Darsamo Bites Documentation

Daniel Darsamo

**Feature Documentation**

**1. Current Features**

**1.1 Menu System**

* **Categorized Display**: The menu is divided into distinct categories, including Drinks (Coffee, Cold Drinks, Teas) and Snacks.
* **Dynamic Filtering**: Users can seamlessly switch between categories with smooth animations.
* **Item Details**: Each menu item is displayed with its name, price, and an "Add to Cart" button.

**1.2 Cart Functionality**

* **Add/Remove Items**: Users can modify quantities of selected items using increment (+) and decrement (-) buttons.
* **Real-time Total Calculation**: The order total is dynamically updated as items are added or removed.
* **Visual Feedback**: The cart counter reflects changes instantly, ensuring a smooth user experience.

**1.3 User Experience Enhancements**

* **Dark Mode**: A toggle feature allows users to switch themes, with preferences saved using localStorage.
* **Mobile Responsiveness**: The application is fully functional on both mobile and tablet devices.
* **Animations**: Transitions between sections are animated for a visually appealing interaction.

**2. Technical Implementation**

**2.1 Frontend Development**

The entire system is built using pure **HTML, CSS, and JavaScript**, without reliance on external frameworks.

**2.2 State Management**

* **menuData Object**: Stores details of all available menu items.
* **cart Array**: Maintains the current list of selected orders.
* **LocalStorage**: Preserves user preferences for dark mode.

**3. Code Structure**

**3.1 Key Files**

1. **index.html** – Defines the main layout and structure.
2. **style.css** – Manages styling using CSS variables for consistent theming.
3. **script.js** – Handles all interactive logic and state updates.

**3.2 Core Functions**

|  |  |
| --- | --- |
| Function Name | Purpose |
| initializeMenu() | Loads and displays menu items dynamically. |
| updateCartDisplay() | Refreshes the cart UI based on changes. |
| addToCart() | Adds selected items to the shopping cart. |

This document serves as a comprehensive guide to the current features and technical aspects of DARSAMO BITES.

Darsamo Bites Documentation V.1.2.0

## 1. Current Features

### 1.1 Enhanced Order System

- Multi-Order Management:  
 - Create and track multiple orders simultaneously  
 - Edit pending ("preparing") orders  
 - Merge current cart into existing orders

- Order Tracking:  
 - Visual status indicators (preparing/completed)  
 - Timestamped order history  
 - Quick-view for order contents

### 1.2 Advanced Cart Functionality

- Edit Mode:  
 - Load existing orders into cart for modification  
 - Real-time quantity adjustment with +/- buttons  
 - Visual indicators for edit state

- Smart Calculations:  
 - Automatic per-item and order total updates  
 - Persistent cart across sessions

### 1.3 Notification System

- Action Feedback:  
 - Success toasts for completed operations  
 - Error alerts for invalid actions  
 - Status messages during workflows

- Visual Design:  
 - Auto-dismissing after 3 seconds  
 - Animated progress bar  
 - Mobile-optimized positioning

### 1.4 User Experience Enhancements

- State Persistence:  
 - Dark mode preference  
 - Active orders and cart contents  
 - Edit states preserved during navigation

- Accessibility:  
 - ARIA live regions for screen readers  
 - Keyboard-navigable interface

- Enhanced Animations:  
 - Smooth category transitions  
 - Notification entrance/exit effects

## 2. Technical Implementation

### 2.1 Frontend Development

Built with pure HTML5, CSS3, and Vanilla JavaScript featuring:  
- CSS variables for dynamic theming  
- Flexbox/Grid layout systems  
- Responsive design principles

### 2.2 State Management

- State Object:

{  
 activeOrders: Order[],  
 currentCart: CartItem[],  
 nextOrderId: number,  
 editingOrderId: number|null  
}

- Persistence:  
 - Full state serialization to localStorage  
 - Version-controlled data structure

## 3. Code Structure

### 3.1 Key Files

index.html  
 - Main application structure  
 - Notification container DOM  
 - Semantic HTML5 elements

- style.css  
 - Notification animations  
 - Edit state visual cues  
 - Dark mode refinements

- script.js  
 - Modular architecture with:  
 - Order management system  
 - Notification controller  
 - State persistence handler

### 3.2 Core Functions

|  |  |  |
| --- | --- | --- |
| Function | Parameters | Description |
| createNewOrder() | None | Generates complete order from current cart |
| addToOrder() | orderId: number | Merges cart into specified order |
| startEditing() | orderId: number | Enters order modification mode |
| showNotification() | message: string, type: enum | Displays contextual feedback toasts |
| persistState() | None | Saves full state to localStorage |
| calculateTotal() | items: CartItem[] | Computes order totals |

## 4. Version History

- v2.1 (Current)  
 - Added notification system  
 - Enhanced order editing  
 - Improved state persistence

- v1.0  
 - Base cart functionality  
 - Dark mode toggle

## Appendix A: Data Structures

interface CartItem {  
 name: string;  
 price: number;  
 quantity: number;  
}  
  
interface Order {  
 id: number;  
 items: CartItem[];  
 total: number;  
 status: 'preparing'|'completed';  
 timestamp: string; // ISO-8601 format  
}

## Appendix B: