

# **FOODGO - A WEB BASED FOOD DELIVERY PLATFORM**

PROJECT REPORT SUBMITTED IN PARTIAL FULFILMENT OF THE  
REQUIREMENTS FOR THE AWARD OF

## **BACHELOR OF COMPUTER APPLICATIONS**

To

**MARIAN COLLEGE KUTTIKKANAM AUTONOMOUS**

Affiliated to

MAHATMA GANDHI UNIVERSITY, KOTTAYAM

By

**VYSAKH B**

(Reg.No:23UBC260)

Guided By

**Dr. SHEELA S**



**DEPARTMENT OF COMPUTER APPLICATIONS**

**MARIAN COLLEGE KUTTIKKANAM AUTONOMOUS**

**PEERMADE – 685531**

## DECLARATION

I, Vysakh B [Reg.no 23UBC260], certify that the Mini Project Report entitled "FoodGo" is an authentic work carried out by me at Marian College Kuttikkanam Autonomous. The matter embodied in this project work has not been submitted elsewhere for the award of any degree or diploma to the best of my knowledge and belief.

Signature of the Student:

Vysakh B

Vysakh B

Date: 24 - 10 - 2025

## BONAFIDE CERTIFICATE

This is to certify that the project work entitled "FoodGo" is a bonafide record of work done by Mr. Vysakh B [Reg.no 23UBC260] at Marian College Kuttikkanam Autonomous in partial fulfilment for the award of the Degree of Bachelor of Computer Applications of Mahatma Gandhi University, Kottayam.

This work has not been submitted elsewhere for the award of any degree to the best of our knowledge.

### Head of the Department

Dr Rajimol A

Dept. of Computer Applications

Marian College Kuttikkanam

Peermade – 685531

### Internal Guide

Dr. Sheela S

Dept. of Computer Applications

Marian College Kuttikkanam

Peermade – 685531

Submitted for the Viva-Voce Examination held on



EXTERNAL EXAMINER

## **ACKNOWLEDGEMENT**

“Gratitude is a feeling which is more eloquent than words, more silent than silence.” In undertaking this project, I needed the direction, assistance, and cooperation of various individuals and organizations, which I received in abundance with the grace of God. Without their unconstrained support, the project could not have been completed. If words are considered as the symbol of approval and token of acknowledgement, then let the following words play the heralding role of expressing my gratitude.

I wish to acknowledge my sincere gratitude to our Manager, **Rev. Fr. Boby Alex Mannamplackal**, and **Prof .Dr. Ajimon George**, Principal **Marian College Kuttikkanam Autonomous**, for all their efforts and administration in educating me in their premier institution.

I extend my gratitude to **Dr. Rajimol A**, Head of the Department of Computer Applications, who has been a constant source of inspiration and whose advice helped me to complete this project successfully. I express my deep sense of gratitude to my internal project guide, **Dr. Sheela S**, for her profound guidance for the successful completion of this project.

With great enthusiasm, I express my gratitude to all the faculty members of the Department of Computer Applications for their timely help and support.

Finally, I express my deep appreciation to all my friends and family members for the moral support and encouragement they have given to complete this project successfully.

Vysakh B

# **ABSTRACT**

## ABSTRACT

**FoodGo** is a web-based food delivery platform developed to connect users with nearby restaurants across Kerala. The application aims to simplify the process of ordering food online by providing a clean, intuitive interface that allows users to explore restaurant listings, browse menus, place orders, and track delivery status in real-time.

The system mainly consists of three user roles Admin, Restaurant Owner, and Customer. Each user has a unique set of permissions and functionalities. Customers can register, log in, view detailed food menus, check availability, place orders from any available restaurant, and track their order status. Restaurant owners can register their restaurants, manage food items, update availability, and view incoming orders, while also having the option to act as customers and place orders from other restaurants on the platform. Admins have full control over the system, including managing user accounts and restaurants, deleting listings, and restricting user access when necessary. The platform implements role-based authentication and uses session and cookie handling to ensure secure and personalized access throughout the application. To complete the ordering process, the platform features a simulated payment module that demonstrates a basic checkout flow without involving real transactions.

The platform uses HTML, CSS, and JavaScript for the front end, providing an intuitive user interface. The back end, developed with PHP and MySQL, ensures robust data management and seamless functionality.

## TABLE OF CONTENTS

<b>1. INTRODUCTION</b>	<b>2</b>
<b>1.1 ABOUT THE PROJECT</b>	<b>2</b>
<b>1.2 EXISTING SYSTEM</b>	<b>2</b>
<b>1.3 PROPOSED SYSTEM</b>	<b>3</b>
<b>2. SYSTEM ANALYSIS</b>	<b>5</b>
<b>2.1 PROBLEM DEFINITION</b>	<b>5</b>
<b>2.2 FEASIBILITY ANALYSIS</b>	<b>5</b>
<b>2.3 RECOMMENDED IMPLEMENTATIONS</b>	<b>6</b>
<b>3. SOFTWARE REQUIREMENT SPECIFICATIONS</b>	<b>9</b>
<b>3.1 PURPOSE</b>	<b>9</b>
<b>3.2 SCOPE</b>	<b>9</b>
<b>3.3 TECHNICAL OVERVIEW</b>	<b>10</b>
<b>3.4 FUNCTIONAL REQUIREMENTS</b>	<b>10</b>
<b>3.5 NON-FUNCTIONAL REQUIREMENTS</b>	<b>12</b>
<b>3.6 STATED REQUIREMENTS</b>	<b>13</b>
<b>3.7 EXTERNAL INTERFACE REQUIREMENTS</b>	<b>17</b>
<b>4. SYSTEM DESIGN</b>	<b>20</b>
<b>4.1 DESIGN METHODOLOGY</b>	<b>21</b>
<b>4.2 SYSTEM ARCHITECTURE AND PROCESS FLOW</b>	<b>24</b>
<b>4.3 MODULE DETAILS</b>	<b>32</b>
<b>4.4 PERFORMANCE CONSIDERATIONS</b>	<b>33</b>
<b>4.5 SECURITY CONSIDERATIONS</b>	<b>34</b>
<b>4.6 TABLE DESIGN</b>	<b>34</b>

<b>5. CODING</b>	<b>39</b>
<b>5.1 SELECTION OF SOFTWARE</b>	<b>39</b>
<b>5.2 CODING PHASE</b>	<b>40</b>
<b>6. TESTING</b>	<b>42</b>
<b>6.1 TESTING</b>	<b>42</b>
<b>6.2 TESTING METHODS</b>	<b>43</b>
<b>6.3 IMPLEMENTATION</b>	<b>46</b>
<b>7. MAINTENANCE AND ENHANCEMENT</b>	<b>48</b>
<b>7.1 MAINTENANCE</b>	<b>48</b>
<b>7.2 ENHANCEMENT</b>	<b>48</b>
<b>8. CONCLUSION</b>	<b>51</b>

# **INTRODUCTION**

# **1. INTRODUCTION**

## **1.1 ABOUT THE PROJECT**

The FoodGo platform is a digital food delivery system designed to connect customers with restaurants seamlessly. It provides a centralized and user-friendly platform where customers can explore restaurants, browse food menus, place orders, and track deliveries in real-time. Restaurant owners can manage their restaurants, food items, and customer orders efficiently, while admins oversee the overall system for smooth operations.

The platform's intuitive interface ensures an easy and smooth navigation experience for all types of users—customers, restaurant owners, and admins. It allows customers to explore food items with images and descriptions, submit reviews with photos, and securely complete online orders, ensuring both convenience and satisfaction.

### **1.1.1 THE PURPOSE AND SCOPE**

The primary purpose of the FoodGo project is to establish a reliable, efficient, and convenient online food delivery system that benefits all stakeholders - customers, restaurant owners, and administrators.

The scope of the project includes:

- Enabling customers to explore restaurants and browse food menus with detailed item descriptions and images.
- Allowing customers to place food orders, track order status, and securely complete payments.
- Providing restaurant owners with tools to manage their restaurants, food menus, and incoming customer orders.
- Facilitating customer engagement through reviews, ratings, and images of food items.
- Implementing an admin panel to manage users, restaurants, food listings, and customer support efficiently.

## **1.2 EXISTING SYSTEM**

In the current scenario, food ordering is often done through traditional and fragmented methods. Customers usually rely on physical restaurant visits, phone calls for orders, or

limited third-party options. These manual and scattered approaches can lead to inefficiencies such as:

- Limited restaurant and food choices.
- Communication gaps between customers and restaurants.
- Delays in order processing.
- Lack of centralized customer support.
- This makes the current process time-consuming, less reliable, and inconvenient for both customers and restaurant owners.

### 1.3 PROPOSED SYSTEM

The proposed FoodGo system eliminates the limitations of the existing manual and fragmented methods by providing a centralized, digital platform for food delivery.

Key advantages of the proposed system include:

**Efficiency & Accuracy:** Orders are placed, processed, and tracked in real-time without confusion or redundancy.

**User-Friendly Experience:** Customers can easily search, filter, and order their favorite food items from multiple restaurants.

**Secure Data Handling:** Digital storage ensures customer data, restaurant details, and orders are managed securely.

**Faster Access:** Since information is stored digitally, searching and retrieving food or restaurant details is quick and reliable.

**Operational Support:** Restaurant owners and admins can efficiently manage the platform with minimal resources.

Overall, the proposed FoodGo platform enhances user convenience, data security, customer engagement, and operational efficiency while providing a modern solution for the food delivery ecosystem.

# **SYSTEM ANALYSIS**

## **2. SYSTEM ANALYSIS**

### **2.1 PROBLEM DEFINITION**

Most food delivery platforms provide digital ordering but lack efficiency, smooth integration, and advanced features. Many small restaurants still depend on manual order-taking, leading to errors, delays, and poor customer experience. Existing systems also miss essentials like role-based authentication, restaurant management, and real-time updates. To overcome these gaps, FoodGo is introduced as a complete platform offering seamless restaurant onboarding, easy food item management, live order tracking, and better customer engagement.

#### **2.1.1 ADVANTAGES OF PROPOSED SYSTEM**

- **Avoids Redundancy:** Uses validations (unique email, phone, restaurant name, food items) to eliminate duplicate records.
- **Quick Access and Processing:** Provides faster access to restaurants, food items, and orders with efficient database queries.
- **Time Efficiency:** Reduces manual work for restaurant owners and ensures instant ordering and tracking for customers.
- **Reduces Paperwork:** Digitalizes orders, invoices, and customer records, minimizing the need for paper-based systems.
- **Provides Accurate Information:** Ensures reliable details like food availability, order status, and restaurant ratings, helping both customers and owners.

### **2.2 FEASIBILITY ANALYSIS**

Feasibility study evaluates whether the proposed system is practical, effective, and resource-friendly. The goal is not to build the system immediately, but to check if it can be implemented successfully. The feasibility study for FoodGo is classified into:

- Operational Feasibility
- Technical Feasibility
- Economic Feasibility

## **2.2.1 OPERATIONAL FEASIBILITY**

The proposed FoodGo system offers a user-friendly interface with quick navigation. Customers can easily search food/restaurants, owners can manage restaurants and food menus, and admins can control the platform. Since the workload is distributed across roles (admin, owner, customer), the system is operationally feasible and reduces manual effort significantly.

## **2.2.2 TECHNICAL FEASIBILITY**

FoodGo is built using PHP, MySQL, HTML, CSS, JavaScript, Bootstrap, and AJAX. These technologies are widely supported and run smoothly on any basic hosting/server environment. Most users already have access to mobile phones or PCs with internet, so no special hardware is required. Thus, the system is technically feasible with minimal setup.

## **2.2.3 ECONOMIC FEASIBILITY**

Since FoodGo is developed using free and open-source technologies, the cost is minimal. Restaurants save money by avoiding third-party commission-heavy apps, and customers benefit from direct access to restaurants. The only initial investment is for hosting and domain, which is negligible compared to the long-term benefits. Hence, FoodGo is economically feasible and cost-effective.

## **2.3 RECOMMENDED IMPLEMENTATIONS**

To design the FoodGo system effectively, data was collected from multiple sources:

- Written Documents (sample menus, order records, existing restaurant data)
- Discussions with Stakeholders (restaurant owners, customers, admin perspective)

Fact-finding techniques used:

- Questionnaires – Simple surveys were designed to understand customer preferences (e.g., veg/non-veg options, delivery speed, favorite cuisines).
- Personal Interviews – Restaurant owners were interviewed to understand challenges in managing orders, menus, and delivery. Customers were asked about issues faced in existing apps (such as high costs, late delivery, and lack of transparency).

- Observations – Observing small restaurants' manual workflows highlighted the need for digital menu management, order tracking, and receipts. Based on this, FoodGo was designed as an efficient, scalable, user-friendly food delivery platform.

# **SOFTWARE REQUIREMENT SPECIFICATIONS**

## **3.SOFTWARE REQUIREMENT SPECIFICATIONS**

Requirements specification is the starting step for the development activities. It is currently one of the weak areas of software engineering. During requirement specification, the goal is to produce a document of the client's requirements. This document forms the basis of development and software validation. The basic reason for the difficulty in software requirements specification comes from the fact that there are three interested parties the client, the end users, and the software developer.

### **3.1 PURPOSE**

The origin of most software systems is in the need of a client, who either wants to automate an existing manual system or desires a new software solution. The software system itself is created by the developer. Finally, the completed system will be used by the end users. Thus, there are three major parties interested in a new system: the client, the users, and the developer. A basic purpose of software requirements specification is to bridge the communication gap. SRS is the medium through which the client and user needs are accurately specified. Indeed, SRS forms the basis of software development. A good SRS should satisfy all the parties something very hard to achieve and involves trade-offs and persuasion.

Another important purpose of developing an SRS is helping the clients understand their own needs. Advantages are:

- An SRS establishes the basis for agreement between the client and the supplier on what the software product will do
- An SRS provides a reference for validation of the final product
- A high-quality SRS is a prerequisite to high-quality software
- A high-quality SRS reduces the development cost

### **3.2 SCOPE**

#### **3.2.1 SYSTEM STATEMENT OF SCOPE**

The FoodGo Food Delivery Website is developed to provide a seamless platform for discovering, ordering, and managing food delivery services. The system offers features such

as browsing nearby restaurants, exploring digital menus, placing real-time orders, posting reviews, and tracking deliveries. Additionally, restaurant owners can manage their listings and menu items, while administrators oversee platform operations. Reports can be generated to analyze order trends, user activity, and restaurant performance.

### **3.3 TECHNICAL OVERVIEW**

#### **3.3.1 USER CHARACTERISTICS**

The system can be accessed by three types of users: Admin, Restaurant Owner, and Customer. Admins have exclusive access to the admin dashboard for managing users, restaurants, and orders. Restaurant Owners can log in to manage their menu, view incoming orders, and update restaurant details. Customers have access to the website for browsing restaurants, placing orders, writing reviews, and tracking deliveries. The functionalities of each user type are restricted and invisible to others based on their roles.

### **3.4 FUNCTIONAL REQUIREMENTS**

The functional requirements of this website are as follows:

#### **1. User Management**

User Registration Process:

- Users provide essential details like name, email, phone number, user type, and password to create an account.

User Authentication:

- Users log in by entering their registered email and password.
- The system verifies credentials against stored data to grant access.
- Users can securely log out, ending their active sessions to prevent unauthorized access.

#### **2. Restaurant Management**

- Restaurant owners can register and manage their restaurant profiles.

- They can add, update, or remove restaurant details, such as address, logo, and contact information.

### **3. Menu Management**

- Restaurant owners can add, update, or delete food items, including names, prices, and images.
- Menu items can be categorized and sorted for user-friendly browsing.

### **4. Cart Management**

- Customers can add or remove food items from their cart.
- The cart automatically updates item quantities and total price.
- Users can view a summary of their cart before proceeding to checkout.

### **5. Order Management**

- The system collects billing and delivery information during checkout.
- Each order stores food item details, quantities, customer data, and timestamps.
- Customers can track the status of their orders in real time.

### **6. Review and Rating System**

- Customers can submit reviews and ratings for restaurants and menu items.
- Reviews can be edited or deleted by the user.
- Admins can moderate inappropriate reviews.

### **7. Admin Dashboard**

- Admins can manage user accounts, restaurant listings, and orders.
- Admins can oversee review moderation and resolve reported issues.
- Admins have access to analytics and reporting tools for platform performance.

### **8. Order History**

- Users can view their past orders, including order ID, food items, total amount, and delivery status.

- Restaurant owners can view order history related to their restaurant.

## 3.5 NON-FUNCTIONAL REQUIREMENTS

### 1. Usability:

- Intuitive and user-friendly interface for customers, restaurant owners, and admins.
- Clean layout with clear navigation and accessible design across all modules.

### 2. Reliability:

- The system must perform reliably in 95% of use cases each month.
- Ensure accurate and consistent data processing for all transactions and activities.

### 3. Availability:

- The platform must be accessible globally 24/7 for ordering and administration.
- Backup and recovery mechanisms should restore operations within one hour of failure.

### 4. Security:

- Secure handling of user credentials, payment details, and personal data.
- Regular database backups and protection against unauthorized access or data breaches.

### 5. Performance:

- Fast response times for user interactions, including real-time updates and page loads.
- Optimized performance to handle high volumes of concurrent users and transactions.

### 6. Scalability:

- Capable of supporting increasing numbers of users, restaurants, and orders over time.
- Designed with modular architecture to allow easy feature additions and system upgrades.

## **7. Maintainability:**

- Well-documented, modular codebase to simplify debugging, updates, and feature changes.
- Regular maintenance routines to ensure continued performance and system stability.

## **3.6 STATED REQUIREMENTS**

### **3.6.1 GENERAL REQUIREMENTS**

The system has 8 functional modules divided among admins, restaurant owners, and users.

#### **1. Login**

- Only registered users, admins, and restaurant owners can log in to access their respective dashboards.
- All users log in using their registered email ID and password.
- The user ID must be a valid email address.
- Passwords must include uppercase and lowercase letters, numbers, and special characters.
- Admins are redirected to the admin dashboard upon successful login using the same login form.
- Restaurant owners are redirected to their restaurant dashboard.
- Users are taken to the main homepage upon successful login.

#### **2. Sign Up**

- New users must register to use the platform's services.
- During registration, users select their account type (Customer or Restaurant Owner).
- The registration form includes the following fields:
  - Name
  - Email Address
  - Phone Number
  - User Type (Customer / Restaurant Owner)
  - Password

- Confirm Password

### **3. Admin Panel**

The admin panel includes the following functionalities:

- Managing user and restaurant accounts
- Overseeing orders and transactions
- Moderating reviews and content
- Accessing analytics and reports

### **4. Browse Restaurants**

Users can explore available restaurants based on location, cuisine, or rating.

### **5. Browse Menu Items**

Users can view and select food items from the digital menu of each restaurant.

### **6. Cart and Checkout**

- Users can add or remove items from their cart.
- A summary of selected items, quantities, and total price is shown before placing the order.
- Users proceed to secure checkout and payment.

### **7. Order Tracking**

- Users can track the real-time status of their food orders.
- Updates include order confirmation, preparation, dispatch, and delivery.

### **8. Review and Rating System**

- Users can rate restaurants and menu items after receiving their order.
- Reviews help improve service and inform future customers.

#### **3.6.2 INPUTS**

The FoodGo platform will take various inputs to provide its services:

### **User Registration:**

- Name
- Email Address
- Phone Number
- User Type (Customer / Restaurant Owner)
- Password, Confirm Password

### **Login:**

- Email Address (User ID)
- Password

### **Restaurant Registration:**

- Restaurant Name
- Owner Name
- Email Address
- Contact Number
- Restaurant Address
- Password, Confirm Password

### **Menu Management:**

- Dish Name
- Price
- Description
- Category
- Food Image

### **Order Placement:**

- Selected Food Items
- Quantity
- Delivery Address
- Payment Method

#### **Review Submission:**

- Rating (1 to 5)
- Comment
- Restaurant or Dish Selection

### **3.6.3 PROCESSING**

The system will perform the following key processing tasks:

#### **Data Validation:**

- Validate email format and password complexity during registration and login.
- Ensure all required fields (e.g., name, address, payment method) are properly filled.

#### **User and Restaurant Management:**

- Users register and manage their profiles.
- Restaurant owners manage their restaurant information and menus.
- Admins oversee account activity and approve restaurants.

#### **Order Management:**

- Users place and manage orders in real-time.
- System updates order status throughout preparation, dispatch, and delivery stages.

#### **Review Moderation:**

- Users submit reviews and ratings for restaurants and dishes.
- Admins monitor and manage reported or flagged reviews.

#### **E-commerce and Payment Processing:**

- Users add food items to their cart and place orders.
- System handles secure payment transactions and updates order history.

#### **Content Management:**

- Admins manage platform content including banners, promotions, and user-reported issues.

### **3.7 EXTERNAL INTERFACE REQUIREMENTS**

#### **3.7.1 USER INTERFACES**

All user interfaces will be GUI-based, designed for ease of use, responsiveness, and smooth functionality. The interfaces will maintain a professional and intuitive layout for all users.

##### **Design and Appearance:**

- The interface will incorporate appealing design elements and color schemes to offer a visually comfortable experience.
- A consistent visual theme will be applied across all pages and user roles.

##### **Usability Components:**

Textboxes, dropdowns, buttons, and other form elements will enable easy data entry and interaction.

- Each input field will include clear labels and user-friendly prompts.
- Intuitive menus and navigational elements will allow seamless movement across the platform.

##### **Responsive Design:**

The user interface will be fully responsive to ensure a smooth experience across all devices, including desktops, laptops, tablets, and smartphones.

#### **3.7.2 HARDWARE INTERFACES**

The system requires basic computing or smart devices with internet access to operate the FoodGo web application. No additional hardware components are necessary.

## **Hardware Specification**

- Processor: Intel Pentium or higher
- RAM: 512 MB or higher
- Hard Disk Drive: 200 MB required on disk
- Keyboard: Standard QWERTY keyboard (for desktop users)
- Network: Stable internet connection

## **Implementation Specification**

- Operating System: Windows, Linux, Android, or iOS

### **3.7.3 SOFTWARE INTERFACES**

#### **Software Specification**

- Operating System: Windows 10 / 11 or equivalent Linux distribution
- DBMS: MySQL
- Backend: PHP
- Frontend: HTML, CSS, JavaScript
- Server: Apache (via XAMPP or hosting server)
- Browser Support: Google Chrome, Mozilla Firefox, Microsoft Edge

# **SYSTEM DESIGN**

## 4. SYSTEM DESIGN

The design phase aims to develop a solution for the problems identified during the analysis phase. This phase marks the transition from understanding the problem to creating a solution. System design details the required features and operations, including screen layouts, business rules, process diagrams, pseudocode, and other relevant documentation.

During this phase, the overall structure and specifics of the FoodGo platform are defined. This includes determining the number of tiers needed for the architecture, designing inputs and outputs, and establishing database and data structure designs. Proper analysis and design are critical to the development cycle since errors in this phase can be costly to fix later. Therefore, careful attention is given during the design phase.

The logical framework of the product and its physical attributes are outlined during this stage. The operating environment is set up, and key resources are identified. Any element that requires user input or approval must be documented and reviewed by the user. The physical aspects of the system are specified, and a detailed design is prepared.

Subsystems identified during design are used to create a detailed system structure. Each subsystem is divided into one or more design units or modules, and detailed logic specifications are created for each module. The module logic is typically described in a high-level design language, which is independent of the final implementation language.

A good design should consider:

- **Promptness:** The design should be straightforward and clear, guiding users to their desired outcomes intuitively.
- **Memory Load:** Research shows that users can retain about six words in their short-term memory. The number of choices presented to users should ideally be four or fewer to prevent confusion and forgetfulness.
- **Service Reachability:** Users dislike going through many steps to access a service. More than five steps can cause impatience, so minimizing the number of steps helps reduce frustration.
- **Navigation:** Users should easily navigate back and forth between different steps, allowing them to access various parts of the dialog seamlessly.

- **Phonetic Similarity:** Avoid choices with similar pronunciations to reduce confusion and ensure users can clearly distinguish between options.
- **Error Handling:** Implementing graceful error handling helps decrease dependency on operators by managing mistakes effectively.
- **User Updates:** Keep users informed about the ongoing process to maintain their engagement and understanding.

For the general design, one or more potential designs are proposed and broadly sketched. These alternatives are then presented to the users, who choose the design that best suits their requirements while staying within project constraints.

The detailed design stage specifies the user interface, database, programs, hardware, and training and system documentation. Several structured techniques are used during the design phase. To design the software components, the designer transforms the automated processes in the physical data flow diagram into a program structure chart, which decomposes software processes into detailed modules and shows control paths between modules.

## 4.1 DESIGN METHODOLOGY

### 4.1.1 INPUT DESIGN

Input design focuses on converting user-oriented inputs into a format recognizable by the computer. Collecting input data is one of the most costly parts of the system in terms of equipment, time, and user involvement. The goal of input design is to make data entry as simple, logical, and error-free as possible.

In FoodGo, inputs include:

- User registration details (name, email, phone, password)
- Restaurant details (name, place, phone)
- Food items (name, category, price, description)
- Customer orders (selected items, quantity, payment details)

System analysis determines the following input design details:

- **What data to input**
- **What medium to use**
- **How the data is arranged and coded**

➤ **Data items and transactions requiring validation to detect errors**

Activities involved in input design include:

- **Data Recording:** Collecting data for input.
- **Data Verification:** Ensuring the accuracy of the collected data.
- **Data Conversion:** Transforming data into the required format.
- **Data Validation:** Checking data for errors.
- **Data Correction:** Fixing any identified errors.

#### **4.1.2 OUTPUT DESIGN**

Output design involves creating necessary outputs tailored to meet the requirements of various users. Outputs in FoodGo include order receipts, restaurant listings, food item listings, and admin reports. Outputs can be displayed on dashboards, as alerts/messages, or as downloadable receipts.

When designing outputs, system analysts must achieve the following objectives:

- Determine the information to be presented
- Decide whether to display, print, or communicate the information, and select the appropriate output medium
- Format the information in an easily understandable manner

Output design is critical to the success of FoodGo as it bridges the gap between the user and the system's operations. Effective output design ensures clarity, accuracy, and user satisfaction by delivering meaningful results such as order confirmations, status updates, and restaurant performance reports.

#### **4.1.3 CODE DESIGN**

The coding phase transforms the detailed software design into a programming language. FoodGo uses a modularized approach with:

- PHP – Server-side logic
- MySQL – Database interaction
- AJAX – Seamless data flow without page reloads
- Bootstrap + JS – Frontend interactivity

Each module is divided based on responsibilities, such as authentication, restaurant management, food management, order management, and reviews.

Code design aims to minimize complexity by modularizing the implementation. Modules hide complexity by encapsulating executable statements under named functions or procedures. Effective information hiding enhances program understanding at higher abstraction levels.

#### **4.1.4 DATABASE DESIGN**

Database design identifies relevant data relationships and defines tables using standard methods. Each table's attributes are carefully defined to optimize performance, ensure integrity, and minimize redundancy.

A database system is a computer representation of an information system designed to handle integrated data efficiently. It minimizes redundancy to provide quick, flexible, and cost-effective information access. Database design considers several specific objectives:

Database design considers the following objectives:

- Controlled redundancy
- User-friendly interface
- Data independence
- Cost-effective data retrieval
- Accuracy and integrity
- Failure recovery
- Privacy and security
- Performance optimization

Each table in the database typically includes a primary key, a unique column (or combination of columns) that uniquely identifies each record. Normalization is applied to minimize redundancy and anomalies during insertion, update, and deletion operations.

## 4.2 SYSTEM ARCHITECTURE AND PROCESS FLOW

### UML DIAGRAMS

#### 4.2.1 USECASES

##### 1. USER REGISTRATION

Use Case Id:	FG_UC_01
Use Case Name:	User Registration
Created by:	Vysakh B
Date Created:	23-06-2025
Description:	Allows new users to register for an account on the FoodGo platform.
Primary actor:	User
Secondary actor:	None
Precondition:	User navigates to the registration page.
Postcondition:	User account is successfully created.
Main flow:	<ol style="list-style-type: none"><li>1. User navigates to the registration page.</li><li>2. User provides required details: name, email, phone, and password.</li><li>3. User submits the registration form.</li><li>4. System sends a verification email.</li><li>5. User verifies email via the verification link..</li><li>6. System creates the user account.</li><li>7. Use case ends.</li></ol>

##### 2. USER LOGIN

Use Case Id:	FG_UC_02
Use Case Name:	User Login
Created by:	Vysakh B
Date Created:	23-06-2025
Description	This use case allows the user to log in using registered credentials. Different roles (User/Admin/Restaurant Owner) can access specific dashboards.
Primary actor:	User/Admin/Restaurant Owner
Secondary actor:	None
Precondition:	User should have a valid account.
Postcondition:	The system redirects to the appropriate dashboard.

Main flow:	<ol style="list-style-type: none"> <li>1. User navigates to the login page.</li> <li>2. User enters email and password.</li> <li>3. User submits the login form.</li> <li>4. System validates credentials.</li> <li>5. If valid, user is redirected to their dashboard.</li> <li>6. Use case ends.</li> </ol>
------------	---

### 3. RESTAURANT MANAGEMENT

Use Case Id:	FG_UC_03
Use Case Name:	Restaurant Management
Created by:	Vysakh B
Date Created:	23-06-2025
Description	This use case allows the admin to add, update, or delete restaurant profiles.
Primary actor:	Admin
Secondary actor:	None
Precondition:	Admin should be logged in.
Postcondition:	Changes are reflected in the restaurant listing.
Main flow:	<ol style="list-style-type: none"> <li>1. Admin navigates to the restaurant management section.</li> <li>2. Admin selects to add, edit, or delete a restaurant.</li> <li>3. Admin enters the required details.</li> <li>4. Admin submits the changes.</li> <li>5. System updates the restaurant database.</li> <li>6. Use case ends.</li> </ol>

### 4. FOOD ITEM MANAGEMENT

Use Case Id:	FG_UC_04
Use Case Name:	Food Item Management
Created by:	Vysakh B
Date Created:	23-06-2025
Description	This use case allows restaurant owners to manage food items listed under their restaurant.
Primary actor:	Restaurant Owner
Secondary actor:	None
Precondition:	Restaurant owner must be logged in.
Postcondition:	Food item details are updated in the system.

Main flow:	<ol style="list-style-type: none"> <li>1. Restaurant owner navigates to the food item management page.</li> <li>2. Owner selects add, edit, or delete item.</li> <li>3. Enters or updates item name, price, image, and description.</li> <li>4. Submits the form.</li> <li>5. System updates the database.</li> <li>6. Use case ends.</li> </ol>
------------	--

## 5. PLACE ORDER

Use Case Id:	FG_UC_05
Use Case Name:	Place Order
Created by:	Vysakh B
Date Created:	23-06-2025
Description	This use case allows users to place food orders from selected restaurants.
Primary actor:	User
Secondary actor:	None
Precondition:	User must be logged in and have items in the cart.
Postcondition:	Order is successfully placed and recorded.
Main flow:	<ol style="list-style-type: none"> <li>1. User navigates to the cart.</li> <li>2. Reviews selected items.</li> <li>3. Selects delivery address and payment method.</li> <li>4. Clicks 'Place Order'.</li> <li>5. System saves the order and notifies the restaurant.</li> <li>6. Use case ends.</li> </ol>

## 6. ORDER PROCESSING

Use Case Id:	FG_UC_06
Use Case Name:	Order Processing
Created by:	Vysakh B
Date Created:	23-06-2025
Description	This use case allows restaurants to process and update the status of received orders.
Primary actor:	Restaurant Owner
Secondary actor:	None
Precondition:	Restaurant must be logged in and have a new order.
Postcondition:	Order status is updated (accepted,

Main flow:	prepared, delivered). <ol style="list-style-type: none"> <li>1. Restaurant logs into the dashboard.</li> <li>2. Views incoming orders.</li> <li>3. Updates status: Accept, Preparing, Out for delivery, Delivered.</li> <li>4. System updates order status.</li> <li>5. User is notified.</li> <li>6. Use case ends.</li> </ol>
------------	--

## 7. REVIEW SYSTEM

Use Case Id:	FG_UC_07
Use Case Name:	Review System
Created by:	Vysakh B
Date Created:	23-06-2025
Description	This use case allows users to leave ratings and reviews (with images) for orders.
Primary actor:	User
Secondary actor:	None
Precondition:	User must be logged in and have a completed order.
Postcondition:	Review is visible on the respective restaurant or item page.
Main flow:	<ol style="list-style-type: none"> <li>1. User navigates to past orders.</li> <li>2. Selects the order to review.</li> <li>3. Enters rating, comment, and optionally uploads an image.</li> <li>4. Submits the review.</li> <li>5. System stores and displays the review.</li> <li>6. Use case ends.</li> </ol>

## 8. VIEW USER FAVORITES

Use Case Id:	FG_UC_08
Use Case Name:	View User Favorites
Created by:	Vysakh B
Date Created:	23-06-2025
Description	This use case allows users to view the favorite food items.
Primary actor:	User
Secondary actor:	None
Precondition:	User must be logged in.
Postcondition:	User favorites is displayed.

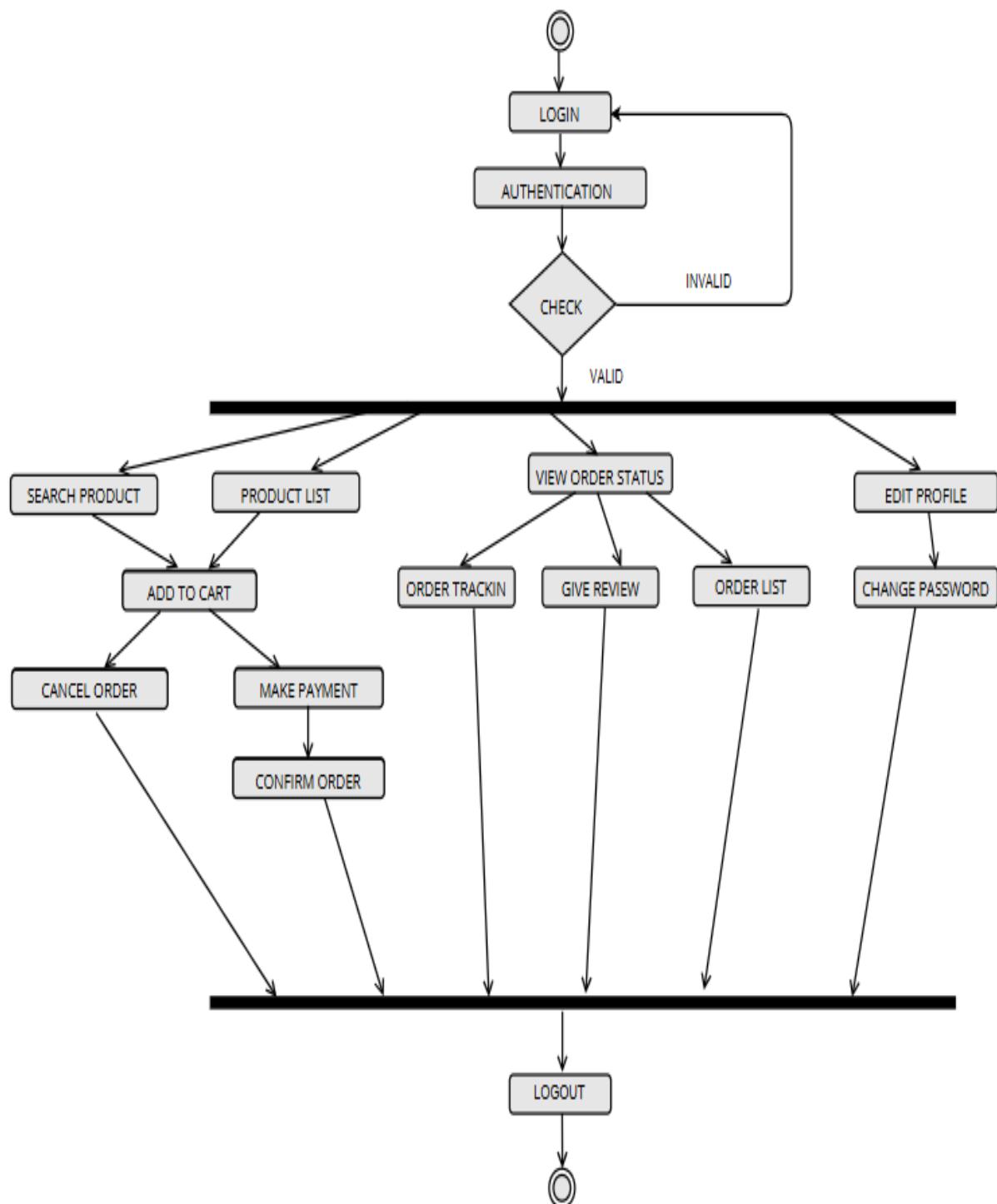
Main flow:	<ol style="list-style-type: none"> <li>1. User logs in.</li> <li>2. Navigates to ‘Favorites’.</li> <li>3. Views a list of all favorite food items.</li> <li>4. Use case ends.</li> </ol>
------------	--

## 4.2.2 USECASE DIAGRAM

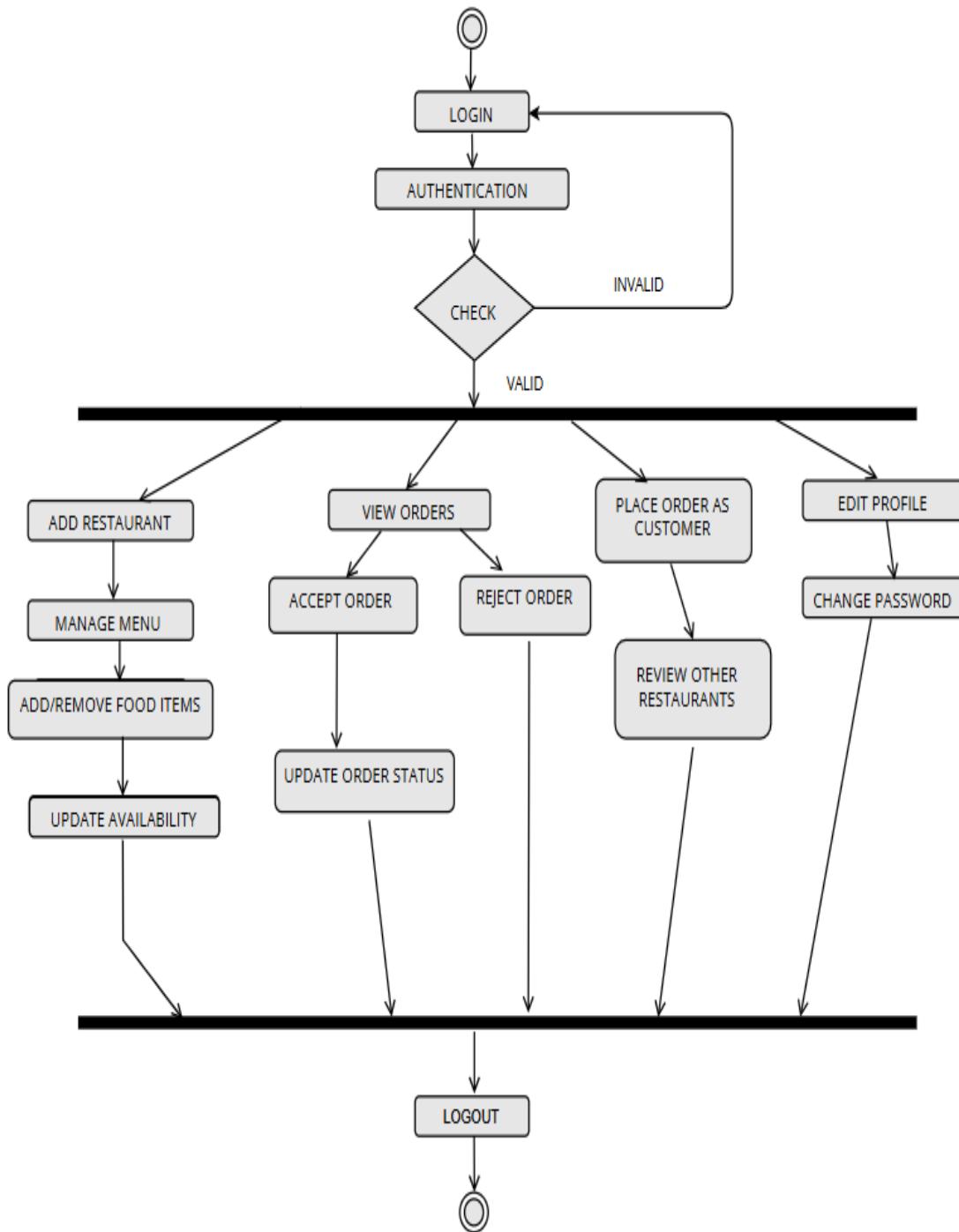


### 4.2.3 ACTIVITY DIAGRAMS

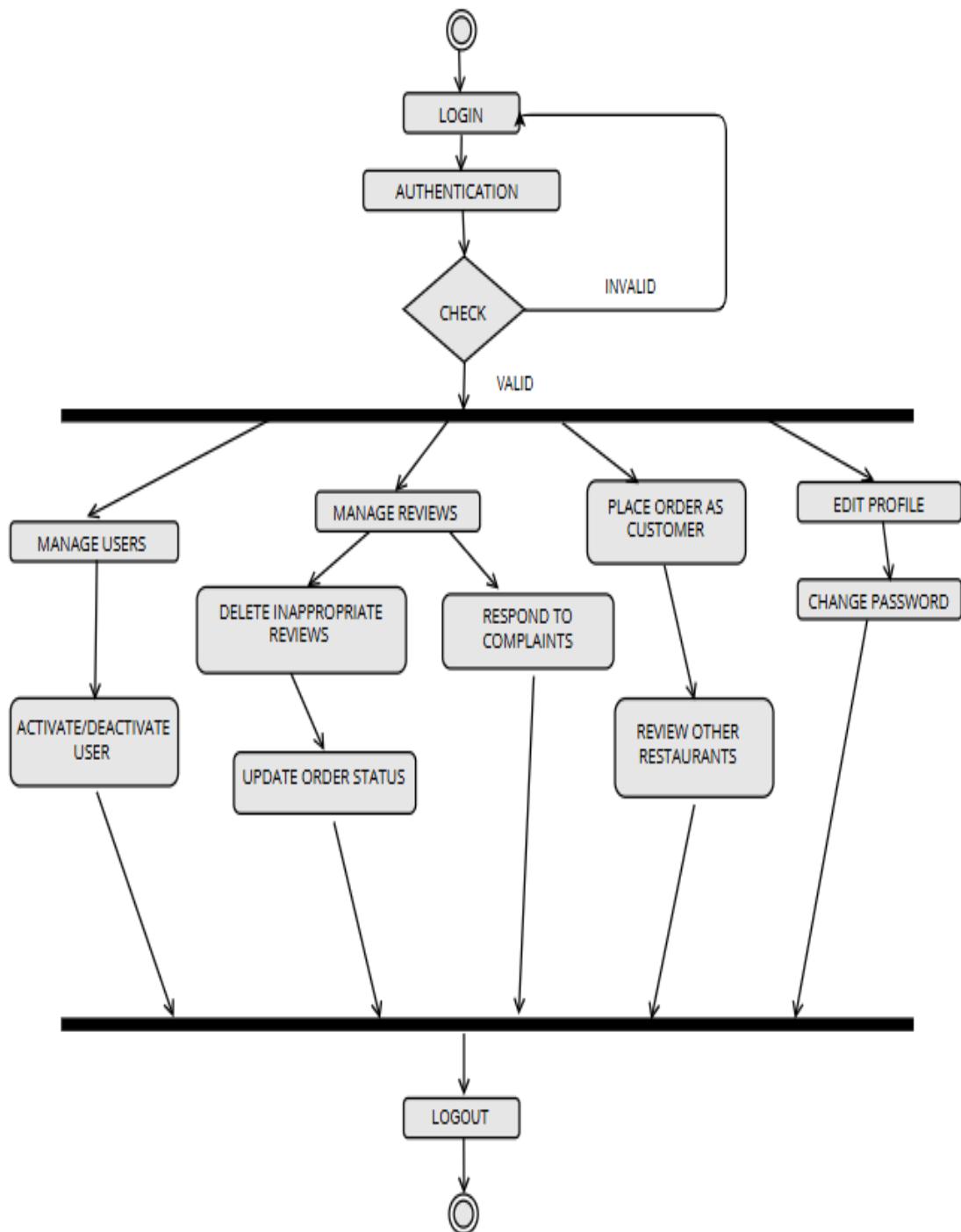
#### 1. CUSTOMER



## 2. RESTAURANT OWNER



### 3. ADMIN



## **4.3 MODULE DETAILS**

There are seven main modules in the FoodGo Food Delivery Platform:

- Login Module
- Sign-Up Module
- User Profile Module
- Restaurant & Food Management Module
- Search & Listings Module
- Order Management Module
- Admin Dashboard Module

### **1. Login Module**

The login module allows registered users (customers, restaurant owners, and admins) to securely access the FoodGo platform. It verifies user credentials (email/phone and password) and grants role-based access.

- Customers can browse restaurants, search food, and place orders.
- Owners can manage restaurants, menus, and orders.
- Admins can oversee the platform, approve restaurants, and monitor activity.

### **2. Sign-Up Module**

The sign-up module enables new users (customers or restaurant owners) to create accounts. Users can register with details such as name, email, phone number, and password. Once registered, users gain access to features like searching food/restaurants, managing restaurants, or placing and tracking orders.

### **3. User Profile Module**

The user profile module allows both customers and restaurant owners to manage their personal details and preferences. Customers can update personal details, view their order history, reviews , etc.

### **4. Restaurant & Food Management Module**

This module allows restaurant owners to Add, edit, and delete restaurant details (name, location, contact info), Manage menus by adding food items, categories, descriptions, images,

and prices. It ensures that customers see updated and accurate restaurant/food listings when browsing.

## **5. Search & Listings Module**

The search & listings module provides a powerful search feature for customers. Search by food item or restaurant (with AJAX auto-suggestions). Filters such as Veg / Non-Veg can be applied. Customers can view listings with food images, restaurant names, prices, locations, and ratings. This module ensures customers can quickly find food or restaurants that match their needs.

## **6. Order Management Module**

This module manages the complete food ordering cycle:

- Customers place orders by selecting food items from restaurants.
- Orders are processed and stored in the system.
- Owners can update order status (Pending → Approved/Rejected → Delivered).
- Customers can track order progress in real time.

This module ensures smooth transactions, transparency, and timely order fulfillment.

## **7. Admin Dashboard Module**

The admin dashboard equips administrators with essential tools to manage the FoodGo platform. It allows them to approve or reject restaurant requests, monitor user activity, generate reports, and ensure smooth operation with data integrity and security.

## **4.4 PERFORMANCE CONSIDERATIONS**

### **Hardware Requirements**

FoodGo runs smoothly with at least 4 GB RAM, modern browsers on Windows, Linux, or macOS, and a basic PHP-MySQL hosting. It is scalable to handle many users efficiently.

## 4.5 SECURITY CONSIDERATIONS

### Access Control

- **Authorized Access:** Only users with valid usernames and passwords are allowed to access the FoodGo platform.
- **Login Security:** The login process includes encryption and validation to authenticate users and prevent unauthorized access.

## 4.6 TABLE DESIGN

### 1. USERS

No.	Field_Name	Data_Type	Description
1	user_id	INT(11), Primary Key, Auto_Increment	Unique user ID
2	name	VARCHAR(100)	Full name of the user
3	email	VARCHAR(100), Unique	Login email
4	phone_number	VARCHAR(15), Unique	Phone number
5	password	VARCHAR(255)	Hashed password
6	role	ENUM('admin','owner','customer')	User role type
7	user_status	ENUM('active','ban') DEFAULT 'active'	User status
8	created_at	DATETIME DEFAULT CURRENT_TIMESTAMP	Account creation timestamp

### 2. RESTAURANTS

No.	Field_Name	Data_Type	Description
1	restaurant_id	INT(11), Primary Key, Auto_Increment	Restaurant ID
2	owner_id	INT(11), Foreign Key	References Users(user_id)

3	name	VARCHAR(100)	Restaurant name
4	phone_number	VARCHAR(15)	Restaurant phone number
5	address_line1	VARCHAR(255)	Primary address
6	address_line2	VARCHAR(255)	Secondary address
7	district	VARCHAR(50)	District name
8	offers_delivery	BOOLEAN DEFAULT FALSE	Supports delivery or not
9	status	ENUM('pending','approved','rejected','ban') DEFAULT 'pending'	Approval status

### 3. FOOD\_ITEMS

No.	Field_Name	Data_Type	Description
1	item_id	INT(11), Primary Key, Auto_Increment	Food item ID
2	restaurant_id	INT(11), Foreign Key	References Restaurants(restaurant_id)
3	name	VARCHAR(100)	Food item name
4	description	TEXT	Food description
5	price	DECIMAL(6,2)	Price of food
6	is_veg	BOOLEAN	True if vegetarian
7	categories	ENUM('Starters','Main Course','Curries','Desserts','Snacks','Beverages')	Category type
8	availability	BOOLEAN DEFAULT TRUE	Availability of item

### 4. FOOD\_IMAGES

No.	Field_Name	Data_Type	Description
1	image_id	INT(11), Primary Key, Auto_Increment	Image ID

2	item_id	INT(11), Foreign Key	References Food_Items(item_id)
3	image_path	VARCHAR(255)	Path or URL of image

## 5. ORDERS

No.	Field_Name	Data_Type	Description
1	order_id	INT(11), Primary Key, Auto Increment	Order ID
2	customer_id	INT(11), Foreign Key	References Users(user_id)
3	restaurant_id	INT(11), Foreign Key	References Restaurants(restaurant_id)
4	delivery_type	ENUM('deliver_to_me', 'make_the_parcel')	Order type
5	delivery_address	TEXT	Delivery address
6	status	ENUM('pending','confirmed','ready', 'rejected','delivered','cancelled','tim eout') DEFAULT 'pending'	Order status
7	reject_reason	VARCHAR(255)	Reason for rejection
8	created_at	DATETIME DEFAULT CURRENT_TIMESTAMP	Order timestamp

## 6. ORDER\_ITEMS

No.	Field_Name	Data_Type	Description
1	order_item_id	INT(11), Primary Key, Auto Increment	Unique order item ID
2	order_id	INT(11), Foreign Key	References Orders(order_id)
3	item_id	INT(11), Foreign Key	References Food_Items(item_id)
4	quantity	INT(11)	Quantity ordered

## 7. REVIEWS

No.	Field_Name	Data_Type	Description
1	review_id	INT(11), Primary Key, Auto_Increment	Review ID
2	order_id	INT(11), Foreign Key	References Orders(order_id)
3	customer_id	INT(11), Foreign Key	References Users(user_id)
4	rating	INT CHECK (1-5)	Review rating
5	comment	TEXT	Review comment
6	created_at	DATETIME DEFAULT CURRENT_TIMESTAMP	Review timestamp

## 8. REVIEW\_IMAGES

No.	Field_Name	Data_Type	Description
1	review_image_id	INT(11), Primary Key, Auto_Increment	Review image ID
2	review_id	INT(11), Foreign Key	References Reviews(review_id)
3	image_path	VARCHAR(255)	Path of review image

## 9. USER\_FAVORITES

No.	Field_Name	Data_Type	Description
1	favorite_id	INT(11), Primary Key, Auto_Increment	Favorite ID
2	user_id	INT(11), Foreign Key	References Users(user_id)
3	item_id	INT(11), Foreign Key	References Food_Items(item_id)
4	restaurant_id	INT(11), Foreign Key	References Restaurants(restaurant_id)
5	created_at	DATETIME DEFAULT CURRENT_TIMESTAMP	Timestamp of favorite

# CODING

## **5. CODING**

Coding section is where the magic happens. All the planning and the designing done in the previous sections come to life in this section. After this section can only the programmer enjoy the result of his/her hard work when he/she runs the FoodGo system for the first time.

### **5.1 SELECTION OF SOFTWARE**

#### **PHP**

PHP, an acronym for Hypertext Preprocessor, is a versatile server-side scripting language that falls under the broader category of software development. It is widely recognized for its pivotal role in web development and boasts several essential features that make it a preferred choice for building dynamic websites and web applications such as FoodGo. Here are some of its key features:

- Open Source
- Database Integration
- Embedded in HTML
- Cross-Platform Compatibility
- Security

#### **MYSQL**

MySQL is an open-source relational database system, widely used for web development tasks like data storage, manipulation, and retrieval. It seamlessly integrates into web applications, eliminating the need for complex setup. MySQL is embedded within web development environments, making administrative tasks effortless. It operates as an SQL-based database, storing data in text files on the device. Unlike systems like JDBC, MySQL simplifies data access with its broad range of relational database features. Its features are:

- Zero configuration
- Server less
- Stable cross platform database file
- Less memory
- Self-contained

- Transactional

## 5.2 CODING PHASE

The goal of the coding or programming phase is to translate the design of the FoodGo system produced during the design phase into code in a given programming language, which can be executed by a computer and that performs the computation specified by the design. The coding phase affects both testing and maintenance profoundly.

The coding phase does not affect the structure of the system; it has great impact on the internal structure of modules, which affects the testability of the system. The goal of the coding phase is to produce clear simple programs. The aim is not to reduce the coding effect, but to program in a manner so that testing and maintenance costs are reduced.

Programs should not be constructed so that they are easy to write; they should be easy to read and understand. Reading programs is a much more common activity than writing programs. Hence, the goal of the coding phase is to produce simple programs that are clear to understand and modify.

### 5.2.1 CODING STANDARDS

The standard used in the development of the FoodGo system is Microsoft Programming standards. It includes naming conventions of variables, constants and objects, standardized formats for labelling and commenting code, spacing, formatting and indenting.

#### Naming Conventions

The controls are prefixed to indicate their functions. The frames are prefixed with frm, textboxes are prefixed with txt, command buttons with cmd, label boxes with lbl, list boxes with lst, comboboxes with cmb, Date Time Pickers with DTP, Grid with grid and so on.

#### Labels and Comments

The functions of each control are labelled clearly in the GUI. The code also includes comments so that other developers using the source code in future might understand the module functions better.

# **TESTING & IMPLEMENTATION**

## **6. TESTING**

Software testing is a critical element of software quality assurance and represents the ultimate review of specifications design and coding. Testing presents an interesting anomaly for the software. Testing is a quality measure process, which reveals the errors in the FoodGo system. During testing, the program is executed with a set of test cases and the output of the program for the test cases is evaluated to determine if the FoodGo system is performing as it is expected. Testing plays a very critical role in determining the reliability and efficiency of the FoodGo system and it is a very important stage in software development.

### **6.1 TESTING**

System testing is actually a series of different tests whose primary purpose is to fully exercise the FoodGo system. Although each test has a different purpose, all work to verify that all system elements have been properly integrated and perform allocated functions. System testing is done in order to ensure that the FoodGo system developed doesn't fail at any point. Before implementation, the system is tested with experimental data to ensure that it will meet the specified requirements, special test data are input for processing and results examined.

#### **6.1.1 TEST PLAN**

##### **Preparation of test data**

Taking various kinds of test data does the testing. Preparation of test data plays a vital role in the system testing. After preparing, the FoodGo system under study is tested using that test data. While testing the system by using test data errors are again uncovered and corrected by using above testing steps and correction are also noted for future use. Two kinds of test data were collected and used:

##### **Using live test data**

Live test is those that are actually extracted from system files. After FoodGo was partially constructed, programmers or analyst often ask users to key in a set of data from their normal activities. Then, the system person uses this data as a way to partially test the system. In other instance, programmers or analysts extract a set of live data from the files and have entered themselves.

## **Using artificial test data**

Artificial test data are created solely for test purpose, since they can be generated to test all combinations of formats and values. In other words, the artificial data, which can quickly be prepared by a data generating utility program in the information system department, make possible the testing of all login and control paths through the program. The most effective test program uses artificial test data generated by person other than those who wrote the program. In this project invalid data was entered to test whether the FoodGo system would break or not. These invalid data entries were randomly generated using random people. Many people were given the FoodGo system for testing the program. They use gibberish values to test if every validation holds strong.

## **6.2 TESTING METHODS**

Testing is generally done at two levels-testing of individual modules and testing of the entire system. During system testing, the FoodGo system is used experimentally to ensure that the software does not fail that is, that it will run according to its specifications and the results examined. A limited number of users may be allowed to use the system so analysis can see whether they use it in unforeseen ways. It is preferable to discover any surprise before the organization implements the system and depends on it. Testing is done throughout system development at various stages. It is always a good practice to test the system at many different levels at various intervals, that is, sub systems, program modules as work progresses and finally the system as a whole. During testing the major activities are concentrated on the examination and modification of the source code. Usually, this testing is to be performed by the person other than the person who has really coded it. This is done in order to ensure more complete and unbiased testing for making the software more reliable.

There are two types of testing:

- Black box testing
- White box testing

### **6.2.1 WHITE BOX TESTING**

In white box testing, the internal logic of the modules is considered. Following levels of testing are performed for the developed FoodGo system:

### **6.2.1.1 Unit Testing**

This involves the tests carried out on modules programs, which make up the system. This is also called as a program testing. The units in a large system many modules at different levels are needed. Unit testing focuses on the modules, independently of one another, to locate errors. The program should be tested for correctness of logic applied and should detect errors in coding. Before proceeding one must make sure that all the programs are working independently.

## **6.2.2 BLACK BOX TESTING**

The concept of the black box is used to represent a system who's inside workings are not available for inspection. In a black box, the test item is treated as "black", since its logic is unknown; all that is known is what goes in and what comes out, or the input and output.

### **6.2.2.1 System Testing**

The system testing is conducted on a complete, integrated FoodGo system to evaluate the system's compliance with its specified requirement. It falls within scope of black box testing so no knowledge of inner design or logic is needed. As a rule, system testing takes, as its input, all of the integrated software components that have passed integration testing and also the software system itself integrated with any applicable hardware system. The purpose of the integration testing is to detect any inconsistencies between software units. System testing is the stage of implementation, which is aimed at ensuring that the system works accurately and efficiently before live operation commence. The logical design and the physical design should be thoroughly and continually examined on paper ensure that they will work when implemented.

### **6.2.2.2 Integration Testing**

Integration testing is a systematic technique for constructing the program structure, while at the same time conducting tests to uncover errors associated with interfacing. This is the program is constructed and tested in small segments, which makes it easier to isolate and the following common types of integration problems may be observed:

- Version mistakes
- Data integrity violations

- Overlapping function
- Resource problems especially in memory handling
- Wrong type of parameter in function calls

### **6.2.2.3 Validation Testing**

At the culmination of the integration testing, the software was completely assembled as a package, interfacing errors have been uncovered and corrected and a final series of software validation testing began. In validation testing we test the FoodGo system functions in a manner that can be reasonably expected by customer, the system was tested against system requirement specification. Different unusual inputs that the users may use were assumed and the outputs were verified for such unprecedented inputs. Deviation or errors discovered at this step are corrected prior to the completion of this project with the help of user by negotiating to establish a method for resolving deficiencies. Thus, the proposed system under consideration has been tested by using validation testing and found to be working satisfactorily. Validation checking is performed on the:

**Numeric Field:** - The numeric field can contain only numbers from 0 to 9. An entry of any character flashes an error message. The individual modules are checked for accuracy and what it has to perform. Each module is subjected to test run along with sample data. The individually tested module are integrated into a single system.

**Character Field:** - This field can only contain letters from A-Z and a-z. It is useful for name, address fields and so on.

**Check Null Fields:** - Before entering values into the database or when updating, a validation is done to check whether any NULL fields are present.

**Email Fields:** - A email only field with a limit of characters. All the necessary validation checks were verified to see if invalid data ever enters into the database. Null values in fields were also treated as invalid values.

**Password Fields:** - A password only field with a limit of characters. All the necessary validation checks were verified to see if invalid data ever enters into the database. Null values in fields were also treated as invalid values.

### **6.2.3 OUTPUT TESTING**

After performing validation test, the next phase is output test of the system, since no system could be useful if it does not produce the desired output in the desired format. By consideration the format of the report/output was generated or displayed and was tested. Here output format was considered in one way: on the display screen.

### **6.2.4 USER ACCEPTANCE TESTING**

User acceptance test of a system is the key factor for the success of the system. The FoodGo system under consideration was listed for user acceptance by keeping constant touch with the perspective user of the system at the time of design, development and making changes whenever required. This was done with the regards of the following points:

- Input screen design
- Output design

Users from each of the 2 user types (Admin, Customer) were selected for user acceptance testing. The Admin was given the FoodGo system for testing with his username and password. The admin actions are performed and see whether all details are entering into the database and working properly as expected. The customer side is tested using a customer name and password by registering to the system and see he can search food items and place orders.

## **6.3 IMPLEMENTATION**

Implementation is the stage where the theoretical design is transformed into a working system. It is a project in itself, requiring careful planning, analysis of the existing system and its constraints, designing methods for changeover, staff training, and evaluation of the process. The first task is planning—deciding on the methods and timeline to be used. Once this is done, the main focus is ensuring that the FoodGo programs function correctly and users are properly trained. When both system and users work together effectively, the entire process can run smoothly. Therefore, clear plans are essential. Successful implementation of the new FoodGo system design is a crucial phase in the system life cycle, as it converts a new or revised design into an operational system.

# **MAINTENANCE & ENHANCEMENT**

## 7. MAINTENANCE AND ENHANCEMENT

### 7.1 MAINTENANCE

This software can be modified as need occurs. Maintenance includes all the activities after installation of the FoodGo system that is performed to keep the system operational. The process of maintenance involves:

- Understanding the existing software
- Understand the effect of change
- Test for satisfaction

The FoodGo system requires little to no maintenance. During the testing phase most maintenance duties are performed. If a maintenance requirement occurs, it can be solved with ease.

### 7.2 ENHANCEMENT

The FoodGo system is built with a modular architecture, allowing for easy expansion and additional functionalities. As the business grows and customer demands evolve, the platform can seamlessly integrate new features to enhance the user experience.

Future enhancements to the FoodGo system could include:

- **AI-Based Food Recommendations:** Introduce smart suggestions for customers based on their order history, browsing patterns, and popular food trends. This will make the platform more personalized and engaging.
- **Discount and Loyalty Programs:** Add reward points, coupons, and referral bonuses to attract new users and retain existing ones. This feature can also improve customer satisfaction and loyalty.
- **Multi-Language Support:** Provide the FoodGo platform in multiple regional languages to ensure accessibility for a wider audience, especially in diverse communities.
- **Push Notifications:** Implement real-time notifications for offers, discounts, and order updates directly to the customer's device. This will keep users informed and connected with the platform

- **Integration with Delivery Partners:** Collaborate with third-party delivery services for extended coverage and faster order fulfillment, especially in regions where in-house delivery is not available.

These future developments will help enhance FoodGo's service offerings, attract new customers, and provide a seamless, user-friendly experience that stays competitive as technology advances.

# **CONCLUSION**

## **8. CONCLUSION**

In today's fast-growing food delivery industry, technology plays a pivotal role in shaping customer experiences and improving service delivery. Traditional methods of ordering food are being transformed by innovative digital solutions. The FoodGo system exemplifies this evolution, providing a seamless bridge between customers and restaurants. With a user-friendly, interactive platform, FoodGo harnesses modern web technologies to allow users to search, order, and engage with services from the comfort of their homes, eliminating the barriers of traditional methods.

The primary goal of FoodGo has always been to empower customers with easy access to high-quality food delivery services while helping restaurants expand digitally. FoodGo streamlines food ordering, enhances transparency, and fosters trust within the service community, setting new standards for customer interaction in the food delivery sector.

As the industry continues to evolve, FoodGo remains committed to further enhancements, embracing new technologies, and meeting the growing needs of its customers. Looking ahead, FoodGo aims to continue revolutionizing food delivery by integrating technology and services to create even more seamless and impactful customer experiences.

# **BIBLIOGRAPHY**

## **BIBLIOGRAPHY**

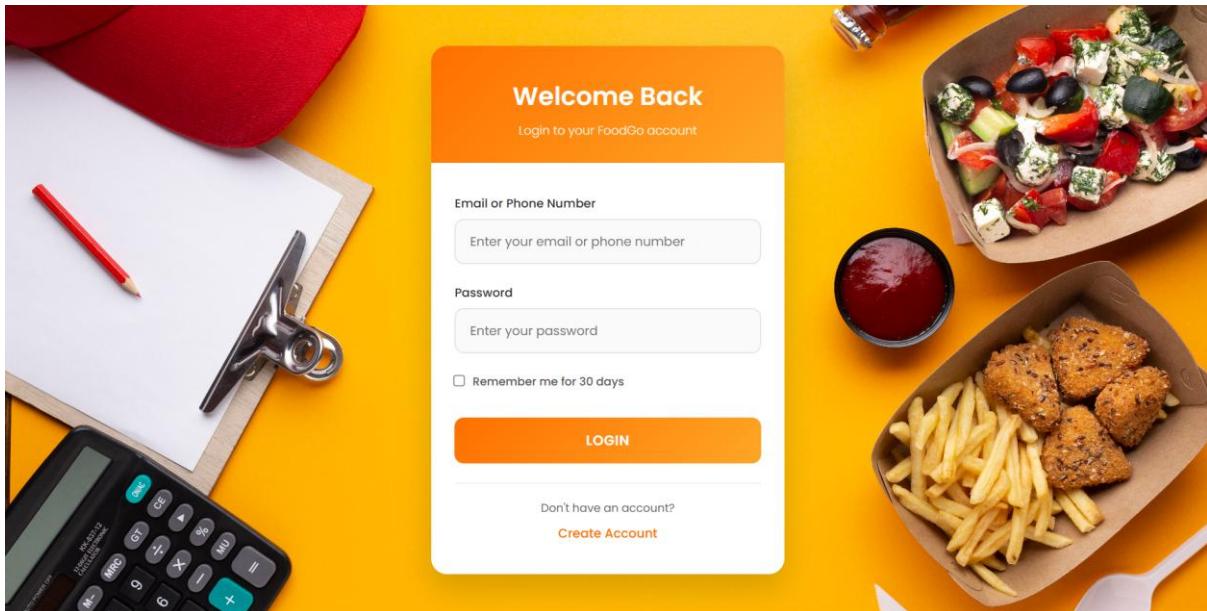
### **WEBSITES**

- ChatGPT
- Claude.ai
- www.pexels.com
- www.freepik.com

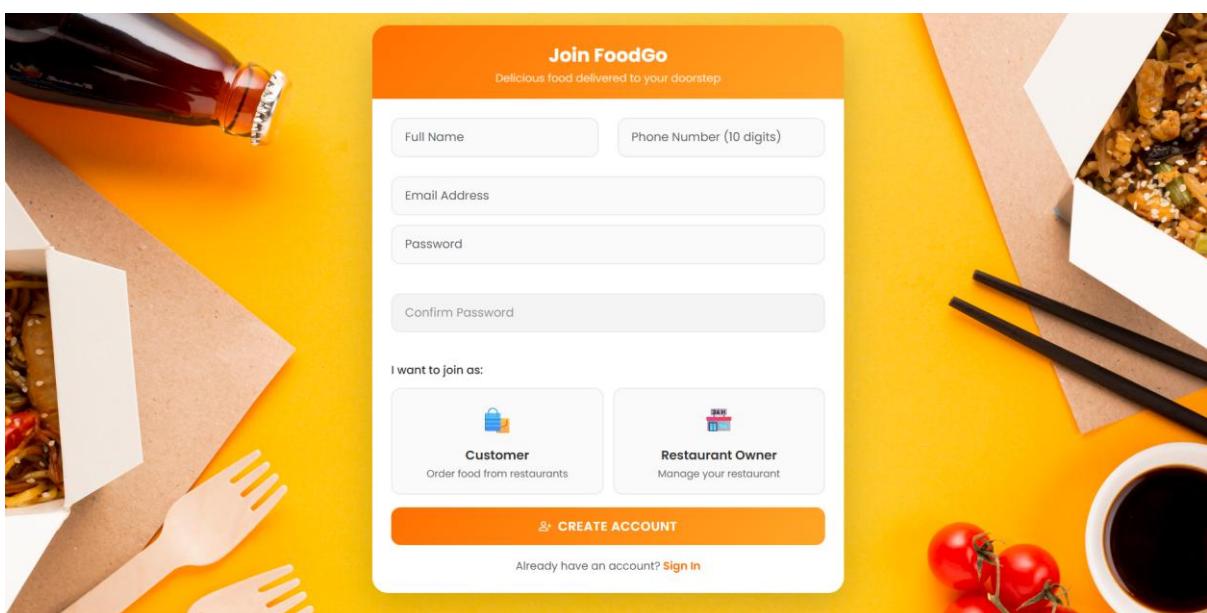
# **APPENDIX**

## SCREENSHOTS

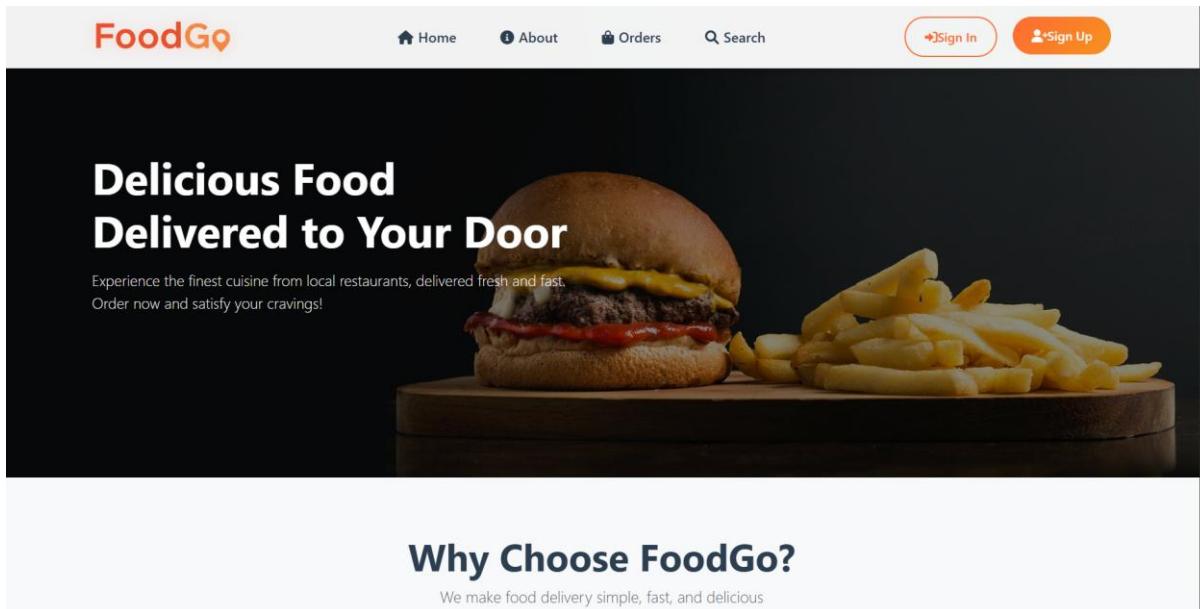
### 1.SIGNIN PAGE



### 2.SIGNUP PAGE



### 3.HOME PAGE



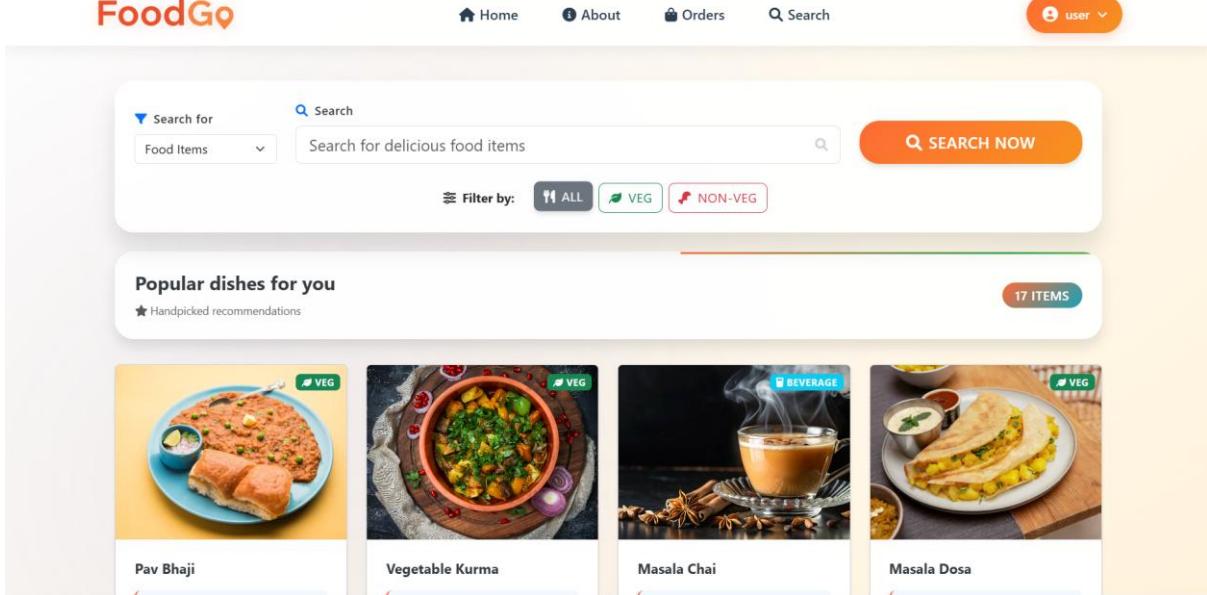
The screenshot shows the FoodGo home page. At the top, there's a navigation bar with the FoodGo logo, a search bar, and buttons for "Sign In" and "Sign Up". Below the header is a large banner featuring a burger and fries on a wooden board. The banner has the text "Delicious Food Delivered to Your Door" and a subtitle: "Experience the finest cuisine from local restaurants, delivered fresh and fast. Order now and satisfy your cravings!". Underneath the banner, there's a section titled "Why Choose FoodGo?" with the subtitle "We make food delivery simple, fast, and delicious".

### 4.ABOUT PAGE



The screenshot shows the FoodGo about page. At the top, there's a navigation bar with the FoodGo logo, a search bar, and a user profile button for "owenrtwo". Below the header is a section titled "About FoodGo" with the subtitle "Delicious food, delivered fresh & fast to your door." To the left of this text is an illustration of a red scooter with a chef's hat and a tray. To the right is a section titled "Who We Are" which includes a paragraph about the company's mission and a smaller paragraph about their partnership with restaurants and delivery experts.

## 5.SEARCH PAGE



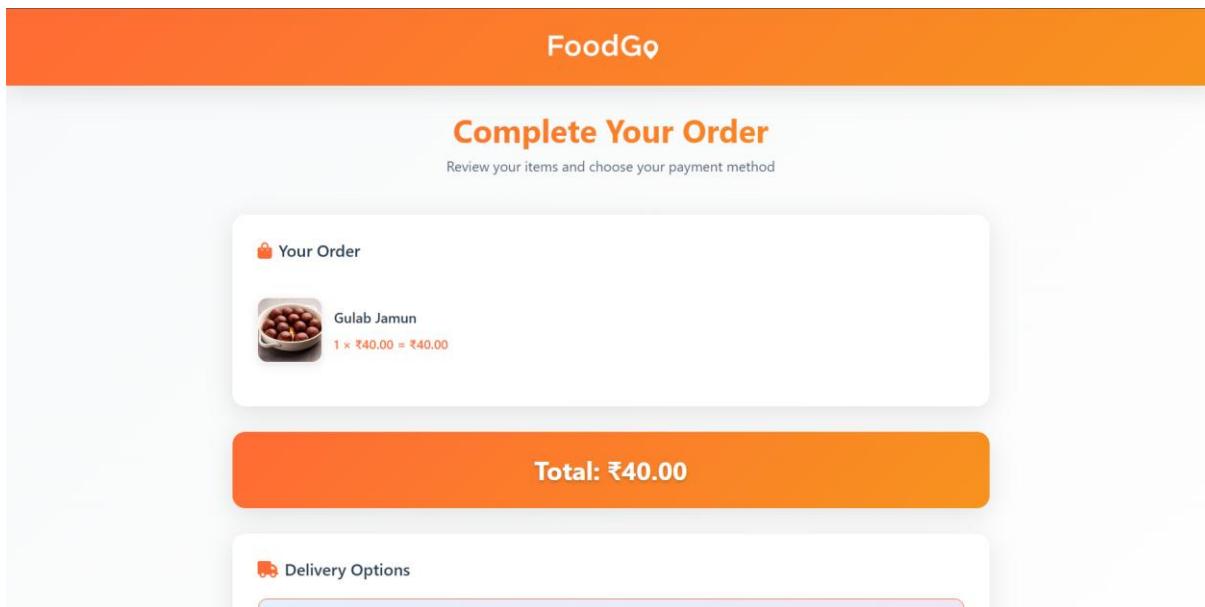
The screenshot shows the FoodGo search interface. At the top, there's a navigation bar with links for Home, About, Orders, Search, and a user profile. Below the navigation is a search bar with dropdown options for 'Search for' (Food Items) and a 'Search' button. There are also filter buttons for 'ALL', 'VEG', and 'NON-VEG'. A section titled 'Popular dishes for you' displays four items: Pav Bhaji, Vegetable Kurma, Masala Chai, and Masala Dosa, each with a small image and a 'VEG' badge.

## 6.FOOD DETAILS PAGE

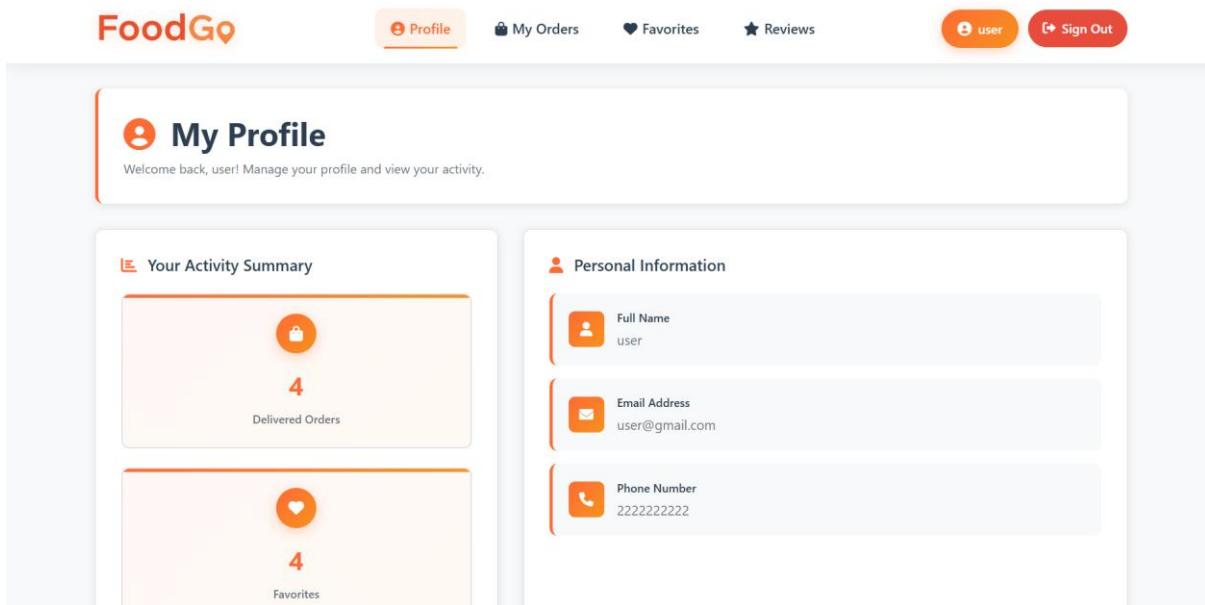


The screenshot shows the FoodGo details page for 'Gulab Jamun'. It features a large image of the dessert in a white bowl. The title 'Gulab Jamun' is displayed with a 'VEG' badge. Below the title, it says 'No reviews yet'. The dish is attributed to 'Ammachi's Table - Ernakulam'. A description states 'Soft milk sweets in syrup.' The price is listed as ₹40.00, and there's a 'Add to Cart' button. At the bottom, there's a section for 'Customer Reviews' with a 'Show Reviews' button.

## 7.PAYMENT PAGE



## 8.PROFILE PAGE



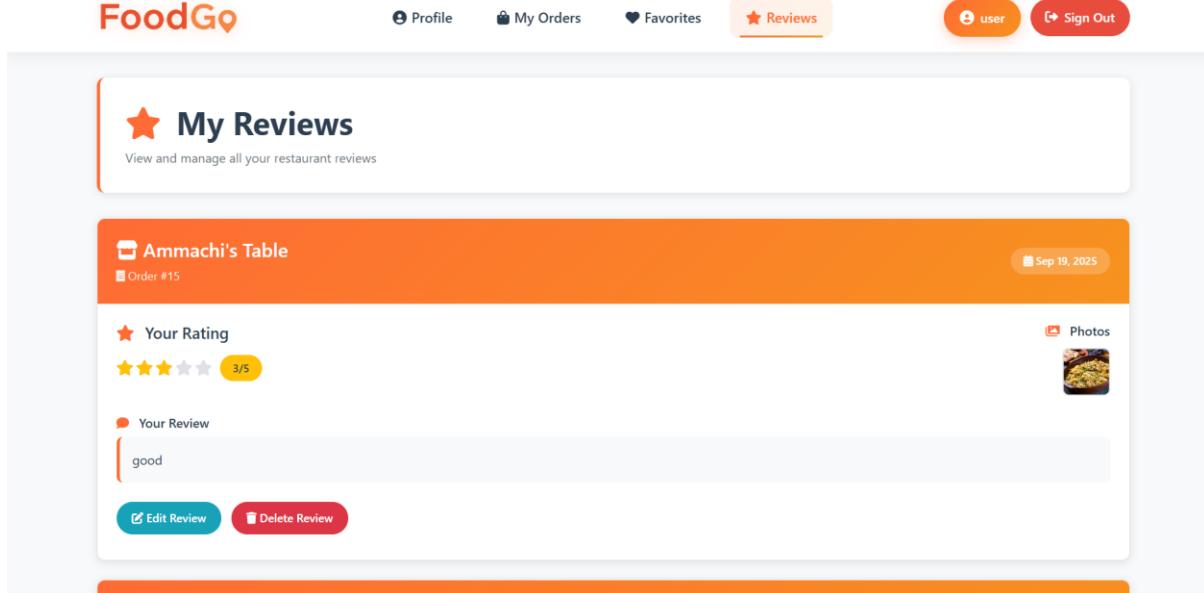
## 9.USER ORDERS PAGE

The screenshot shows the 'My Orders' section of the FoodGo app. At the top, there is a header with the FoodGo logo, navigation links for Profile, My Orders (which is highlighted in orange), Favorites, and Reviews. On the right, there are buttons for 'user' and 'Sign Out'. Below the header, a section titled 'My Orders' displays a single order. The order summary is for 'Order #17 - Ammachis Table' placed on 'Sep 19, 2025 at 4:45 PM'. A 'TIMEOUT' indicator is shown in the top right of this summary box. The order details show two items: 'Chicken Biryani' (x2) and 'Samosa' (x1). The total amount for the order is '₹390.00'.

## 10.FAVORITES PAGE

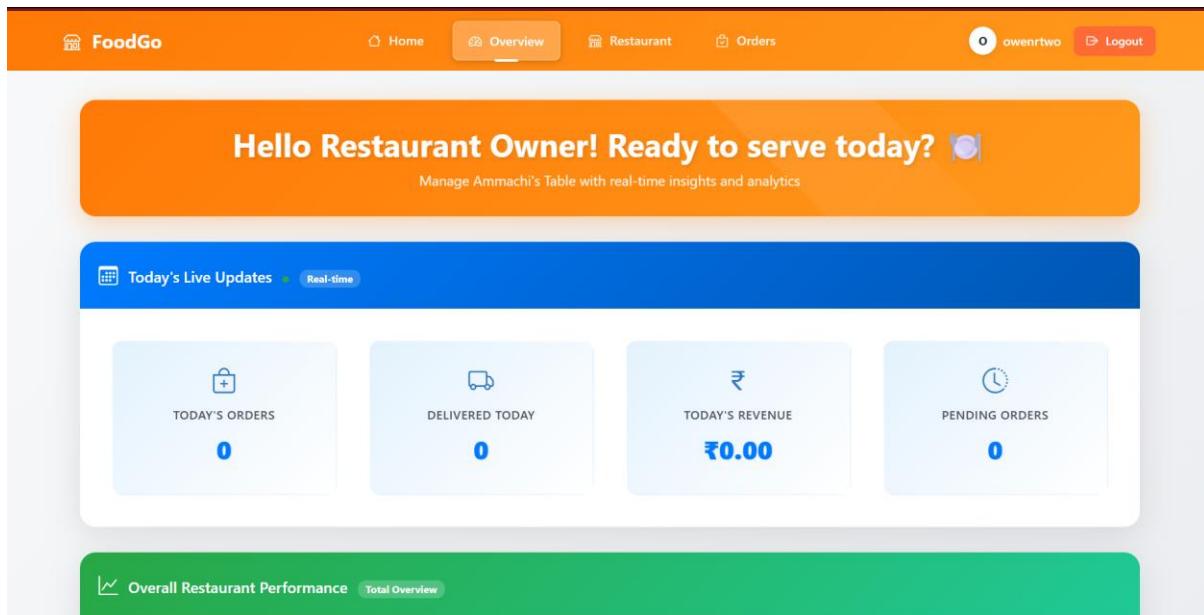
The screenshot shows the 'My Favorites' section of the FoodGo app. At the top, there is a header with the FoodGo logo, navigation links for Profile, My Orders, Favorites (which is highlighted in orange), and Reviews. On the right, there are buttons for 'user' and 'Sign Out'. Below the header, a section titled 'My Favorites' displays a collection of three favorite food items. Each item is shown with an image, name, price, and a note about the source restaurant. There is also a 'Remove from Favorites' button for each item. The items listed are: 'Medu Vada' (Rs 45.00) from 'Ammachi's Table', 'Idli & Chutney' (Rs 40.00) from 'Ammachi's Table', and 'Samosa' (Rs 30.00) from 'Ammachi's Table'.

## 11.REVIEWS PAGE



The screenshot shows the FoodGo platform's reviews section. At the top, there is a navigation bar with links for Profile, My Orders, Favorites, and Reviews (which is highlighted with an orange border). Below the navigation is a header titled "My Reviews" with a star icon. A sub-header says "View and manage all your restaurant reviews". The main content area displays a review for "Ammachi's Table" (Order #15). The review includes a rating of 3/5 stars, a text input field containing "good", and two buttons: "Edit Review" and "Delete Review". To the right of the review, there is a link to "Photos" with a thumbnail image of a dish.

## 12.RESTAURANT OVERVIEW PAGE



The screenshot shows the FoodGo platform's restaurant overview page. At the top, there is a navigation bar with links for Home, Overview (which is highlighted with an orange border), Restaurant, and Orders. The user profile "owenrtwo" is also visible. The main content area features a prominent orange banner with the text "Hello Restaurant Owner! Ready to serve today? 🍔" and a sub-instruction "Manage Ammachis Table with real-time insights and analytics". Below the banner, there is a blue header bar with the text "Today's Live Updates" and a "Real-time" button. The main dashboard displays four key metrics in light blue boxes: "TODAY'S ORDERS" (0), "DELIVERED TODAY" (0), "TODAY'S REVENUE" (₹0.00), and "PENDING ORDERS" (0). At the bottom, there is a green footer bar with the text "Overall Restaurant Performance" and a "Total Overview" button.

## 13. RESTAURANT MANAGEMENT PAGE

FoodGo

Home Overview Restaurant Orders

owenrtwo Logout

### Restaurant Management

Manage your restaurant and menu items

APPROVED

**Ammachi's Table**

AT 1234567891 Convent Road, Fort Kochi, Kochi, Ernakulam Delivery Not Available

EDIT DELETE ADD FOOD

Menu Items

## 14. VIEW ORDERS PAGE

FoodGo

Home Overview Restaurant Orders

owenrtwo Logout

### Customer Orders

Manage and track your restaurant orders from the last 24 hours

Filter by Status:

All Orders Pending Confirmed Ready Rejected Delivered Cancelled Timed Out

Search Orders:

Search by customer, items, phone...

CUSTOMER	PHONE	ITEMS	TOTAL	ADDRESS	STATUS	DATE/TIME	ACTIONS
----------	-------	-------	-------	---------	--------	-----------	---------

No orders found

Orders from the last 24 hours will appear here when customers place them.

## 15.ADMIN OVERVIEW PAGE

The screenshot shows the FoodGo Admin Overview Page. At the top, there is a navigation bar with the FoodGo logo, a user icon labeled 'admin', and a 'Logout' button. Below the navigation bar is a large orange header section with the text 'Welcome back, Admin!' and a small crown icon. Underneath this, it says 'Manage your FoodGo platform with comprehensive insights and real-time data'. The main content area has a blue header bar with 'Today's Live Updates' and 'Real-time' status. Below this are four light blue cards with icons and counts: 'NEW CUSTOMERS' (0), 'APPROVED RESTAURANTS' (0), 'TODAY'S ORDERS' (0), and 'DELIVERED TODAY' (0). At the bottom of the page is a green footer bar with 'Overall Platform Statistics' and a 'Total Overview' link.

## 16.RESTAURANT REQUESTS PAGE

The screenshot shows the FoodGo Restaurant Requests Page. At the top, there is a navigation bar with the FoodGo logo, a user icon labeled 'admin', and a 'Logout' button. Below the navigation bar is a large orange header section with the text 'Restaurant Requests' and a small checkmark icon. It says 'Manage pending restaurant approval requests'. To the right of this is a white box with a large '0' and the text 'PENDING REQUESTS'. The main content area below has a large white box containing a checkmark icon and the text 'All Caught Up!'. Below this, it says 'No restaurant requests pending approval at the moment.'

## 17.RESTAURANT MANAGEMENT PAGE

FoodGo

Overview Restaurant Requests View Restaurants View Users admin Logout

### Restaurant Management

Manage and monitor all registered restaurants

Search by: Name, District, Query

User ID	Restaurant ID	Owner Name	Restaurant (rating)	District	Delivery	Status	Actions
7	3	ownerthree	Ammachi's Table (★ 0.0)	Malappuram	Yes	APPROVED	<button>Menu</button> <button>View</button> <button>Ban</button>
4	2	owenrtwo	Ammachi's Table (★ 2.5)	Ernakulam	No	APPROVED	<button>Menu</button> <button>View</button> <button>Ban</button>

## 18.CUSTOMER MANAGEMENT PAGE

FoodGo

Overview Restaurant Requests View Restaurants View Users admin Logout

### Customer Management

Name, Enter search term...

Customer Name	Email	Phone	Registration Date	Role	Status	Actions
zam zam	zam@gmail.com	8787887787	Sep 16, 2025	Owner	Active	<button>View</button> <button>Ban</button> <button>Delete</button>
ownerthree	owner6@gmail.com	6666666666	Sep 16, 2025	Owner	Active	<button>View</button> <button>Ban</button> <button>Delete</button>
Mathew Selvi	mathewselvi29@gmail.com	9633035175	Sep 11, 2025	Customer	Active	<button>View</button> <button>Ban</button> <button>Delete</button>
usertwo	user2@gmail.com	5555555555	Sep 07, 2025	Customer	Active	<button>View</button> <button>Ban</button> <button>Delete</button>
owenrtwo	owner2@gmail.com	4444444444	Sep 07, 2025	Owner	Active	<button>View</button> <button>Ban</button> <button>Delete</button>