

# Software Requirements Specification

Version 1.0

## FoodMate

Domain: Cloud Kitchen/Food Delivery

Category: Web Application Development

# Table of Contents

**1.0**

Background and Need for the System

**1.1**

Proposed Solution

**1.2**

Purpose of this Document

**1.3**

Scope of Project

**1.4**

Functional Requirements

**1.5**

Non-Functional Requirements

**1.6**

Interface Requirements

- Hardware
- Software
- Database Design

**1.7**

Project Deliverables

## 1.0

## Background and Need for the System

Eating home-cooked food can sometimes be a luxury or even a time-consuming chore considering the amount of preparation it can take. If one is exhausted after a difficult day's work or one is sick and unable to cook, ordering food online can be a good alternative. Perhaps, a group of people are suddenly visiting your home and there is no time or no available ingredients to cook an elaborate meal quickly for so many persons. Ordering online would be an easy way out here.

Besides these, there are various other circumstances where it makes sense to order food online than cook at home.

As per a recent study conducted abroad, there could be a scenario where by 2030 most meals currently cooked at home are instead ordered online and delivered from either restaurants or central kitchens. When restaurants and kitchens partner with online services to form a collaboration, such services are called as cloud kitchens.

## 1.1

## Proposed Solution

The proposed solution is a Web application that will act as an online bridge between customers/end users and restaurant staff. It will enable end users to browse through restaurant menus online and order food online. It will enable staff to add/modify items in online menus, accept orders, and so on.

The Web application will also enable users to register themselves, login to the application, look for food items, and place orders. Users can submit their feedback regarding the application using a feedback form.





## 1.2

## Purpose of this Document

The purpose of this document is to present a detailed description of the Cloud Kitchen or Online Food Delivery application, titled FoodMate.

Through FoodMate, individuals can browse breakfast, lunch and dinner items as well as snacks and choose whatever items they want and in whatever quantity. They can specify addresses for delivery and view their computed bill.

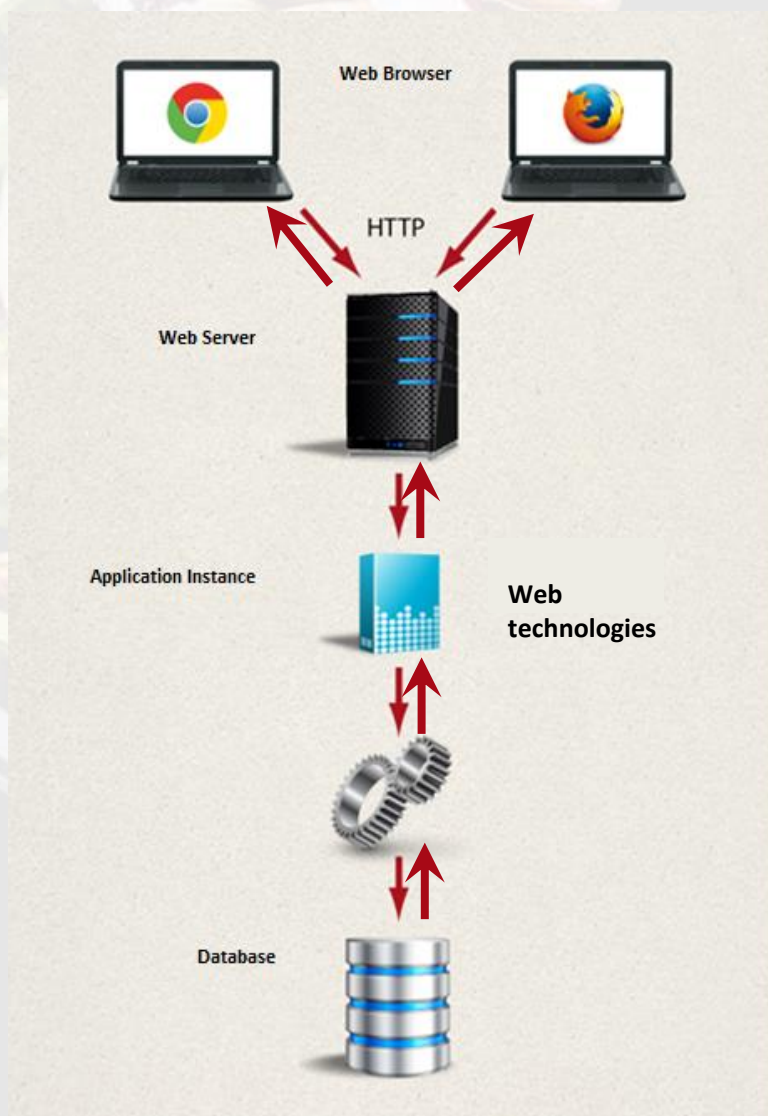
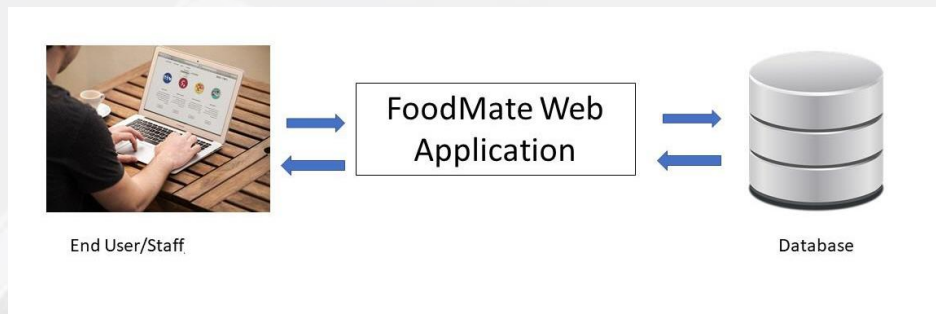
This document explains the purpose and features of the Website, the interfaces of the Website, what the Website will do, and the constraints under which it must operate. This document is intended for both stakeholders and developers of the Website and will be proposed to the client for approval.

## 1.3

## Scope of the Project

This Web portal will be a responsive and visually appealing Web application to be used by individuals. This portal will be designed to enable users to browse through and select food items to be delivered. Payment and delivery functionality is beyond the scope of the project.





## Broad View of the System



## 1.4

## Functional Requirements

Application will be designed with a set of forms/pages with menus representing choice of activities to be performed.

Following are the functional requirements of the system:

- **Home Page:** It will display general purpose information pages through menus such as Register, Login, Search, Place Order, and so on.
- **Register:** It will enable new users to register themselves with FoodMate. At the time of registration, users need to provide Name, Email ID, Contact Number, and Username, and then, select their Password.

Appropriate error-checking must be done on the fields of the form to ensure correct data. For example, email id can be checked to see if it is of appropriate format. (Hint: Use client-side validation).

- Staff need not register, they will be provided with readymade passwords which can be changed later.
- **Login:** It will allow users to login to the FoodMate application and access various features of the application. There will be two types of users for this application:
  - ☐ Customers (End users)
  - ☐ Staff (staff of restaurants)
- Each type of user will be presented with different menus. Login is the common option for both users.

## Features and Functionality for Customers

- **Account Settings:** Customers will be able to manage their accounts by using options such as Create, Update, and Delete Profile. A profile can include detailed information about the user, such as address.
- **Home Page:** The home page for customers should accept first name from the user and display a personalized welcome message. At the top corner, the user's first name should be displayed for the entire duration that the portal is loaded. Popular and featured food items should be displayed on the home page. Each food item can be presented as an image and a caption displaying name, brief description, and price. Also, the restaurant name from where it will be sent should be displayed.
- **Search:** Users can search for various food items such as breakfast, lunch, and dinner items besides snacks. Search can also be performed based on cuisine categories. Each food item can be presented as an image and a caption displaying name, brief description, and price, along with restaurant name from where it will be sent.
- **Filter:** Users can apply various filters on the displayed products. Filters can include vegetarian, vegan, Chinese, Mexican, and so on.
- **View Restaurants:** View the list of all the restaurants sorted by their current location.
- **View Menu:** View all the items available for a restaurant
- **Place an Order:** This menu option upon clicking should enable users to add one or more items to the checkout cart.
- **Checkout:** User must enter delivery address at this point. Total computed bill will be displayed to the user. You need not implement actual payment or delivery functionality.
- **History:** View the list of all the past orders placed till date.
- **Cancel Order:** Cancel an already placed order that is not yet delivered.
- **Submit Feedback:** Users can submit their feedback regarding the application using a feedback form.



## Features and Functionality for Staff of the restaurants:

- **View Orders:** View list of open orders.
- **Search Order:** Staff can search for a specific order by providing the Order ID.
- **Accept Order:** Staff can confirm and accept an open order.
- **Reject Order:** If a specific order cannot be fulfilled staff should reject it and provide the reason for rejection.
- **Generate Reports:** FoodMate will be able to generate and display different kinds of reports such as most ordered item, largest quantity ordered, and so on.

This option will be common to all users:

- **About Us and Contact Us:** This menu option should display Email id, address, and contact number of the organization who is developing the system.





## 1.5

## Non-Functional Requirements

There are several non-functional requirements that should be fulfilled by the system.

The system should be:



- **Safe to use:** The system should not result in any malicious downloads or unnecessary file downloads.
- **Accessible:** The system should have clear and legible fonts, user-interface elements, and navigation elements.
- **User-friendly:** The system should be easy to navigate with clear menus and other elements and easy to understand.
- **Operability:** The system should operate in a reliably efficient manner.
- **Performance:** The system should demonstrate high value of performance through speed and throughput. In simple terms, the system should be fast to load and page redirection should be smooth.
- **Security:** The system should implement adequate security measures such as authentication. For example, only registered users can access certain features.
- **Capacity:** The system should support large number of users.
- **Availability:** The system should be available 24/7 with minimum downtime.
- **Compatibility:** The system should be compatible with latest browsers.



## 1.6

## Interface Requirements

### Hardware

- Intel Core i3/i5 Processor or higher
- 8 GB RAM or above
- Color SVGA
- 500 GB Hard Disk space
- Mouse
- Keyboard

### Software

**Frontend:** HTML5, CSS 3, Bootstrap, JavaScript, jQuery, AJAX, XML

**Client & Server:** Java 8 or higher, Java EE 7 technologies

OR

Visual Studio IDE, C# 7.2 or higher, ASP.NET MVC

OR

PHP 7.2 or higher version with Laravel Framework (optional)

OR

Python 3 or higher and Flask

**Database:** MySQL 5.7 or higher/ SQL Server 2016 or higher

### Database Design

**Data Dictionary:** : Customer, OrderDetails, Item, Cuisine, RestaurantDetails, and so on.

Based on the given specifications, you will define suitable entities, attributes for these entities, and identify relationships between the entities.

For example, some entities along with their attributes can be identified as follows:

| Customer:        |
|------------------|
| 1. CustomerID    |
| 2. Name          |
| 3. Email ID      |
| 4. Contact Phone |
| 5. Username      |
| 6. Password      |
| 7. Address       |

| OrderDetails:   |
|-----------------|
| 1. CustomerID   |
| 2. OrderID      |
| 3. Date         |
| 4. Location     |
| 5. ItemName     |
| 6. ItemQty      |
| 7. RestaurantID |

| Item            |
|-----------------|
| 1. ItemID       |
| 2. ItemName     |
| 3. RestaurantID |
| 4. Description  |
| 5. Price        |

| Billing         |
|-----------------|
| 1. OrderID      |
| 2. CustomerID   |
| 3. RestaurantID |
| 4. BillAmount   |
| 5. Date         |

Similarly, you can define other entities and also relationships between entities and methods representing activities on the entities.

**Note:** These are just examples, you need not adhere to these structures and can design your own table structure with more or less columns.



## 1.7

## Project Deliverables

You will need to design and build the project and submit it along with a complete project report that includes:

- Problem Definition
- Design specifications
- Diagrams such as flowcharts for various activities, Data Flow Diagrams, and so on
- Database Design
- Source Code
- Test Data Used in the Project
- Project Installation Instructions (if any)

The consolidated project will be submitted as a zip file with a ReadMe.doc file listing assumptions (if any) made at your end and SQL scripts files (.sql) containing database and table definitions.

Over and above the given specifications, you can apply your creativity and logic to improve the system.

**Sitemap:** To understand the flow of FoodMate Web application, you will have to create a Sitemap and add it to the home page of your application.

~~~ End of Document ~~~