A Field Project Report on

FOOD QUEST

Submitted

In partial fulfillment of the requirements for the award of the degree

BACHELOR OF TECHNOLOGY

In

COMPUTER SCIENCE and ENGINEERING

By

K.Amrutha	(231FA04A65)
K.V.Lakshmi	(231FA04A78)
R.Sai Deepthi	(231FA04B01)
G.Naga Manju Sravanthi	(231FA04B73)

Under the Guidance of

Name of the Supervisor

Assistant Professor, CSE

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

SCHOOL OF COMPUTING AND INFORMATICS

VIGNAN'S FOUNDATION FOR SCIENCE, TECHNOLOGY & RESEARCH
(Deemed to be University)
Vadlamudi, Guntur -522213, INDIA.

April, 2025

VIGNAN'S
FOUNDATION FOR SCIENCE, TECHNOLOGY & RESEARCH
(Deemed to be University) - Estd. u/s 3 of UGC Act 1956

CERTIFICATE

This is to certify that the field project entitled "FOOD QUEST" is being submitted by **[K.Amrutha], [231FA04A65]**, **[K.V.Lakshmi], [231FA04A78]**, **[R.Sai Deepthi], [231FA04B01]**, and **[G.Naga Manju Sravanthi], [231FA04B73]** in partial fulfilment of the requirements for the degree of **Bachelor of Technology (B.Tech.) in Computer Science and Engineering** at Vignan's Foundation for Science, Technology and Research (Deemed to be University), Vadlamudi, Guntur District, Andhra Pradesh, India.

This is a bonafide work carried out by the aforementioned students under my guidance and supervision.

Guide

Project Review Committee

HoD, CSE

HoD
Dept. of Computer Science & Engineering
VFSTR Deemed to be Univers
VADLAMUDI - 522 213
Guntur Dist., A.P., India.

VIGNAN'S
FOUNDATION FOR SCIENCE, TECHNOLOGY & RESEARCH
(Deemed to be University) - Estd. u/s 3 of UGC Act 1956

DECLARATION

Date:

We hereby declare that the work presented in the field project titled "FOOD QUEST" is the result of our own efforts and investigations.

This project is being submitted under the supervision of Mr. K. Pavan Kumar sir, Designation in partial fulfillment of the requirements for the Bachelor of Technology (B.Tech.) degree in Computer Science and Engineering at Vignan's Foundation for Science, Technology and Research (Deemed to be University), Vadlamudi, Guntur, Andhra Pradesh, India.

K.Amrutha	(231FA04A65)	Signature K. Amrutha
K.V.Lakshmi	(231FA04A78)	Signature K.V. Lakshmi
R.Sai Deepthi	(231FA04B01)	Signature R. Sai Deepthi
G.Naga.Manju.Sravanthi	(231FA04B73)	Signature G. Naga Manju Sravanthi

Contents

S. No	Description	Page. No
1	Introduction:problem definition,existing system,proposed system,literature review	
2	System requirements:hardware&software requirements,SRS	
3	System design:modules of system,uml diagrams	
4	Description	
5	Implementation,Testcases	
6	Conclusion	

1. INTRODUCTION

FOOD QUEST is an online food ordering application designed to make the process of ordering meals easy and efficient. The system provides users with a categorized menu of vegetarian and non-vegetarian dishes, allowing them to select their favorite items quickly. After selecting the dishes, users can fill in their personal details and choose a preferred payment method from options like UPI, Credit/Debit Card, or Cash on Delivery.

PROBLEM DEFINITION

Today, food lovers prefer ordering food online for convenience and time-saving. However, many existing systems are complicated, non-user-friendly, or lack quick ordering features. The need is to develop a simple, intuitive, and interactive food ordering system where users can easily browse a menu, select food, place an order, and make payments effortlessly.

EXISTING SYSTEM

In existing food ordering platforms:

- Users often face lengthy login and checkout procedures.
- Menus are cluttered and confusing.
- Payment gateways are sometimes unstable.
- Lack of lightweight and fast-loading user interfaces.

PROPOSED SYSTEM

The proposed system, FOOD QUEST, is a web-based application designed with a clean interface allowing users to:

- View a categorized food menu (Vegetarian and Non-Vegetarian).
- Select food items by clicking "Order Now."
- Fill a simple order form.
- Choose preferred payment methods like UPI, Credit/Debit Card, or Cash on Delivery.
- Receive order confirmation immediately after submission.

LITERATURE REVIEW

Various food ordering systems such as Zomato, Swiggy, and UberEats were studied. These applications provided insights into user needs, common system flaws, and improvement areas. Research papers on user experience (UX) design and responsive web development were also reviewed to develop a highly usable system.

2. SYSTEM REQUIREMENTS

HARDWARE & SOFTWARE REQUIREMENTS

Hardware Requirements:

- Processor: Intel Core i3/i5 or equivalent
- RAM: 4GB minimum
- Hard Disk: 500GB minimum
- Internet Connection

Software Requirements:

- Web Browser (Chrome, Firefox, Edge)
- Code Editor (VS Code, Sublime Text)
- Frontend Technologies: HTML, CSS, JavaScript
- (Optional) Web server for hosting (Apache, Node.js)

Software Requirements Specification (SRS)

- Functional Requirements:
    - Display food menu.
    - Allow item selection.
    - Fill user information form.
    - Select and process payment method.
    - Show order success or cancellation message.
- Non-Functional Requirements:
    - Fast loading time.
    - Mobile responsiveness.
    - User-friendly UI.
    - Secure form handling.

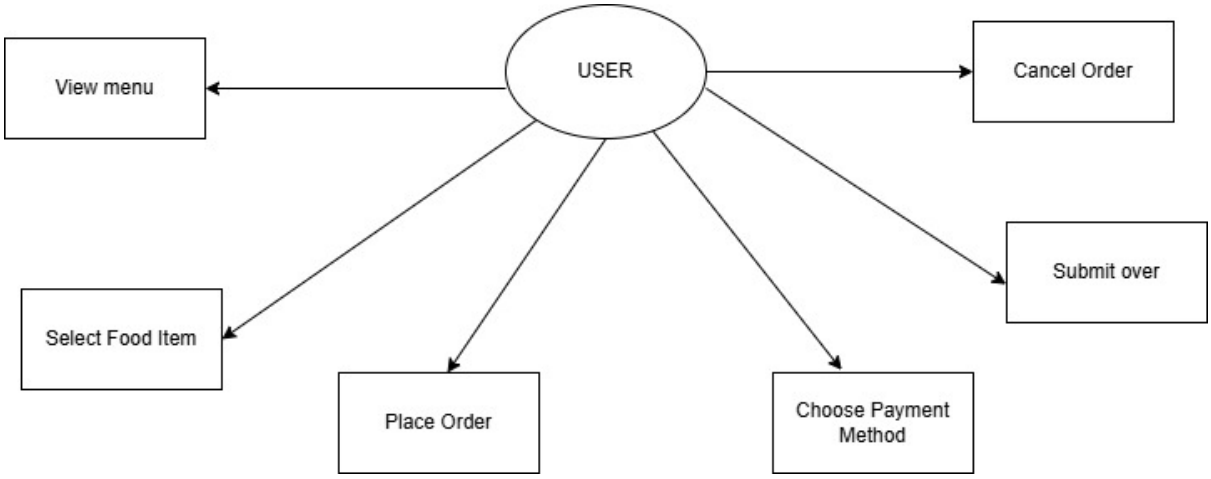
3. SYSTEM DESIGN

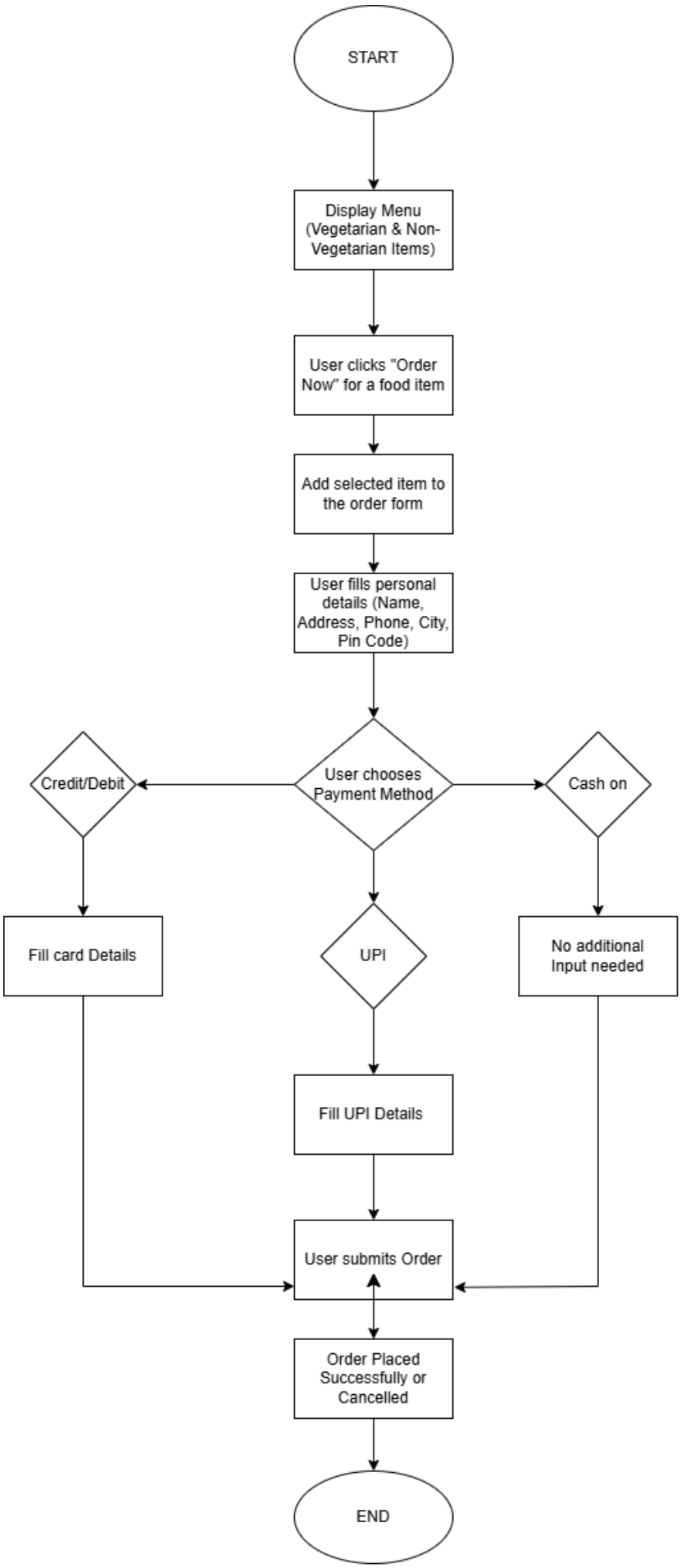
MODULES OF SYSTEM

- Menu Module: Displays the Vegetarian and Non-Vegetarian items.
- Order Form Module: Captures user details and selected food items.
- Payment Module: Provides multiple payment options (UPI, Card, COD).
- Confirmation Module: Displays success or cancellation messages after order actions.

UML DIAGRAMS

- Use Case Diagram: Illustrating user interactions with the system.
- Flowchart: Step-by-step working of the system.
- Sequence Diagram: User interactions with the system sequentially.




START
Display Menu (Vegetarian & Non-Vegetarian Items)
User clicks "Order Now" for a food item
Add selected item to the order form
User fills personal details (Name, Address, Phone, City, Pin Code)
User chooses Payment Method
Credit/Debit
Cash on
Fill card Details
UPI
No additional Input needed
Fill UPI Details
User submits Order
Order Placed Successfully or Cancelled
END

DESCRIPTION:

- Actor:
    - User (depicted as an oval in the center) is the primary actor who interacts with the system.
- Use Cases (Functionalities):
    - View Menu – The user can browse the food items available.
    - Select Food Item – The user can choose one or more food items from the menu.
    - Place Order – After selecting items, the user places an order.
    - Choose Payment Method – The user can choose how to pay (e.g., card, UPI, cash).
    - Cancel Order – If needed, the user can cancel a placed order.
    - Submit Over – Final submission of the order, likely indicating confirmation or end of the transaction.

Connections (Relationships):

- All arrows point from or toward the User, indicating the actions they initiate or are involved in.
- The relationships are associations, showing what the user can do within the system.

This diagram provides a high-level view of how a user interacts with an online food ordering system and the main functionalities available to them.

4. IMPLEMENTATION:

- HTML: Page structure (Menu, Forms)
- CSS: Styling (Color themes, Layout, Animations)
- JavaScript: Interactivity (Add item to cart, Handle forms, Payment selection)

SAMPLE CODE:

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Food Quest Menu</title>
    <style>
        body {
            font-family: 'Arial', sans-serif;
            background-color: #000;
            color: #fff;
            margin: 0;
            padding: 0;
            text-align: center;
        }

        .container {
            max-width: 1200px;
            margin: 0 auto;
            background-color: #333;
            padding: 30px;
            box-shadow: 0 10px 30px rgba(0, 0, 0, 0.1);
        }

        h1 {
            font-family: 'Georgia', serif;
            color: #f04e23;
```

```
    font-size: 36px;
    margin-bottom: 40px;
    text-transform: uppercase;
}

.menu-section {
    margin-bottom: 40px;
}

.menu-section h2 {
    color: #b5d408;
    font-size: 28px;
    margin-bottom: 20px;
    text-align: left;
    font-weight: bold;
}

.menu-items {
    display: flex;
    flex-wrap: wrap;
    justify-content: space-between;
}

.menu-item {
    width: 30%;
    background-color: #444;
    padding: 20px;
    margin-bottom: 20px;
    border-radius: 8px;
    box-shadow: 0 4px 12px rgba(0, 0, 0, 0.1);
    transition: transform 0.3s ease;
    color: #fff;
    text-align: center;
}
```

```
.menu-item:hover {
    transform: translateY(-10px);
    background-color: #555;
}

.menu-item h3 {
    font-size: 24px;
    color: #d40819;
    margin-top: 10px;
}

.menu-item .price {
    font-size: 18px;
    color: #28a745;
    margin-top: 10px;
}

.menu-item p {
    font-size: 16px;
    color: #ddd;
    margin-top: 10px;
}

.order-now {
    color: #010005;
    text-decoration: none;
    margin-top: 10px;
    display: inline-block;
    padding: 8px 20px;
    background-color: #d5eb67;
    border-radius: 5px;
    transition: background-color 0.3s;
}
```

```
.order-now:hover {
    background-color: #218838;
}

.order-form {
    max-width: 600px;
    margin: 30px auto;
    padding: 30px;
    background-color: #FFFFFF;
    border-radius: 8px;
    box-shadow: 0 10px 30px rgba(0, 0, 0, 0.1);
    color: #000;
}

.order-form input,
.order-form button {
    width: 100%;
    padding: 12px;
    margin: 10px 0;
    border-radius: 5px;
    border: 1px solid #ddd;
}

.order-form button {
    background-color: #28a745;
    color: white;
    border: none;
    cursor: pointer;
}

.order-form button:hover {
    background-color: #218838;
}
```

```
        #paymentForm label {
            font-size: 16px;
            display: block;
            margin-bottom: 10px;
            color: #000;
        }

        #paymentForm button {
            margin-top: 10px;
            background-color: #007bff;
            color: white;
            border: none;
            cursor: pointer;
        }

        #paymentForm button:hover {
            background-color: #0056b3;
        }

        #paymentDetails,
        #creditCardDetails {
            display: none;
        }

        footer {
            background-color: #333;
            color: #fff;
            text-align: center;
            padding: 20px;
            margin-top: 50px;
        }
    </style>
</head>
```

```
<body>

  <div class="container">
    <h1>Food Quest</h1>

    <!-- Menu Sections -->
    <div class="menu-section">
      <h2>Vegetarian Menu</h2>
      <div class="menu-items">
        <div class="menu-item">
          <h3>Grilled Veggie Salad</h3>
          <div class="price">Rs. 400/-</div>
          <a href="#order-form" class="order-now" onclick="addToOrder('Grilled Veggie Salad')">Order Now</a>
        </div>
        <div class="menu-item">
          <h3>Panner Tikka</h3>
          <div class="price">Rs. 350/-</div>
          <a href="#order-form" class="order-now" onclick="addToOrder('Panner Tikka')">Order Now</a>
        </div>
        <div class="menu-item">
          <h3>Avocado Bowl</h3>
          <div class="price">Rs. 450/-</div>
          <a href="#order-form" class="order-now" onclick="addToOrder('Avocado Bowl')">Order Now</a>
        </div>
      </div>
    </div>

    <div class="menu-section">
      <h2>Non-Vegetarian Menu</h2>
      <div class="menu-items">
        <div class="menu-item">
          <h3>Butter Chicken</h3>
          <div class="price">Rs. 650/-</div>
```

```
            <a href="#order-form" class="order-now" onclick="addToOrder('Butter Chicken')">Order Now</a>
        </div>
        <div class="menu-item">
            <h3>Fish Tikka</h3>
            <div class="price">Rs. 650/-</div>
            <a href="#order-form" class="order-now" onclick="addToOrder('Fish Tikka')">Order Now</a>
        </div>
        <div class="menu-item">
            <h3>Prawns Curry</h3>
            <div class="price">Rs. 850/-</div>
            <a href="#order-form" class="order-now" onclick="addToOrder('Prawns Curry')">Order Now</a>
        </div>
    </div>
</div>

<!-- Order Form -->
<div id="order-form" class="order-form">
    <h3>Place Your Order</h3>

    <form id="orderForm" onsubmit="showSuccessMessage(event)">
        <div id="selected-items">
            <!-- Selected food items will appear here -->
        </div>

        <input type="text" placeholder="Your Name" required><br>
        <input type="text" placeholder="Your Address" required><br>
        <input type="tel" placeholder="Your Phone Number" required><br>
        <input type="text" placeholder="Your City" required><br>
        <input type="text" placeholder="Your Pin Code" required><br>
        <button type="submit">Submit Order</button>
        <button type="button" onclick="cancelOrder()">Cancel Order</button>
    </form>

    <!-- Payment Section -->
```

```
<h3>Select Payment Method</h3>

<form id="paymentForm" onsubmit="submitPayment(event)">

<label><input type="radio" name="payment" value="UPI" onchange="showPaymentOptions()"> UPI</label><br>

<label><input type="radio" name="payment" value="Credit/Debit Card" onchange="showPaymentOptions()"> Credit/Debit Card</label><br>

<label><input type="radio" name="payment" value="Cash on Delivery" onchange="showPaymentOptions()"> Cash on Delivery</label><br>


<div id="paymentDetails">

<label><input type="radio" name="upi" value="GPay"> Google Pay</label><br>

<label><input type="radio" name="upi" value="PhonePe"> PhonePe</label><br>

</div>


<div id="creditCardDetails">

<input type="text" placeholder="Card Number" required><br>

<input type="text" placeholder="Cardholder Name" required><br>

<input type="text" placeholder="Expiry Date (MM/YY)" required><br>

<input type="text" placeholder="CVV" required><br>

</div>


<button type="submit">Confirm Payment</button>

</form>


<div id="success-message" style="display:none; color: green;">Your order has been successfully placed!</div>

<div id="cancel-message" style="display:none; color: red;">Your order has been canceled.</div>

<div id="payment-success" style="display:none; color: green; margin-top: 10px;">Payment method selected successfully!</div>

</div>


</div>


<!-- Footer -->

<footer>
```

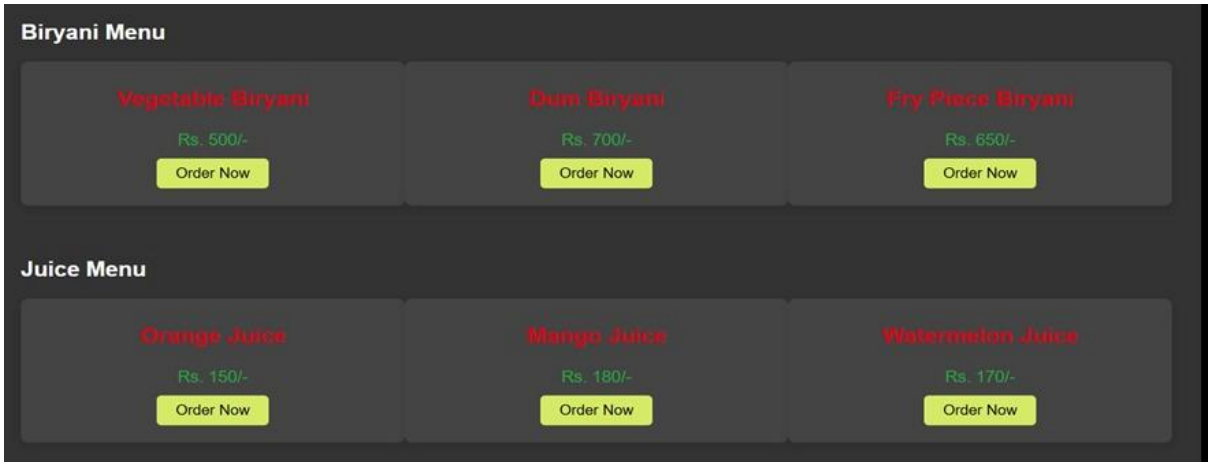
```
    <p>&copy; 2025 Food Quest. All Rights Reserved.</p>
</footer>

<script>
    // Function to add selected item to the order form
    function addToOrder(item) {
        const selectedItemsList = document.getElementById('selected-items');
        selectedItemsList.innerHTML = '';  // Clear any previous selection
        const foodItem = document.createElement('p');
        foodItem.textContent = item;
        selectedItemsList.appendChild(foodItem);
    }

    // Function to show selected items before order submission
    function showSuccessMessage(event) {
        event.preventDefault();

        const selectedItems = document.querySelectorAll('input[name="food"]:checked');
        const selectedItemsList = document.getElementById('selected-items');
        selectedItemsList.innerHTML = '';

        selectedItems.forEach(item => {
            const foodItem = document.createElement('p');
            foodItem.textContent = item.value;
            selectedItemsList.appendChild(foodItem);
        });

        document.getElementById("success-message").style.display = "block";
        document.getElementById("cancel-message").style.display = "none";
    }

    function cancelOrder() {
        document.getElementById("orderForm").reset();
        document.getElementById("selected-items").innerHTML = '';
```

```
        document.getElementById("success-message").style.display = "none";
        document.getElementById("cancel-message").style.display = "block";
    }

    function showPaymentOptions() {
        const paymentMethod = document.querySelector('input[name="payment"]:checked').value;
        if (paymentMethod === "UPI") {
            document.getElementById("paymentDetails").style.display = "block";
            document.getElementById("creditCardDetails").style.display = "none";
        } else if (paymentMethod === "Credit/Debit Card") {
            document.getElementById("creditCardDetails").style.display = "block";
            document.getElementById("paymentDetails").style.display = "none";
        } else {
            document.getElementById("paymentDetails").style.display = "none";
            document.getElementById("creditCardDetails").style.display = "none";
        }
    }

    function submitPayment(event) {
        event.preventDefault();
        document.getElementById("payment-success").style.display = "block";
    }
  </script>

</body>
</html>
```

TEST CASES:

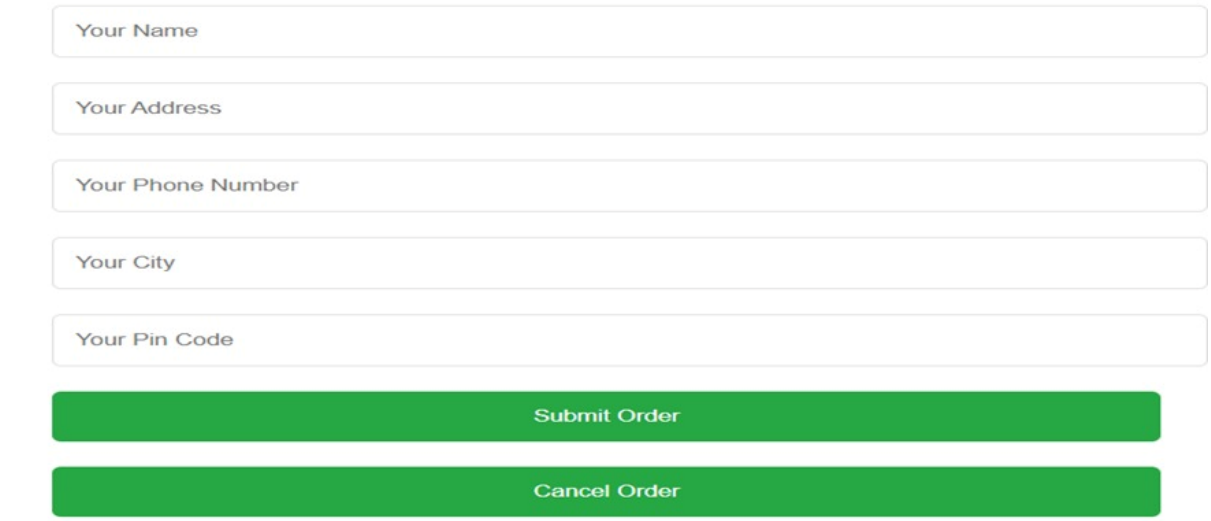
Test Case ID	Test Scenario	Input	Expected Output	Status
TC001	Select Food Item	Click on "Order Now"	Item added to order form	Pass
TC002	Submit Order Form	Fill valid data	Order success message	Pass
TC003	Cancel Order	Click "Cancel Order"	Order canceled message	Pass
TC004	Choose Payment	Select UPI/Card	Relevant payment options appear	Pass

OUTPUT SCREENS:

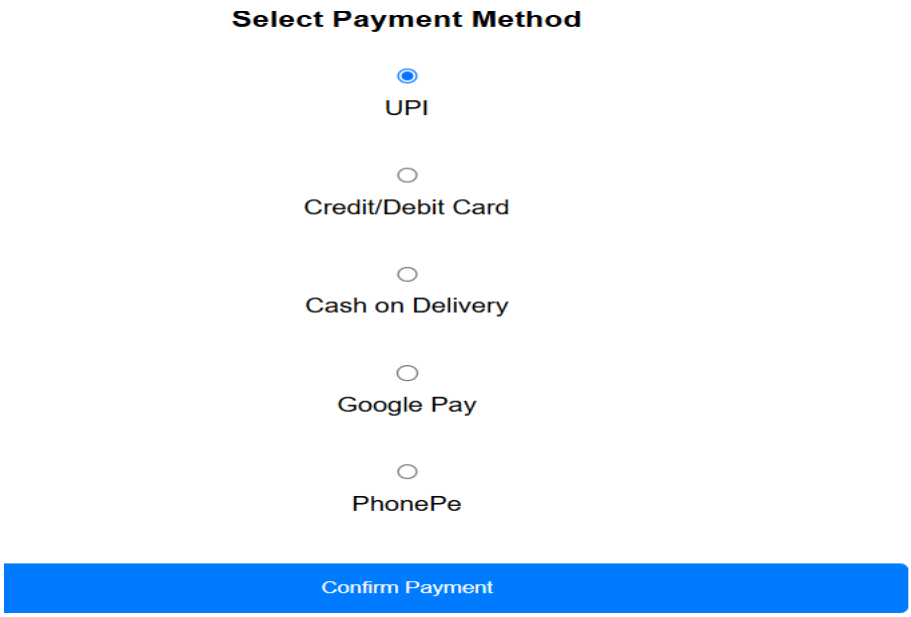


The landing page of the "FOOD QUEST" application displays the available food items neatly categorized into Vegetarian Menu and Non-Vegetarian Menu.

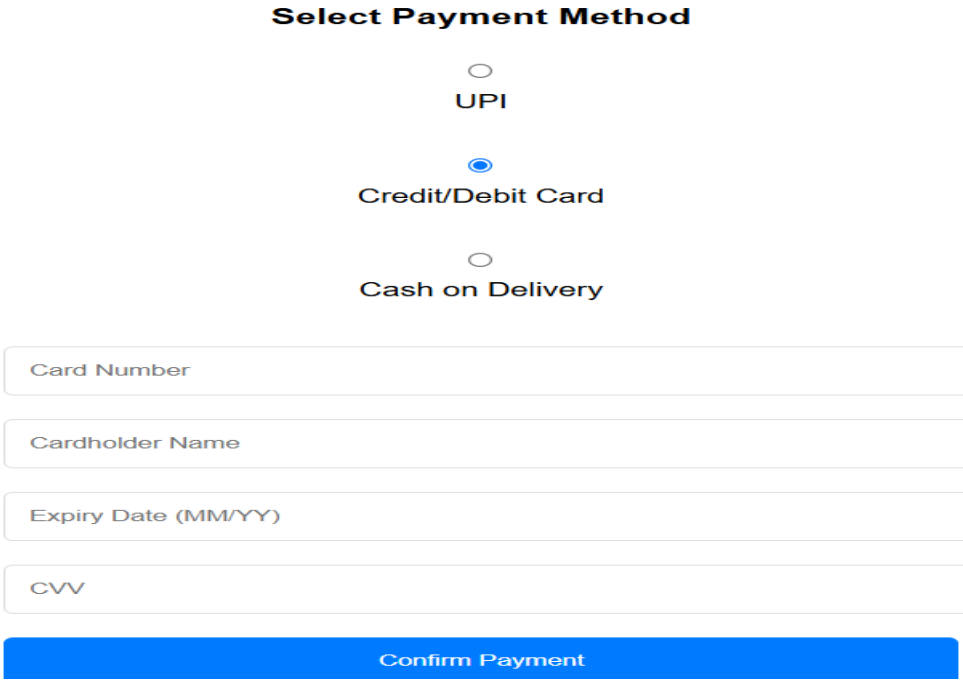
The landing page of the "FOOD QUEST" application displays the available food items neatly categorized into Vegetarian Menu and Non-Vegetarian Menu.



Once a food item is selected, users are redirected to the Order Form. The order form ensures essential delivery information is captured before proceeding to payment.



The payment section adapts based on the user's selection to make the process flexible and quick.



When Credit/Debit Card is selected, users are shown a Card Details Form. This ensures secure and structured payment input for online card transactions.

CONCLUSION:

The FOOD QUEST project successfully achieves its objective of creating a simple, user-friendly, and efficient online food ordering system. By providing a clean and intuitive interface, users can easily browse a categorized food menu, select their desired dishes, fill in delivery details, and complete their orders using a variety of payment methods.

The project demonstrates how frontend technologies like HTML, CSS, and JavaScript can be combined to deliver a smooth and interactive user experience without the need for complex backend operations. Features like real-time order selection, dynamic payment options, and instant order confirmation significantly enhance user satisfaction.

Overall, FOOD QUEST offers a lightweight, responsive, and practical solution for small restaurants or startups looking to implement an easy-to-manage online ordering system.

In the future, this project can be enhanced by:

- Integrating a backend database for managing orders and user data.
- Adding login and registration modules for user accounts.
- Providing real-time order tracking and estimated delivery times.
- Expanding the payment gateway to include wallets and net banking options.

Thus, FOOD QUEST stands as a strong prototype for a modern food ordering system, balancing functionality and simplicity effectively.

REFERENCES:

W3Schools
HTML, CSS, JavaScript Tutorials
https://www.w3schools.com

MDN Web Docs
JavaScript and Web Technologies Documentation
https://developer.mozilla.org

Bootstrap Documentation
for layout inspiration and responsive design ideas
https://getbootstrap.com

GeeksforGeeks
HTML Forms, JavaScript Event Handling and UI Design Concepts
https://www.geeksforgeeks.org

Zomato Website
Study of food ordering system user interface and user experience
https://www.zomato.com

Swiggy Website
Analysis of existing online food ordering workflows
https://www.swiggy.com

Research Papers on UX/UI Design
For studying user interface improvements and responsive web design

- "Principles of User Interface Design" by Bruce Tognazzini
- "Responsive Web Design Basics" from Google Developers

YouTube Tutorials
JavaScript Form Validation and Simple Website Projects

- "Online Food Order Website using HTML, CSS, JavaScript" – Easy Tutorials Channel.