



CHRIST
(DEEMED TO BE UNIVERSITY)
BANGALORE · INDIA

**School of Engineering and Technology,
CHRIST (Deemed to be University)**

Department of Artificial Intelligence, Machine Learning
and Data Science

FRONT END UI/UX DEVELOPMENT

(BTOE361T5P)

CIA 3

PROJECT REPORT 1

Title: Online Food Ordering Website

3 BTCS AIML C

Submitted to: Mr. Narendra

Submitted by:

Ann Elizabeth (2462037) - ann.elizabeth@btech.christuniversity.in

Sasmita S (2462144) - sasmita.s@btech.christuniversity.in

Rachel Febin Pulicken (2462132) - rachel.febin@btech.christuniversity.in

CONTENTS

1. Abstract, Objectives and Scope of the Project.....	01
2. Tools and Technology used.....	02
3. HTML Structure Overview.....	03
4. CSS Style Strategy.....	03
5. Key Features.....	04
6. Challenges faced and Solutions.....	05
7. Future Enhancements.....	05
8. Sample Code.....	06
9. Screenshots of Final Output.....	07
10. Conclusion and References.....	08

ABSTRACT

The FoodieHub — Online Food Ordering System is a responsive web-based application designed to simplify the process of browsing restaurants, selecting dishes, and placing food orders. Built using HTML, CSS (Bootstrap 5), JavaScript, and jQuery, the system provides an interactive user experience with features such as a hero banner, dynamic restaurant listings, a real-time cart, and order tracking. Users can add menu items from different restaurants to their cart, view totals instantly, and proceed to checkout. The system also maintains an order history, simulating real-world delivery status updates (Preparing, Out for Delivery, Delivered). The project demonstrates the integration of modern front-end technologies with dynamic rendering, smooth navigation, and user-friendly design. It serves as a scalable model for developing practical online food delivery platforms.

OBJECTIVES

The main objective of the **FoodieHub — Online Food Ordering System** is to provide a simple, efficient, and interactive platform for users to order food online. The system aims to:

- Offer a user-friendly interface to browse restaurants and menus.
- Enable customers to add, update, or remove items in a real-time cart.
- Provide accurate billing and order summaries before checkout.
- Simulate order tracking with live status updates for better user engagement.
- Demonstrate the use of **web technologies (HTML, CSS, Bootstrap, JavaScript, jQuery)** in building responsive and dynamic applications.

SCOPE OF THE PROJECT

The **FoodieHub — Online Food Ordering System** is designed as a front-end web application that demonstrates the core functionalities of online food delivery platforms. The scope of the project covers the following areas:

- **User Experience:**
Users can explore restaurants, browse menus, and place orders through a clean and responsive interface accessible across devices.
- **Cart Management:**
Customers can add, update, and remove items from their cart, with real-time calculation of total cost.

- Order Tracking:**
 The system provides simulated live order status updates (Preparing → Out for Delivery → Delivered), enhancing realism.
- Dynamic Rendering:**
 Restaurant details and menus are generated dynamically from JavaScript objects, making the system scalable for adding more restaurants and dishes.
- Technology Demonstration:**
 The project showcases the integration of **HTML, CSS (Bootstrap 5), JavaScript, and jQuery** to build a modern, interactive, and responsive web application.
- Academic & Practical Relevance:**
 The project serves as a demonstration model for learning full-stack development concepts and can be extended into a real-world system by connecting it to a backend and database for user authentication, payments, and order persistence.

TOOLS AND TECHNOLOGY USED:

TOOL/TECHNOLOGY	PURPOSE
HTML5	Structuring the content and layout of the FoodieHub application using semantic elements such as <code><header></code> , <code><section></code> , <code><nav></code> , and <code><footer></code> for better accessibility and readability.
CSS3	Styling the webpage with colors, typography, spacing, shadows, and hover effects to enhance the visual hierarchy and user experience.
Bootstrap 5	Utilizing pre-designed components, cards, buttons, and a responsive grid system to create a modern, mobile-friendly layout.
Flexbox & Grid Layout	Arranging and aligning restaurant cards, menu items, and cart sections in a clean and structured way.

Responsive Design	Ensuring the application adapts seamlessly across desktops, tablets, and mobile devices.
JavaScript(ES6)	Adding interactivity such as adding items to the cart, calculating totals, and simulating order status updates.
jQuery	Simplifying DOM manipulation, event handling, and dynamically rendering restaurant menus, cart items, and order history.
External Assets(Texture & Icons)	Using placeholder images, icons, and background textures to improve visual appeal and realism.

HTML STRUCTURE OVERVIEW:

The FoodieHub website is structured using semantic HTML5 elements to ensure clarity, accessibility, and maintainability. The `<head>` section includes metadata, the page title, viewport settings, and links to external CSS and Bootstrap for styling. The `<body>` is divided into distinct sections: a fixed-top navbar for navigation, a hero banner with a call-to-action button, a restaurants section displaying dynamic restaurant cards with menu items, a cart section showing selected items and total price, and an orders section listing past orders with status updates. A footer provides copyright and credits. Interactive functionality, such as adding items to the cart, calculating totals, and simulating order delivery, is handled using JavaScript and jQuery, while Bootstrap ensures the layout is responsive across devices.

CSS STYLE STRATEGY:

The CSS style strategy for FoodieHub focuses on creating a visually appealing, responsive, and user-friendly interface. Bootstrap 5 provides the foundation for a responsive grid layout, pre-designed components, and consistent spacing. Custom CSS3 is used to enhance the design, including hero banner backgrounds, card shadows, hover effects, and typography to establish a clear visual hierarchy. Flexbox is employed to align elements within cards, menus, and cart items, ensuring proper spacing and alignment. The hero section uses a full-screen background with cover and center properties for an attractive layout. Colors, font sizes, and

button styles are chosen for readability and call-to-action clarity. The design also incorporates responsive techniques, ensuring the layout adapts seamlessly across desktops, tablets, and mobile devices, while maintaining usability and aesthetic consistency.

KEY FEATURES :

Feature	Description
Dynamic Restaurant Listings	Displays restaurants and their menus dynamically from JavaScript objects, allowing easy updates and scalability.
Interactive Cart System	Users can add items to the cart, adjust quantities, remove items, and see the total cost in real-time.
Checkout Functionality	Enables users to place orders with a single click, with the checkout button disabled if the cart is empty.
Order Tracking	Simulates live order status updates: Preparing → Out for Delivery → Delivered, enhancing user engagement.
Responsive and Mobile-Friendly Design	Uses Bootstrap 5 grid system and Flexbox to ensure the website adapts seamlessly across desktops, tablets, and mobile devices.
User-Friendly Interface	Clean layout with hero banner, restaurant cards, buttons, and hover effects for easy navigation and interaction.
Visual Appeal	Incorporates images, icons, shadows, and consistent color schemes to make the site attractive and intuitive.
Smooth Navigation	Navbar links allow smooth scrolling to key sections: Home, Restaurants, Cart, and Orders.

Real-Time Updates	Cart totals and order lists update immediately as users interact with the system, providing a dynamic experience.
--------------------------	---

CHALLENGES FACED AND SOLUTIONS

Challenge	Solution
Cluttered restaurant and cart layout	Used Bootstrap grid system and spacing utilities for clean organization of restaurant cards, menu items, and cart entries.
Poor mobile responsiveness	Implemented Bootstrap's responsive classes and Flexbox to ensure proper alignment and adaptability on all devices.
Real-time cart and order updates	Applied JavaScript and jQuery to dynamically update cart totals, item quantities, and order lists instantly.
Simulating order tracking without a backend	Utilized JavaScript timers (setTimeout) to mimic order status progression (Preparing → Out for Delivery → Delivered).
Inconsistent styling across sections	Adopted a unified color palette, consistent fonts, shadows, and spacing for a cohesive visual design.
Low user engagement	Added interactive buttons, hover effects, and smooth scrolling to make navigation and ordering more intuitive.

FUTURE ENHANCEMENTS

In the future, FoodieHub can be improved by integrating a backend system and a database to store user accounts, restaurant details, and order history permanently. This would allow users to maintain their cart, view past orders, and personalize their experience with saved addresses and favorite restaurants. Adding a payment gateway would enable secure transactions, while search and filter functionality could help users quickly find restaurants based on cuisine, rating, or price.

Additionally, the platform could include live delivery tracking with maps for a more realistic experience and enhanced user engagement. Improvements in animations, accessibility, and mobile interactions would further elevate the user interface, making the website more

interactive, inclusive, and professional. These enhancements would transform FoodieHub into a comprehensive, production-ready online food ordering system.

SAMPLE CODE:

```
<!doctype html>
<html lang="en">
<head>
  <meta charset="utf-8"/>
  <meta name="viewport" content="width=device-width, initial-scale=1"/>
  <title>FoodieHub — Online Food Ordering</title>
  <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.2/dist/css/bootstrap.min.css" rel="stylesheet"/>

  <style>
    body {
      scroll-behavior: smooth;
    }
    /* Hero banner */
    .hero {
      background: url('https://picsum.photos/seed/foodhero/1600/600') center/cover no-repeat;
      height: 80vh;
    }
    /* Restaurant cards */
    .restaurant-card img {
      height: 180px;
      object-fit: cover;
    }
    .restaurant-card .card-body {
      display: flex;
      flex-direction: column;
      justify-content: space-between;
    }
  </style>
</head>
<body>
  
  <nav class="navbar navbar-expand-lg navbar-dark bg-primary fixed-top shadow">
    <div class="container">
      <a class="navbar-brand fw-bold" href="#">🍴 FoodieHub</a>
      <button class="navbar-toggler" type="button" data-bs-toggle="collapse" data-bs-target="#navMenu">
        <span class="navbar-toggler-icon"></span>
      </button>
      <div id="navMenu" class="collapse navbar-collapse">
        <ul class="navbar-nav ms-auto">
          <li class="nav-item"><a class="nav-link active" href="#">Home</a></li>
          <li class="nav-item"><a class="nav-link" href="#restaurants">Restaurants</a></li>
```

```

<li class="nav-item"><a class="nav-link" href="#cart">Cart</a></li>
<li class="nav-item"><a class="nav-link" href="#orders">Orders</a></li>
</ul>
</div>
</div>
</nav>

<!-- Hero Section -->
<header class="hero d-flex align-items-center text-center text-white">
  <div class="container">
    <h1 class="fw-bold">Delicious food, delivered fast</h1>
    <p class="lead">Order from your favorite restaurants near you.</p>
    <a href="#restaurants" class="btn btn-warning btn-lg mt-3">Explore Restaurants</a>
  </div>
</header>

<!-- Restaurants Section -->
<section id="restaurants" class="py-5">
  <div class="container">
    <h2 class="text-center mb-4">Popular Restaurants</h2>
    <div id="restaurantsRow" class="row g-4"></div>
  </div>
</section>

<!-- Cart Section -->
<section id="cart" class="bg-light py-5">
  <div class="container">
    <h2 class="text-center mb-4">Your Cart</h2>
    <div id="cartItems" class="mb-3"></div>
    <div class="d-flex justify-content-between align-items-center">
      <div><strong>Total:</strong> ₹<span id="cartTotal">0</span></div>
      <button id="checkoutBtn" class="btn btn-success" disabled>Checkout</button>
    </div>
  </div>
</section>

<!-- Orders Section -->
<section id="orders" class="py-5">
  <div class="container">
    <h2 class="text-center mb-4">My Orders</h2>
    <div id="ordersList" class="text-muted">No orders placed yet.</div>
  </div>
</section>

<!-- Footer -->
<footer class="bg-dark text-white text-center py-3 mt-5">
  <small>© 2025 FoodieHub | Built with ❤ for demo</small>
</footer>

<!-- Scripts -->

```

```

<script src="https://code.jquery.com/jquery-3.7.1.min.js"></script>
<script
src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.2/dist/js/bootstrap.bundle.min.js"></script>
<script>
$(function(){
  const restaurants = [
    { id:1, name:"Spice Villa", cuisine:"Indian",
img:"https://picsum.photos/seed/indian/400/300",
      menu:[
        { id:101, name:"Butter Chicken", price:240 },
        { id:102, name:"Paneer Tikka", price:180 },
        { id:103, name:"Garlic Naan", price:40 }
      ]
    },
    { id:2, name:"Sushi House", cuisine:"Japanese",
img:"https://picsum.photos/seed/japanese/400/300",
      menu:[
        { id:201, name:"Salmon Nigiri", price:220 },
        { id:202, name:"Miso Soup", price:80 },
        { id:203, name:"California Roll", price:260 }
      ]
    },
    { id:3, name:"La Pasta", cuisine:"Italian",
img:"https://picsum.photos/seed/italian/400/300",
      menu:[
        { id:301, name:"Carbonara", price:230 },
        { id:302, name:"Margherita Pizza", price:270 },
        { id:303, name:"Tiramisu", price:120 }
      ]
    }
  ];
}

let cart = {};
let orders = [];

function renderRestaurants(){
  const $row = $('#restaurantsRow').empty();
  restaurants.forEach(r=>{
    const col = $(`

<div class="card restaurant-card shadow">
        
        <div class="card-body">
          <h5>${r.name}</h5>
          <p class="small text-muted">${r.cuisine}</p>
          <div id="menu-$ {r.id}"></div>
        </div>
      </div>
    </div>`);
    r.menu.forEach(m=>{


```

```

        col.find(`#menu-${r.id}`).append(`
            <div class="d-flex justify-content-between align-items-center my-1">
                <span>${m.name} - ₹${m.price}</span>
                <button class="btn btn-sm btn-primary add-to-cart" data-rid="${r.id}" data-
id="${m.id}">Add</button>
            </div>');
        });
        $row.append(col);
    );
}

function addToCart(rid, mid){
    const r = restaurants.find(x=>x.id==rid);
    const item = r.menu.find(m=>m.id==mid);
    if(!cart[mid]) cart[mid]={...item, qty:0};
    cart[mid].qty++;
    renderCart();
}

function renderCart(){
    const $c = $('#cartItems').empty();
    let total=0, count=0;
    Object.values(cart).forEach(it=>{
        const sub=it.qty*it.price;
        total+=sub; count++;
        $c.append(`
            <div class="d-flex justify-content-between align-items-center border-bottom py-2">
                <div>${it.name} x${it.qty} = ₹${sub}</div>
                <button class="btn btn-sm btn-danger remove" data-
id="${it.id}">Remove</button>
            </div>`);
    });
    $('#cartTotal').text(total);
    $('#checkoutBtn').prop('disabled', count==0);
}

function removeItem(id){
    delete cart[id];
    renderCart();
}

function checkout(){
    if(Object.keys(cart).length==0) return;
    const order = {
        id:"ORD"+Date.now(),
        items:Object.values(cart),
        total:Object.values(cart).reduce((s,it)=>s+it.qty*it.price,0),
        status:"Preparing"
    };
    orders.unshift(order);
}

```

```

        cart={};
        renderCart();
        renderOrders();
        alert("Order placed! ID: "+order.id);
        // simulate status updates
        setTimeout(()=>{order.status="Out for delivery"; renderOrders();},4000);
        setTimeout(()=>{order.status="Delivered"; renderOrders();},8000);
    }

    function renderOrders(){
        const $o = $('#ordersList').empty();
        if(orders.length==0){$o.text("No orders placed yet."); return;}
        orders.forEach(or=>{
            $o.append(` 
                <div class="border rounded p-2 mb-2">
                    <strong>${or.id}</strong> - ₹${or.total}
                    <div class="small text-muted">${or.items.map(it=>it.qty+"x "+it.name).join(",")}</div>
                    <div class="fw-bold text-primary">Status: ${or.status}</div>
                </div>');
        });
    }

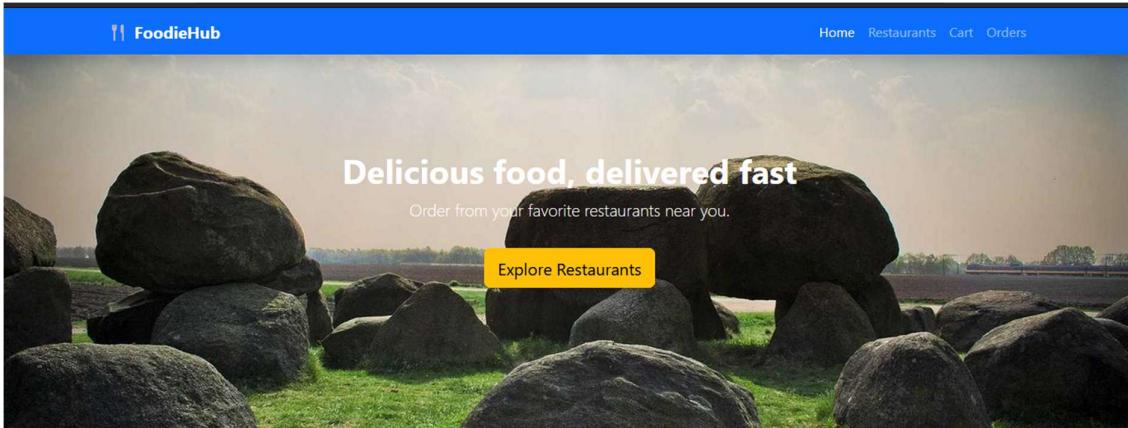
    // Events
    $(document).on('click','.add-to-cart',function(){
        addToCart($(this).data('rid'),$(this).data('id'));
    });
    $(document).on('click','.remove',function(){ removeItem($(this).data('id'))});
    $('#checkoutBtn').click(checkout);

    // init
    renderRestaurants();
    renderCart();
    renderOrders();
});

</script>
</body>
</html>

```

SCREENSHOTS OF FINAL OUTPUT:



Popular Restaurants

This screenshot displays a section titled 'Popular Restaurants' on the FoodieHub website. It features three cards, each representing a different restaurant with its name, cuisine type, menu items, and small 'Add' buttons. The first card is for 'Spice Villa' (Indian cuisine), listing Butter Chicken (₹240), Paneer Tikka (₹180), and Garlic Naan (₹40). The second card is for 'Sushi House' (Japanese cuisine), listing Salmon Nigiri (₹220), Miso Soup (₹80), and California Roll (₹260). The third card is for 'La Pasta' (Italian cuisine), listing Carbonara (₹230), Margherita Pizza (₹270), and Tiramisu (₹120).

Your Cart

The screenshot shows the 'Your Cart' page of the FoodieHub app. It lists three items: Butter Chicken (₹240), Miso Soup (₹80), and Tiramisu (₹120). Each item has a red 'Remove' button to its right. At the bottom, it shows a total of ₹440 and a green 'Checkout' button.

My Orders

The screenshot shows the 'My Orders' page. It displays a single order entry in a light gray box: 'ORD1758858195444 - ₹240', '1x Butter Chicken', and 'Status: Delivered'.

CONCLUSION:

The FoodieHub Online Food Ordering System demonstrates the effective use of modern web technologies, including **HTML5, CSS3, Bootstrap 5, JavaScript, and jQuery**, to build a responsive, interactive, and user-friendly platform. It allows users to browse restaurants, add menu items to a cart, place orders, and track them with simulated real-time updates. The project showcases dynamic content rendering, seamless cart management, and a visually appealing layout, making it an excellent model for online food delivery applications. With future enhancements like backend integration, payment systems, and live delivery tracking, FoodieHub has the potential to evolve into a fully functional and practical food ordering solution.

REFERENCES:

L&T LMS: <https://learn.lntedutech.com/Landing/MyCourse>