Software Requirements Specification   
–

Author(s): Arunadevi S

Date: November ,2024

Contents

[1. Abstract 3](#_toc104)

[2. Modules 3](#_toc115)

[2.1 Home Page 3](#_toc116)

[2.2 Place Order 3](#_toc122)

**3. Functional Requirements 4**

2.1 Login/Sign Up 4

2.2 Select Food Item 4

2.3 Check Out 4

[4. Non-Functional Requirements 5](#_toc104)

[5. High-Level Design 5](#_toc104)

[6. Low-Level Design 6](#_toc104)

7. Use Case Diagram 7

[8. Class Diagram 8](#_toc104)

[9. Sequence Diagram 9](#_toc104)

[10. Flow Chart ……………………………………………………………………………………………………. 10](#_toc104)

**11. ER Diagram …………………………..………………………………………………………………………..11**

# Abstract

The purpose of an online food ordering system is to provide a convenient and efficient way for customers to place their food orders, without having to physically visit a restaurant or call in an order. The web application will allow users to register and log in to the system, browse through various categories of food items, select the desired food items and add them to their cart, review the order details, and proceed to checkout. The application will provide a user-friendly interface for customers to easily navigate and place their food orders.

# Modules

## Home Page

First-time visitors to the food ordering website must create an account before being able to proceed with their order. Returning users can simply log in to access the menu and add items to their cart. Once the user has completed their order, they can proceed to payment.

## Place Order

Once the user has selected the food items they want, they can review the details of their cart, including the name, quantity, and price of each item. From the cart, the user can proceed to checkout, where they will be prompted to fill in their delivery address. The delivery address is a crucial step, as it helps ensure that the food is delivered to the correct location. Once the user has entered their delivery address, they can proceed to make payment using their debit or credit card. The website uses secure payment processing to ensure the safety and confidentiality of the user's financial information. The user can also choose to save their payment information for future transactions. After the payment is processed, the user will receive a confirmation, with the details of their order.

# Functional Requirements

## Login/Sign Up

The Login/Sign Up Page is the first step for users who want to access the website's features and order food online. If the user is a new customer, they can click on the "Sign Up" button, which will redirect them to the registration page. Here, the user will be asked to provide their personal information, including their name, email address, and password. After submitting the information, the user will receive a confirmation email, and upon clicking the link in the email, their account will be created. For returning customers, they can click on the "Login" button, which will redirect them to the login page. On this page, the user will enter their email address and password, and upon successful authentication, they will be redirected to the Home Page.

## Select Food Item

The Select Food Item page is where users can browse and choose the food items they want to order. The page is designed with a clean and intuitive layout, making it easy for users to find the food items they are looking for. The page features a search bar that allows users to search for specific food items by name or keyword, as well as food categories that allow users to browse items by meal type, such as breakfast, lunch and dinner. The food items are displayed with clear and eye-catching images, along with the name, description, and price of each item. In addition to browsing and selecting food items, the Select Food Item page also features a filtering panel, which allows users to narrow their search based on the food name. This makes it easy for users to find the food items that meet their specific needs and preferences. Once the user has selected the food items they want, they can add them to their cart, which will be displayed at the top of the page.

## Check Out

The Check Out page is the final step in the food ordering process. On this page, users can review their order details, including the items they have selected, the quantity of each item, and the total cost of their order. They can also enter their delivery address, which is a critical step in ensuring that their food is delivered to the correct location. This page features a secure payment system, which allows users to make payment using their credit or debit card. The website uses encrypted payment processing to ensure the safety and confidentiality of the user's financial information. The user can also choose to save their payment information for future transactions. Once the payment is processed, the user will receive a confirmation, with the details of their order.

# Non-Functional Requirements

## Reusable Code

Reusable code should always be a key factor that needs to be considered. The code should be reusable because it reduces the time it takes to develop application. The ASP.NET MVC framework provides a number of features and patterns that support reusability, such as the Model-View-Controller (MVC) design pattern, which separates the application logic, data, and presentation layers into distinct components that can be easily reused in other projects.

## Security

In an ASP.NET Core MVC web application that uses SQL Server as its database and implements stored procedures instead of hard-coded queries, security can be effectively maintained by adhering to industry standards and best practices. The use of stored procedures can help to mitigate the risk of SQL injection attacks by encapsulating and separating the database logic from the application code.

## Compatibility

The ASP.NET Core framework provides a flexible and scalable architecture that allows for easy integration with different operating systems and hardware configurations, and the use of a modular design and cross-platform compatibility can further enhance the application's compatibility across different environments.

# High-Level Design

* The home page of the food ordering system allows users to either sign up or log in. Once logged in, users can select their desired food items by filtering through various food categories.
* On the select food item page, users can browse through the available food items and add them to their cart for checkout. The page should display clear and concise information about the food items, including images, descriptions, and prices.
* The checkout page allows users to review their cart, make any necessary changes, and proceed to make payment using a credit or debit card.
* The page should also require users to fill out their delivery address. The checkout process should be secure, user-friendly, and easy to navigate.

# Low-Level Design

## Home Page

* Design a login and sign up form that captures user information and stores it in a database.
* Create a user interface that allows users to browse and filter food categories to select their desired food items.
* Implement a session management system to keep track of the user's session and allow them to access their food cart.

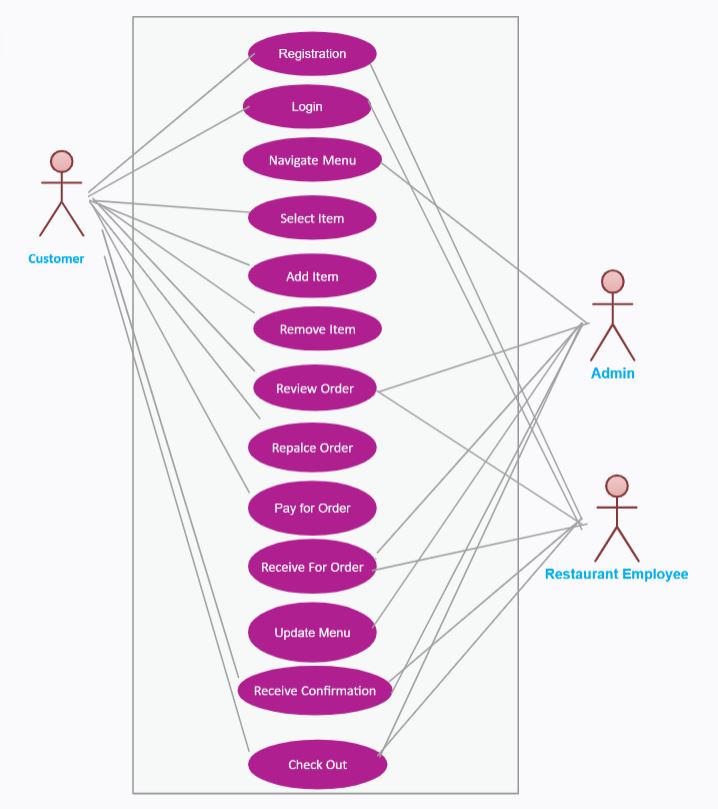
## Select Food Item

* Display food items with clear and concise information such as images, descriptions, and prices.
* Implement a system to add food items to the user's cart and calculate the total cost of their order.
* Provide an interface that allows users to modify the quantity of items in their cart and remove items as needed.

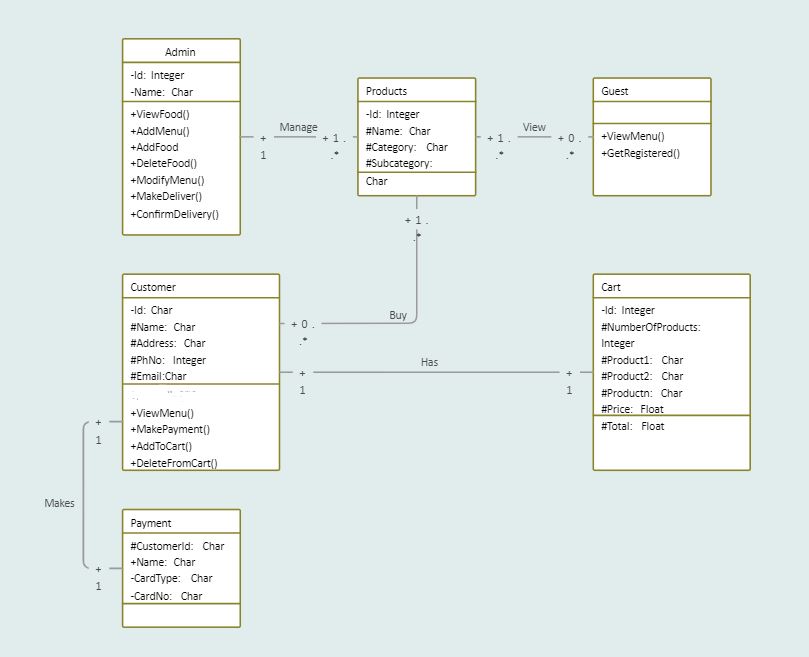
## Checkout

* Design a secure checkout process that requires users to enter their payment information, including their credit or debit card details.
* Implement a secure payment gateway to process the user's payment and complete the transaction.
* Store the user's delivery information, including their name, address, and phone number, and use it to generate the delivery address for the food order.
* Design a confirmation page that provides a summary of the user's order and the total cost, including any applicable taxes and delivery charges.

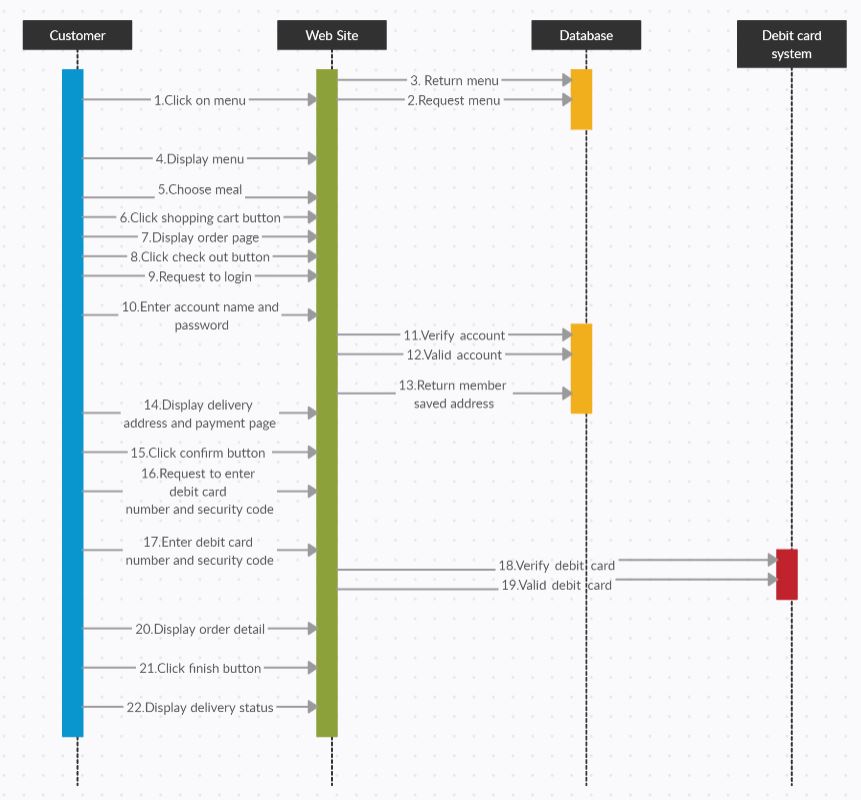
# Use Case Diagram



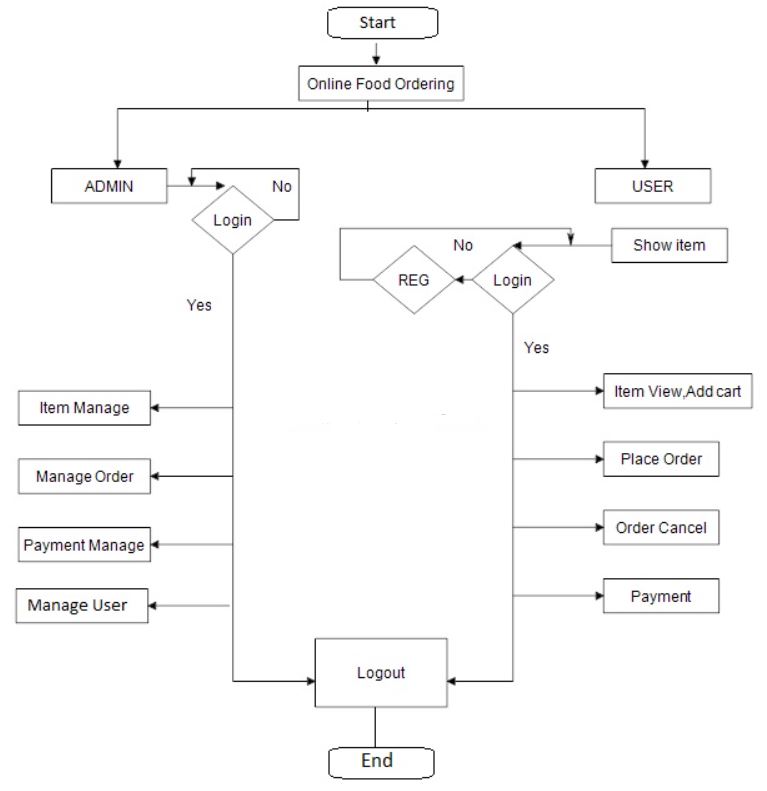
# Class Diagram



# Sequence Diagram



# Flow Chart



# ER Diagram

