SOFTWARE REQUIREMENTS SPECIFICATION

for

Restuarant Management System

Version 1.0

Prepared by : 1.Neel K. Parikh (190001044)
2.Yash Suvarna (190001068)
3.Bhoomil Gohel (190001008)
4.Parul Mogre (190001037)
5.Shivesh Dave (190001057)

May 2, 2021

Contents

1 Introduction

The following section gives an overview of the subject Restaurant Management System's derived Software Requirements Specification (SRS) . To begin, the document's intent is explained, as well as its intended audience. Following that, the paper specifies the project scope, with a specific emphasis on what the resultant programme can do and the relevance.

1.1 Purpose

The aim of this SRS is to lay out the subject RMS's functional and non-functional specifications. The paper also includes a comprehensive profile of the external interfaces, performance considerations, and design limitations that will be implemented on the subsequent implementation. The following attributes are intended to be present in the presented set of requirements: correctness, verifiability, modifiability, and traceability. Foodex is an web application that aims to digitalize the process of a restaurant management operations including ordering and delivery management and status.
The development team responsible for implementing the listed RMS will be the primary audience of this SRS text. The target audience for the secondary document is the project's stakeholders, which include restaurateurs and their workers.

1.2 Document Conventions

Font Conventions: All user entities are capitalised in this text, with the first letter capitalised. In addition, every significant word mentioned in the glossary is bolded and italicised in the text. Those words, on the other hand, that are important (but not essential). In the text, words that aren't described in the glossary are bolded.

1.3 Intended Audience and Reading Suggestions

The purpose of this document is to provide a comprehensive overview of the programme specifications for "Foodex." It will show the intent, scope, and full explanation of the system's creation. It will also go over the system's specifications for external interfaces. Both functional and non-functional specifications This document is mainly intended for submission.

1.4 Project Scope

Foodex is a restaurant management system that was created with the goal of automating common restaurant tasks such as order and inventory management and feedback collection. This version of the programme would only deal with these activities.
The key goal is to boost the restaurant's efficiency by eliminating regular paperwork. Tasks will be completed in less time and with greater efficiency with this method.
Another advantage of this programme is that it can easily balance the load during rush hours, allowing restaurants to perform better than normal. Human error, which occurs while performing tasks, is also a factor.Manual labour is also reduced, and the presence of queues in the system for assigning tasks to chefs can help to alleviate kitchen congestion. The device would also result in a reduction in labour, lowering the restaurant's costs.
This concerns the electronic replacement of paper-based menus, a complementary electronic strategy for managing a customer's order at the front of house.

2 Overall Description

2.1 Product Perspective

The planned restaurant management system stores the following information

1. Menu Details: It includes the category, pricing and user_given average rating of each recipe in the menu.
2. Customer Description: It includes the customer name, email, address, phone number. This information may be used for keeping the records of the customer for any emergency of for any other kind of information. The email is also used for sharing the status of the food delivery.
3. Order Description: It includes the customer details, recipe details, time of order, status of the order, quantity ordered and the delivery address.

2.2 Product Features

The major features of restaurant management system as shown in below Relational Schema of Database (Relational Schema):

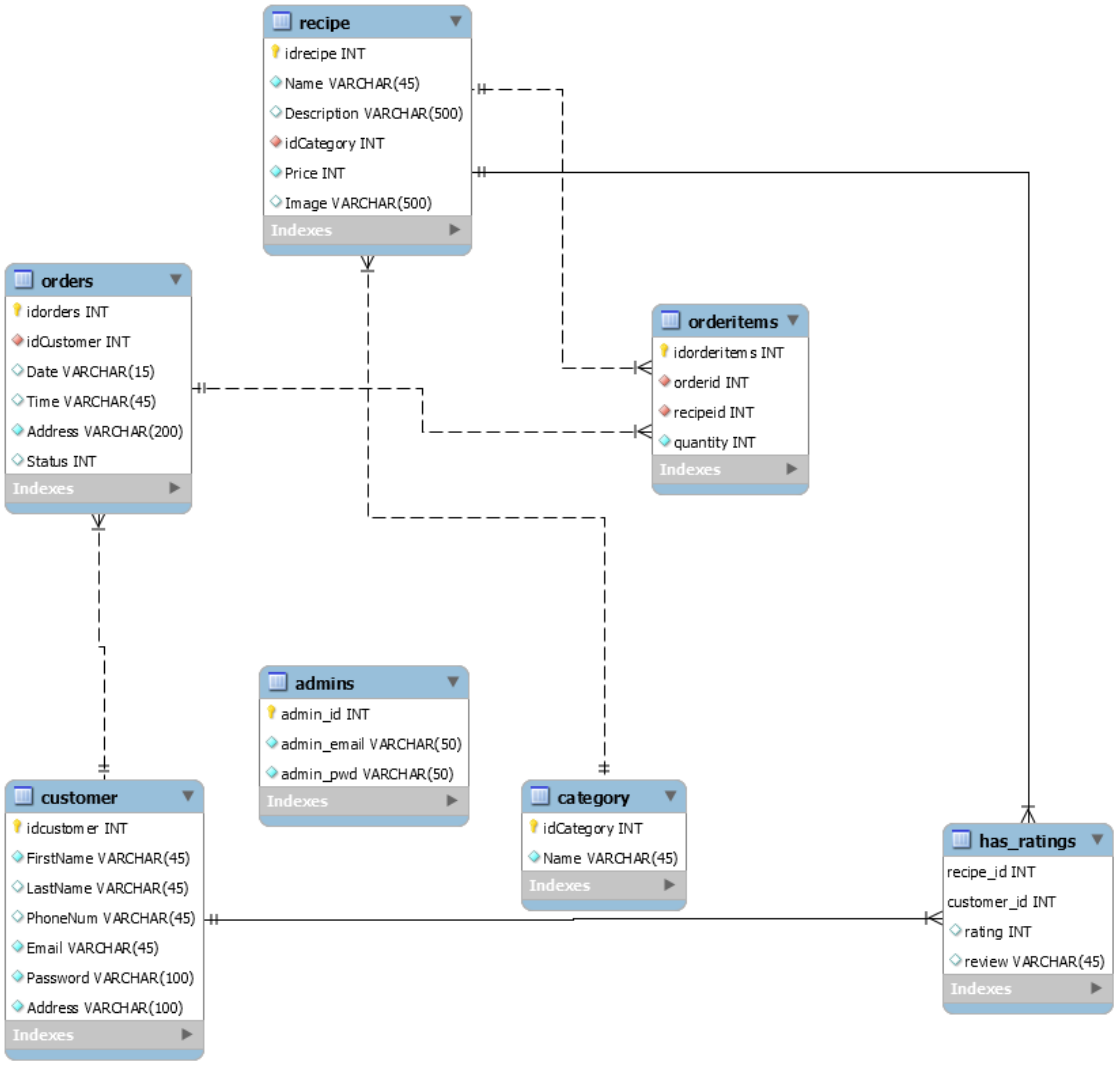

Figure 2.1: Relational Schema of Restaurant Management System

2.3 User Classes and Characteristics

There are different types of services based on the user type. The restaurant owners will be the administrator of the system while the customers will be the users. The features that are available to the Admin are:

1. Add or remove food items/recipes from the menu.
2. Access to all the orders that the system has received (with an additional viewing option ordered by status).
3. Update the status of a particular order.

4. View the details of all the registered customers of the system
5. Add or remove food category.

The features that are available to the customers(registered) are:

1. Look at the current menu and choose what to order from the menu.
2. The customer will have a "my cart" feature where all the items he/she chooses to order will be present(with respective quantity).
3. Place order from the "my cart".
4. View the status of the order in real time
5. Cancel order.
6. Rate food items.
7. Update Profile.

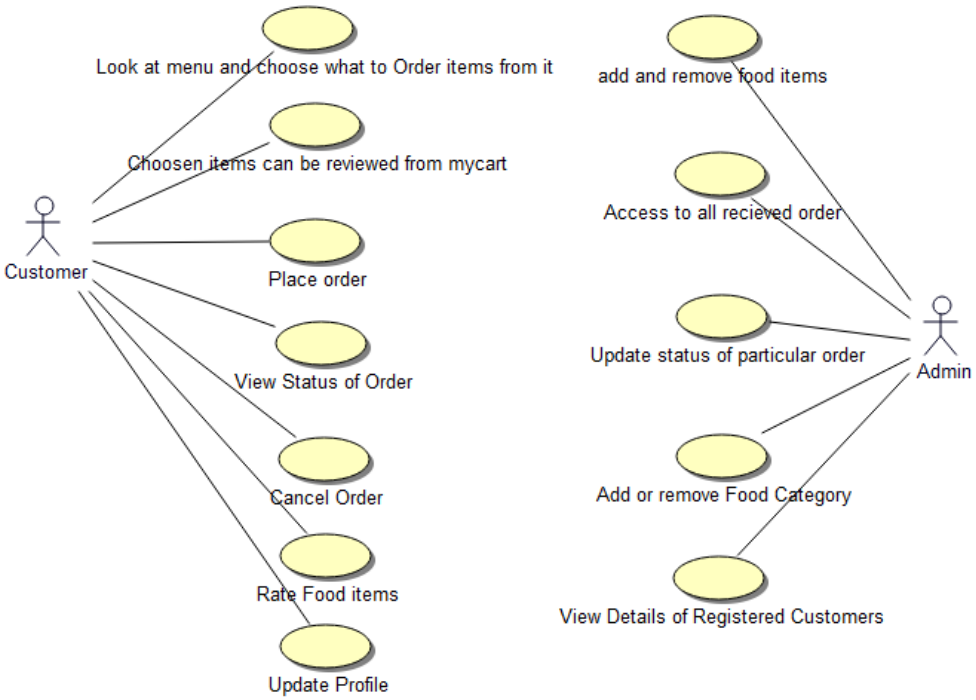

Figure 2.2: Use Case Diagram of Restaurant Management System

2.4 Operating Environment

The website can be used through any popular web browsers on any of the popular operating systems. Most of its features are compatible with the latest standards. The users would require a stable internet connection for accessing the website. The basic computer and its peripherals are need for easy access.

2.5 Software Configuration

This software package is developed using javascript as front end.
Microsoft SQL Server as the back end to store the database.
Operating System: Windows
Language: JavaScript, HTML5, CSS3 (front end)
Database: MY SQL
Server (back end): NodeJS
Hardware Configuration :
Processor: Pentium(R)Dual-core CPU
Hard Disk: 40GB
RAM: 256 MB or more

2.6 Assumptions ans Dependencies

Assumptions:

1. The coding should be error free
2. The system should be user-friendly so that it is easy to use for the users.
3. The information of all users, recipes and orders must be stored in a database that is accessible by the website.
4. The system should have more storage capacity and provide fast access to the database
5. The system should provide search facility and support quick transactions
6. The system is running 24 hours a day.

Dependencies:

1. The specific hardware and software due to which the product will be run
2. On the basis of listing requirements and specification the project will be developed and run
3. The end users (admin) should have proper understanding of the product

4. The system should have the general report stored
5. The information of all the users must be stored in a database that is accessible by the system.
6. Any update regarding men is to be recorded to the database and the data entered should be correct.

3 System Requirements

3.1 Place order

3.1.1 Description and Priority

The system allows a user to place an order using the website. It will display a list of available dishes. Customers will be able to select multiple dishes and their quantities for a particular order.
Priority: high

3.1.2 Stimulus/Response sequences

When a user enters the order page, the initial system displays a list of available dishes along with their prices.

1. Stimulus: The customer chooses an available dish.
   Response: The system shows the name of the dish and the price per serving. Also, it contains a text box for the customer to enter the quantity, OK button and a Cancel button.

2. Stimulus: The customer enters the quantity and presses the OK button.
   Response: The system closes the popup, shows a small green tick mark at the side of the dish. Below the tick mark, it shows the quantity selected and the total price of that dish.

3. Stimulus: Customer taps on confirm order button at the bottom
   Response: The system closes the order screen and displays a timer along with a "Cancel Order" button and "Edit Order" button

3.1.3 Functional Requirements

REQ-1: The system will show a list of cards (UI element) of dishes. Each card will have a picture of the dish. Below the dish, it shows the price in Rupees per serving and the option to rate and review the dish.
REQ-2: The user will be able to order multiple quantities for the same dish.

3.2 Admin Order

3.2.1 Description and Priority

Whenever a new order is placed by the customer, the dishes in the orders are classified into categories, which are visible to the admin. The admin can update the status of the order.
priority: high

3.2.2 Stimulus/Response sequences

Stimulus: Customer taps the "Confirm Order" button in the "Place Order screen."
Response: Display the dishes of the order to the admin.

3.2.3 Functional Requirements

REQ-1: System will classify the dishes in order according to category and Add this dish to a particular chef's queue on the kitchen screen.

3.3 Edit Order

3.3.1 Description and Priority

The customer can edit the order any time before the serving. In editing mode, the customer can change the quantity of the food ordered and add or remove dishes from the order.
priority: high

3.3.2 Stimulus/Response sequences

The screen shows two buttons "Cancel Order" and "Edit Order" button

1. Stimulus: Customer taps on the "Edit Order" button.
   Response: The user will be able to remove a dish that they had ordered or change the quantity of the dish ordered.

2. Stimulus: Customer taps on any new dish which was not previously selected
   Response: Stimuli/Responses of the "Place Order" feature will be followed.

3.3.3 Functional Requirements

REQ-1: The system must allow the customer to increase, decrease or even remove the dish from the order any time before serving.

3.4 Cancel Order

3.4.1 Description and Priority

We will also provide an option to cancel the current order. The customer can cancel an order until it has been served.
priority: high

3.4.2 Stimulus/Response sequences

1. Stimulus: Customer taps on the "Cancel Order" button
   Response: the system responds with a popup "Order canceled successfully" or order cannot be canceled."

3.4.3 Functional Requirements

REQ-1: The system must allow the customer to cancel the order at any time before serving.

3.5 Change the status of an order

3.5.1 Description and Priority

The admin can change the status of a particular order complete, out for delivery, etc.
priority: high

3.5.2 Stimulus/Response sequences

1. Stimulus: Admin chooses a new status for an order.
   Response: The system notifies the user via email about any change in the status of the order.

3.5.3 Functional Requirements

REQ-1: The system must send a notification to the hall manager once all the dishes of a particular order has been marked "cooked."
REQ-2: The system must replace the timer screen with a new screen having feedback and request bill options.

3.6 Customer Rating and Review

3.6.1 Description and Priority

The system will give customers the ability to provide feedback for the food. The customer can rate a particular food item. They can also give a text review about that item.
priority: high

3.6.2 Stimulus/Response sequences

1. Stimulus: The customer chooses to submit a review about a menu item.
   Response: The system takes input for the values

3.6.3 Functional Requirements

REQ-1: System must display the rating option to the user.
REQ-2: System must display the write a review option to the user.

3.7 Add/Edit/Delete Menu Items/Categories

3.7.1 Description and Priority

The system gives the ability to the admin to add, edit and delete menu items. Using In this feature, an admin can add menu categories, menu items.
priority: high

3.7.2 Stimulus/Response sequences

Admin screen shows all the previously added dishes. It also offers an "Add Dish" button along with "Edit" and "Remove" with all the available dishes

1. Stimulus: Admin taps on the "Add Dish" button
   Response:

   The system opens another screen with a form

2. Stimulus: Admin fills the information and hit submit
   Response: The system responds with "¡Dish¿ added successfully."

3. Stimulus: Admin taps on edit button
   Response: The system opens a screen with a form prefilled with the existing values.

4. Stimulus: Admin taps on remove button on a particular row
   Response: responds with a "¡Dish¿ removed successfully."

3.7.3 Functional Requirements

REQ-1: Admin should be able to add all necessary information about the menu category.
REQ-2: System must give admin the ability to edit information about any menu item
REQ-3: System must give the admin the ability to remove menu items and menu categories.

4 Other Non-Functional Requirements

4.1 Performance Requirement

The food management system will be used by various people and presumably at the time, so the system will be accountable to cater for the needs of many users and it is expected of the system to not crash at many simultaneous requests to the server.

1. The system would be required to handle large amounts of user data, process updates from the admin side and handle customer orders, thus it should work without fail.

2. The food Management System should be able to handle errors that were not encountered before and be reliable so that it doesn't cause long durations of downtime. This can be done by having inbuilt systems for error testing like catching invalid usernames or passwords.

3. If any corruption or failure happens refreshment page must be shown. Back up of the database must be retrieved from the software and saved by the admin.

4. The application is mainly HTML and scripting language-dependent, so the end-user part is fully portable and any system or software, the web browser should be able to deploy the software. The software can run on any laptop, computers, etc.

5. The system should be fast and accurate.

4.2 Safety Requirement and Reliability

In any software, the overall reliability of the system depends on the stability of the container and its underlying OS. The database may crash because of a virus or operating system failure. The main pillar of reliability of the system is the backup of the database which is constantly maintained and updated to reflect the changes. A UPS/Power supply backup would be worth investing in. The software must be available at all times to all users meaning it can be accessed at any time. If any corruption or failure happens refreshment page must be shown. Back up of the database must be retrieved from the software and saved by the admin.

4.3 Security Requirement

1. User accounts are perfectly secure with hashed passwords.

2. The user is properly authenticated during signup. An email is sent from the server to the registered user account during sign-up or in case the user forgets the password.

3. Admin has the sole control over the database and edits access. However, the user passwords are securely encrypted and hashed, thus user accounts cannot be accessed by anyone but the user.

4. The normal user can only read information but cannot edit it except their personal info.

5. There should be separate accounts for admin and members such that no member can access the database and only admin has the rights to update the database.

6. In case of system failure, incomplete or cancelled transaction requests must be rolled back to avoid any monetary loss to either side.

7. The system user Secured Socket Layer (SSL) in all transactions to secure the confidential transactions and information sensitive data will be encrypted.

4.4 Requirement Attributes

1. There are many admins working on the system development, so all of them will have edit and update access to the database. There can be different admin accounts or a single account can be shared by all the admins depending upon the need.

2. The database should be designed in such a way that is comprehensible and easy to use for the admin.

3. The system should be easy to install and use in the user's system.

4.5 Business Rules

These include the rules and regulations this the System users should abide by. Illegal rules and practices must be avoided by the user and admin. No one should break the rules and indulge in any kind of malpractices.

4.6 User Requirement

The users of this system are regular customers and presumably the owner of the food chain restaurant. The users are expected to have a basic knowledge of mobile applications and internet browsing. The software must be installed onto their devices to access the functionalities of the same. The admin or the owner must have the knowledge of handling databases and should be able to add more admins if needed. The proper user interface, user manual. online help and the guide to install and maintain the system must be

sufficient to educate the users on how to use the system without any problems.
The system should be capable of:

1. Backup and Recovery
2. Data migration i.e, whenever user registers for the first time then the data is stored in the server
3. Setting new passwords in case the user forgets the existing one.
4. Data replication i.e. If the data is lost in one branch, it is still stored with the server
5. Auto Recovery i.e. frequently auto saving the information
6. Maintaining files i.e. File Organization

5 Other Requirements

5.1 Copyright,Legal and Other Notices

Our Foodex team reserves all rights.

5.2 Applicable Standards

It varies with industry standards.

5.3 Licensing Requirements

These are required to run a restaurant, so it is not applicable here.

5.4 Further Requirements

Further Requirements will change with changes in the restaurant.