

Front end Mini Project

Restaurant Menu

- **Submitted By:**
 - o *Team Members Joe Francis Sebastina(2461013)*
 - o *[:joe.franics@btech.christuniversity.in](mailto:joe.franics@btech.christuniversity.in)*
 - o *Joel George(2462539)*
 - o *joelgeorge.jomon@btec.hchristuniversity.in*
- **Course:** *LnT Front End UI/UX*
- **Instructor Name:** *.Dhiraj Alate*
- **Institution:** *e.g., Christ University*
- **Date of Submission:** *11/08/2025*

Resturant menu

Abstract

This project is an interactive restaurant menu web application built entirely with HTML, CSS, and JavaScript. It allows customers to browse dishes by category, search items, adjust quantities, and add selections to a cart. The system calculates subtotal, tax, service charges, discounts, and total in real time. It also includes a checkout feature that generates a printable receipt. The design is responsive, visually appealing, and optimized for a smooth user experience on both desktop and mobile devices

Objectives

1. To create a user-friendly restaurant menu interface that enables easy browsing and selection of food items.
- 2.To implement real-time price calculation, including tax, service charges, and discounts.
3. To allow customers to customize their orders by adjusting item quantities before checkout.
- 4.To generate a printable receipt summarizing the order details for the customer or restaurant.

Tools & Technologies Used

HTML5 for structuring the web page content and layout.

CSS3 for styling, layout design, and creating a visually appealing UI.

Vanilla JavaScript (ES6) for dynamic interactivity, cart logic, and receipt generation.

Google Fonts for typography enhancement.

Unsplash Images for high-quality placeholder food item images.

HTML Structure Overview

The HTML is organized into three main sections: a sidebar for category filtering and search, a central grid for displaying menu items, and a right-hand cart summary.

Semantic HTML elements such as `<header>`, `<aside>`, `<main>`, and `<section>` are used for better structure and accessibility. Each menu item is represented in a reusable card layout containing the image, name, category, price, and action buttons. The cart area includes dynamically updated order details, summary totals, and checkout controls.

CSS Styling Strategy

The CSS follows a clean, modern design approach with a focus on readability and user comfort. A custom color palette with accent and muted tones creates a professional yet friendly atmosphere. Grid and flexbox layouts are used to achieve responsive designs that adapt to various screen sizes. Box shadows, border radiuses, and gradient backgrounds enhance visual appeal without overloading the interface. Interactive elements like buttons and hover states have subtle animations for better user feedback.

Key Features

Category Filtering: Quickly filter dishes by category such as Pasta, Pizza, Desserts, etc.

Search Functionality: Real-time search for menu items by name or ingredients.

Dynamic Cart: Add, remove, and adjust quantities of items while automatically updating totals.

Price Calculation: Includes tax, service charges, and conditional discounts.

Printable Receipt: Checkout generates a detailed receipt for customers or staff

Challenges Faced & Solutions

Challenge 1: Managing cart state dynamically without a backend database.

Solution: Used a JavaScript object to store cart data in memory and update the UI in real time.

Challenge 2: Keeping the UI responsive across different screen sizes.

Solution: Implemented CSS grid and flexbox layouts with media queries for adaptive design.

Challenge 3: Handling multiple filters and search together.


Solution: Created a rendering function that applies category, search, and sort operations before generating HTML.


Sample code

```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4 <meta charset="UTF-8">
5 <title>Restaurant Menu</title>
6 <
7 b The title element represents the document's title or name. Authors should use titles that identify their documents even
8 h when they are used out of context, for example in a user's history or bookmarks, or in search results. The document's
9 . title is often different from its first heading, since the first heading does not have to stand alone when taken out of
10 . context.
11 . ✓ Widely available across major browsers (Baseline since 2015)
12 1 MDN Reference
13 .total {margin-top:20px; font-size:18px; color:■ #4caf50;}
14 button {margin-top:10px; padding:10px 20px; border:none; background:■ #ff5722; color:■ white; border-radius:5px; cursor:pointer;
15 button:hover {background:■ #e64a19;}
16 </style>
17 </head>
18 <body>
19
20 <h1>🍽️ Our Menu</h1>
21 <div class="menu">
22   <div class="item">
23     <h3>Burger - ₹120</h3>
24     <label><input type="number" min="0" value="0" data-price="120"> Qty</label>
25   </div>
26   <div class="item">
27     <h3>Pizza - ₹200</h3>
28     <label><input type="number" min="0" value="0" data-price="200"> Qty</label>
29   </div>
30   <div class="item">
31     <h3>Pasta - ₹150</h3>
32     <label><input type="number" min="0" value="0" data-price="150"> Qty</label>
```

```
21 <div class="menu">
34   <div class="item">
36     <label><input type="number" min="0" value="0" data-price="00" /> Qty</label>
37   </div>
38 </div>
39
40 <div class="total">Total: ₹<span id="total">0</span></div>
41 <button onclick="calcTotal()">Check Order</button>
42 <p id="summary"></p>
43
44 <script>
45 function calcTotal(){
46   let total=0, summary="";
47   document.querySelectorAll('input[type=number]').forEach(i=>{
48     let qty=i.value, price=i.dataset.price;
49     if(qty>0){
50       total += qty*price;
51       summary += `${qty} x ₹${price}<br>`;
52     }
53   });
54   document.getElementById("total").innerText=total;
55   document.getElementById("summary").innerHTML=summary || "No items selected";
56 }
57 </script>
58
59 </body>
60 </html>
```

Output


 restaurantify


[Home](#) [Orders](#) [Menu Management](#) [Reports](#) [Reservations](#) 


Menu Categories

- Breakfast Dishes
- Vegan
- Vegetarian
- Appetizers
- Pasta
- Pizza


Menu Categories > Pasta









Cheese Ravioli
\$15.00




Lasagna
\$18.00




Pesto Spaghetti
\$14.00



Margherita Pizza
\$15.00




Tiramisu
\$8.00




Caesar Salad
\$12.00


Order

Fettuccine Bolognese	\$20,00
Subtotal	20,00
Tax	1.50
Service Charge	3,00
Discount	-2,00
Total	\$22,50

 Checkout



Fettuccine Bolognese \$20,00



Outcome

The project successfully delivers a fully functional, interactive menu system that can be used in real restaurant settings or as a prototype for larger systems. Users can easily browse, select, and customize their orders, and restaurant staff benefit from quick, clear order summaries. The lightweight design ensures fast load times and smooth operation, even on mobile devices. Feedback from initial testing indicates high usability and visual appeal, making it a strong candidate for real-world adoption.

Conclusion

This interactive restaurant menu project demonstrates how a well-structured front-end application can streamline the dining experience without relying on heavy back-end infrastructure. By combining responsive design, real-time cart calculations, and receipt generation, it offers both functionality and aesthetics. The modular structure makes it easy to extend with additional features like payment integration, inventory tracking, or multi-language support. Overall, it serves as a practical, adaptable solution for modern restaurants seeking to digitize their menu systems.

Reference

1.MDN Web Docs - HTML, CSS, and JavaScript tutorials and documentation.
Available at: <https://developer.mozilla.org/>

2.W3Schools - HTML, CSS, and JavaScript references with examples.
Available at: [<https://www.w3schools.com/>]