

# Day 4 - Dynamic Frontend Components - Foodtuck.com

## Table of Contents

1. [Product Listing Component](#)
2. [Product Detail Component](#)
3. [Category Component](#)
4. [Search Bar Component](#)
5. [Cart Component](#)
6. [Checkout Flow Component](#)
7. [Reviews and Ratings Component](#)
8. [Pagination Component](#)
9. [Filter Panel Component](#)
10. [Related Products Component](#)
11. [Footer and Header Components](#)
12. [Notifications Component](#)
13. [Analytics Dashboard Component \(with Sanity Studio Integration\)](#)
14. [Additional Components \(Planned/Not Implemented\)](#)
15. [Summary & Next Steps](#)

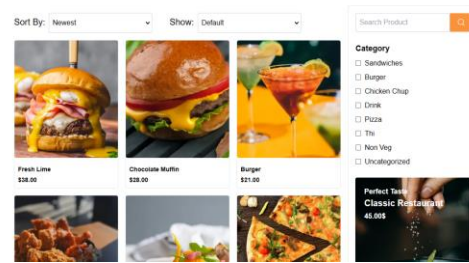
---

## 1. Product Listing Component

**Status:** Implemented

**Details:**

- Displays product cards in a responsive grid layout on the shop page.
- Each card contains key product details such as product image, name, price, and stock status.
- Every product card is clickable and routes users to the corresponding product detail page.



### Key Considerations:

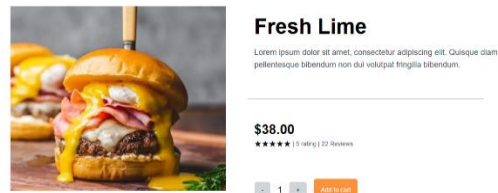
- Responsiveness for both mobile and desktop.
- Efficient data fetching from Sanity CMS or APIs to populate the grid.

## 2. Product Detail Component

**Status:** Implemented

**Details:**

- Dynamic routing is used to render individual product detail pages.
- Displays comprehensive product information, including descriptions, pricing, available options (like sizes or colors), and additional images.
- Ensures that all the relevant details are fetched from the backend and displayed properly.



### Key Considerations:

- Integration with Sanity CMS for dynamic content.
- SEO-friendly URL structures and dynamic meta information.

## 3. Category Component

**Status:** Implemented

**Details:**

- Provides category filtering and navigation.
- Allows users to select a category which updates the products displayed on the shop page.
- Enhances user experience by segmenting products into logical groups.

#### Key Considerations:

- Seamless integration with the filtering system.
- Efficient data handling to reflect real-time category changes.

**Category**

- ☐ Sandwiches
- ☐ Burger
- ☐ Chicken Chup
- ☐ Drink
- ☐ Pizza
- ☐ Thi
- ☐ Non Veg
- ☐ Uncategorized

## 4. Search Bar Component

**Status:** Partially Implemented

#### Details:

- A search bar component is present on the site.
- Designed to filter products based on name or tags.
- Currently, the functionality is not fully operational, and further work is needed to make the search feature robust.

#### Key Considerations:

- Completion of search logic and integration with the product data.
- Consideration for fuzzy search and real-time suggestions.


## 5. Cart Component

**Status:** Implemented

#### Details:

- A fully functional shopping cart system is in place.
- Supports adding and removing items, displays item quantities, and calculates total prices.
- Integrates with the rest of the components to maintain cart state across user sessions.
- 

#### Key Considerations:

Product	Price	Quantity	Subtotal	Remove
 Homemade Pizza	\$20.00	<input type="text" value="1"/>	\$20.00	<input type="button" value="X"/>

Coupon Code (uzair) to get 10% discount

**Order Summary**

<b>Subtotal</b>	<b>\$20.00</b>
Shipping Charge	\$5.00
Discount (If you applied couponCode uzair)	10%
<b>Total Amount</b>	<b>\$25.00</b>

- State management (using React state or a global context) to ensure consistency. • Potential enhancements with persistence (e.g., local storage or session storage).
- 

## 6. Checkout Flow Component

**Status:** Implemented

**Details:**

- Provides a multi-step checkout process:
  - When a user clicks the “Go to Checkout” button, they are either directed to the checkout page (if logged in) or to an authentication flow before proceeding.
  - The checkout page collects shipping details and allows order placement.
- Ensures a smooth transition between cart, authentication, and checkout flows.

**Key Considerations:**

- Secure handling of user data during the checkout process.
  - Future integration with real payment gateways or mock payment methods.
- 

## 7. Reviews and Ratings Component

**Status:** Implemented

**Details:**

- Allows users to view and submit reviews and ratings on product detail pages.
- Displays aggregated ratings and individual user reviews. • Enhances product credibility and user interaction.

**Key Considerations:**

- Moderation features for reviews could be added.
  - Validation and error handling for review submissions.
- 

## 8. Pagination Component

**Status:** Implemented

**Details:**

- Implements pagination on product listings.
- Displays a maximum of nine products per page.
- Users can navigate through multiple pages to view additional products.

**Key Considerations:**

- Ensuring smooth transitions between pages.
  - Performance optimizations for large datasets.
  -
- 

## 9. Filter Panel Component

**Status:** Implemented

**Details:**

- Provides advanced filtering options on the shop page.
- Users can adjust filters such as price range, availability, and other product attributes.
- The filter panel updates the product listing state in real time based on the selected filters.

**Key Considerations:**

- User-friendly design and responsiveness.
  - Optimizing performance when applying multiple filters.
- 

## 10. Related Products Component

**Status:** Implemented

**Details:**

- Displays related products on the product detail page.
- Selection is based on the category of the current product, enhancing cross-selling opportunities.
- Improves user engagement by suggesting complementary items.

**Key Considerations:**

- Dynamic fetching and rendering based on the current product's data.
  - Ensuring relevance and freshness of the related products list.
-

# 11. Footer and Header Components

Status: Implemented

SHOP.CO

Shop

On Sale

New Arrivals

Brands

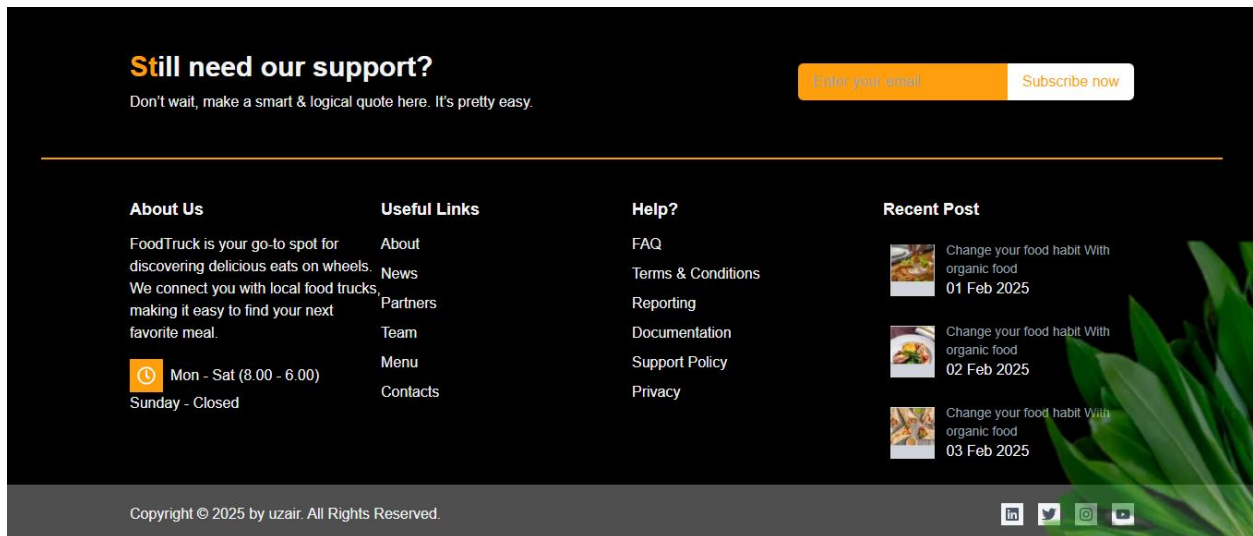
Search for products...



## Details:

- Provides consistent navigation and branding across the entire site.
- Both components are designed to be fully responsive.
- Includes key links such as Home, About, Contact, and other navigational elements

## Key Considerations:



- Accessibility compliance.
- Maintaining a consistent look and feel across devices.

# 12. Notifications Component

Status: Implemented

## Details:

- Toast notifications are used to alert users about important events:
  - Item added to cart.
  - Item removed from cart.
  - Success messages during order placement.
  - Error notifications when something goes wrong.

- Provides immediate feedback, enhancing user experience.

**Key Considerations:**

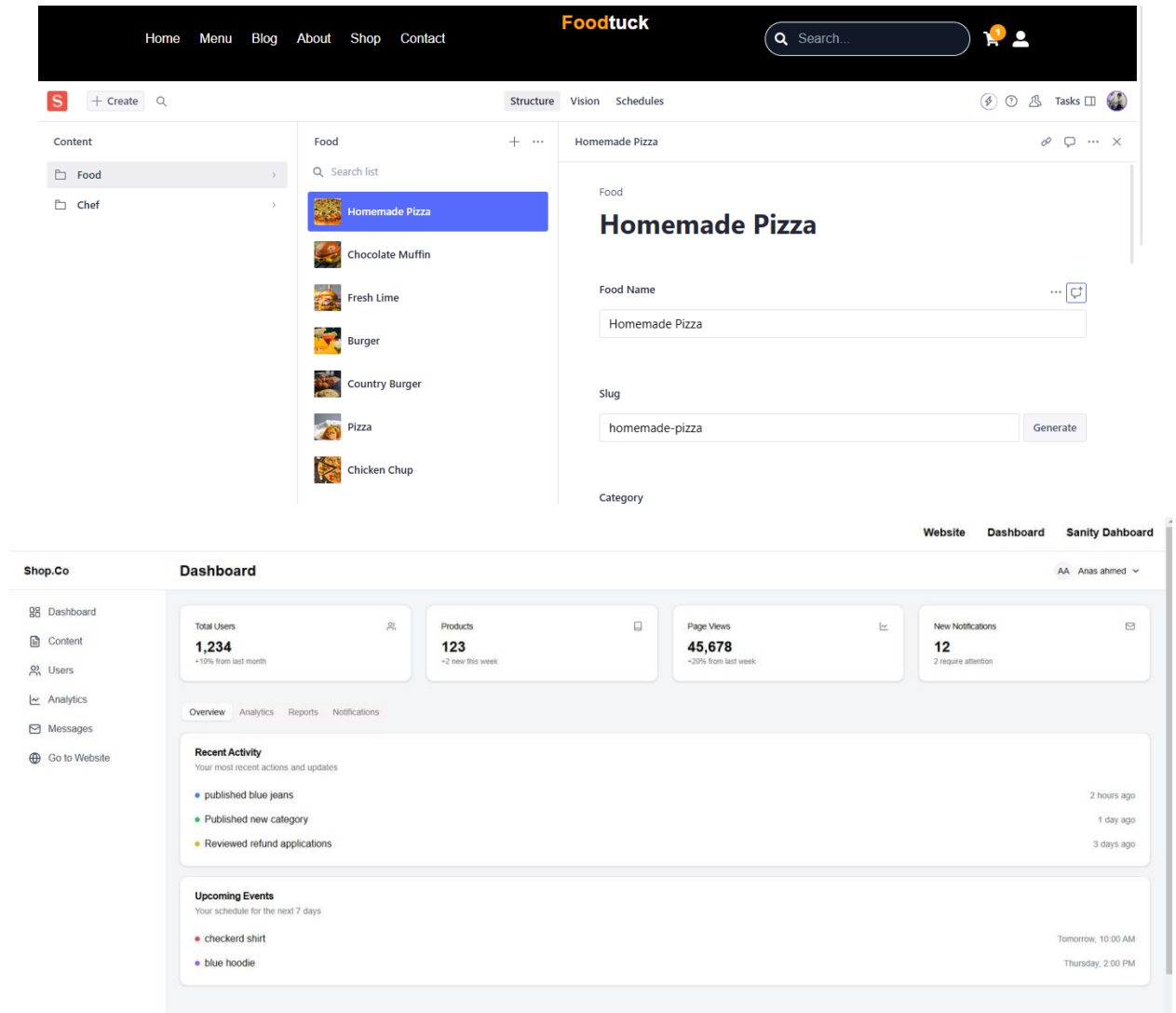
- Ensuring notifications do not obstruct important content.
  - Customizing notification duration and style.
- 

## 13. Analytics Dashboard Component (with Sanity Studio Integration)

**Status:** Implemented

**Details:**

- A basic analytics dashboard has been built, which also serves as a vanity studio.
- Displays key performance indicators such as total users, products, page views, and recent activities.
- Integrates with **Sanity Studio** for managing products, orders, and reviews.
- Data retrieval is handled using the Sanity client, ensuring real-time updates.



### Key Considerations:

- Future enhancement: Integration of charting libraries (like Recharts or Chart.js) for visual data representation.
- Streamlining data for better performance and clarity in analytics.

## 14. Additional Components (Planned/Not Implemented)

### Components Not Implemented:

- **Wishlist Component:** Functionality for saving products for later use.
- **User Profile Component:** User-specific data management.
- **Product Comparison Component, Multi-Language Support, Order Tracking, FAQ and Help Center, Subscription Management, Discount and Promotion, Social Media Sharing, Bulk Upload, AI Recommendations, Gift Card and Voucher, Customer Feedback, Advanced Search Component:** These components were listed



as possibilities but have not been implemented in the current build. They remain on the roadmap for future enhancements based on the marketplace's evolving needs.

---

## Summary & Next Steps

### Accomplishments:

- The core shopping experience is robust, with dynamic product listings, detailed product pages, and a smooth checkout flow.
- Essential user interaction features such as cart management, toast notifications, and filtering have been effectively implemented.
- The admin-side analytics and management features are supported by Sanity Studio, providing a solid backend foundation.

### Areas for Future Development:

- **Enhance Search Bar:** Improve functionality for product filtering by name/tags.
- **Implement Wishlist and User Profile:** Introduce features that allow users to save favorite products and manage their profiles.
- **Advanced Analytics:** Integrate charting libraries to convert raw analytics data into visually appealing insights.
- **Additional Components:** Consider rolling out components like product comparison, multi-language support, and order tracking based on user feedback and business needs.

### Conclusion:

The current implementation of Shop.Co covers a broad spectrum of functionalities crucial for a dynamic and responsive marketplace. With a strong foundation in modular component design, there is ample opportunity for future expansion and refinement. The architecture is well-positioned to support additional features as the project scales.