"ONLINE RESTAURANT MANAGEMENT SYSTEM"

Advance Web Programming Project

report

submitted

In the partial fulfilment the award of degree of

BACHELOR OFTECHNOLOGY

IN

COMPUTER SCIENCE AND ENGINEERING (2022-2023)

By

K. JANET CHRISTINA (211801390006)

Under the esteemed Guidance of G. Rama devi, Asst. Professor

Centurion
UNIVERSITY

CENTURION UNIVERSITY OF TECHNOLOGY & MANAGEMENT

Vizianagaram, Andhra Pradesh

Centurion
UNIVERSITY

CENTURION UNIVERSITY OF TECHNOLOGY & MANAGEMENT

Vizianagaram, Andhra Pradesh

BONAFIDE CERTIFICATE

This is to certify that the project work entitled **"**ONLINE RESTAURANT MANAGEMENT SYSTEM**"** is a fulfilment of project work done by **K. JANET CHRISTINA (211801390006)** for the awardthe degree of **BACHELOR OF TECHNOLOGY** in **COMPUTER SCIENCE AND ENGINEERING,** during academic year 2022-2023.

INTERNAL GUIDE
G. Rama Devi
M.TECH, (Ph.D)
Asst. Professor
Dept. Of CSE

HEAD OF THE DEPARTMENT
R. LAXMAN RAO
M.TECH, (Ph.D.)
Asst. Professor
Dept. Of CSE

ACKNOWLEDGEMENT

It is with at most pleasure and excitement we submit our project partial fulfilment of the requirement for the award of Bachelor of Technology.

The project is a result of the cumulative efforts, support, guidance, encouragement and inspiration from many of those for whom we must give our truthful honor and express gratitude through bringing out this project at the outset as per our knowledge.

We convey special thanks to our Project Guide **G. RAMA DEVI, MTech, (Ph.D.)** Asst.Professor who has guided us and encouraged us to enhance our knowledge with present working of this project to enrich the quality of project.

We expressed our appreciation **to R. LAKSHMANA RAO, M.Tech,(Ph.D)** Asst.Professor and Head of the Department, who facilitated us to providing a friendly environment which helped to enhance our skills in the present project.

DECLARATION

We hereby declare that the project entitled “University Management System” submitted to the fulfilment of award the degree of B.TECH.,(CSE) at CENTURION UNIVERSITY OF TECHNOLOGY AND MANAGEMENT (A.P).

This project work in original has not been submitted so far in any part or full for any other university or institute for the award of any degree or diploma.

K. JANET CHRISTINA	211801390006
V. BHANU SRITEJA	211801340002
G. INDU	211801340005
P. SUNANDITHA	211801390024
P. SRILAKSHMI	211801390036

Table of Contents

Table of Figure(s)

1.1 INTRODUCTION

1.1 Purpose

The purpose of this SRS is to outline both the functional and non-functional requirements of the subject RMOS. In addition to said requirements, the document also provides a detailed profile of the external interfaces, performance considerations and design constraints imposed on the subsequent implementation. It is the intention that the presented set of requirements possesses the following qualities; correctness, unambiguousness, completeness, consistency, verifiability, modifiability and traceability. Consequently, the document should act as a foundation for efficient and well-managed project completion and further serve as an accurate reference in the future. The primary audience of this SRS document will be the development team employed to implement the specified RMOS. It will not only provide an extensive capacity for project planning and progress assessment but it will further assist with developer/stakeholder interactions. The secondary document audience comprises the stakeholders of the project, that is, restaurateurs and associated staff. To this audience group, this SRS should convey and confirm the required functionality and represent a contractual agreement between the involved parties

1.2 Scope

In current formal dining environments, some form of physical static menu is utilised to convey the available food and beverage choices to customers. Said menus are generally paper based and hence impose restrictions on the textual real estate available and the ability a restaurateur has to update them. This document specifies the requirements for a restaurant paper menu and ordering replacement strategy to alleviate the problems associated with the current archaic method. Three related concepts are encompassed by the general scope of the Restaurant Menu and Ordering System. The first pertains to the replacement of paper-based menus using an electronic format, the second relates to a complementary electronic strategy for the front of house handling of a customer's order and the third surrounds the process of transferring said electronic orders to the kitchen for preparation. It should be noted that while the suggested strategy incorporates the use of various hardware components, the primary focus of the presented SRS relates to the constituent software elements.

1.3 Definitions, Acronyms and Abbreviations

- RFOS- restaurant Food Ordering System

- SRS - Software Requirements Specification
- DBMS Database Management System
- LAN Local Area Network
- IP Internet Protocol
- TCP Transmission Control Protocol
- UDP User Datagram Protocol
- IEEE 802.11 Wireless Local Area Network Standard
- Outstanding dues- Pending amount of money to be paid by the customer.

1.4 REFERENCES

- The website of the newspaper delivery service
- C. Larman, APPLYING UML AND PATTERNS An Introduction to Object-Oriented Analysis and Design and Iterative Development, 3rd ed., Massachusetts: Pearson Education, 2005.D. Carrington, CSSE3002 Course Notes, School of ITEE University of Queensland, 2008IEEE Recommended Practice for Software Requirements Specifications, IEEE Standard 830, 199.

1.5 OVERVIEW

This document will provide a detailed description of the functional and non-functional requirements of the online restaurant management system. It will also include system models and diagrams to help understand the system.

2.OVERALL DESCRIPTION

2.1 Product Perspective

The software described in this SRS is the software for a complete Restaurant food ordering system. The system merges various hardware and software elements and further interfaces with external systems. it relies on a number of external interfaces for persistence and unhandled tasks, as well as physically interfacing with humans.

2.1.2 User Interface

This interface uses the surface computer paradigm - users interact with the system by dragging 'objects' around on the flatscreen touch-sensitive display. For the ORMS, users can manipulate objects such as items of food, dietary requirements, tips and menus on the surface of their table. Such objects can be moved into static objects such as meals and payments to perform various functions. In addition to this object manipulation paradigm, a limited system menu is necessary.

Users will summon their restaurant menu, which is combined with a system/command menu, using an easy touch gesture, a double-tap on the touch surface, and dismiss it with a similar gesture or by tapping a close button GUI element.

2.1.3 Hardware Interfaces

These devices are the surface computers, the wireless tablets and the touch displays. All three devices must be physically robust and immune to liquid damage and stains. The devices (with the possible exception of displays) must also have good industrial design aesthetics, as they are to be used in place of normal restaurant tables and notepads and will be in direct contact with customers.

2.1.4 Software Interfaces

The RFOS will interface with a Database Management System (DBMS) that stores the information necessary for the RMOS to operate. The DBMS must be able to provide, on request and with low latency, data concerning the restaurant's menu, employees (and their passwords) and available dietary requirements

2.1.5 Communications Interfaces

The RFOS will interface with a Local Area Network (LAN) to maintain communication with all its devices. It should use a reliable-type IP protocol such as TCP/IP or reliable-UDP/IP for maximum compatibility and stability. All devices it will interface with should contain standard Ethernet compatible, software accessible LAN cards to maintain communication between the server and the surface computers, tablets, displays and the external payment system

2.2 PRODUCT FUNCTIONS

The functions of the product are given below:

Online Restaurant System Login Information System

- Description -The system will maintain the login information of its user to enter in to the software
- Validating Checks -Administrator need to login the unique id and password. -Contact number should have maximum 10 digits. -All the details must be fill up. -Email address should be in the proper format.
- Sequencing information -Login information should be filled before the user allowed.

- Error Handling -If user doesn't fill up validate information, then the system display error message for user and request to enter the validate information.
- Performance required Security 10 -System should be Protected from unauthorized access Where the validate Username and Password are required so no other can access.
- Maintainability -System should be design in a maintain order. So, it can be easily modified.

2.3 USER CHARACTERISTICS

Customer:

A customer shall be able to engage their menu by double tapping the activated surface computer in their table. A customer shall be able to add an item to a pending order by dragging the item from the engaged menu onto the order. A customer shall be able to remove an item from a pending order by dragging the item off the order. A customer shall be able to add a special dietary requirement to an order by dragging the requirement from the engaged menu onto the order. When in billing mode, a surface computer shall display a representation of a bankcard payment foreach custome

Waiter:

A waiter assigned to a table shall be alerted via their wireless tablet when: An order is placed from that table an item ordered by that table is rejected by the kitchen an item ordered by that table is ready to be served the table has requested waiter assistance tablet shall allow a waiter to accept, reject and modify an order placed by a customer through a surface computer. A tablet shall allow a waiter to process a payment using cash or a bankcard

Chef:

A chef shall be able to accept or reject a customer's order item through a display. A chef shall be able to indicate that a customer's order item is ready to be served through a display

Supervisor: A supervisor shall be able to abort/purge a table's account/meals from the active system with no expectation of payment. A supervisor shall be able to issue a refund for one or more items to a customer.

2.4 USER CONSTRAINTS

•GUI is only in English

- Login and Password for the identification of Users
- Delivery Person:
- cannot view the complete list of Customers
- cannot add/delete customers/products.
- cannot modify subscriptions (can only add a request).
- cannot generate bills/reminders.
- Manager:
- cannot enter requests of Customers.

2.5 ASSUMPTIONS & DEPENDENCIES

ASSUMPTIONS:

- Customers will pay the bills in online payment or through the delivery person.
- Customer can modify their subscription list by sending a request through the delivery person.
- It is assumed that the system never crashes.
- The manager and delivery boy understand English.
- The manager and delivery boy should be well versed with using a computer.

DEPENDENCIES:

- Modification of subscriptions depends on customer's request.
- The publications subscribed by the customers depends upon available products in the product list.

2.6 APPORTIONING REQUIREMENT

- We can develop our system even further for the customer to avail their subscription facility and modify it online.
- Online Payment of bills by customer.
- Text-to-speech feature.

3.SPECIFIC REQUIREMENTS

3.1 EXTERNAL INTERFACE:

3.1.1: WEB SERVER:

- The web server chosen is Apache:
- Using HTML forms, the user submits data to the web server

The web server runs PHP as a module, and if the post data is accessible, the Popcraft obtains it.

- The PHP script provides data back to the web server.
- The end-user sees an HTML page as a result from the web server.

3.1.2: PHPAPPLICATION:

PHP was used to create the actual programme that will carry out the procedures. A database will be used to store all the data.

3.1.3: MYSQL DATABASE:

It’s an open-source SQL database to store all data which communicates with the application on the server.

3.2 PERFORMANCE REQUIREMENTS:

Performance requirements are a set of criteria or specifications that specify the speed, capacity, and efficiency with which a system or application must operate. These specifications, which are frequently established by users or stakeholders of the system or application, are used to assess the system's performance and make sure that it satisfies its users' needs.

3.2.1 LOGICAL DATABASE SPECIFICATIONS:

All information, with the exception of files that are stored on the disc, will be saved in the database, including user accounts and profiles, discussion data, messages, etc. A solid database architecture is necessary for the database to support concurrent access and maintain consistency at all times.

3.2.2 DESIGN CONSTRAINTS:

1. SQL will be used for all communication between the portal programmed and the database.

2. HTML/CSS will be used to create the portal layout.

3. PHP will be used to create the product.

4. The output needs to be W3C XHTML 1.0 compliant.

5. The source code must adhere to PHP's coding standards.

6. Complete documentation must be available to system administrators.

3.3 SOFTWARE SYSTEM CHARACTERISTICS:

The components of the software are as follows:

1.the PHP program and

2.the Apache web server

3. MySQL, the database

4.REQUIREMENT SPECIFICATION

4.1 FUNCTIONAL REQUIREMENT

The requirement specification aims to outline the essential features and functionalities needed for an effective online restaurant management system.

- Allow users to create accounts and register to the system.
- Verify user identity through email or phone verification.
- Allow users to login and logout securely.
- Optimize delivery routes to ensure efficient and timely delivery.
- Allow users to modify, pause, or cancel their subscriptions.
- Provide subscription renewal reminders and alerts.
-

4.2 NON-FUNCTIONAL REQUIREMENTS:

Following Non-functional requirements will be there in the online shopping portal.

- Secure access of confidential data (customer’s details).
- 24 X 7 availability.

Better component design to get better performance at peak time.

Flexible service-based architecture will be highly desirable for future extension Non-functional requirements define system properties and constraints It arise through user needs, because of budget constraints or organizational policies, or due to the external factors such as safety regulations, privacy registration and so on.

Various other Non-functional requirements are:

1. Security
2. Reliability
3. Maintainability
4. Portability
5. Extensibility
6. Reusability
7. Application Affinity/Compatibility

5.DATA FLOW DIAGRAM

What it is?

The Data Flow Diagram shows the flow of data or information. It can be partitioned into single processes or functions. Data Flow Diagrams can be grouped together or decomposed into multiple processes. There can be physical DFD's that represent the physical files and transactions, or they can be business DFD's (logical, or conceptual). Dataflows DFDs show the flow of data from external entities into the system, showed how the data moved from one process to another, as well as its logical storage.

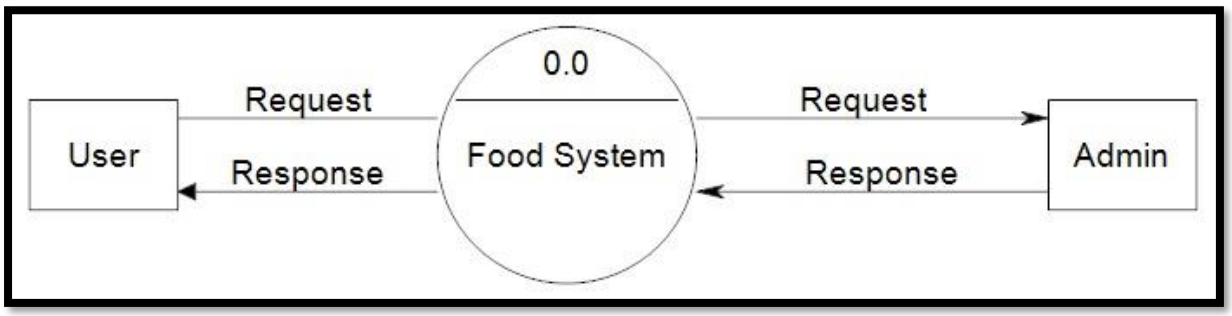


Figure :1.1Flow chart

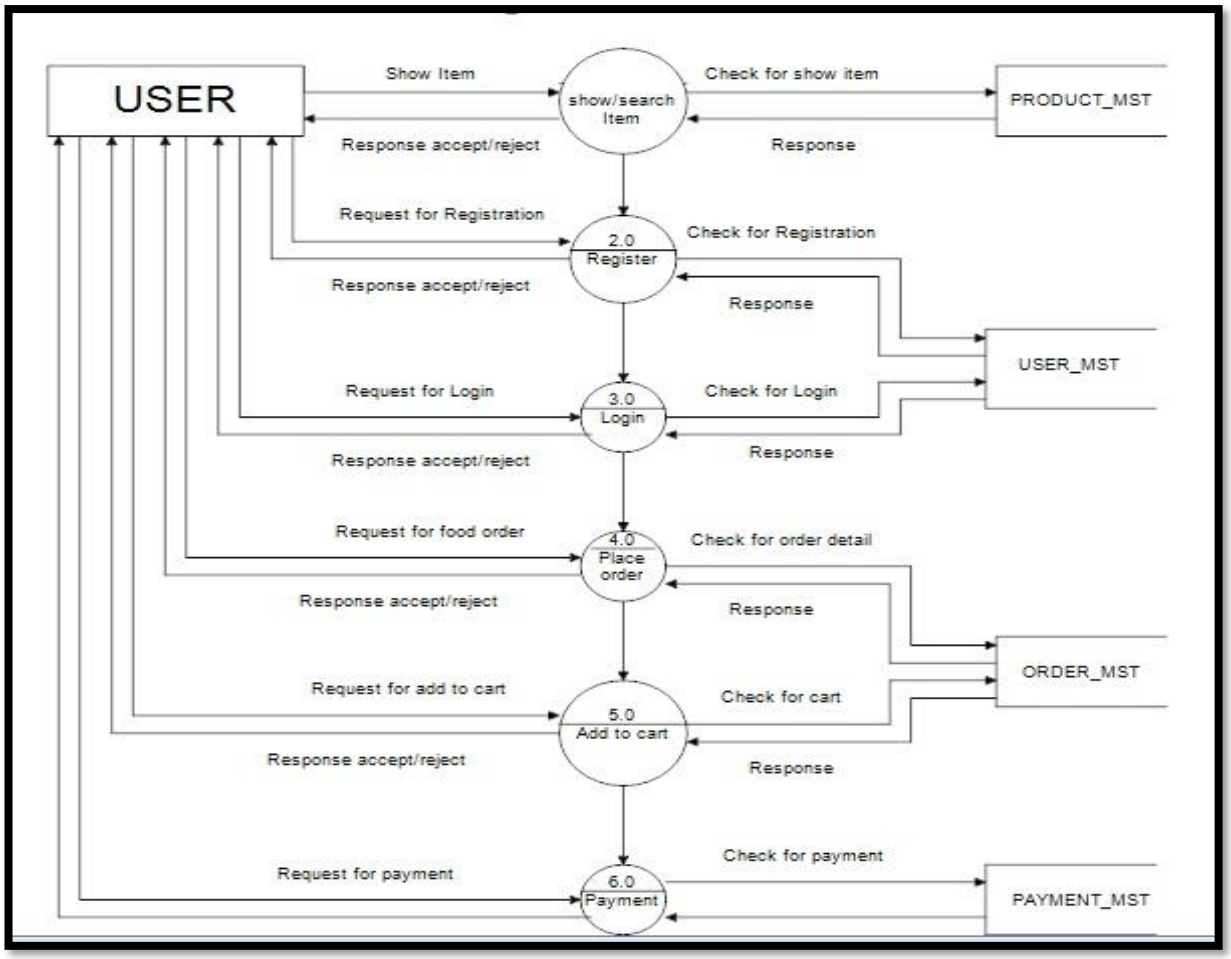
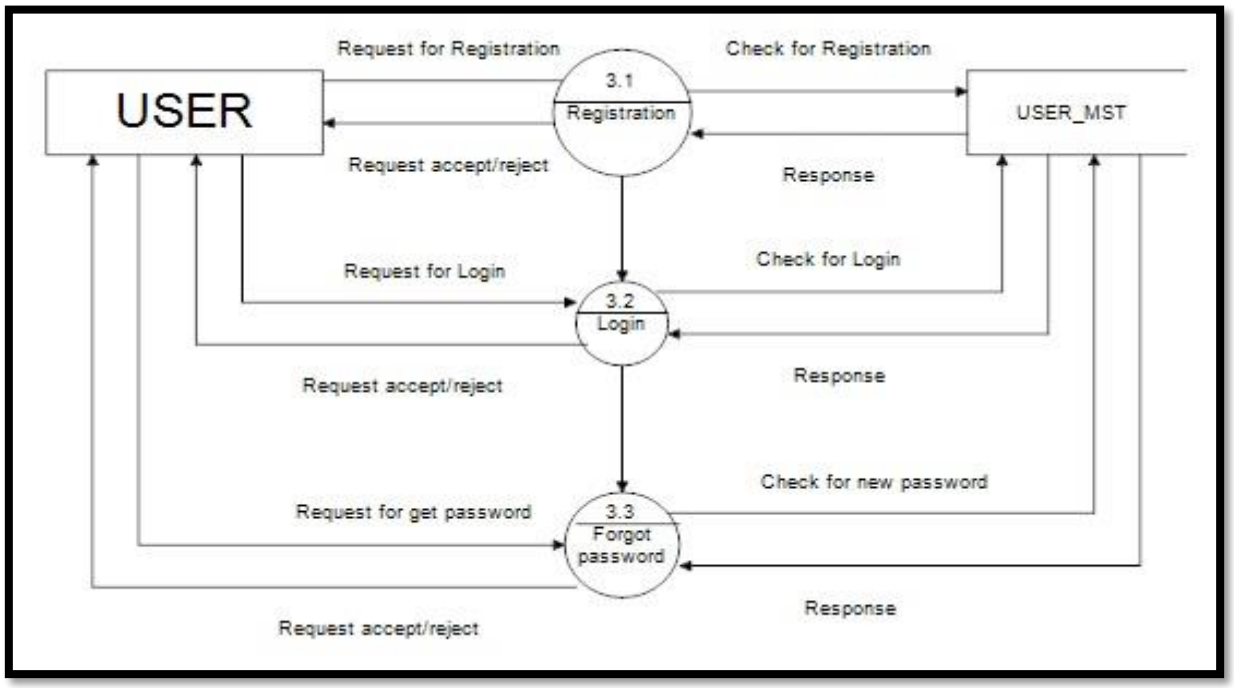


Figure :1.2 Flow chart for user side



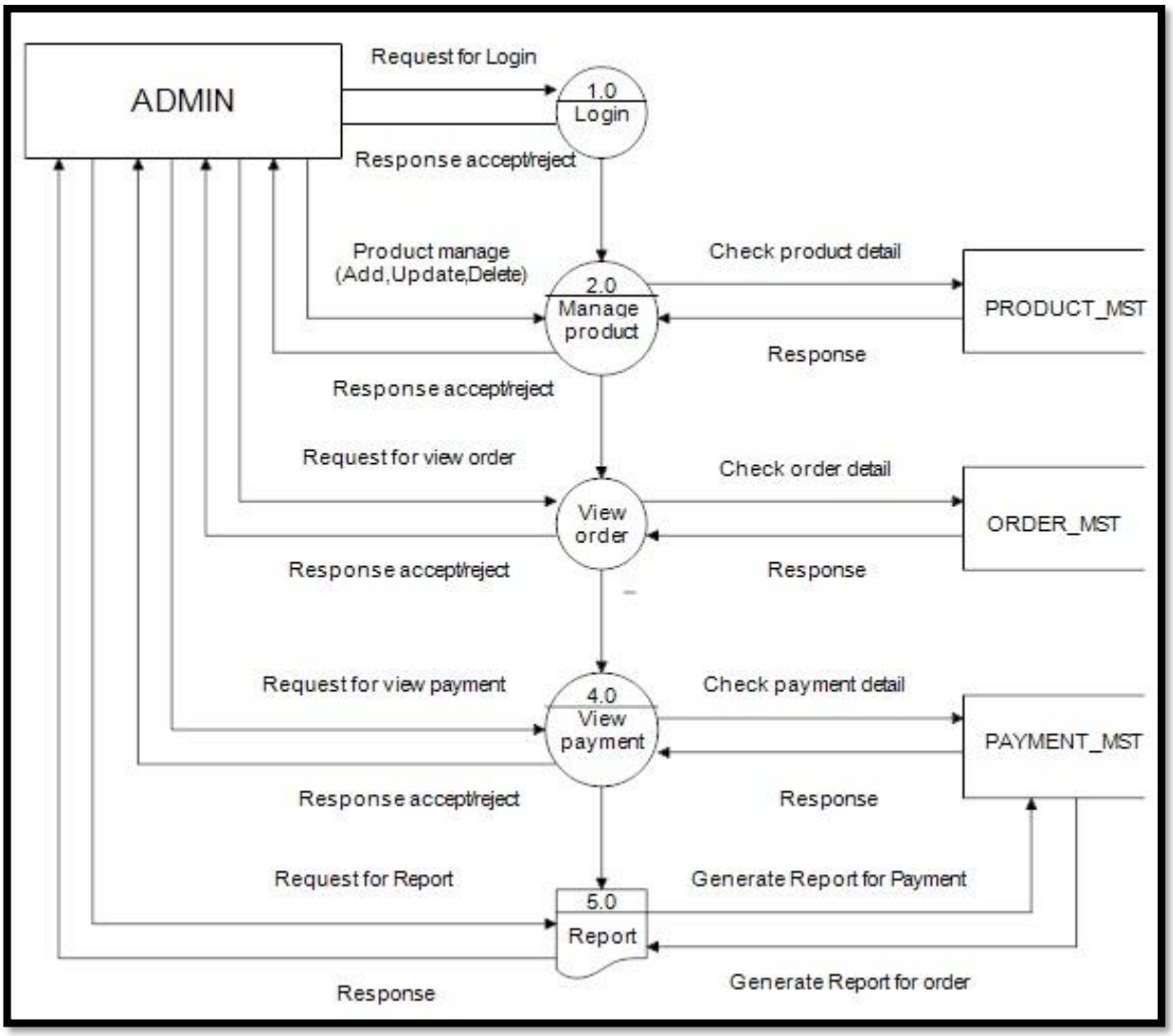


Figure :1.3 Flow chart for admin

5.1 Entity-Relationship Model

Simply stated the ER model is a conceptual data model that views the real world as entities and relationships. A basic component of the model is the Entity-Relationship diagram which is used to visually represent data objects.

Basic Constructs of E-R Modeling

The ER model views the real world as a construct of entities and association between entities.

Entities

Entities are the principal data object about which information is to be collected. Entities are classified as independent or dependent (in some methodologies, the terms used are strong

and weak, respectively). An independent entity is one that does not rely on another for identification. A dependent entity is one that relies on another for identification.

Relationships

A Relationship represents an association between two or more entities. Relationships are classified in terms of degree, connectivity, cardinality, and existence.

Attributes

Attributes describe the entity of which they are associated. A particular instance of an attribute is a value. The domain of an attribute is the collection of all possible values an attribute can have. The domain of Name is a character string.

Classifying Relationships

Relationships are classified by their degree, connectivity, cardinality, direction, type, and existence. Not all modeling methodologies use all these classifications.

Degree of a Relationship

The degree of a relationship is the number of entities associated with the relationship. The n-ray relationship is the general form for degree n. Special cases are the binary, and ternary, where the degree is 2 and 3 respectively.

Connectivity and Cardinality

The connectivity of a relationship describes the mapping of associated entity instances in the relationship. The values of connectivity are "one" or "many". The cardinality of a relationship is the actual number of related occurrences for each of the two entities. The basic types of connectivity for relations are: one-to-one, one-to-many, and many-to-many.

Direction

The direction of a relationship indicates the originating entity of a binary relationship. The entity from which a relationship originates is the parent entity; the entity where the relationship terminates is the child entity.

The direction of a relationship is determined by its connectivity type. An identifying relationship is one in which one of the child entities is also a dependent entity. A non-identifying relationship is one in which both entities are independent.

Existence

Existence denotes whether the existence of an entity instance is dependent upon the existence of another, related, entity instance. The existence of an entity in a relationship is defined as either mandatory or optional.

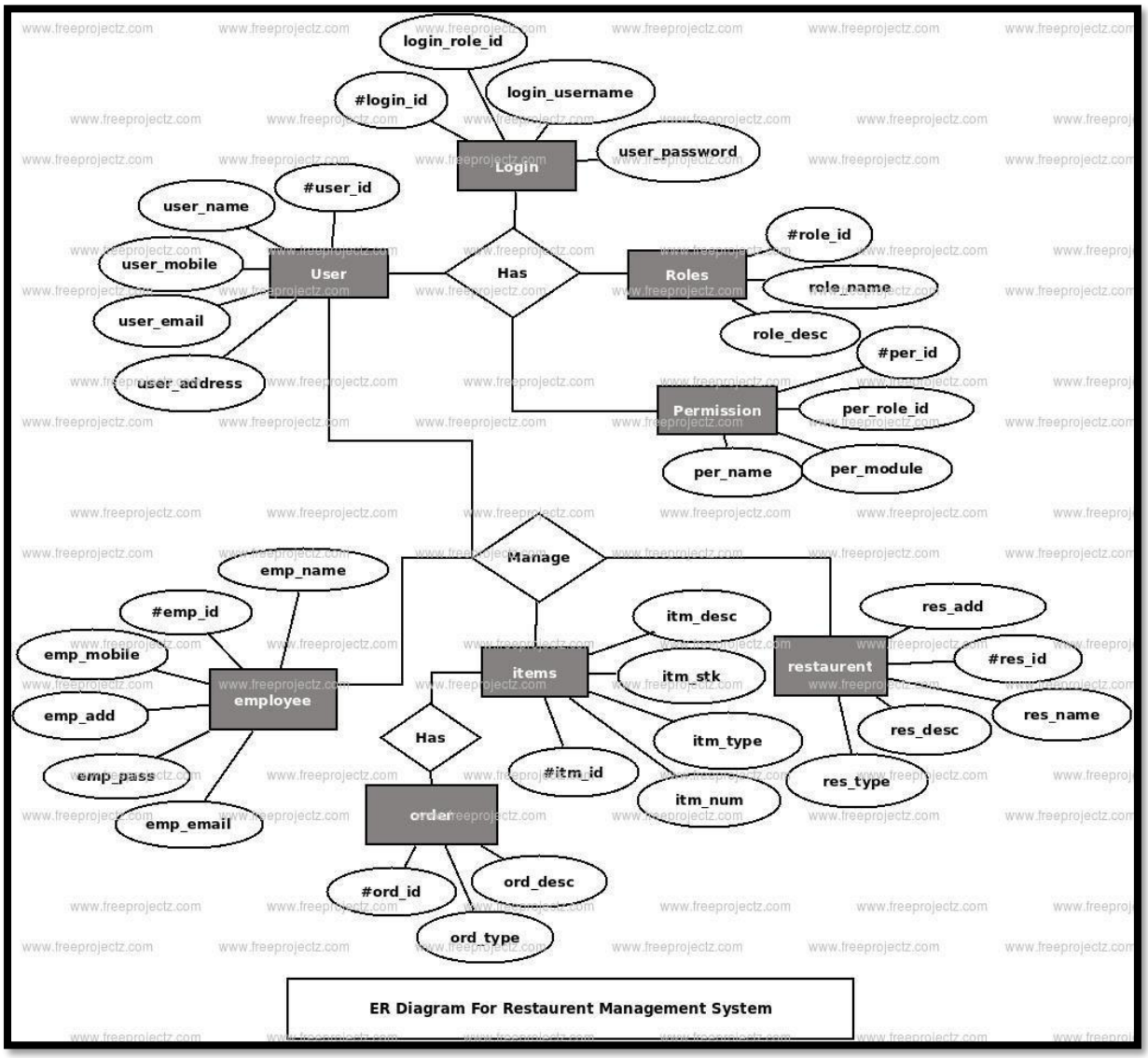


Figure 2.1: ER Diagram

Use Case Diagram

A use case describes a sequence of actions that provide something of measurable value to an actor and is drawn as a horizontal ellipse an actor is a person, organization, or external system that plays a role in one or more interactions with your system.

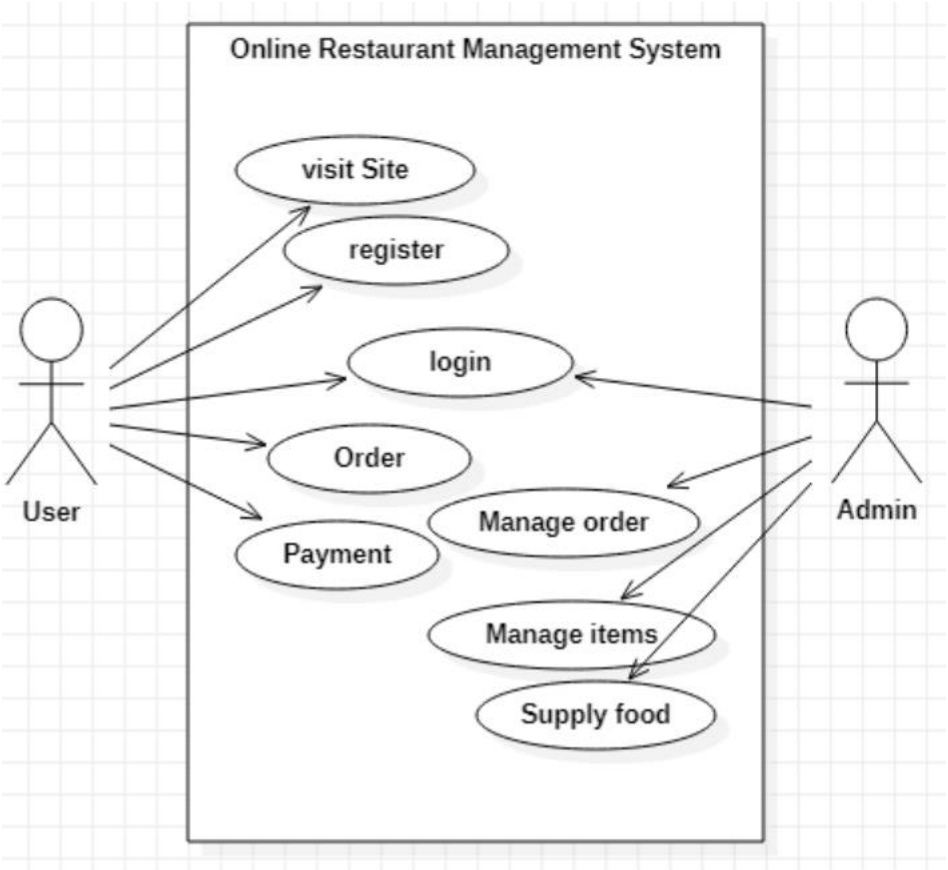


Figure 2.2: Use Case Diagram

6.Software System Attributes

6.1 Reliability:

- The system should operate consistently and reliably without unexpected errors or failures.
- It should be able to handle high loads and maintain performance under various conditions.

6.2 Availability:

The system should be available to users as per the defined service level agreements (SLAs).

It should minimize downtime and provide a high level of uptime.

6.3 Security:

The system should implement robust security measures to protect user data and ensure confidentiality.

It should have mechanisms to prevent unauthorized access, data breaches, and other security threats.

6.4 Maintainability:

The system should be designed to be easily maintainable and modifiable.

It should allow for efficient bug fixing, updates, and enhancements without disrupting the overall functionality.

6.5 Portability:

The application is windows -based and should be compatible with other systems. Apache, PHP and MySQL programs are practically independent of the OS-system which they communicate with. The end-user part is fully portable and any system using any web browser should be able to use the features of the application.