Loyola ICAM College of Engineering and Technology

SOFTWARE REQUIREMENTS SPECIFICATION

For

Food Delivery System

Prepared by:

Bhavesh G Sadhwani

Dhanush Kabilan D

Luxon C L

Academic Year: *2023*

Department of Information Technology

1. Introduction

1.1 Purpose

1.2 The main objective of this document is to outline the requirements for the Food Delivery System project. This document provides a detailed description of both functional and non-functional requirements as per the client's needs. The purpose of this project is to create a user-friendly platform for ordering and delivering food. It aims to streamline the food ordering and delivery process using computers and to generate various reports related to food delivery. The document also describes the hardware and software interface requirements using diagrams.

1.3 Document Conventions

- Entire document should be justified.
- Convention for Main title
  - Font face: Times New Roman
  - Font style: Bold
  - Font Size: 14
- Convention for Sub title
  - Font face: Times New Roman
  - Font style: Bold
  - Font Size: 12
- Convention for body
  - Font face: Times New Roman
  - Font Size: 12

1.4 Scope of Development Project

The Food Delivery System project aims to create an online platform for food ordering and delivery. This system will allow users to browse food options, place orders, and have food delivered to their location. It will be accessible to both customers and restaurant owners. The project can be implemented in various situations and can be customized to meet specific requirements. The software will be developed using a technology stack suitable for web-based applications, ensuring performance, cross-platform compatibility, and ease of use.

1.5 Definitions, Acronyms and Abbreviations

UI: User Interface
API: Application Programming Interface
GPS: Global Positioning System
SQL: Structured Query Language
ER: Entity Relationship
UML: Unified Modeling Language
SRS: Software Requirement Specification

1.6 References

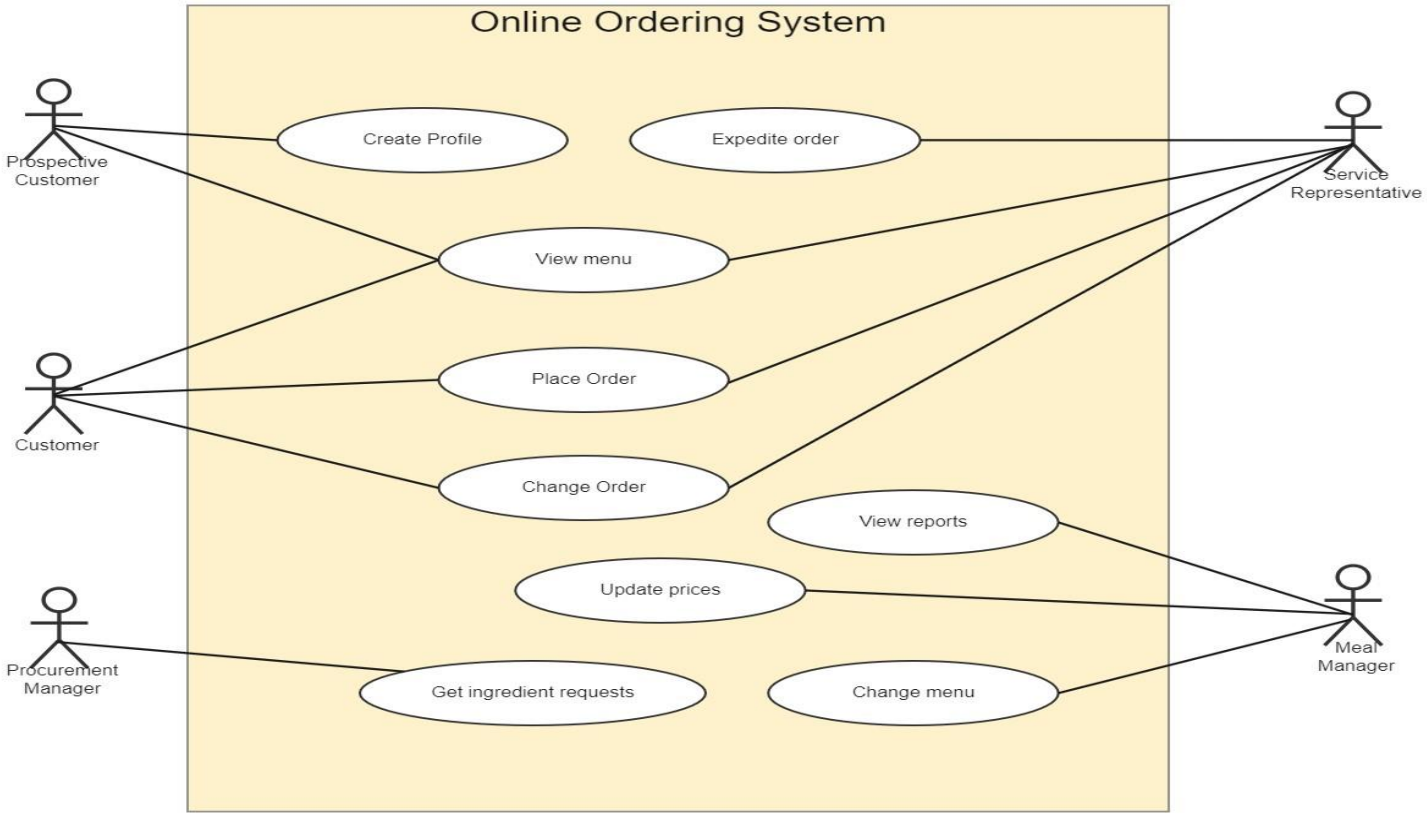
Books

- "Software Requirements and Specifications: A Lexicon of Practice, Principles, and Prejudices" by Michael Jackson
- "Software Requirements" (Microsoft) Second Edition by Karl E. Wiegers
- "Software Engineering: A Practitioner's Approach" Fifth Edition by Roger S. Pressman

2. Overall Descriptions

2.1 Product Perspective

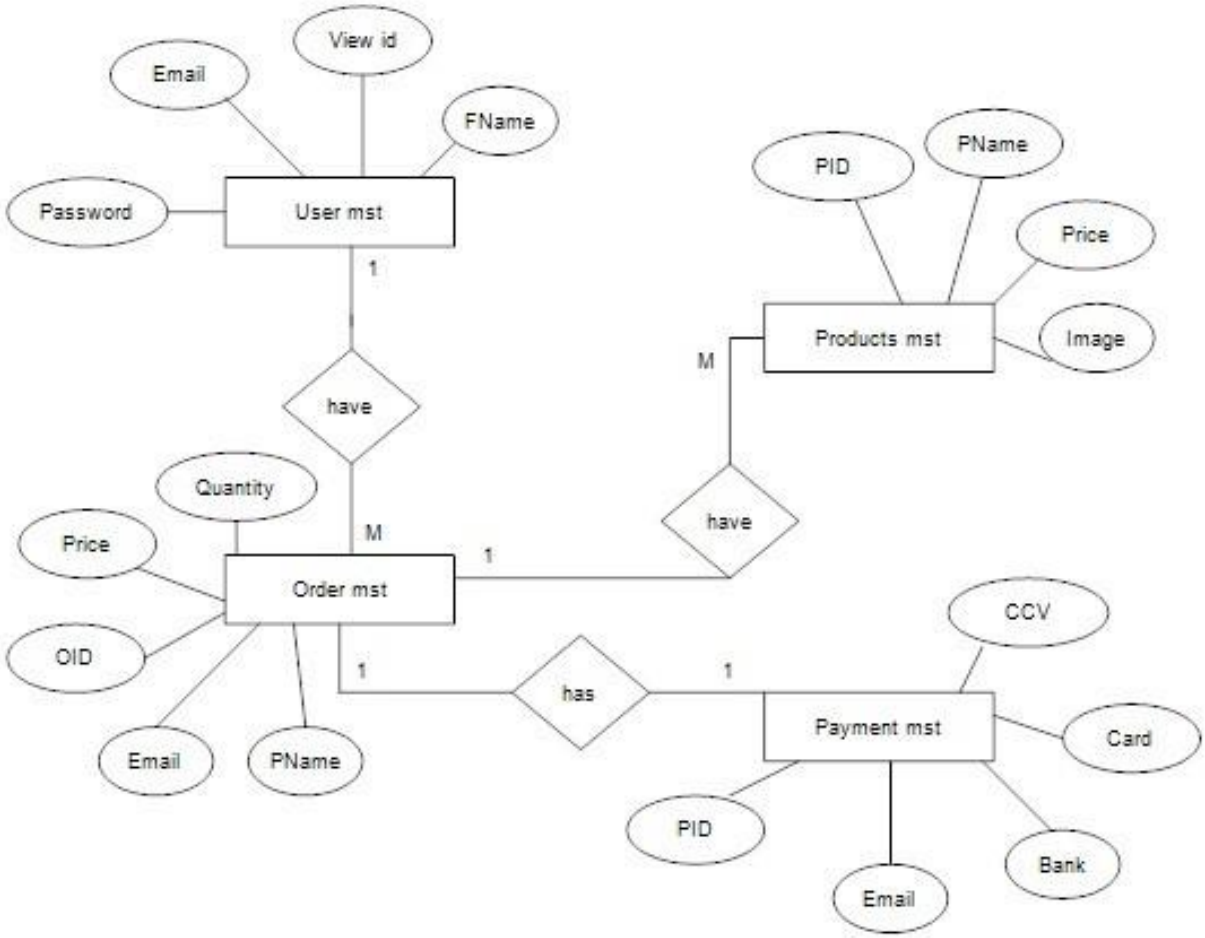
Use Case Diagram of Food Delivery System



The Food Delivery System is a web-based platform that connects customers with restaurants for food orders and delivery. The system includes features such as food search, order placement, order tracking, and payment processing. Users can be either customers or restaurant owners. The system facilitates search for food options, placing and managing orders, and coordinating food delivery.

2.2 Product Function

Entity Relationship Diagram of Food Delivery System



The Food Delivery System provides real-time information about available food options and facilitates food ordering, delivery, and payment. It reduces manual processes and streamlines the food delivery process. The system is designed to be user-friendly and efficient for both customers and restaurant owners.

2.3 User Classes and Characteristics

The system serves two main user classes:

Customers

Can browse food options.

Can place orders.

Can track orders.

Can make payments.

Can provide feedback and reviews.

Restaurant Owners

Can list their food menu.

Can accept and manage orders.

Can update menu items and prices.

Can view order history and sales data.

2.4 Operating Environment

The product operates as a web application and is accessible from various web browsers such as Chrome, Firefox, and Internet Explorer. It requires an internet connection for both customers and restaurant owners. The hardware requirements include a computer or mobile device with internet connectivity.

Assumptions and Dependencies

Assumptions:

The system will be error-free.

The system will provide a user-friendly experience.

All user data and food information will be stored in a database accessible via the website.

The system should be capable of handling a large number of users and orders.

Dependencies:

The system relies on specific hardware and software components to operate.

The project development and execution depend on the defined requirements and specifications.

Users (admins) should have a good understanding of the system.

The system should maintain general records and backup data.

Any updates to the restaurant's menu should be accurately recorded.

2.5 Requirement

Software Configuration:
The software is developed using technology stack, including frontend technologies like HTML, CSS, and JavaScript. The backend is supported by a server-side scripting language such as Python or Node.js. A relational database management system (e.g., MySQL) is used for data storage.

Hardware Configuration:
Processor: Dual-core CPU
Hard Disk: 40GB or more
RAM: 256 MB or more

2.6 Data Requirement

The system will have a database that stores information about food items, restaurants, customers, orders, and payments. Inputs include user queries and orders, and outputs include order details, delivery information, and payment confirmation.

3. External Interface Requirement

3.1 GUI

The software provides a user-friendly graphical interface for customers to browse food options, place orders, and track deliveries. The restaurant owners can use a separate interface to manage their menu and orders.

User Interface
User registration and login.
Browsing food options with images and descriptions.
Placing orders and making payments.
Tracking order status and delivery.
Providing feedback and reviews.
Restaurant Owner Interface
Listing food menu items with descriptions and prices.
Managing incoming orders.
Updating menu items and prices.
Viewing order history and sales data.

3.2 API

The system may require integration with third-party services for payment processing and mapping for delivery tracking. APIs for payment gateways and GPS services may be needed.

4. System Features

User authentication for secure accounts.

Real-time order tracking.

Payment processing and confirmation.

Restaurant menu management.

User feedback and reviews.

Food search and filtering options.

Delivery address validation and mapping.

5. Other Non-functional Requirements

5.1 Performance Requirement

The system should be fast and responsive.

Handle errors gracefully to prevent data loss or downtime.

Scalable to accommodate a large number of users and orders.

5.2 Safety Requirement

Regular database backups to prevent data loss.

UPS or inverter support for power supply failures.

5.3 Security Requirement

Secure user authentication.

Access control for different user types.

Encrypted communication.

Protect user data and payment information.

Regular system updates and security patches.

5.4 Requirement attributes

Regular database backups to prevent data loss.

UPS or inverter support for power supply failures.

5.5 Business Rules

The system enforces business policies for payment, refunds, and delivery.

Users must abide by the system's terms of service.

Rules and protocols for order placement, cancellation, and refunds are in place.

5.6 User Requirement

Customers and restaurant owners should have access to user manuals and online help for system usage.

Administrators should be knowledgeable in system maintenance and troubleshooting.

6. Other Requirements

6.1 Data and Category Requirement

The system should categorize food items and display relevant data based on categories.

User access rights are determined by their category (customer or restaurant owner).

The system should code categories and related data in a standardized format.

6.2 Appendix

A: Admin, Abbreviation, Acronym, Assumptions
B: Business rules
C: Class, Client, Conventions
D: Data requirement, Dependencies
G: GUI
K: Key
L: Library, Librarian
M: Member
N: Non-functional Requirement
O: Operating environment
P: Performance, Perspective, Purpose
R: Requirement, Requirement attributes
S: Safety, Scope, Security, System features
U: User, User class and characteristics, User requirement

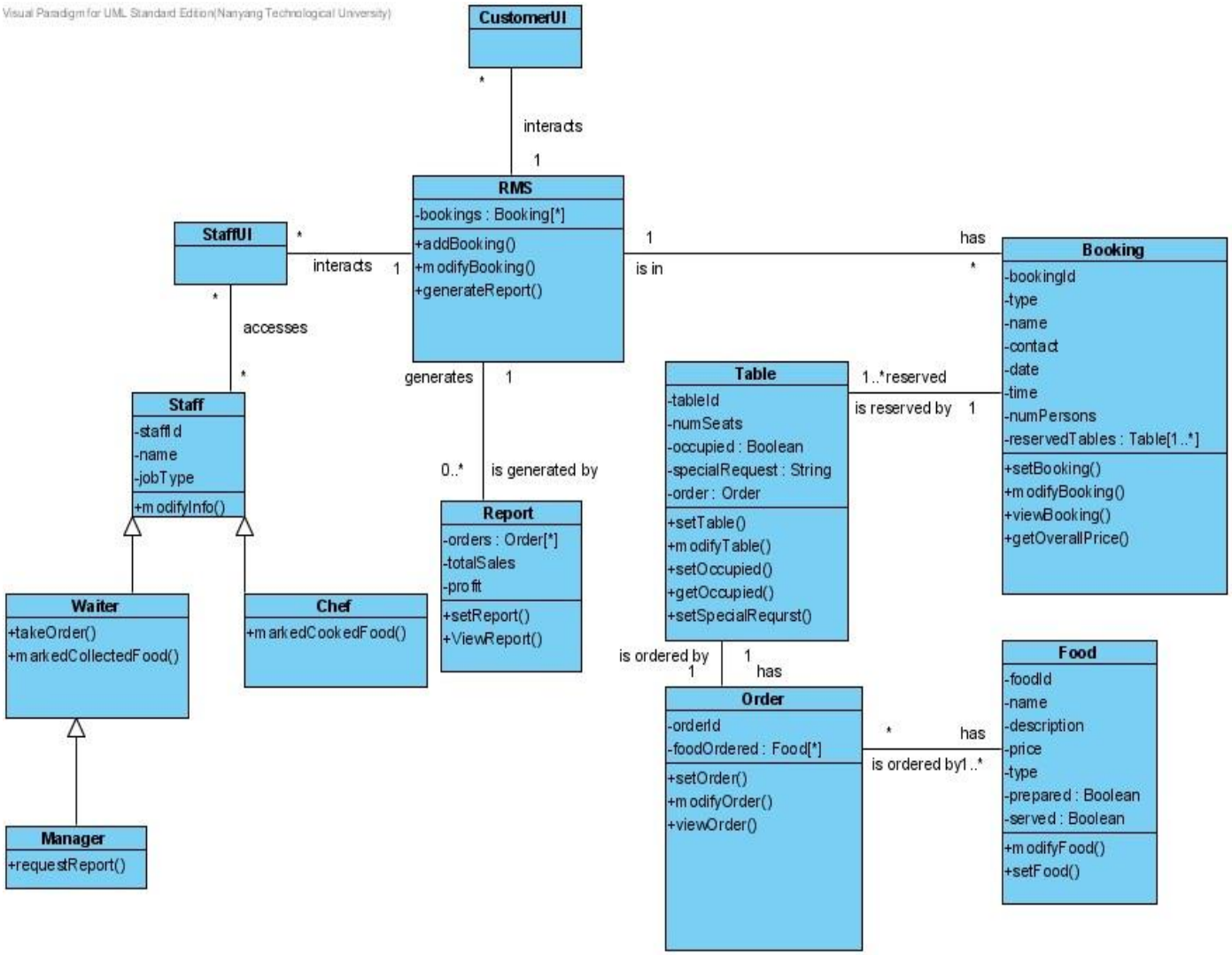
6.3 Glossary

UI: User Interface
API: Application Programming Interface
GPS: Global Positioning System
SQL: Structured Query Language
ER: Entity Relationship
UML: Unified Modeling Language
SRS: Software Requirement Specification

7. Other Requirements

7.1 Class Diagram for Food Delivery System

A class is an abstract, user-defined description of a type of data. It identifies the attributes of the data and the operations that can be performed on instances (i.e. objects) of the data. A class of data has a name, a set of attributes that describes its characteristics, and a set of operations that can be performed on the objects of that class. The classes' structure and their relationships to each other frozen in time represent the static model. In this project there are certain main classes which are related to other classes required for their working. There are different kinds of relationships between the classes as shown in the diagram like normal association, aggregation, and generalization. The relationships are depicted using a role name and multiplicities. Here 'Librarian', 'Member' and 'Books' are the most important classes which are related to other classes



THANK YOU