****

**CentRes**

**Software Requirements Specification**

**Version #1**

**Team Number: 10**

**Project Manager: Katherine Landsman**

**Mentor: Susan Rizzo**

**Team Members: Carson Pribble, Chase Jamison, David Utshudiema, Mohamed Ibensilalen, Sergio Mainville**

**Revisions**

| **Version** | **Primary**  **Author(s)** | **Description of Version** | **Date Completed** |
| --- | --- | --- | --- |
| V.1.0 | Carson, Chase, David, Mohamed, Sergio,  Katherine | First draft of Software Requirements | 10/13/22 |
|  |  |  |  |

**Review History**

| **Reviewer** | **Version Reviewed** | **Date** |
| --- | --- | --- |
|  |  |  |
|  |  |  |

**Table of Contents (Version 1)**

1. Introduction
   1. Project Objectives
   2. Project Scope
   3. Project Overview
2. General Description
   1. Project Features / Functions
   2. User Stories
   3. Use Cases
3. Team Collaboration and Documentation Tools
4. Project Management Plan
5. Business Requirements
6. User Requirements
7. Functional Requirements
8. Non-Functional Requirements
9. Implementation (Performance) Requirements (Optional)
10. **Introduction**
    1. Project Objectives:

Allow for employees to easily interact with customers’ orders via an app and reduce miscommunication. Provide centralization of all server tables allowing servers to better manage and prioritize orders. App will help cut down on customer wait time by transmitting information directly to the kitchen.

* 1. Project Scope:
* App will not automatically rotate servers, this will be manually achieved by the hostess
* App will provide order capability
* App will provide order management including creating, deleting, adding, and removing items from an order.
* App will provide server management
* Managers will have upgraded admin privileges within the app to edit customer orders
* Menu items will be stored in a database
* App will keep track of volume of items sold each day for tracking purposes
* App will not have timesheet functionality. (Keeping track of hours worked)
* App will not keep track of employee’s weekly schedules.
* App will not have the functionality for communication between users (servers, managers, etc.)
  1. Project Overview:

Servers will have an easy-to-read display that keeps track of things such as menu items, prices, order time, and order status. They can also view their current tables and see the current bill/tab for each table, see the average order time, and display information associated with a table. Line cooks/Chefs will see the current orders placed and the time they were placed. They can notify the server when an order is ready. The application would display the menu and allow the ability to add or edit items in an order. When viewing a table, the user can view or distribute an itemized bill to the customer.

1. **General Description**
   1. Project Features / Functions
   2. User Stories
      1. As a server, I want to only see available tables so I can reserve it for the customer.
      2. As a server, I want to place an order in my device and send it directly to the kitchen.
      3. As a line cook, I want to receive notification for every new order and can mark it ready after preparation.
      4. As a server, I want to receive notification from the kitchen on every order that is ready so I can mark it completed after serving it to the right table.
   3. Use Case
      1. **Name:** Server

**Actor:** Server

**Basic Use Case Description:**

* Server reserve available table for the customer.
* Server sends order to the kitchen.
* Server got notification off the order and marked it completed after serving the order to the right table.
  + 1. **Name:** Line Cook

**Actor:** Line Cook

**Basic Use Case Description:**

* Line cooks receive placed order with time stamps
* Line cooks check off individual items within order
* Line cooks mark the order ready after preparation.
  + 1. **Name**: Host

**Actor:** Host

Basic Use Case Description:

* Host can manage table availability
* Host can distribute tables to servers
  + 1. **Name:** Manager

**Actor:** Manager

**Basic Use Case Description:**

* Manager can edit orders
* Manager can alter bill
* Manager has an overview of wait times
* Manager can view time stamps on orders

1. **Team Collaboration and Documentation Tools**

* GitHub
* GitHub Projects
* Microsoft Teams
* Practera
* Email

1. **Project Management Plan**

* Agile Principles
* Scrum Methodology

1. **Business Requirements**

| Requirement ID | Requirement Description | MOSCOW |
| --- | --- | --- |
| BR1 | System must be available on both IOS and Android | M |
| BR2 | Users leave a feedback | C |
| BR3 | Weekly efficiency reports are generated |  |
|  |  |  |

1. **User Requirements**

| Requirement ID | Requirement Description | MOSCOW |
| --- | --- | --- |
| UR1 | e.g. Screen A accepts production information, including Lot, Product Number, and Date. | M |
| UR2 | The system will provide the user a guideline | S |
| UR3 |  |  |

1. **Functional Requirements**

| Requirement ID | Requirement Description | MOSCOW |
| --- | --- | --- |
| FR1 | e.g. Send email when a new customer signs up. | M |
| FR2 | Send alteration to user(server) if order is ready | M |
| FR3 |  |  |

1. **Non-Functional Requirements**

| Requirement ID | Requirement Description | MOSCOW |
| --- | --- | --- |
| NFR1 | e.g. User must have a unique username when registering for an account | M |
| NFR2 | The system information can only be accessed by the user and the restaurant management | M |
| NFR3 |  |  |

1. **Implementation (Performance) Requirements (Optional)**

| Requirement ID | Requirement Description | MOSCOW |
| --- | --- | --- |
| IR1 | Different database for employee | S |
| IR2 |  |  |
| IR3 |  |  |