Project Name : Restaurant system

Table of Participants/ Workload distribution

Table of Content

1. Preface:

Welcome to the documentation of the Restaurant Management Software (RMS) project. In this document, we delve into the intricacies of our innovative software solution tailored specifically for the restaurant industry.

The Restaurant Management Software project aims to address the challenges faced by restaurants, cafes, and eateries in managing their operations efficiently. By leveraging modern technology and automation, our software seeks to streamline various aspects of restaurant management, including order processing, inventory control, billing, and reporting.

Throughout this documentation, we will provide insights into the features, functionalities, and technical aspects of our RMS project. Whether you're a restaurant owner seeking to enhance operational efficiency or a developer interested in understanding the architecture of restaurant management software, this document will serve as a comprehensive guide.

Join us on this journey as we explore the transformative potential of the Restaurant Management Software project.

2-Introduction: Restaurant Management Software Project

In the digital age, the hospitality industry is undergoing a transformation with the integration of sophisticated software solutions. The Restaurant Management Software (RMS) project endeavors to develop a comprehensive digital platform tailored to the unique needs of restaurants, cafes, and eateries. This software aims to streamline operations, enhance efficiency, and elevate the overall dining experience for both customers and restaurant staff.

3-Glossary

3.1 Order Management: The RMS will feature an intuitive interface for seamless order processing, modification, and tracking. It will support various order types including dine-in, takeout, and delivery, facilitating smooth communication between front-of-house and kitchen staff.

3.2 Inventory Control: Effective inventory management is crucial for controlling costs and minimizing wastage. The software will provide tools for real-time inventory tracking, automatic stock alerts, and supplier management, ensuring optimal stock levels and reducing operational inefficiencies.

3.3 Billing and Payment : Simplifying the payment process is essential for customer satisfaction. The RMS will include a secure payment gateway for processing transactions, supporting multiple payment methods such as cash, credit/debit cards. It will also generate detailed invoices and receipts for accurate financial records.

.

3.4 Analytics and Reporting: Data-driven insights are invaluable for informed decision-making. The RMS will provide comprehensive reporting tools for analyzing sales trends, monitoring key performance indicators (KPIs), and identifying opportunities for growth and optimization.

The RMS project will leverage modern technologies and frameworks to ensure scalability, reliability, and security. This may include:

User Interface: C#

Program: c#

Conclusion:

In conclusion, the Restaurant Management Software project aims to revolutionize the way restaurants operate by offering a feature-rich digital platform tailored to their specific needs. By embracing technology and automation, the RMS will empower restaurants to enhance efficiency, improve customer satisfaction, and drive business growth in an increasingly competitive market landscape.

4. **User Requirements**

The User Requirements section outlines the needs and expectations of different user roles.

4.1-Manager Requirements

1-Role: The manager oversees the entire restaurant operation.

2-Responsibilities:

3-Staff management

4-Inventory control

5-inancial management

6-Customer service

7-Table reservations

8-Reporting

4.2-Cashier Requirements

1-Role: Cashiers handle transactions and interact with customers.

2-Responsibilities:

3-Order processing

4-Payment handling

5-Receipt generation

6-Cash management

7-Customer service

5. System Requirements

The System Requirements section details the technical specifications and features of the restaurant management system.

Features

User Authentication:

Secure login for managers and cashiers.

Order Management:

Taking and processing orders.

Inventory Tracking:

Monitoring stock levels.

Reporting:

Generating sales reports, expense summaries, and performance metrics.

Reservation System:

Managing table reservations.

Menu Management:

Updating menu items and prices.

Payment Integration:

Integrating with payment gateways.

Notifications:

Alerts for reservations, order status, etc

5-System requirements

Functional Requirements

User Authentication and Authorization:

Description: Users (managers, cashiers, storekeepers) must log in with appropriate credentials.

1- Features:

2- Secure login

3- Role-based access control (RBAC)

4- Order Management:

5- Description: Efficiently process customer orders.

Features:

1- Order creation

2- Order modification (add, edit, delete)

3- Order status tracking

4- Inventory Management:

Menu Management:

Description: Maintain an up-to-date menu.

Features:

Add, edit, or remove menu items

Set prices and descriptions

Categorize items (appetizers, entrees, desserts)

Table Reservations:

Description: Manage table bookings.

Features:

Reservation creation

Availability check

Reporting and Analytics:

Description: Generate reports for decision-making.

Features:

Sales reports

Expense summaries

Performance metrics

Payment Integration:

Description: Handle various payment methods.

Features:

Cash, credit card

Integration with payment gateways

* **Non-Functional Requirements***

Performance:

The system should handle concurrent users efficiently during peak hours.

Response time for order processing should be minimal.

Security:

User data (credentials, personal information) must be encrypted and stored securely.

Access controls should prevent unauthorized actions.

Reliability:

The system should be available …

Backup and recovery mechanisms should be in place.

Usability:

The user interface should be intuitive and easy to navigate.

Training for staff should be minimal.

Scalability:

The system should accommodate future growth (new branches, increased user load).

6. **apandcies**

A- Source Code: The complete C++ code for a restaurant management system.

B- Data Structures

Provide detailed explanations of the data structures used in your system, including their purpose and relationships. For example:

c- user Structure:

Fields: username, password, role

Purpose: Store user information for authentication

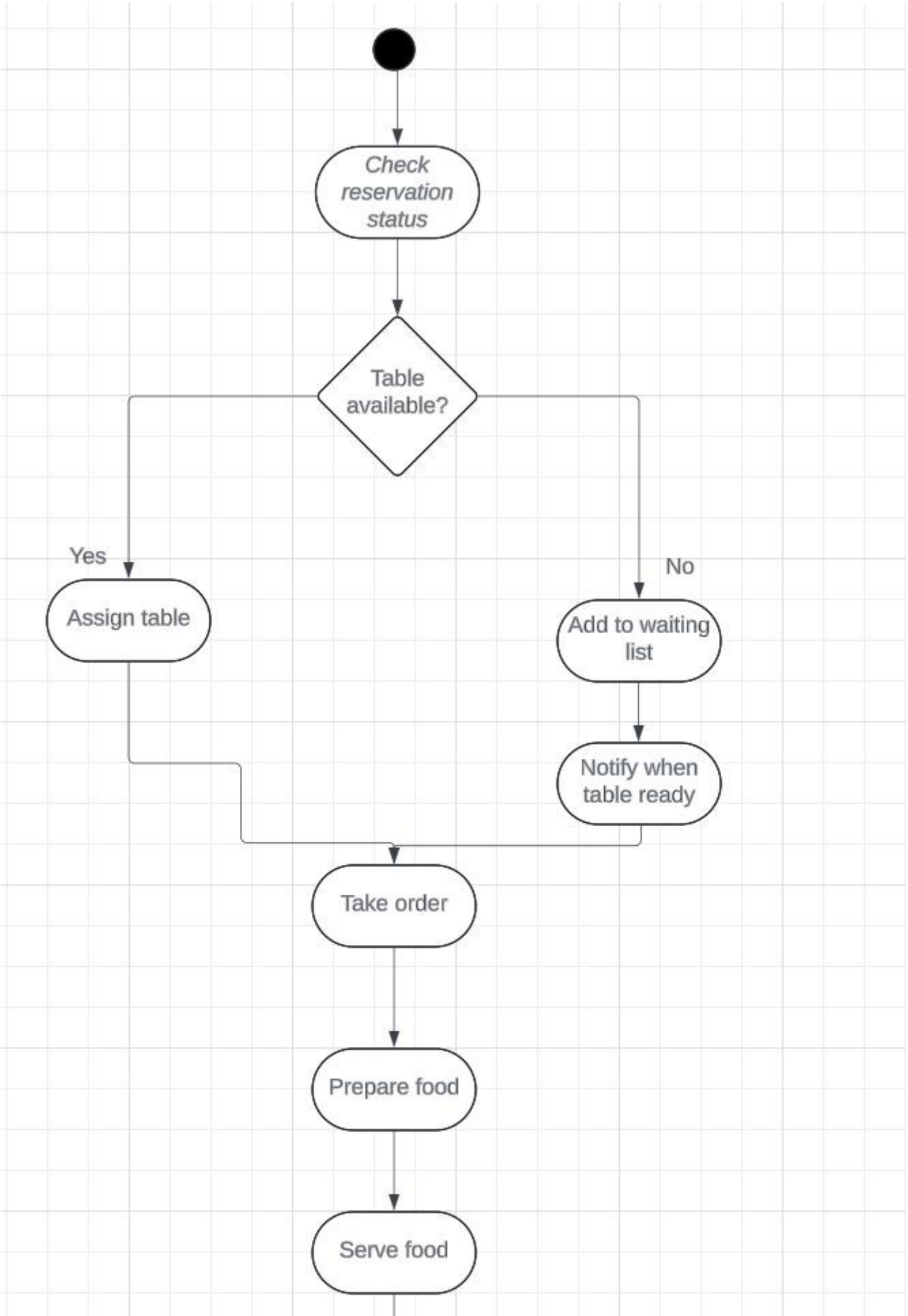
d-Menu Item Structure:

Fields: id, name, price

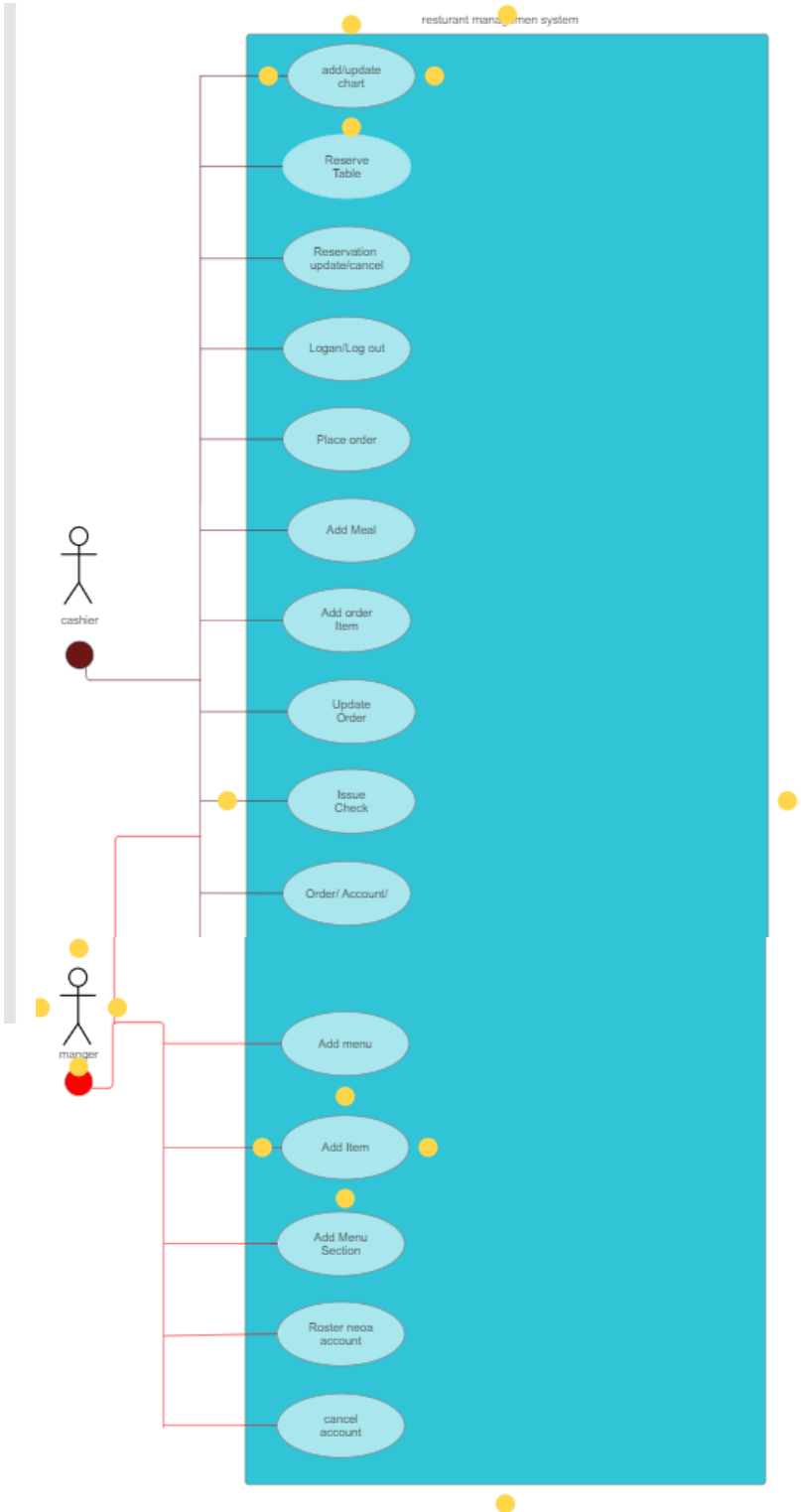
Purpose: Represent menu items and their details

This document aims to provide a clear understanding of the program's functionality, requirements, and relevant terms, making it easier for users to utilize the restaurant system

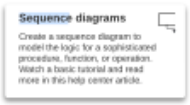
* Make the activity diagram for your project

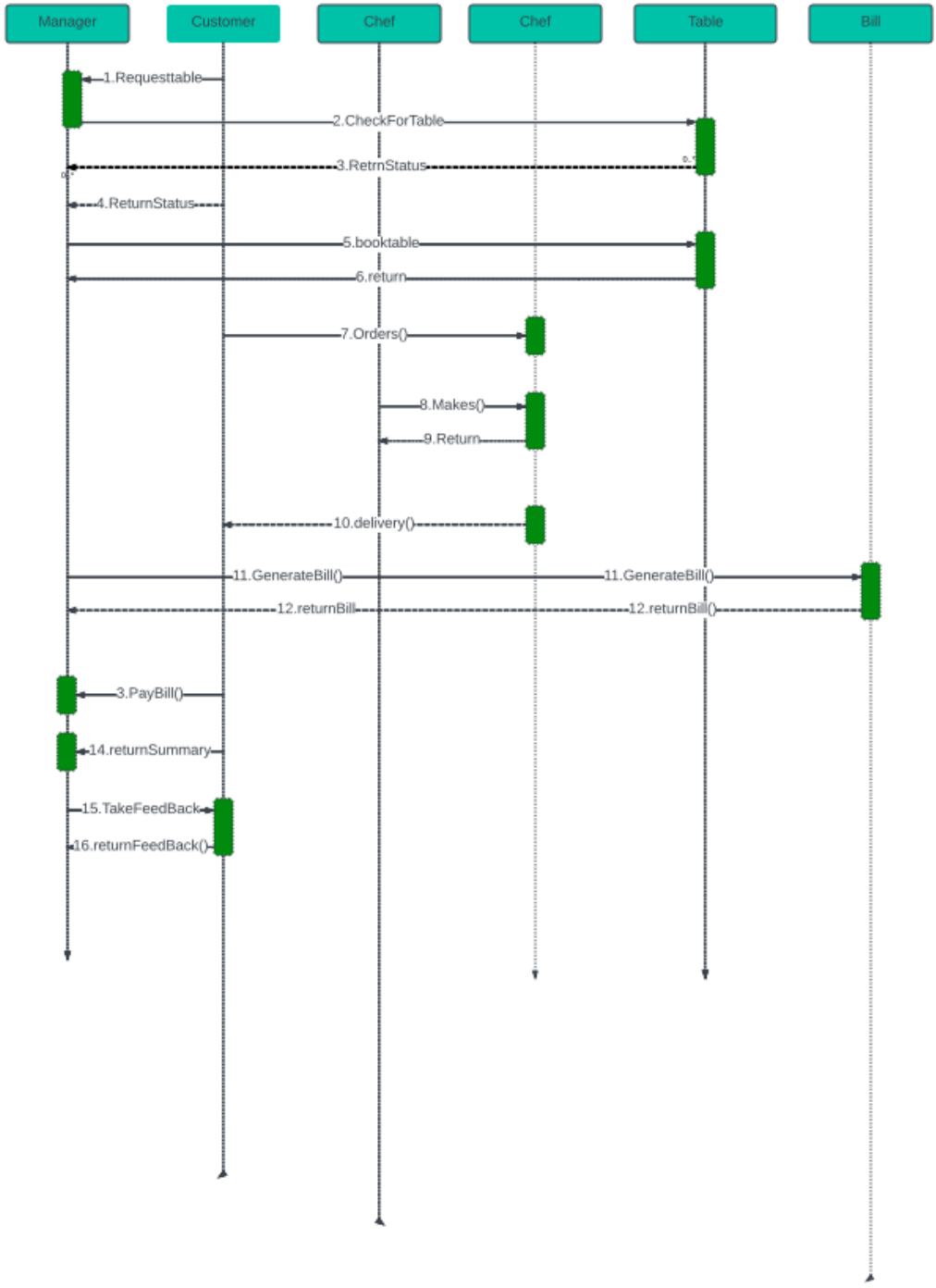

Check reservation status
Table available?
Yes
No
Assign table
Add to waiting list
Notify when table ready
Take order
Prepare food
Serve food

* **Make the Use case diagram for your project**

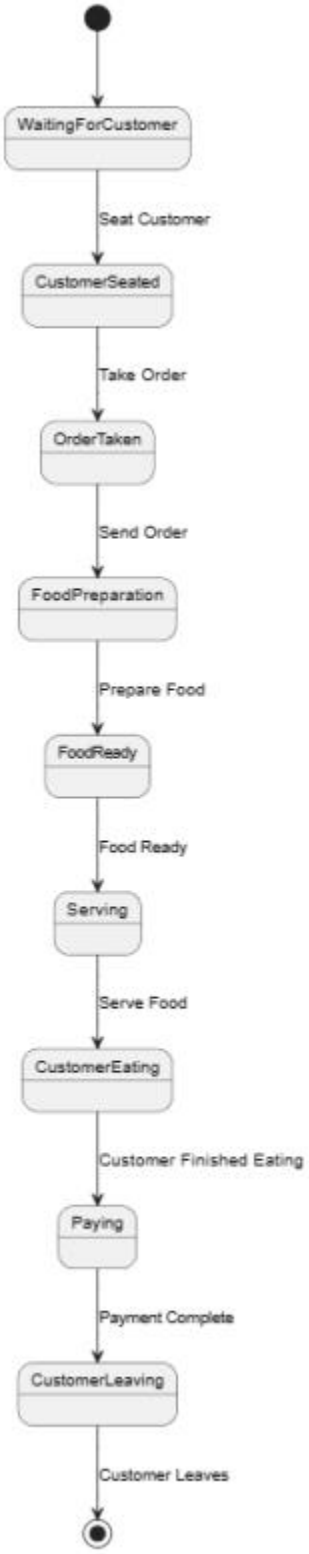


* Make the sequence Diagram for your project





state Diagram



Class Diagram