Kitchen Sync
Software Requirements Specification
CS 3398
Software Engineering
Summer 2025

Wyatt Churchman
Christopher De Sousa
Aslan Delroba
Johan Ejiasi

June 2025

Contents

1 Introduction

Primary responsibility: Wyatt Churchman

1.1 Purpose of the SRS

The purpose of this Software Requirements Specification (SRS) is to have a comprehensive detailed description of the KitchenSync Commercial Kitchen Managing System. This document is the agreement between the development team and the stakeholders, recording what the system will do, how it will work, and the constraints in which it will work. This document is aimed at the restaurant owners, system developers, testers, and project managers.

1.2 Problem Statement

Commercial restaurants face acute operational problems at peak service hours that result in lost sales, food spoilage, and customer discontent. Current manual and semi-automated solutions fail to address some major problems:

- **Inventory Control Problems**: Restaurants often over-sell menu items, leaving customers disappointed and losing sales when kitchen staff discover products 86'd only after orders are taken
- **Kitchen Communication Breakdown**: Servers must walk physically the kitchen to share special instructions and allergy information, disrupting workflow and providing opportunity for miscommunication
- **Ineffective Order Routing**: Kitchen stations receive unnecessary tickets for items they do not prepare, resulting in clutter and confusion during high volume times
- **Manual Staff Tracking**: Paper-based time tracking systems are prone to errors and making payroll processing time-consuming and inaccurate
- **Isolated Payment Processing**: Disconnected order entry and payment processing systems result in tip calculation discrepancies and sales reporting
- **Simultaneous Access Issues**: Multiple servers attempting to place orders simultaneously frequently causes system conflicts as well as mislaid orders

KitchenSync is able to resolve such problems through a comprehensive solution that manages real-time stock, directs orders intelligently to the right kitchen stations, tracks staff hours automatically, and processes payments seamlessly with the ability to handle multiple users at once.

1.3 Product Scope

This requirements document provides scope of KitchenSync Software. The system has the following capabilities:

- Input orders being dispatched to the kitchen and printers located within the kitchen.

- Allow employees to clock in and out while specifying the number of hours logged in per day.
- Allow managers to input the amount of stock of a unit of food being sold on the day.
- Allow card and cash payments that can be charged to customers by servers for their meals.
- Saves all the orders and transactions received by servers throughout the day in order to automate the tip out.
- Allows servers to enter their own comments on an order so that they are able to tell the kitchen without having to go to the kitchen.

1.4 Intended Audience

Restaurant owners are the target audience for this software. Owners require a system that is free of bugs and errors since any time a system is closed for repair costs money to repair and loss of earning by service.

1.5 Definitions, Acronyms, Abbreviations

Terms	**Definition**
Server	Someone who serves food and beverages to customers. They ring out customers and take orders.
Cook	Only clock in and out functions available on the main computer.
Manager	Can view all orders made and can enter stock quantity into main computer.
86'd	A kitchen phrase when a dish on the menu is out of stock no longer and should not be served for sale.
Main Computer	The computer that accepts input orders. Also computer that performs transactions, stock data, and 86'd items.
Expo Station	Station that accepts all tickets and checks food for quality.
Grill Station	Station that accepts all proteins and Sauté station tickets.
Sauté Station	Station handling all orders in case there is a side that goes together with a protein from the Grill station, appetizers, and pastas.
Salad Station	Station handling all salads and desserts.

1.6 Overview/Document Conventions

This document employs the outline style provided by IEEE to demarcate information into disparate sections and subsections to aid readability and comprehension of functions in the KitchenSync system. For specification of physical objects are given subsections in a way that the operations are identifiable with the object.

2 Overall Description

Primary responsibility: Christopher De Sousa

2.1 Document Perspective

The document of software specification describes the non-functional and functional requirements of a kitchen stock information, transaction database, and order creation for kitchen software. If the requirements are functional, the document first describes a straightforward path taken by a user and then describes the requirements the software shall implement in order to adhere to the fundamental path. The document further outlines some non-functional but necessary traits that the software shall have.

2.2 Document Functions

This document gives the client a simple outline of how the user will use the software. The developers receive a simple template for the software.

2.3 User Characteristics

The system is built with users who know how to use a simple computer and operating systems.

2.4 User Interface

The software should allow users to view available menu items. The software should also allow users to conduct transactions with cash and card. Allow users to input quantity of food items in stock. Finally, allow the user to clock in and out.

2.5 End-User Operating Environment

The software can be accessed by anyone who has purchased the program with a computer and is internet-linked. The users can access the application and begin using the service to take orders, clock in and out, close transactions, and stock food items.

2.6 Design And Implementation Constraints

The user must have a computer and printers with an internet connection to be able to access the software.

3 Statement of Functional Requirements

Primary responsibility: Aslan Delroba

The functional requirements are listed in terms of activities that can be accomplished by the users of the KitchenSync system. Every task definition includes the user workflow, GUI and system requirements descriptions needed to effectively execute the task.

3.1 Task 1: Manager Daily Stock Management

User Task: Restaurant manager enters daily stock levels for each menu item before opening of service.

Task Workflow:

1. Manager logs in to system with administrative credentials
2. Manager goes to Stock Management interface
3. Manager enters quantity available for each menu item by category
4. System checks input and saves stock quantities
5. System provides feedback that the stock has been input successfully

GUI Description: Stock Management GUI displays a tabbed list of all menu items with entry fields for quantity. Items are categorized under Entrees, Appetizers, Desserts, and Drinks with low stock (yellow) and 86'd items (red). The GUI contains "Save All", "Reset", and "Generate Stock Report" buttons.

Requirements:

1. System will store quantity of stock for each menu item every day
2. System will prevent input of negative stock values
3. System will automatically place items in 86'd status once stock reaches zero
4. System will accommodate adjustments of stock within service time
5. System will maintain stock history for reporting

3.2 Task 2: Server Order Creation

User Task: Server enters customer orders with special requests and modifications.

Task Workflow:

1. Server logs in with specific employee ID
2. Server selects table number and enters party size
3. Server browses menu categories and adds items to order
4. Server adds special instructions or allergy alerts
5. Server sends order to kitchen stations

GUI Description: Order entry screen has large category buttons with running order summary on right-hand side. Special instruction fields are easily visible with each item. Interface shows real-time inventory with unavailable items grayed out.

Requirements:

1. System will permit concurrent order input by several servers
2. System will associate orders with an individual server ID to track tips
3. System will verify menu items against existing inventory
4. System will allow partial orders to be saved to be completed later
5. System will recognize special instructions and food allergy information

3.3 Task 3: Kitchen Ticket Distribution

User Task: Kitchen staff receives appropriately routed tickets to their station.

Task Workflow:

1. Tickets print automatically in sequence at relevant kitchen stations
2. Staff verifies ticket data such as special instructions
3. Staff prepares items according to specifications
4. Expo station coordinates final quality check

GUI Description: Kitchen stations are provided with printed tickets containing clear formatting, high-visibility timestamps, and highlighted special instructions. Tickets are color-coded by priority sequence and include all preparation data needed.

Requirements:

1. Expo Station will accept each order for final quality coordination

2. Grill Station receives tickets for sautéed and grilled food
3. Sauté Station receives hot appetizers, pasta, and sides
4. Salad Station receives salad orders and all desserts
5. System receives timestamp, server name, table number on every ticket

3.4 Task 4: Payment Processing

User Task: Server processes payments from customers with automatic tip calculation.

Task Workflow:

1. Server selects finished order for checkout
2. System computes total including tax
3. Server chooses payment method (cash or credit)
4. System processes payment and adjusts tip amount
5. System generates customer receipt

GUI Description: Payment panel displays clear order summary with detailed pricing and tax calculation. Buttons to choose payment method are large along with fields to accept tip input and change computation.

Requirements:

1. System must be connected to PCI-compliant payment processing
2. System must calculate credit transactions tips automatically
3. System must provide manual tip input for cash transactions
4. System to generate detailed end-of-shift tip reports

3.5 Task 5: Employee Time Tracking

User Task: Employees clock in and out with automatic hour calculation.

Task Workflow:

1. Employee enters unique ID at main computer
2. System verifies credentials and records timestamp
3. For clock-out: System calculates total hours worked
4. System prints time summary slip for clock-out only

GUI Description: Time clock interface has large-button design with numeric ID entry keypad. System displays the date, time, and employee name upon successful authentication.

Requirements:

1. System will verify employees against unique ID codes
2. System will track precise clock-in and clock-out times
3. System will prevent duplicate clock-in or clock-out entries
4. System will maintain employee time history for payroll linking

4 Non-Functional Requirements

Primary responsibility: Johan Ejiasi

4.1 Reliability

The system should operate 24/7, for any time the restaurant is open. The system shall function as needed, if there is an available reliable internet connection. Other specifications:

- System uptime will be 99.5% of running time
- System will recover automatically from network loss in 30 seconds
- Transactional data shall be backed up in real-time
- System will maintain data integrity on power failure

4.2 Robustness

The application should display errors if there is no internet connection. The system will gracefully recover from error conditions:

- System will indicate clear-cut error messages for connectivity issues
- System will maintain order entry in case of individual printer failures
- System will prevent data corruption on unexpected shutdowns
- System will maintain inventory integrity on multiple accesses

4.3 Performance

The following performance specifications are made:

- System clock-in/out will be under 5 seconds
- System will support 30 users concurrently
- Order submission will be performed in under 3 seconds
- Printing tickets in the kitchen will occur within 10 seconds
- Payment processing will occur in 15 seconds

4.4 Maintainability

Periodically, the program will be tested to maintain connectivity with the internet and updated whenever the link is modified. Other requirements:

- System will enable diagnostic reports automatically
- Software updates will be deployable during off-peak hours
- System will maintain complete logs for debugging
- Database maintenance will occur automatically

4.5 Security

The app must only allow users in the manager category to adjust/change orders already entered via the system and add stock items. Other security needs:

- System must authenticate employee ID for all access
- Payment data must be encrypted according to PCI compliance standards
- System must audit log for all user activity
- System must automatically log out idle users after 10 minutes

4.6 Usability

The application should be intuitive and user friendly. Special requirements:

- New users to complete basic training within 15 minutes
- System interface legible in restaurant lighting
- All key operations within 3 screen navigations
- System to provide audio feedback for operations
- Error messages to provide corrective action

5 Design and Implementation Constraints

Primary responsibility: Christopher De Sousa

5.1 Standards Compliance

- **PCI DSS**: Payment processing must comply with Payment Card Industry Data Security Standards
- **ADA Compliance**: Interfaces must be accessible and compliance with appropriate contrast and text sizes
- **Food Service Standards**: System must support typical restaurant operational processes

5.2 Development Constraints

- User must have computer and printers with internet access to use the software
- System must run on Windows 10 or greater operating systems
- Minimum of 8GB RAM required for multi-user operation
- Receipt printers for all the kitchen stations with network connectivity
- Reliable internet connection with a minimum of 10 Mbps for payment processing via Stripe or other third-party vendor
- Development limited to 4 individuals within academic semester time period
- Technology stack: Vue.js frontend, Node.js with Express.js backend, MongoDB database
- Vue.js chosen for its lightweight nature and ease of learning, making it ideal for small to medium-scale projects
- Payment processing integration with Stripe or similar PCI-compliant third-party vendors

6 References and Sources of Information

Compiled by: All team members

1. IEEE 830-1998 Recommended Practice for Software Requirements Specifications
2. "Writing Quality Requirements" by Karl S. Weigers
3. Introduction to the Team Software Process, Watts S. Humphrey, Addison Wesley, 2000, page 113
4. PCI Data Security Standard Documentation
5. Restaurant industry best practices for kitchen management systems
6. MySQL Documentation for database design
7. Node.js and React.js Documentation for web development