, etc., to enhance user navigation and experience.
4. **Cart Component**:

     a. Allows users to add products to their shopping cart, view items in the cart, and proceed to checkout.

5. **Wishlist Component**:
     a. Enables users to save products for later purchase.

6. **User Profile Component**:
     a. Displays the user's personal details, order history, and preferences.

7. **Reviews and Ratings Component**:
     a. Allows users to leave reviews and rate products.

8. **Related Products Component**:
     a. Displays products similar to the one the user is viewing, enhancing cross-selling opportunities.

9. **Header Components**:
     a. Provide navigation links, contact information, and additional utility features throughout the application.

10. **Footer Components**:
     a. Provide navigation links, contact information, and additional utility features throughout the application.

Self-Validation Checklist for Day 4

- All components fetch data dynamically from Sanity CMS or APIs.
- Components are modular, reusable, and easy to maintain.
- State management is implemented effectively for smooth data flow.
- UI is responsive and adheres to UX/UI best practices.
- The overall design and workflow align with real-world professional standards.

Conclusion

On Day 4, students gained valuable experience in building dynamic frontend components using real-time data fetched from Sanity CMS or APIs. The focus on modularity, state management, and responsive design ensured that the components were scalable and production-ready. By following best practices, students are now equipped to build dynamic and interactive web applications suitable for real-world projects.