## **Grocery List App**

The Grocery List App is designed to help users manage their grocery shopping efficiently. The application includes three main pages:

1. **Product List** - Displays a list of available products and allows users to search and filter items.
2. **Cart** - Shows items added to the cart and allows users to review their selections and checkout.
3. **Add Item** - Provides a form for users to add new products to the list.

## **Page Details**

### **1. Product List**

#### **HTML Structure**

* **<nav>**: Contains navigation links with icons for various sections including the Add Item page, Cart page, and Login/Register pages.
* **<section>**: Contains the main content of the page, including:
  + A search bar for filtering products.
  + A dropdown filter to sort products by category (All, Fruits, Vegetables, Beverages).
  + A <div class="product-content" id="product-list"> where product items are dynamically added by JavaScript.

#### **CSS File: style.css**

* Styles the product list, search bar, and filters.
* Ensures responsiveness and proper layout for different devices.

#### **JavaScript File: style.js**

* Handles dynamic addition of products.
* Implements search and filter functionalities.
* Manages interactions with the cart.

### 

### 

### 

### **2. Cart**

#### **HTML Structure**

* **<nav>**: Contains navigation links similar to those on the Product List page.
* **<h1>**: Displays the title "Your Cart".
* **<div id="cart-list">**: Displays items currently in the cart.
* **<div class="total-section hidden">**: Shows the total price and a checkout button, which is hidden until items are added to the cart.

#### **CSS File: cart.css**

* Styles the cart items, total section, and ensures the layout is clear and user-friendly.

#### **JavaScript File: cart.js**

* Manages adding, updating, and removing items from the cart.
* Calculates the total price of the items in the cart.
* Handles the checkout process.

### **3. Add Item**

#### **HTML Structure**

* **<nav>**: Contains navigation links to the Product List and Cart pages.
* **<section>**: Contains a form to add new items with the following fields:
  + Product Name
  + Brand
  + Price
  + Weight/Volume
  + Quantity
  + Store
  + Category (dropdown)
  + Product Image (file upload)

#### **CSS File: pradacct.css**

* Styles the form and input fields to ensure a user-friendly experience.

#### **JavaScript File: pradacct.js**

* Handles the functionality of the "Add Item" button.
* Validates input and adds the new item to the product list.

## 

## 

## **Functionality**

1. **Product List Page**:
   * Users can search for products using the search bar.
   * Filter products by category using the dropdown menu.
   * Add products to the cart using the "Add to Cart" button.
2. **Cart Page**:
   * Users can view items added to their cart.
   * Update quantities and remove items as needed.
   * The total price is automatically updated.
   * Proceed to checkout when ready.
3. **Add Item Page**:
   * Users can add new products by filling out the form.
   * All details of the product are captured, including image and category.
   * Submitted items are dynamically added to the Product List page.

## **Dependencies**

* **HTML**: Provides the structure for the application.
* **CSS**: Styles the layout and appearance of each page.
* **JavaScript**: Manages dynamic content and user interactions.

## **Usage**

1. **Navigating the App**:
   * Use the navigation bar to switch between pages.
   * On the Product List page, search and filter products as needed.
   * Add items to the cart and review them on the Cart page.
   * Add new items via the Add Item page.
2. **Adding and Managing Items**:
   * On the Product List page, click "Add to Cart" to add products to your cart.
   * On the Cart page, update quantities and remove items as necessary.
   * On the Add Item page, complete the form and click "Add Item" to include new products.

## 

## **Notes**

* Ensure that JavaScript files are correctly linked and functioning.
* CSS files should be updated to reflect any design changes.
* Test responsiveness on various devices to ensure compatibility.

This documentation provides an overview and guide for using and managing the Grocery List App, ensuring a smooth and efficient user experience.