Software Requirements Specification
for

Restaurant Management System

SUBMITTED IN
PARTIAL FULFILLMENT OF

PG - DIPLOMA IN ADVANCED COMPUTING (PG-DAC)

BY

Tejas Gopal Badgujar
Suruchi Sudhir Sonone
Shivam Gaur
Saurabh Anand Ladi
Ghanshyam Sahebrao Mali
Latika Kashinath Mitkari

AT

CDAC Kharghar, Mumbai
March-2023

INDEX

1.	**INTRODUCTION**	4
	1.1 Introduction	
2.	**PRODUCT OVERVIEW AND SUMMARY**	5
	2.1 Purpose	
	2.2 Scope	
	2.3 User Classes and Characteristics	
	2.4 Design and Implementation Constraints	
3.	**REQUIREMENTS**	6
	3.1 Functional Requirements	
	3.1.1 Use case for Administrator.	
	3.1.2 Use case for Customer.	
	3.1.3 Use case for Manager.	
	3.1.4 Use case for Waiter	
	3.2 Non - Functional Requirements	11
	3.2.1 Entity Relationship Diagram	
	3.2.2 Performance Requirement	
	3.2.3 Safety Requirements	
4.	**PROJECT DESIGN**	
	4.1 Class diagram	13
	4.2 Schema Diagram	14
5.	**REFERENCES**	15

LIST OF FIGURES

Sr. No	Figure Title	Page
1	Use case for owner	6
2	Use case for customer	7
3	Use care for manager	8
4	Use case for waiter	9
5	E-R Diagram	10
6	Class diagram	13
7	Schema Diagram	14

1: INTRODUCTION

1.1 Introduction

The main objective of this project is to develop a client/server model which deals with “Restaurant Management System”. This system has two parts, one part is for management side and other part is for customers.

After successful login the customer side allows customer to reserve table, view menu according to category and price and add food items to their cart and also check previous orders.

At the management side the staff is allowed to edit information regarding menu list, food price, assign chef, take ratings and reviews and also maintain information regarding placed orders.

2: PRODUCT OVERVIEW AND SUMMARY

2.1 Purpose

The main purpose of this system is to manage restaurant in an easy and less time consuming way. It is created to avoid rush by letting customer reserve table in advance and also place orders.

2.2 Scope

The primary scope of the restaurant management system project is well depicted on the user case diagrams that are well showcased in this report. However, the central system functionalities of the restaurant management system comprise generating reports, inventories, employee records, and managing orders. The order management involves the creation and deleting of orders, removing and adding food staff from a request, and closing the orders. All the orders made should be stored in the system's database and connected to the significant project tasks, with testing and implementation that consume marginally more time than design.

2.3 User Classes and Characteristics

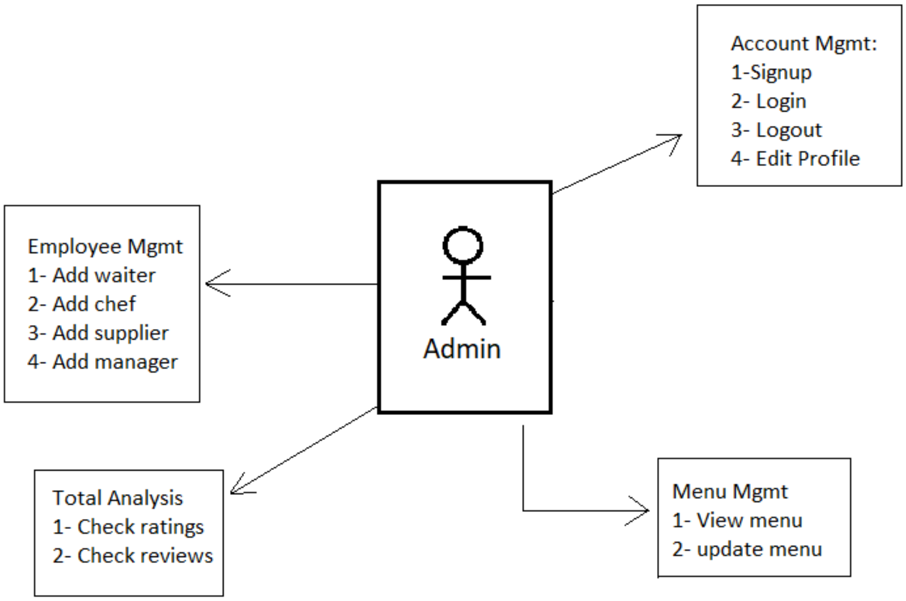
The Restaurant Management System has 6 actors and 1 co-operating system. The actors involved are: Owner, Manager, Customer, Waiter.

1- The Owner has access to full system and can manage employees, salaries, inventory and revenue management.
2- The Manager can manage orders, table reservations, orders and total analysis.
3- The customer can reserve table, view and order food items from menu, update orders, pay bill and give rating.
4- The Waiter can access table and order management.
5- The Supplier can access inventory and supply ingredients as per restaurant’s needs.

3: REQUIREMENTS

3.1 Functional Requirements

3.1.1- Use case for Owner



A- User Management

1. Customer
2. Manager
3. Staff

B- Menu Management

1. View menu
2. Update menu

C- Payment Management

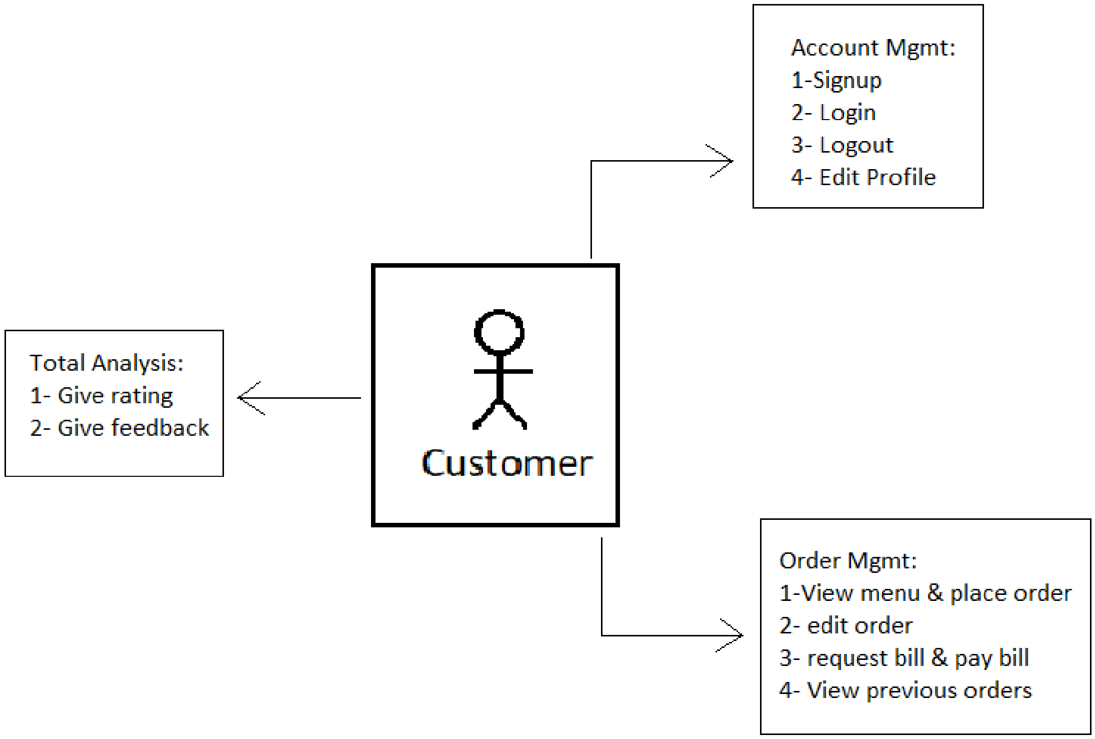
1. Manage salary(Waiter, chef, manager)
2. Inventory Payment

D- Edit Profiles

E- Total Analysis

1. Check ratings
2. Check reviews

3.1.2- Use case for Customer



A- Account Management

1- Customer can create account by signing up.
2- Customer can Login and also edit information as needed.
3- Customer can logout as per their need.

B- Order Management

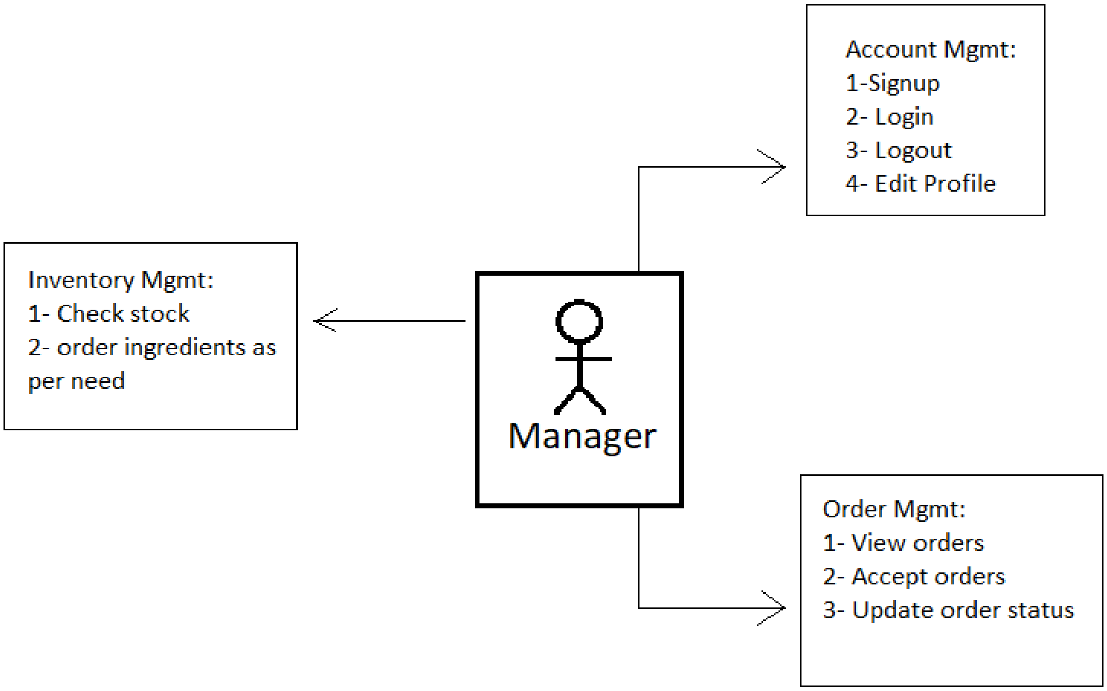
1- Customer can view menu.
2- Customer can place order according to category and price.
3- View previous orders.

C- Customer should pay the bill

D- Analysis

1- Customer can give rating
2- Customer can give feedback.

3.1.3- Use case for Manager



A- Account Management

4- Manager can create account by signing up.
5- Manager can Login and also edit information as needed.
6- Manager can logout as per their need.

B- Order Management

1. Check Table reservation
2. Check Orders
3. Assign chef and waiter for an occupied table.
4. Cancel order incase needed.

C- Inventory Management

1. Check Inventory and order ingredients accordingly.

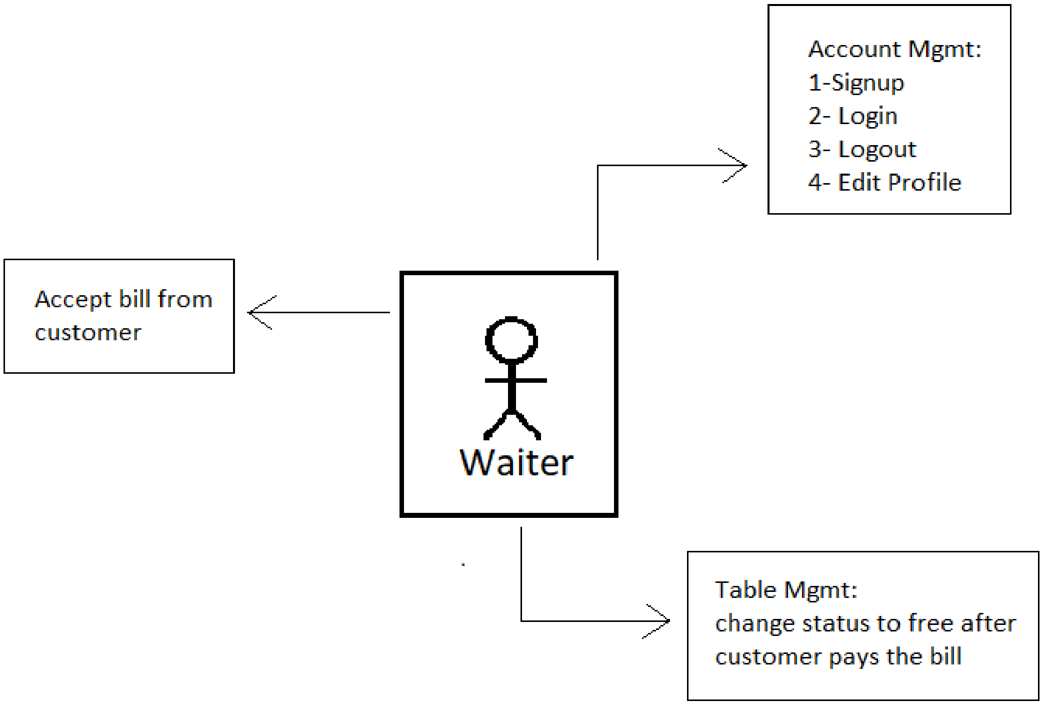
D- Bill Payments

1. Check bill and collect payment from customer

E- Analysis

1. Ask for rating
2. Take feedback.

3.1.4- Use case for Waiter


Account Mgmt:
1-Signup
2- Login
3- Logout
4- Edit Profile
Accept bill from customer
Waiter
Table Mgmt:
change status to free after customer pays the bill

A- Account Management

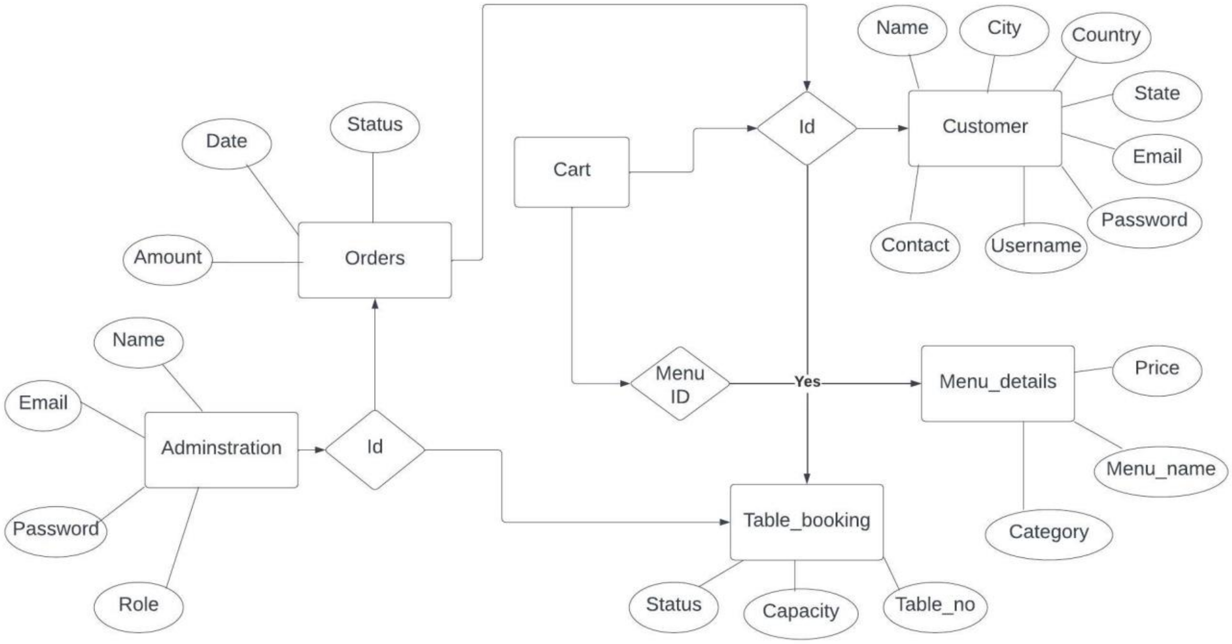
1- Waiter can create account by signing up.
2- Waiter can Login and also edit information as needed.
3- Waiter can logout as per their need.

B- Table Management

1- Waiter should clear previous table.
2- Waiter should take customer's order.
3- Change table status to free after bill paid by customer.

3.2: Non-functional Requirements

3.2.1- ER Diagram



3.2.2- Performance Requirement

1 Response Time

The application should respond to user interactions promptly. User commands and inputs should be processed within a maximum of 2 seconds to ensure a smooth user experience.

2 Search Performance

The search functionality should return results within a reasonable time frame, depending on the database size and complexity of search criteria. The search response time should be kept under 3 seconds.

3.2.3- Safety Requirements

1 Data Security

The application should ensure the security and confidentiality of user data, including login credentials and personal information. User passwords should be stored securely using encryption or hashing techniques.

2 Authorization

Access to certain features, such as job posting, should require additional authorization to prevent unauthorized usage.

3 Privacy

The application should enforce communication between owner and customer within the application to maintain user privacy and security.

3.2.4- Security Requirements

1 Data Transmission

User data should be transmitted securely over the internet using encryption protocols such as SSL/TLS to protect it from unauthorized access.

2 Vulnerability Prevention

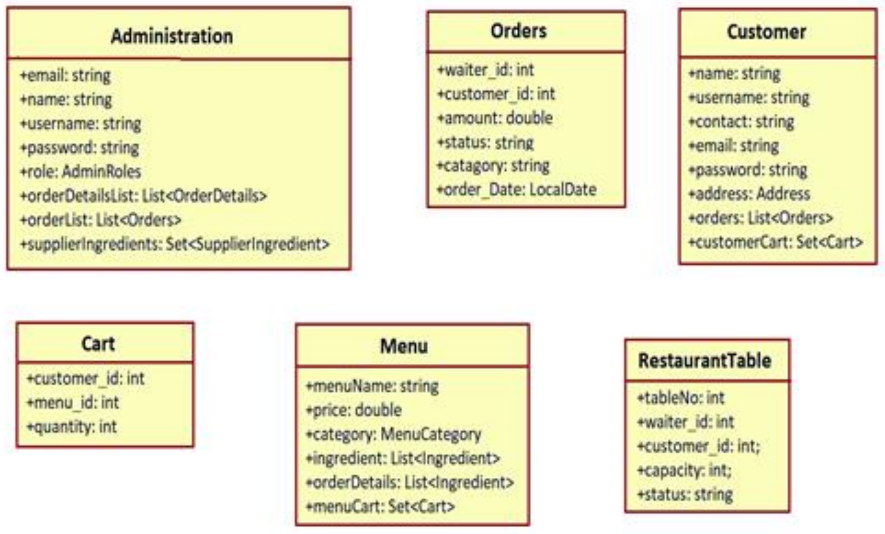
The application should implement measures to prevent common security vulnerabilities, such as SQL injection and cross-site scripting .

3 Access Control

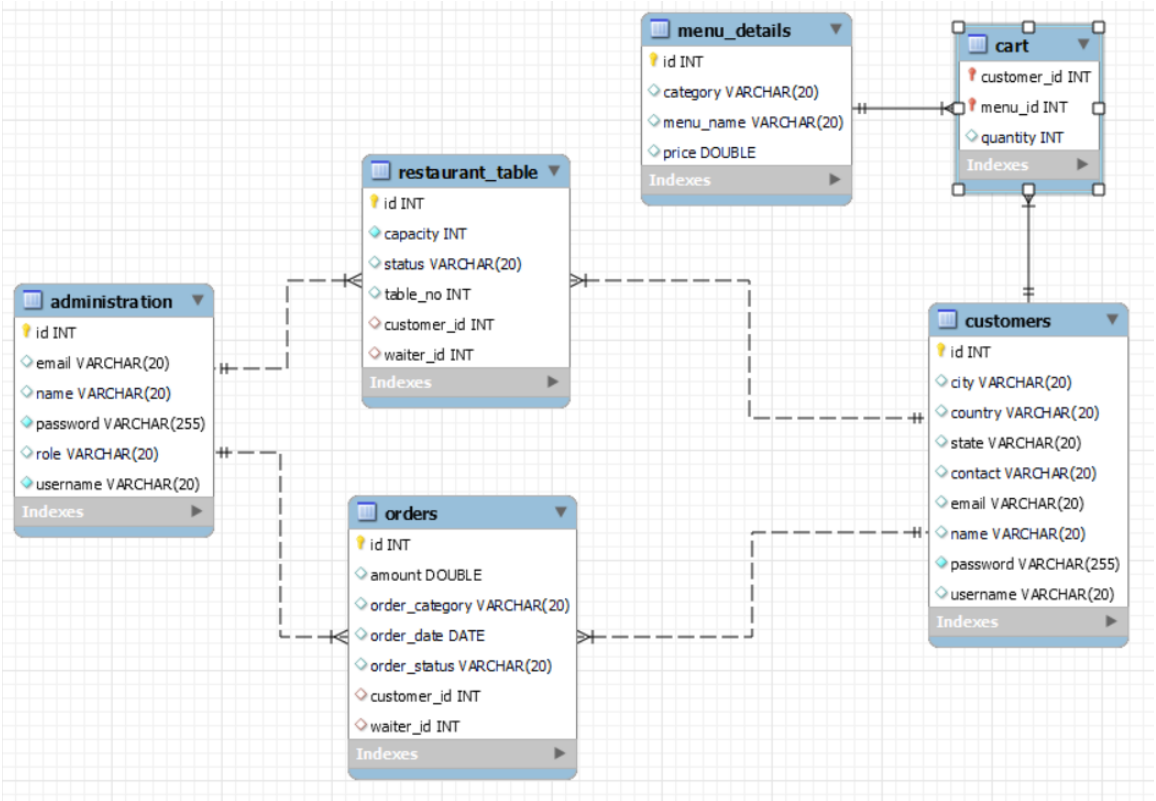
The system should have mechanisms in place to prevent unauthorized access to sensitive data and features.

4: PROJECT DESIGN

Class Diagram:



SCHEMA DIAGRAM:



5: REFERENCES

http://www.google.com

http://www.webdevelopersjournal.com/

http://www.w3.org

http://www.wikipedia.org

http://reactjs.org

http://getbootstrap.com