Day 4 Summary: Building Dynamic Frontend Components for FoodTuck Marketplace

Prepared By: Rabia (Development Team)

Objective

Day 4 focused on designing and developing dynamic frontend components for the FoodTuck marketplace. The goal was to fetch and display data dynamically using APIs or Sanity CMS and ensure components are modular, reusable, and responsive.

Key Learning Outcomes

1. Developed dynamic frontend components to fetch and display marketplace data.
2. Designed reusable and modular components for scalability.
3. Improved understanding of state management techniques.
4. Applied responsive design principles for better user experience.
5. Followed professional workflows to mimic real-world development practices.

Components Developed

1. Product Listing Component

- **Functionality:** Displayed a grid of products dynamically fetched from the backend.
- **Fields Included:** Product Name, Price and Image.
- **Example:** Grid layout with product cards.

2. Product Detail Component

- **Functionality:** Created detailed pages for each product using dynamic routing.
- **Fields Included:** Description, Price, Available .

3. Category Component

- **Functionality:** Dynamically displayed product categories and enabled filtering.

4. Search Bar

- **Functionality:** Allowed users to search for products by name or id.

5. Cart Component

- **Functionality:** Displayed selected items, quantities, and total price.
- **State Management:** Tracked cart items using React Context.

6. Filter Panel Component

- **Functionality:** Allowed users to filter products by Category.

7. Responsive Header and Footer

- **Functionality:** Included navigation links, branding, and key contact details.
- **Responsiveness:** Adjusted design dynamically for desktop, tablet, and mobile views.

Frontend Best Practices Followed

1. **Reusable Components**
   - Designed modular components like `ProductCard` and `CategoryFilter` to be used across pages.
   - Passed data via props for flexibility.
2. **State Management**
   - Used React state and context for efficient data management.
3. **Styling**
   - Implemented responsive designs using Tailwind CSS.
   - Ensured consistency across desktop and mobile views.
4. **Performance Optimization**
   - Lazy-loaded images and assets to improve load time.
   - Implemented pagination for large datasets.

Challenges Faced

1. **Dynamic Routing Issues:**
   - Initial challenges with setting up product detail pages using Next.js dynamic routing.
   - **Solution:** Reviewed and corrected file structure to enable correct routing.
2. **API Data Fetching:**
   - Encountered delays in rendering data due to API response time.
   - **Solution:** Used loading states to improve user experience.
3. **Responsive Design Adjustments:**
   - Ensuring the UI adapted perfectly on smaller screens like the iPhone 12.
   - **Solution:** Adjusted breakpoints in Tailwind CSS for seamless responsiveness.

Next Steps

1. Implement advanced features like wishlist and order tracking.
2. Optimize the search bar functionality for better performance.
3. Test components for cross-browser compatibility.
4. Document all steps and challenges for future reference.

Conclusion

Day 4 marked significant progress in creating a dynamic and professional frontend for the FoodTuck marketplace. The components are now modular, scalable, and responsive, laying a strong foundation for future development and real-world deployment.