Software Requirements Specification

for

Automated Restaurant System

Version 1.0

Prepared by Spice Boys

Stephen F. Austin State University Computer Science Department

9/22/2021

Table of Contents

Table of Contents ii

Revision History ii

1. Introduction 1

1.1 Purpose 1

1.2 Document Conventions 1

1.3 Intended Audience and Reading Suggestions 1

1.4 Product Scope 1

1.5 References 1

2. Overall Description 2

2.1 Product Perspective 2

2.2 Product Functions 2

2.3 User Classes and Characteristics 2

2.4 Operating Environment 2

2.5 Design and Implementation Constraints 2

2.6 User Documentation 2

2.7 Assumptions and Dependencies 3

3. External Interface Requirements 3

3.1 User Interfaces 3

3.2 Hardware Interfaces 3

3.3 Software Interfaces 3

3.4 Communications Interfaces 3

4. System Features 4

4.1 System Feature 1 4

4.2 System Feature 2 (and so on) 4

5. Other Nonfunctional Requirements 4

5.1 Performance Requirements 4

5.2 Safety Requirements 5

5.3 Security Requirements 5

5.4 Software Quality Attributes 5

5.5 Business Rules 5

6. Other Requirements 5

Appendix A: Glossary 5

Appendix B: Analysis Models 5

Appendix C: To Be Determined List 6

Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Date** | **Reason For Changes** | **Version** |
| Spice Boys | 9/30/21 | Completed Document | 1.0 |
|  |  |  |  |

# Introduction

## Purpose

The purpose of this document is to present a detailed description of the software that will be used to automate certain tasks within a restaurant operation. There will be descriptions of the software that will streamline handling tables when they are seated (whether by walk-in or reservation), empty, or on standby to pay and leave, the software that handles the payments made by customers, as well as the software that is used for ordering food.

## Document Conventions

The font that we used for this SRS is Times font size 12 and the header is Times font size 14. The requirement that would take precedence over would be the User interface. Once created, the other parts that we would need to put on the user interface can be added on to it so the requirements would be inherited on to the user interface.

## Intended Audience and Reading Suggestions

This document is intended for the developers and those who use the system. The SRS would consist of an introduction of the product such as its purpose and scope. The overall description of the product, external interface requirements, system features, and nonfunctional requirements.

## Product Scope

The product is a software application that is accessed from a website, the website is displayed on a table where the customers interact with it. The goal of this is to decrease the inefficiency in the way a restaurant handles customer service which can enable the restaurant to serve more customers minimizing service time. The software allows the customer to make reservations, create their orders and payments. The more efficient the restaurant can be run, the more customers they can serve, the more revenue is generated. Having a self-service system allows customers to make their needs met easier which can promote the restaurant by how unique the system is.

# Overall Description

## Product Perspective

This product is a first-generation system that will have many subsystems that communicate both with each other as well as with the restaurant staff. There will be an interactive seating system that ensures people are sat in a way that optimizes the efficiency of the wait staff as well as keep customers at a comfortable distance from other guests when the restaurant is not full. It will interact with a food order placement system that gives each table its own documentation detailing the items that were purchased as well to pay the bill electronically. Both previously mentioned systems will interact with associates of the restaurant for when a customer has specific requests including but not limited to: paying in cash, requesting a specific table or wait staff, and splitting bills amongst the members of a party. To state the perspective broadly, this system will be comprised of smaller subsystems that are dynamically interacting with each other as well as the staff.

## Product Functions

* Provide a status on each table for use by staff
* Allow for reservation for specific tables depending on availability
* Provide a way for users to order food from the table or upon reservation
* Supplement payments for checks, which can be parsed among users upon payment
* Create a system by which users can flag waiters

## User Classes and Characteristics

* Customer - has a name, can reserve tables at a minimum of one hour in advance, pick up or dine in, food order, food bill, food paid, call server, can reserve tables at a minimum of one hour in advance, customers will be alerted through their phone.
* Employee (store computers) - get and set table status, can print physical receipt, names for tables, get/set food order, get/set food bill, and get/set food paid, turn off get server
* Manager – edit menu, refund, change order, change table status
* Table – status (seated, eating, calling waiter, check paid, need cleaning, awaiting reservation), name

## Operating Environment

The application is accessed by a web browser with an internet connection. The hardware used to access the website is desktop computers, Macs, and mobile devices with web browsers. Browsers to be supported are Chrome, Microsoft Edge, Safari, and any browser that is chromium based.

## Design and Implementation Constraints

Limitation constraints that need to be taken into consideration would be the budget of the shareholder and development team. Existing hardware would also be taken into consideration, the product should be compatible with the original hardware so limitations such as memory should be considered. The product should be flexible as well, being able to be compatible with the existing websites and systems.

## User Documentation

Website will follow standard online ordering conventions, with options to reserve tables and order food. An FAQ3 page will be included with anticipated questions, including how to use the website’s systems. A user manual with brief, succinct instructions on staff-side usage will be included for both website and table tablets. The interface of the tablets will walk users through ordering food and payment.

## Assumptions and Dependencies

It is assumed that the customers will be able to either place an order through a website, or through a tablet on the tables. It is also assumed that both restaurants and customers will be able to access the internet.

# External Interface Requirements

## User Interfaces

There will be three main interfaces, each serving a specific purpose:

* Ordering/Payment system – Small tablet interface used by patrons
  + Allows customers to order food
  + Provides a way to request assistance from an associate
  + Provides a means of electronic payment
  + Capable of splitting tickets amongst the members of a party
* Seating System – an interface that will have a visual layout of the restaurant with all tables
  + Tables will be color coded to indicate a status for each table
    - Red – occupied
    - Blue – ticket paid and on standby for customers to leave so that a busser can prepare a table for next occupants
    - Green – empty and ready to be seated
    - Tan – expecting a party by reservation
  + Allows an associate to move connect or disconnect tables from others to accommodate special seating requests
* Website – an interface that can be accessed by customers to perform certain tasks
  + Order takeout food
  + Make table reservations
  + Display menu items

## Hardware Interfaces

The website can only be accessed by devices that have a wired or wireless internet connection. Computers and mobile devices must have a web browser. The host device will have a secure connection to the restaurant's servers. Since the software is a web application, there is no need to download any application to use the website.

## Software Interfaces

Application will use a website to host the GUI5, using a local host that can be accessed by the restaurant's personal hostname/port number using the Node.js8 library. The website uses HTML6, and CSS2/JavaScript and uses libraries like Angular1 for the function of the website and bootstraps to make the website look nice. The operating system will be iOS, Android, and Windows which can be accessed using all devices. The databases would be created using SQL10, which contains information about the menu’s items and table accessibility.

## Communications Interfaces

When connected to the website the user will be connected by a HTTPS7 connection. This protocol encrypts the user’s information such as their name, phone number and party size.

# System Features

## Table Status Feature

4.1.1 Description and Priority

The Table Status Feature provides associates with a visualization of the physical layout of tables in the restaurant, each assigned a color-coded status. Status can be changed manually by an associate. Some status updates happen automatically.

4.1.2 Stimulus/Response Sequences

Customer places reservation online and picks empty table for a specified time. When customers are seated at a table the system will then set the status as “Seated”. Once food is served the system will set the status to “Eating”. Anytime the customer needs a waiter, the customer will request one through the tablet interface, and the status will be set to “Calling Waiter”. A waiter will update that status back to “Eating” once the customer has been assisted. When the customer is finished with their meal, they will interact with the tablet to pay. Once the payment has been processed the status will be set to “Check Paid”, telling staff to clean the table. As soon as the table is cleaned the staff will set the status as “Empty”.

4.1.3 Functional Requirements

41A: The Table Status System (TSS)

**4.2 Reservation and Seating Feature**

4.2.1 Description and Priority

The Reservation and Seating Feature is a means of remotely reserving table(s). The online service will provide the customer with a visualization of what tables are open and which ones reserved or currently occupied. In addition, there will be a visual that displays typical heavy business hours for each day of the week.

4.2.2 Stimulus/Response Sequences

Customer selects time of reservation. WebApp displays available tables for specified time. Customer chooses table or tables to be reserved. Table status is changed to reserved 15 minutes ahead of reservation time. Associate prepares tables for Customer’s arrival. Customer arrives at reservation time.

4.2.3 Functional Requirements

42A: WebApp shall allow the customer to specify time of arrival

42B: WebApp shall display all available tables at the customer’s specified time of arrival in the physical layout of the restaurant.

42C: WebApp shall notify Table Status System that the selected tables are reserved for the specified time.

**4.3 Order Feature**

4.3.1 Description and Priority

For better time management, customers will be able to have access to a point-of-sale device, in this case a tablet. This would have a high priority to eliminate wait time on waiters bringing checks. Component Ratings Benefit: 8, Cost: 8, Risk: 5

4.3.2 Stimulus/Response Sequences

User inputs order(s), system adds order(s) to restaurant queue, the order is placed on a running tab for the customer, once food is made the order is removed from the restaurant’s queue. Once the customer is satisfied the customer will press a button to end his running tab, the system then totals the tab. The system then shows a slider, allowing for user to increase or decrease the tip while displaying the customers total with the tip. The system then asks the user what their preferred method of payment would be. Cash payments would be processed through a waiter. Card payments would be inserted into the tablet by the users. The users would input the necessary information to process their payment. The system would process the payment, then ask the user if they need a receipt. If the user wants a receipt the system would print it out, then clear the tab.

4.3.3 Functional Requirements

43A: The tablet provides a customer with an interactive touch screen.  
44B: The tablet includes a card reader to gather payment information.  
44C: Tablets should have low latency in alerting the status of the table, sending orders, and processing payments.

**4.4 Waiting Feature**

4.4.1 Description and Priority

The Waiting Feature provides the customers a way to contact an associate in the case that the customer needs assistance through a notification system on the tablet. Sending this notification will update the tables status to “calling waiter”. When customers have paid all pending tabs through the tablet, the Seating Status System is notified so that it can change the status of table to “check paid”.

4.4.2 Stimulus/Response Sequences

Customer needs assistance. Customer notifies that they need assistance. Seating Status System updates tables status to “calling waiter”. Wait staff respond to customers' needs.

4.4.3 Functional Requirements

44A: Tablet at table shall allow customer to notify an associate for assistance.

# Other Nonfunctional Requirements

## Performance Requirements

The website must handle at least 400 users at the same time. The table tablets must be able to communicate in real time for payments and orders. Any requests made by the user will be pushed to the server.

## Safety Requirements

Possible loss and damage include device or accessory structural failure and theft of the tablets. Besides the device breaking, there is nothing that can cause harm to the customers when using the device. The safety precautions to prevent the device from breaking would be providing a case, screen protector, and stationary stand that will prevent device theft. When accepting payments, FDIC4 guidelines will be followed to ensure that the payment is valid and harm the consumers that are submitting payment. The tablet will also follow SEC9 guidelines to ensure that the transfer of data from customers/bank to retail is secure, so confidential information will not be exposed.

## Security Requirements

The website is encrypted with HTTPS7. We will not make or store any user data outside of sales. The need to protect user privacy is not needed to a certain degree.

## Software Quality Attributes

To keep our software up to date, we would have monthly updates for the website. By taking in data from customers we can also create a recommendation system based on the most ordered item. This would make the software more adaptable and maintained. The software can be continuously reused while the server is up and running, since the website is the host.

## Business Rules

Customers can order, call waiter, and pay from the tablet. Waiters can update table status given service conditions, such as when a customer has been seated, answering call from customer, and cleaning table(s) after customer has left.

**Other Requirements**

Other systems necessary include a menu system and Graphical User Interface (GUI5). The menu system is composed of a list of restaurant items to be ordered by the customer, and items will be created by the restaurant staff. The GUI5 will allow users to select orders and call the waiter when necessary.

Appendix A: Glossary

1. *Angular* - a TypeScript-based, free, and open-source web application framework used for web application development
2. *CSS (Cascading Style Sheets)* - A style sheet language used for designing the way that the text looks on websites which use other languages, such as HTML6.
3. *FAQ (Frequently Asked Questions)* - A list of questions and answers that allows
4. *FDIC (Federal Deposit Insurance Corporation)* - The Federal Deposit Insurance Corporation is one of two agencies that provide deposit insurance to depositors in American depository institutions, the other being the National Credit Union Administration, which regulates and insures credit unions.
5. *GUI/UI (Graphical User Interface/User Interface)* - The interface with which the users will directly interact with to manage the data, which is graphically enhance for a more intuitive and simplified user experience.
6. *HTML (Hypertext Markup Language)* - A markup language used in web browsers to display text in
7. *HTTPS (Hypertext Transfer Protocol Secure)* - a secure communication protocol for computer networks which is encrypted using Transport Layer Security
8. *Node.js* - Node.js is a JavaScript runtime environment that executes JavaScript code outside of a web browser.
9. *SEC (U.S. Securities and Exchange Commission)* - a large independent agency of the United States federal government, created to protect investors; maintain fair, orderly, and efficient markets; and facilitate capital formation.
10. *SQL (Structed Query Language)* - a domain-specific language used for the management of database systems.