

SUPERIOR UNIVERSITY LAHORE



Faculty of Computer Science & IT

Final Year Project

PROJECT REPORT

Restaurant management system

Project ID: FYP-BSCS-F18-005

Project Team

Student Name	Student ID	Program	Contact Number	Email Address
Haris Ahsan	BCSM-F15-249	BSCS	0315-9297108	Harisahsan349@gmail.com
Abdul Ahad	BCSM-F15-293	BSCS	0323-4214124	Ahadhabib79@yahoo.com
Waleed Muaz	BCSM-F15-298	BSCS	0323-4155484	Waleedmuaz2@gmail.com

Project Supervisor

Muhammad Mumtaz Ali Rajput
(Lecturer CS & IT)

Project Report

Restaurant management system

Change Record

Author(s)	Version	Date	Notes	Supervisor's Signature
Abdul Ahad	1.0	10/12/18	Project Report Prepared	
Abdul Ahad Waleed Muaz Haris Ahsan	1.7	15/12/18	Project Plan Done	
Haris Ahsan	1.8	22/12/18	Changes are Advise by Supervisor	
Waleed Muaz Abdul Ahad Haris Ahsan	2.0	27/12/18	Project Management Plan	
Waleed Muaz/Haris Ahsan	2.3.1	3/3/19	Admin Panel Created	
Waleed Muaz	3.1	12/2/19	User Interface Changes	
Waleed Muaz Abdul Ahad Haris Ahsan	4.0	29/4/19	Changes are Advise by Supervisor	

APPROVAL

PROJECT SUPERVISOR

Comments: _____

Name: _____

Date: _____ Signature: _____

PROJECT MANAGER

Comments: _____

Date: _____ Signature: _____

HEAD OF THE DEPARTMENT

Comments: _____

Date: _____ Signature: _____

Dedication

“This work is dedicated to my Parents and teachers. That is all with the prayers of them. They give us blessing and confidence to do this. And on this day, we are able to work in real world just because of my parents and teachers prayers.”

Acknowledgements

“Learning has no limits. We are thankful to our supervisor, Mr.Mumtaz Ali for guidance, awareness, advice and give comments on (ONLINE VEHICLE TRACKING AND MONITORING SYSTEM) project idea, requirement for the system and specification and so forth. Besides, he also guides us about the life cycle of software development and shows us how to build a efficient system. He also gives feedback on our project.Our thanks also go to our CS & IT teacher for their advice and support”

Table of Contents

Contents

Dedication	4
Acknowledgements	4
Table of Contents	5
Chapter 1	7
Introduction	7
1.2 Background	8
1.3 Motivations and Challenges	8
1.4 Goals and Objectives	9
1.5 Literature Review/Existing Solutions	9
1.6 Gap Analysis	9
1.7 Proposed Solution	10
1.8 Project Plan	10
1.8.1 Work Breakdown Structure	10
1.8.2 Roles & Responsibility Matrix	12
1.8.3 Gantt Chart	13
1.9 Report Outline	13
Chapter 2	14
Software Requirement Specifications	14
2.1 Introduction	15
2.1.1 Purpose	15
2.1.2 Document Conventions	15
The symbols used in this document do NOT follow any particular naming conventions nor other conventions .but we use adhere to POSIX.....	15
2.1.3 Intended Audience and Reading Suggestions	15
2.1.4 Product Scope	15
2.2 Overall Description	16
2.2.1 Product Perspective	16
2.2.2 Product Functions	16
2.2.3 User Classes and Characteristics	16
2.2.4 Operating Environment	17
The Cafeteria Ordering System shall operate on a server running the current corporate approved versions of Red Hat Linux and Apache WebServer.	17
2.2.5 Design and Implementation Constraints.....	17
2.2.6 User Documentation	17
2.2.7 Assumptions and Dependencies	17
The restaurant is open for breakfast, lunch, and dinner of every day in which employees are expected to be on site. The operation of the restaurant depends on changes being made in the Payroll System to accept payment requests for meals ordered with the restaurant. The operation of the restaurant depends on changes being made in the Inventory System to update the availability of food items as orders are accepted.	17
2.3 External Interface Requirements	18
2.3.1 User Interfaces	18

2: Register Page	19
2.3.2 Hardware Interfaces.....	24
2.3.3 Software Interfaces	24
Communications Interfaces	24
2.4 System Features.....	25
2.4.1 System Feature 1	25
2.4.1.1 Functional Requirements	25
2.4.2 System Feature 2	25
2.4.2.1 Functional Requirements	25
2.4.3 System Feature 3	25
2.5 Other Nonfunctional Requirements	26
2.5.1 Performance Requirements.....	26
2.5.2 Security Requirements.....	27
2.5.3 Software Quality Attributes.....	28
2.5.4 Business Rules.....	28
Chapter 3.....	29
Use Case Analysis.....	29
Chapter 4.....	37
System Design	37
4.1. Architecture Diagram.....	38
Architecture Diagram below in fig 4.1.1	38
4.2. Domain Model.....	39
Domain model is below in fig 4.2.1.....	39
4.3. Entity Relationship Diagram.....	40
4.4. Class Diagram	41
4.5. Sequence / Collaboration Diagram	42
4.6. Operation Contract	52
4.7. Activity Diagram.....	57
State Transition Diagram	58
4.8. Deployment Diagram	63
4.9. Data Flow diagram	64
4.10. Component Diagram.....	67
Chapter 5.....	68
Implementation	68
5.1. Important Flow Control/Pseudo codes.....	69
5.2. Components, Libraries, Web Services and stubs	69
5.3. Deployment Environment	70
5.4. Tools and Techniques.....	71
5.5. Best Practices / Coding Standards.....	71
5.6 Version Control.....	71
Reference and Bibliography	90

Chapter 1

Introduction

Chapter 1: Introduction

1.1 Introduction

We are going to develop an online Restaurant management system for final year project.

The restaurant management systems help the restaurant manager to manage the restaurant more effectively and efficiently by computerizing meal ordering, billing, and inventory control. It will be all web based that handle all the transaction between customer and sales department and take all the record of stock and create list of the weekly, monthly or yearly sales. In addition to this we run the app on an Android based tablet & not on an iOS based tablet which is more expensive alternative.

1.2 Background

The major problem to manage the restaurant is to maintain the sales record of the restaurant .if you manually maintain record then your staff may be hide or corruption with the sale and you could not be say anything to your staff because you have no clue because they have lot of excuses. If keep the record of sale moment by moment by a proper system then profit must be goes in a good direction.

1.3 Motivations and Challenges

We get motivations for this project from the supervisor and Sir Aziz Jan who purchased our product. When we discussed over idea with him he was very happy with this idea and ask us to do this project but still there no such kind of challenge we are facing Insha'Allah we will do this project on time comfortable.

1.4 Goals and Objectives

Objective of this project is to provide a better handling the customer order, restaurant management and billing system. Goal of this project is to provide the better communication between customer and management.

1.5 Literature Review/Existing Solutions

There are many computerized restaurant management systems available but for each system there exist disadvantages or missing features. The most common type of restaurant management system contains a static order entry computer system usually in the shape of a desktop computer with a touch screen. Typically this common approach is adequate to the restaurants requirements but still requires handwritten orders to be relayed to the order entry computer system.

1.6 Gap Analysis

<p>Strength</p> <ul style="list-style-type: none"> • Time Management • Passion for customer services 	<p>Weakness</p> <ul style="list-style-type: none"> • Competitor is present in market
<p>Opportunity</p> <ul style="list-style-type: none"> • Get all information on time 	<p>Threats</p> <ul style="list-style-type: none"> • We have threat from competitors

1.7 Proposed Solution

I think the best solution for this, is the web based project. It provides the better facility for check and maintain all the sales record and its calculate all the information by quickly. This system creates all the sales record by daily, monthly or yearly that's very helpful for the owner of the company. In which it has two panels one is for sales department and one is for owner who will check the detail of sales and also change the price of their restaurant menu.

1.8 Project Plan

Project plan will be discussed in work break down structure in detail.

1.8.1 Work Breakdown Structure

1- Requirement Gathering

- 1.1- Technical specification
- 1.2- Restricted areas
- 1.3- Available solutions

2- Establish Design

2.1- Website

- 2.1.1- Design elements
- 2.1.2- Banner
- 2.1.3- Footer
- 2.1.4 - Logo
- 2.1.5- Color scheme
- 2.1.6- Font usage
- 2.1.7- Icons
- 2.1.9- Content
- 2.1.10- Header

3- Admin panel:

- 3.1- Admin dashboard
- 3.2- Visit page List
- 3.3- Verify user register page

4- User Panel:

- 4.1-Home/index page
- 4.2-Login/ Signup Page
- 4.3- download data
- 4.4-Contact Us
- 4.5-news updates
- 4.6-Feedback
- 4.7-A bit about us Page
- 4.8-logout

5- Database Module

- 5.1-database operations

6- Documentation

- 6.1- User Documentation

7- Testing

- 7.1 Performance testing
- 7.2- Load testing
- 7.3- Security testing
- 7.4- Unit testing
- 7.5- Integration testing
- 7.6- System testing

1.8.2 Roles & Responsibility Matrix

The purpose of roles & responsibility matrix is to identify who will do what.

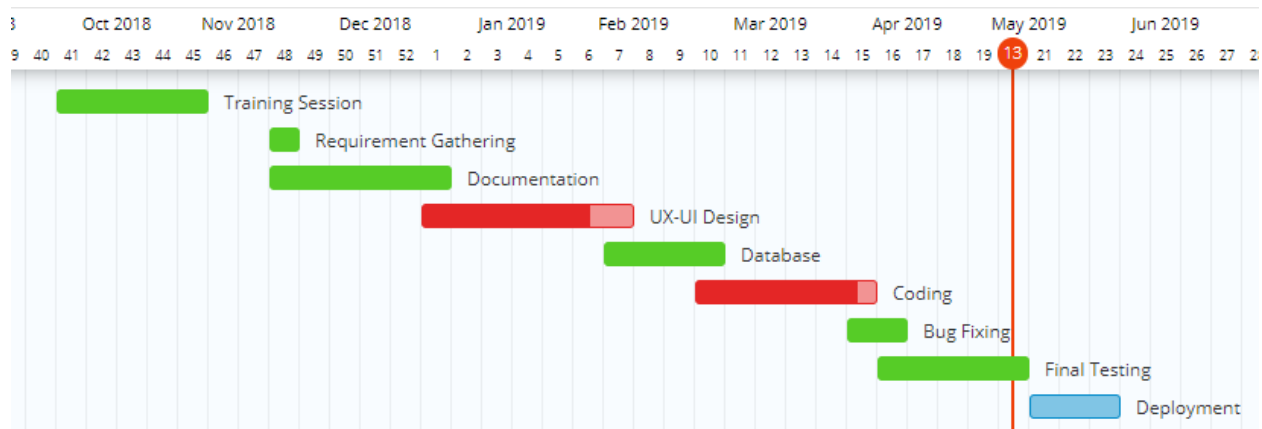
1. WBS #	WBS Deliverable	Activity #	Activity Complete to the Deliverable	Duration (# of Days)	Responsible Team Member(s) & Role(s)
1	Training Session			23	All of us
2	Requirement Gathering			15	Abdul Ahad
3	UX-UI Design			25	Waleed
4	Documentation			30	Ahad/ Haris/Waleed
5	Database			40	Waleed Muaz
6	Coding			60	Waleed Muaz /Haris
7	Bug Fixing			20	Haris Ahsan
8	Final Testing			15	Abdul Ahad
9	Deployment			10	Abdul Ahad

1.8.3 Gantt Chart

Gantt chart is given below

0	✓	Training Session	-	08/Oct	07/Nov	100%
1	✓	Requirement Gathering	-	26/Nov	26/Nov	100%
2	✓	Documentation	-	27/Nov	31/Dec	100%
3	⊗	UX-UI Design	-	01/Jan	11/Feb	80%
4	✓	Database	-	12/Feb	08/Mar	100%
5	⊗	Coding	-	09/Mar	09/Apr	90%
6	✓	Bug Fixing	-	10/Apr	20/Apr	100%
7	✓	Final Testing	-	21/Apr	13/May	100%
8	⊗	Deployment	-	20/May	05/Jun	0%

Fig(1.8.3.1)



S

Fig(1.8.3.2)

1.9 Report Outline

Project is to provide a better handling the customer order, restaurant management and billing system This system creates all the sales record by daily, monthly or yearly that's very helpful for the owner of the company.

Chapter 2

Software Requirement Specifications

Chapter 2: Software Requirement Specifications

2.1 Introduction

2.1.1 Purpose

The purpose of this SRS is to outline both the functional and non-functional requirements. 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

2.1.2 Document Conventions

The symbols used in this document do NOT follow any particular naming conventions nor other conventions .but we use adhere to POSIX

2.1.3 Intended Audience and Reading Suggestions

This Software Requirements document is intended for: –

Developers who can review project’s capabilities and more easily understand where their efforts should be targeted to improve or add more features to it (design and code the application – it sets the guidelines for future development). – Project testers can use this document as a base for their testing strategy as some bugs are easier to find using a requirements document. This way testing becomes more methodically organized. End users of this application who wish to read about what this project can do.

2.1.4 Product Scope

This document specifies the requirements for a restaurant paper menu and ordering replacement strategy to solve the problems. Three related concepts are encompassed by the general scope of the Restaurant Menu and Ordering System. The first priority 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.

2.2 Overall Description

2.2.1 Product Perspective

The software described in this SRS is the software for a complete RMOS system. The system merges various hardware and software elements and further interfaces with external systems. Thus, while the software covers the majority of the system's functionality, it relies on a number of external interfaces for persistence and unhandled tasks, as well as physically interfacing with humans.

2.2.2 Product Functions

- **Stock & Inventory Management:** Manages and maintains a record of everything you need as a raw material in the business operations.
- **Customer Relationship Management:** Keep relationship with both new & existing customers.
- **Automatic Billing:** Automatic Bill generation, cost calculation & expense monitoring.
- **Graphical & Analytical Marketing:** Marketing your website through an automated platform using Emails.
- **Easy to use Interface:** Cloud Backup, Works both online & offline & Integration with real time systems.

2.2.3 User Classes and Characteristics

- **User**

Users of our website are Management/customer they will login using their Id's and password and perform the task.

- **Admin**

Administration will update information for RMS.

- **Database**

All actions will perform in database .Data will be stored and retrieved.

2.2.4 Operating Environment

The Cafeteria Ordering System shall operate on a server running the current corporate approved versions of Red Hat Linux and Apache WebServer.

The software will operate with the following software components and applications: The software being developed will be running under Windows and embedded operating system. The hardware that will be running these programs is being developed for this specific project, and will follow the specifications that appear in this document. The synchronization procedures will be written to interface with Windows XP, Windows.

2.2.5 Design and Implementation Constraints

The project intends to follow SCRUM (AGILE METHODOLOGY) for its development. Agility helps in time boxed iterative development, suggests adaptive planning and promote incremental delivery. To deal with changing requirements and yield frequent and small software increments which can be adjusted, documented and built on the project intends to use SCRUM as the process model.

2.2.6 User Documentation

The product will include user manual. The user manual will include product overview, complete configuration of the used software, technical details, backup procedure and contact information which will include email address. The product will be compatible with the Google Chrome.

2.2.7 Assumptions and Dependencies

The restaurant is open for breakfast, lunch, and dinner of every day in which employees are expected to be on site. The operation of the restaurant depends on changes being made in the Payroll System to accept payment requests for meals ordered with the restaurant. The operation of the restaurant depends on changes being made in the Inventory System to update the availability of food items as orders are accepted.

2.3 External Interface Requirements

2.3.1 User Interfaces

Application will be accessed through a Browser Interface. The interface would be viewed best using 1024 x 768 and 800 x 600 pixels' resolution setting. The software would be fully compatible with Google Chrome. No user would be able to access the application without logging on to the system

1: User Interface

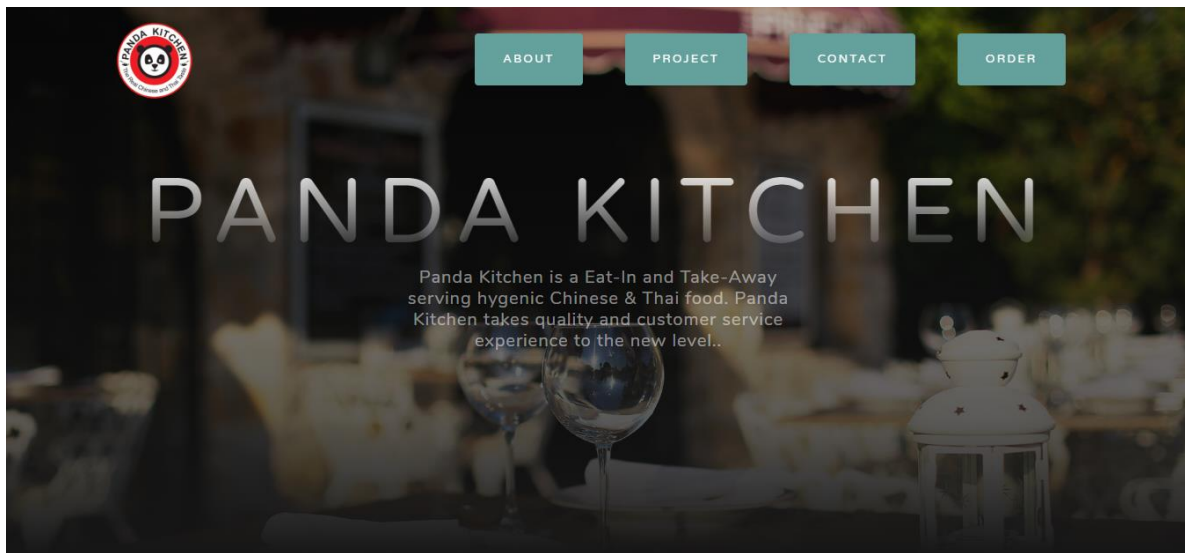


Fig (2.3.1.1)

2: Register Page

The screenshot shows a registration form titled "Register". It contains the following fields and elements:

- Name:
- E-Mail Address:
- Password:
- Confirm Password:
- Address:
- Account Type: (dropdown menu)
- Register:

Fig (2.3.1.2)

3: Login Page

The screenshot shows a login form titled "Login". It contains the following fields and elements:

- E-Mail Address:
- Password:
- Remember Me:
- Login:
- Forgot Your Password?: [Forgot Your Password?](#)

Fig (2.3.1.3)

4: Dashboard

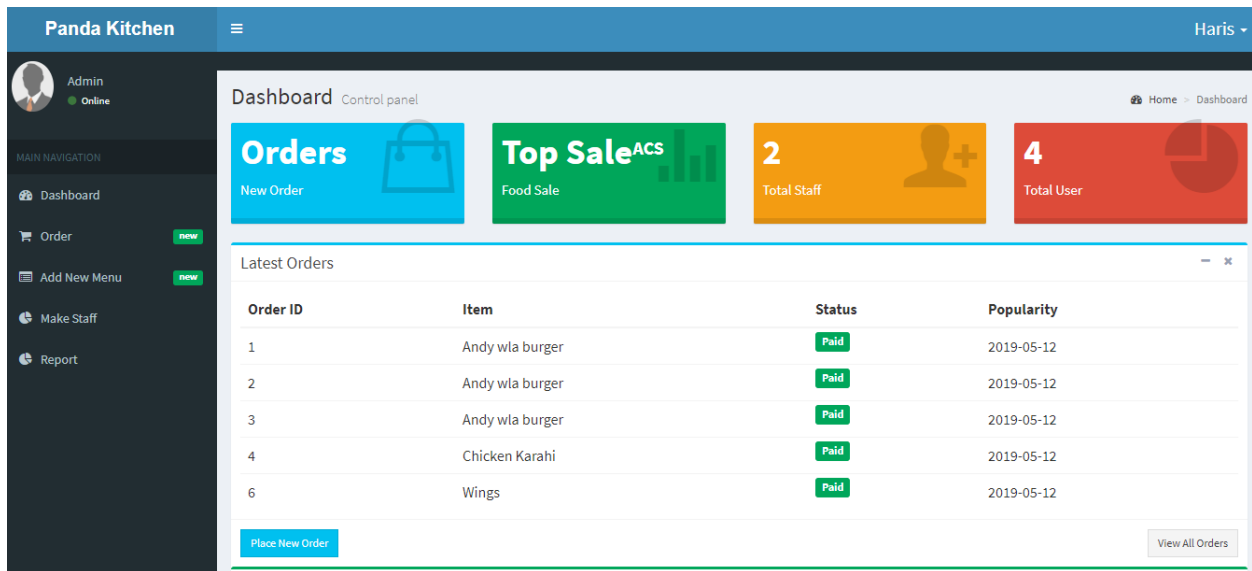


Fig (2.3.1.4)

5: Feedback



Fig (2.3.1.5)

6: Menu Items

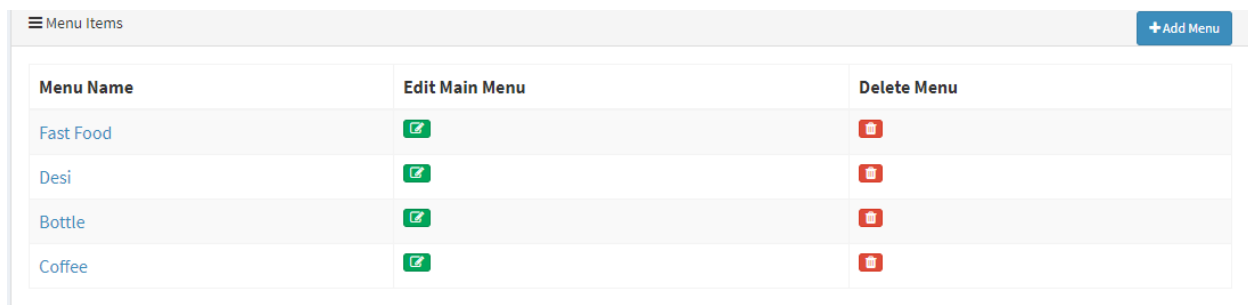


Fig (2.3.1.6)

7: Sub Menu

Item Name	Price	Quantity	Update	Remove
Zinger Burger	120	98	Edit	Delete
Pizza	320	84	Edit	Delete
Shawarma	100	45	Edit	Delete
Paratha Roll	190	178	Edit	Delete
Wings	35	147	Edit	Delete

Fig (2.3.1.7)

8: Add Menu Item

Fig (2.3.1.8)

9: Update Sub Menu Item

Fig (2.3.1.9)

10: Add Staff

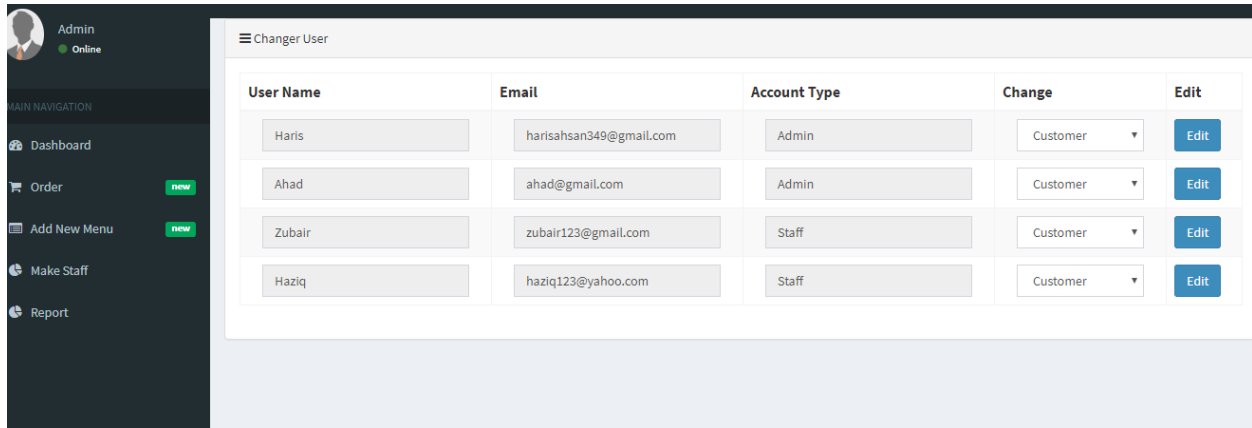


Fig (2.3.1.10)

11: Order

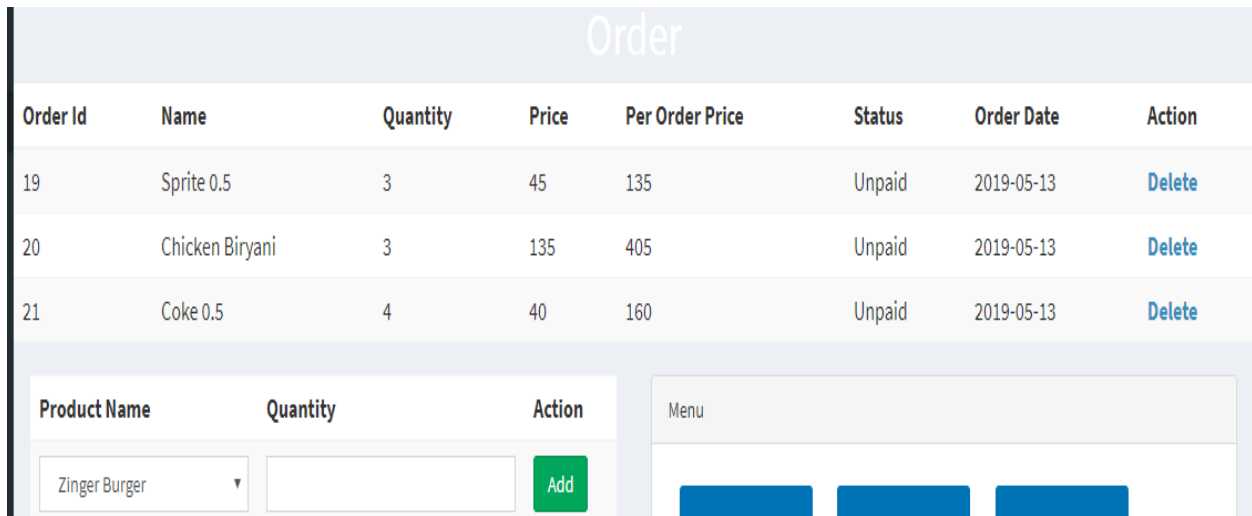


Fig (2.3.1.11)

12: Menu

Menu		
Zinger Burger Rs 120	Pizza Rs 320	Shawarma Rs 100
Paratha Roll Rs 190	Wings Rs 35	Chicken Karahi Rs 800
Mutton Karahi Rs 1400	Chicken Biryani Rs 135	Sindhi Biryani Rs 145
Bar B Que Rs 340	Coke 1.5 Rs 90	Coke 0.5 Rs 40

Fig (2.3.1.12)

13: Bill

The screenshot shows a modal window titled "Modal Header" with a close button (X) in the top right corner. Inside the modal, there is a table with the following columns: Order Id, Name, Quantity, Price, and Per Order Price. The table contains three rows of data. Below the table, there is a red button labeled "Total Bill : 700". At the bottom right of the modal, there is a button labeled "Order Paid".

Order Id	Name	Quantity	Price	Per Order Price
19	Sprite 0.5	3	45	135
20	Chicken Biryani	3	135	405
21	Coke 0.5	4	40	160

Total Bill : 700

Order Paid

Fig (2.3.1.13)

2.3.2 Hardware Interfaces

Customer, management or staff can use this website by using 80 GB hard drive and 2 GB ram and embedded system. If customer will use better than that hardware system then they will get better performance.

2.3.3 Software Interfaces

The system shall be capable of running on any version of window OS, web Application Android base Application . The system shall make use of the operating system.

Communications Interfaces

The communication between the different parts of the system is important since they depend on each other. However, in what way the communication is achieved is not important for the system and is therefore handled by the underlying operating systems for the web Application and Android Application system.

2.4 System Features

2.4.1 System Feature 1

This feature is for Customer.

2.4.1.1 Functional Requirements

- 1) Register
- 2) Login
- 3) Give Order
- 4) Form Address
- 5) Bill
- 6) Feedback
- 7) logout

2.4.2 System Feature 2

This feature is for Saff.

2.4.2.1 Functional Requirements

- 1) Login
- 2) Take order
- 3) Bill
- 4) Logout

2.4.3 System Feature 3

This system feature is for Admin

2.4.3.1 Functional requirements

- 1) Login
- 2) Update list of Inventory System
- 3) Add staff
- 4) Update Menu

- 5) Check Feedback
- 6) Generate Report
- 7) logout

2.5 Other Nonfunctional Requirements

2.5.1 Performance Requirements

Usability:

The system's user interface will be very simple and self-explanatory. The Staff will interact with the system through Physical Hardware. They are very easy to adapt to and use. The users just have to click the option they want. The menus for the restaurant will show up as drop-down menus on the Screen. The staff only have to select the item that the customer ordered from the drop-down list.

Reliability:

The system is guaranteed to be reliable. All inputs to the system will be selections from options that the system will show on the screen. Since the system will only provide valid options, there is absolutely no possibility that invalid inputs can be entered. Each user will have a unique username and password. This removes the risk of unauthorized access to the system. Also, users can only access parts of the system that they require for their job. For example, a staff cannot access the inventory management or the payroll feature of the system. Those parts among a few others can only be accessed by the admin. All these constraints will ensure the reliability of the system.

Performance:

The system will be used by many staff of the restaurant at the same time and can handle it without any errors. However, for this to happen, the server at the restaurant should be able to handle all the traffic without creating any problems. The restaurant should also have a high

speed wireless connection for the system to perform its best. Since some tasks like placing order are sequential, the staff has to update the system with the order first before the cook will see it. In order to for this to happen efficiently, the internet connection needs to be fast so that changes are reflected on the website instantly without any delay.

Supportability:

The system will support changes that the restaurant might need to make in future. The admin will have the ability to modify items from the menu. He will also be able to add or delete staff to the system for new hires.

Implementation Requirements:

We plan to utilize PHP for the front-end and MySQL as our database in the back-end. The server should have the ability to run both PHP , MySQL and android.

Interface Requirements:

The system requires both the admin and the staff to have system. The Restaurant should also have a high-speed wireless connection be connected to the system.

Operations Requirements:

The restaurant admin will be the administrator of the system. They can modify the database and add or delete users from the system. The manager can also compile reports to analyze the restaurant performance in general or to see the bestselling item or to compare different Staff.

2.5.2 Security Requirements

Security. Provider systems SHALL resist unauthorized, accidental or unintended usage and provide access only to users.

- Volume and performance
- Capacity
- Scalability

- Availability
- Recoverability
- Audit & provenance
- Maintainability

2.5.3 Software Quality Attributes

- Performance
- Scalability.
- Capacity.
- Availability.
- Reliability.
- Recoverability.
- Maintainability.
- Serviceability.

2.5.4 Business Rules

- Performance
- Scalability
- Capacity
- Availability
- Reliability
- Serviceability
- Security
- Manageability
- Data Integrity
- Usability

Chapter 3

Use Case Analysis

Chapter 3: System Analysis

The system Analysis Use case analysis is a technique used to identify the requirements of a system (normally associated with software/process design) and the information used to both define processes used and classes (which are a collection of actors and processes) which will be used both in the use case and the overall use case in the development or redesign of a software system or program. A use case analysis is the primary form for gathering usage requirements for a new software program or task to be completed. The primary goals of a use case analysis are designing a system from the user's perspective, communicating system behavior in the user's terms, and specifying all externally visible behaviors

3.1 Use Case Model:

Use Case 2.0

The use case of RMS is below in fig (3.1.1)

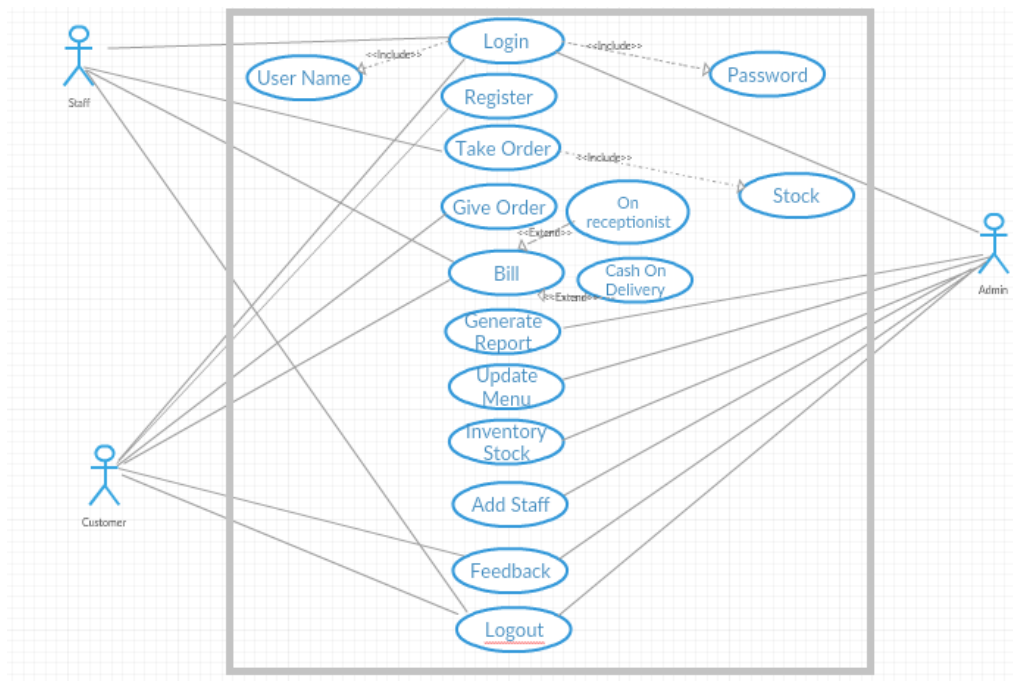


Fig (3.1.1)

3.2 Fully Dress use case:

Use Case Name: Register

Scope:

The Scope of registration is only Authorized person can access and use the facility to book the order.

Primary Actor:Customer

Pre-Conditions:

The Customer should have a valid information to Login.

Main Scenario:

1. Firstly the customer has to register.
2. If it don't register then it must be register before login.
3. After login the User get our services.
4. After successful registration the Actor providing by unique id.

Failure Scenario:

1. The actor has failed to register by providing wrong input.
2. Failure may occur through server problem.
3. Show the Error if the User donot providing correct input.
4. If the there is no server problem or it is already register then it should be contact with the administrator.

Use Case Name: Login

Scope:

The Scope of Login is only Authorized person can access the system.

Primary Actor:Admin, staff and customer.

Pre-Conditions:

The Customer should have already register by his/her correct information.

Main Scenario:

1. The user have email and password to login the website
2. If the user haven't register and want to check the rate then they can also do it.

Failure Scenario:

1. The user input wrong email but correct password
2. The user have wrong password but wrong email
5. The actor have failed to login by providing wrong input
6. Failure may occur through server problem
7. Show the Error if the User cannot providing correct input.

8. If there is no server problem or it is already registered then it should be contacted with the administrator

Use Case Name:Take order

Scope:

The Scope of taking order is only Authorized staff can take the order. The staff can take the order according to the valid criteria and also change the order if the staff wants to change/delete the order then the staff can do it.

Primary Actor:staff

Pre-Conditions:

The staff must have a Login. System must be connecting with the high speed internet.

Main Scenario:

1. Firstly, the User Staff has to login
2. After login the staff get our services
3. The staff has enough information to book the order.
4. After taking the order the staff can change the order.
5. The staff can also delete the order..

Alternative Scenario:

1. The staff edits the information after submitting.
2. Failure may occur through server problem in submission then the page reloads in the same tab.

Failure Scenario:

1. The staff has failed to click submit button
2. Failure may occur through server problem
3. Show the Error if the User does not provide the correct information.

Use Case Name:Give order

Scope:

The Scope of given order is only Authorized customer can give the order. The customer can give the order according to the valid criteria and also change the order if the customer wants to change/delete the order then the customer can do it.

Primary Actor:customer

Pre-Conditions:

The customer must have register or Login. System must be connecting with the high speed internet.

Main Scenario:

1. Firstly, the customer has to login.
2. After login the customer get our services.
3. The customer have enough information to booked the order.
4. After given the order the customer can change the ordrer before confirmation.
5. The customer can also delete the order before confirmation.

Failure Scenario:

1. The customer have failed to click submit button
2. Failure may occur through server problem
3. Show the Error if the customer do not provide the correct information.

Use Case Name: Bill**Scope:**

The Scope of payment bill is only Authorized customer can pay the the bill customer can pay the bill on receptionist or delivery.

Primary Actor:Staff and Customer.

Pre-Conditions:

The customer must have the order to get bill. .

Main Scenario:

1. Firstly, the customer and Staff has to login
2. Select the order from the dropdown list

Alternative Scenario:

1. If the user give order on receptionist then it pay bill on the spot.
2. If user booked order from mobile application then it pay the bill on delivery.

Use Case Name:Give order**Scope:**

The Scope of given order is only Authorized customer can give the order. The customer can give the order according the valid criteria and also change the order if the customer wants to change/delete the order then the customer can do it.

Primary Actor:customer

Pre-Conditions:

The customer must have register or Login. System must be connecting with the high speed internet.

Main Scenario:

- 1.1. Firstly, the customer has to login.
- 1.2. After login the customer get our services.
- 1.3. The customer have enough information to booked the order.
- 1.4. After given the order the customer can change the ordrer before confirmation.
- 1.5. The customer can also delete the order before confirmation.

Failure Scenario:

- The customer have failed to click submit button
- Failure may occur through server problem
- Show the Error if the customer do not provide the correct information.

Use Case Name: Add staff

Scope:

The Scope of given add staff is only Authorized admin can give the make new entry of add staff. The staff can change the information of given customer order .The valid criteria and also change the order if the customer wants to change/delete the order then the Staff can do it.

Primary Actor: Admin

Pre-Conditions:

The Admin must have Login.

Main Scenario:

1. Firstly, the Admin has to login.
2. The Admin can add and delete the staff responsibility.

Failure Scenario:

1. The Admin have failed to click submit button
2. Failure may occur through server problem
3. Show the Error if the admin do not provide the correct information.

Use Case Name: Inventory Stock

Scope:

The Scope of given Inventory stock is only admin can update inventory database system. The admin can view the report of total information of stock in restaurant.

Primary Actor: Admin

Pre-Conditions:

The Admin must have Login

Main Scenario:

1. Firstly, the Admin has to login.
2. The Admin can add in inventory database.
3. View the total Stock

Failure Scenario:

1. The Admin have failed to click submit button
2. Failure may occur through server problem
3. Show the Error if the admin do not provide the correct information.

Use Case Name: Generate Report

Scope:

The Scope of Generate Report is only admin can View the whole year monthly and daily reports for the Restaurant Owner. The admin can view the report of total information about income from restaurant.

Primary Actor:Admin

Pre-Conditions:

The Admin must have Login

Main Scenario:

1. Firstly, the Admin has to login.
2. The Admin can View the Monthly record monthly yearly and daily
3. View the total income

Failure Scenario:

1. The Admin have failed to click submit button
2. Failure may occur through server problem
3. Show the Error if the admin do not provide the correct information.

Use Case Name: Logout

Scope:

The Scope of Logout is all actors of restaurant management who login can must be logout.

Primary Actor:Admin, staff and customer

Pre-Conditions:

The All actors must have login

Main Scenario:

1. Firstly, the all actors has to login.
2. The all actors can be logout after they done their work.

Failure Scenario:

1. The Admin have failed to click submit button
2. Failure may occur through server problem
3. Show the Error if the admin do not provide the correct information.

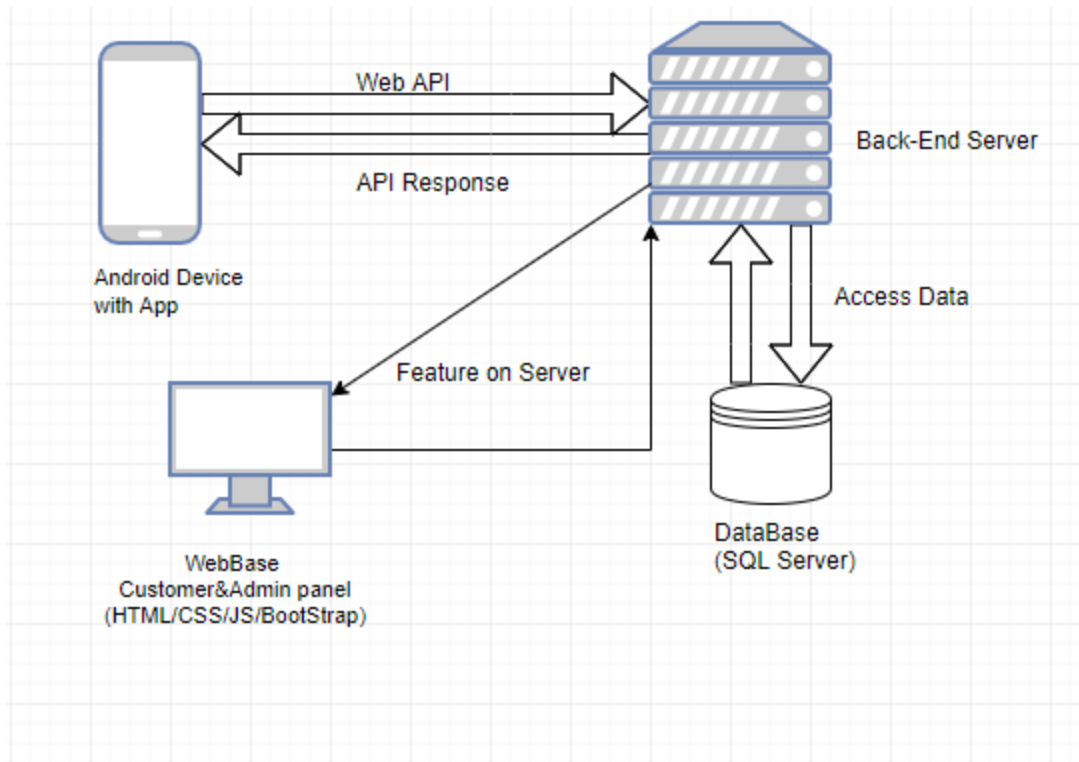
Chapter 4

System Design

Chapter 4: System Design

4.1. Architecture Diagram

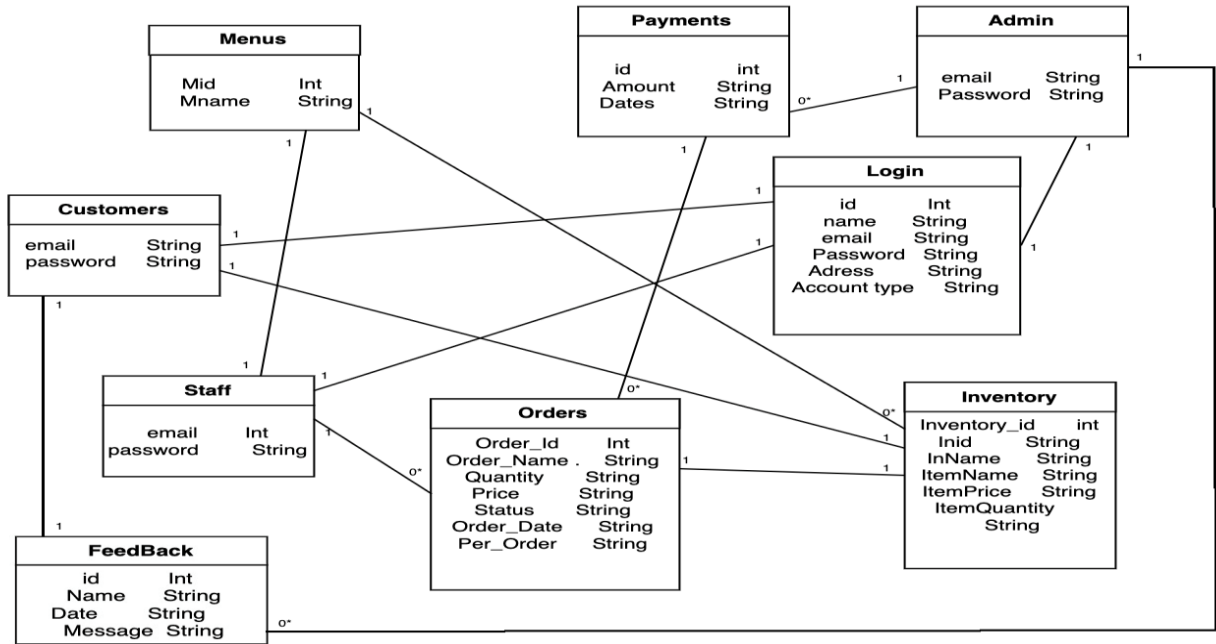
Architecture Diagram below in fig4.1.1



Fig(4.1.1)

4.2. Domain Model

Domain model is below in fig4.2.1



Fig(4.2.1)

4.3. Entity Relationship Diagram

Entity relationship diagram in below fig 4.3.1

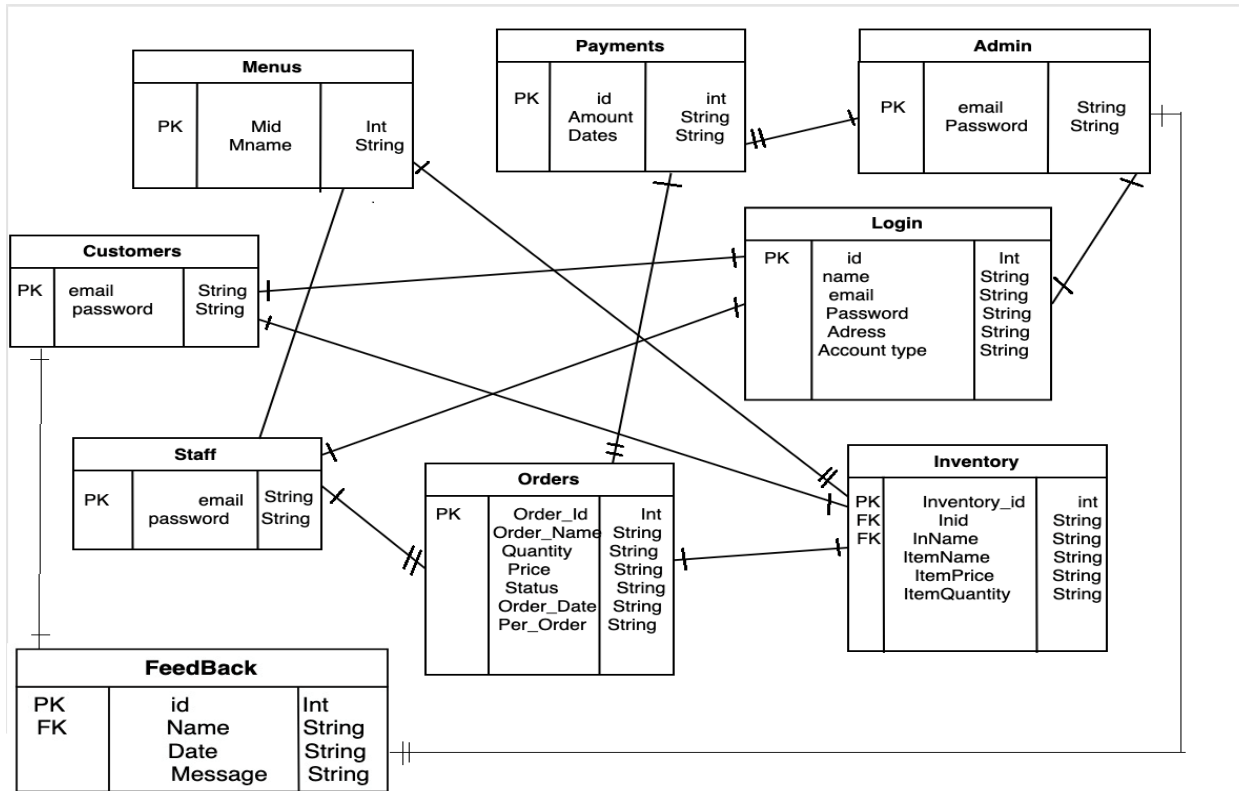


Fig (4,3,1)

4.4. Class Diagram

Class Diagram in below fig 4.4.1

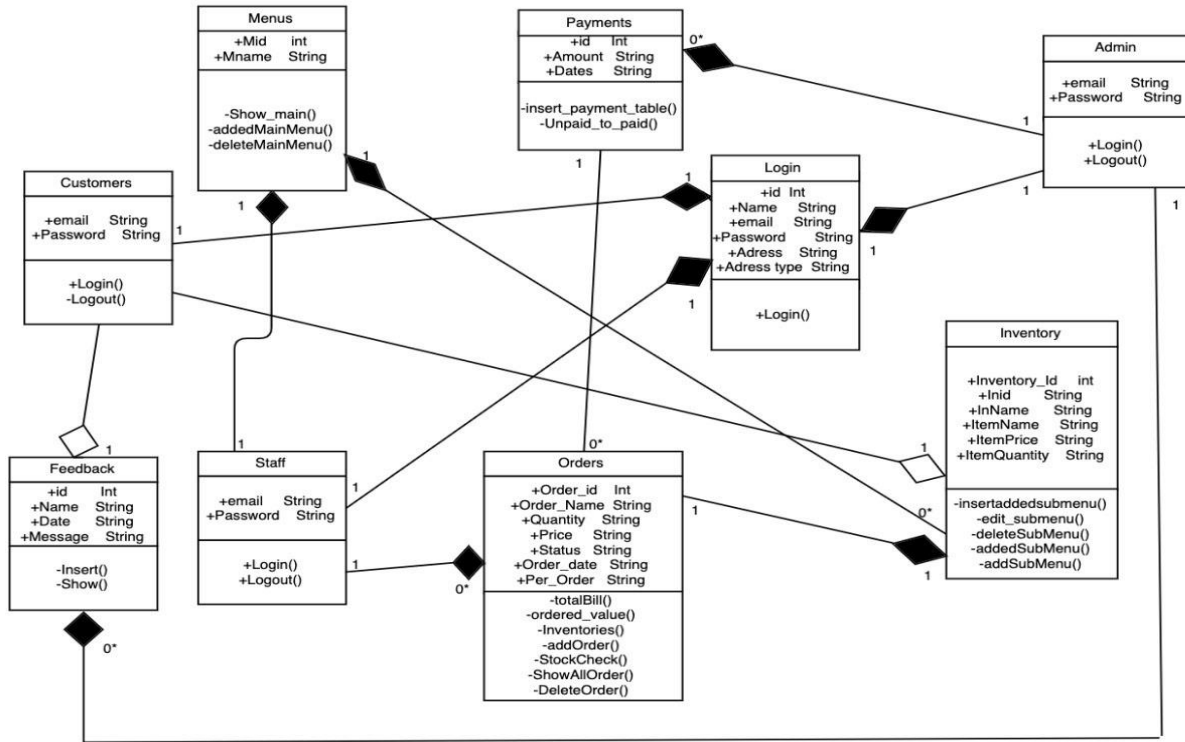


Fig (4.4.1)

4.5. Sequence / Collaboration Diagram

Registration

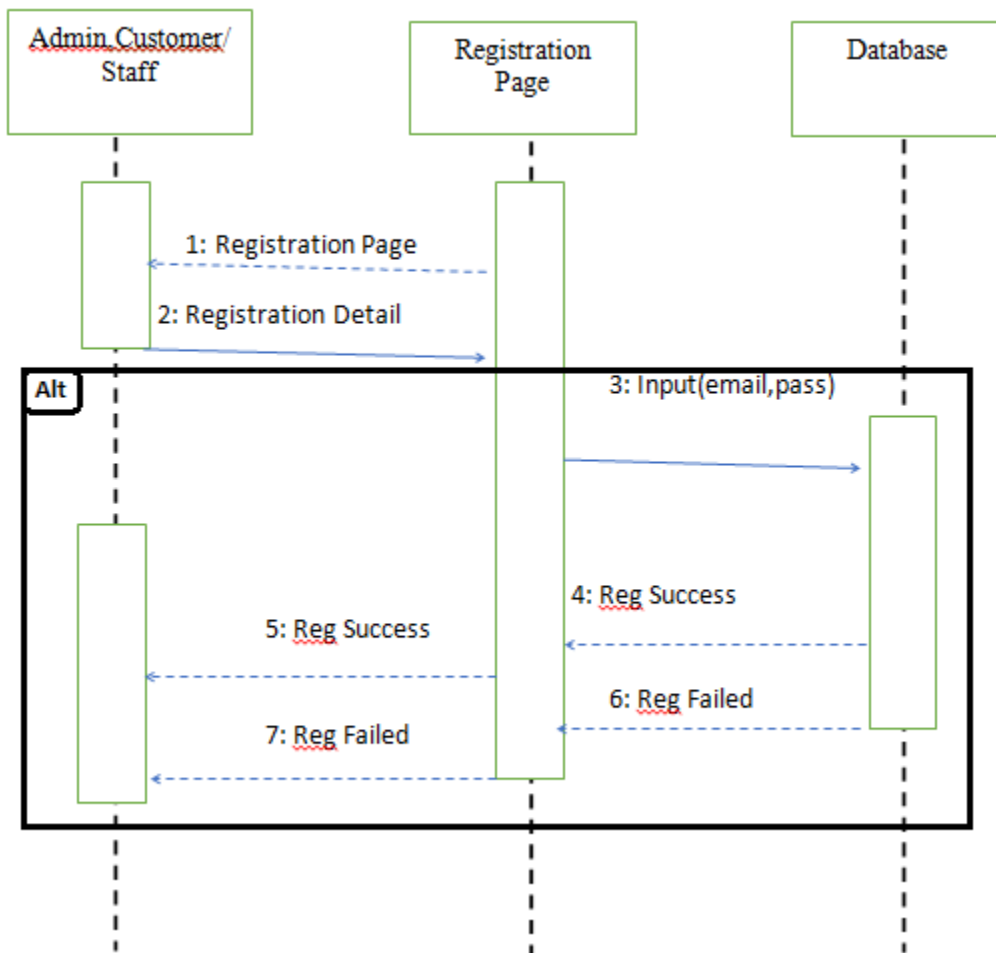


Fig (4.5.1)

Login

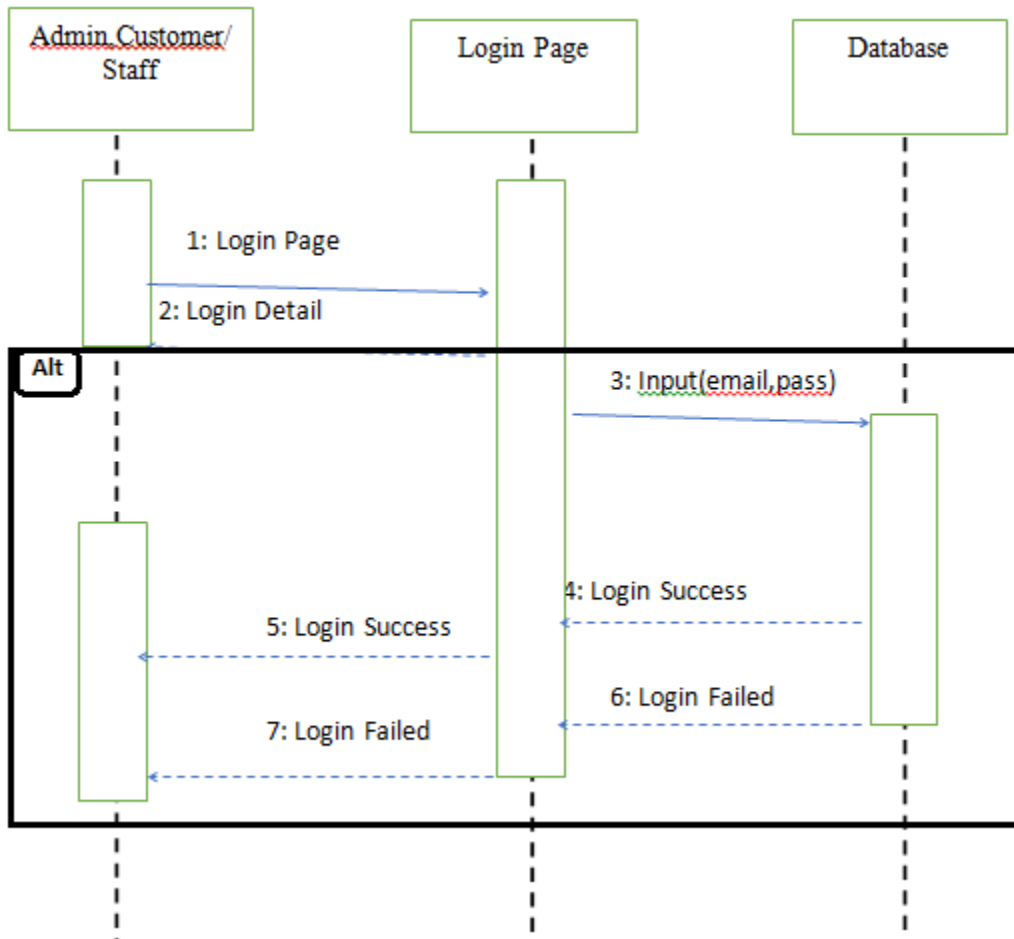
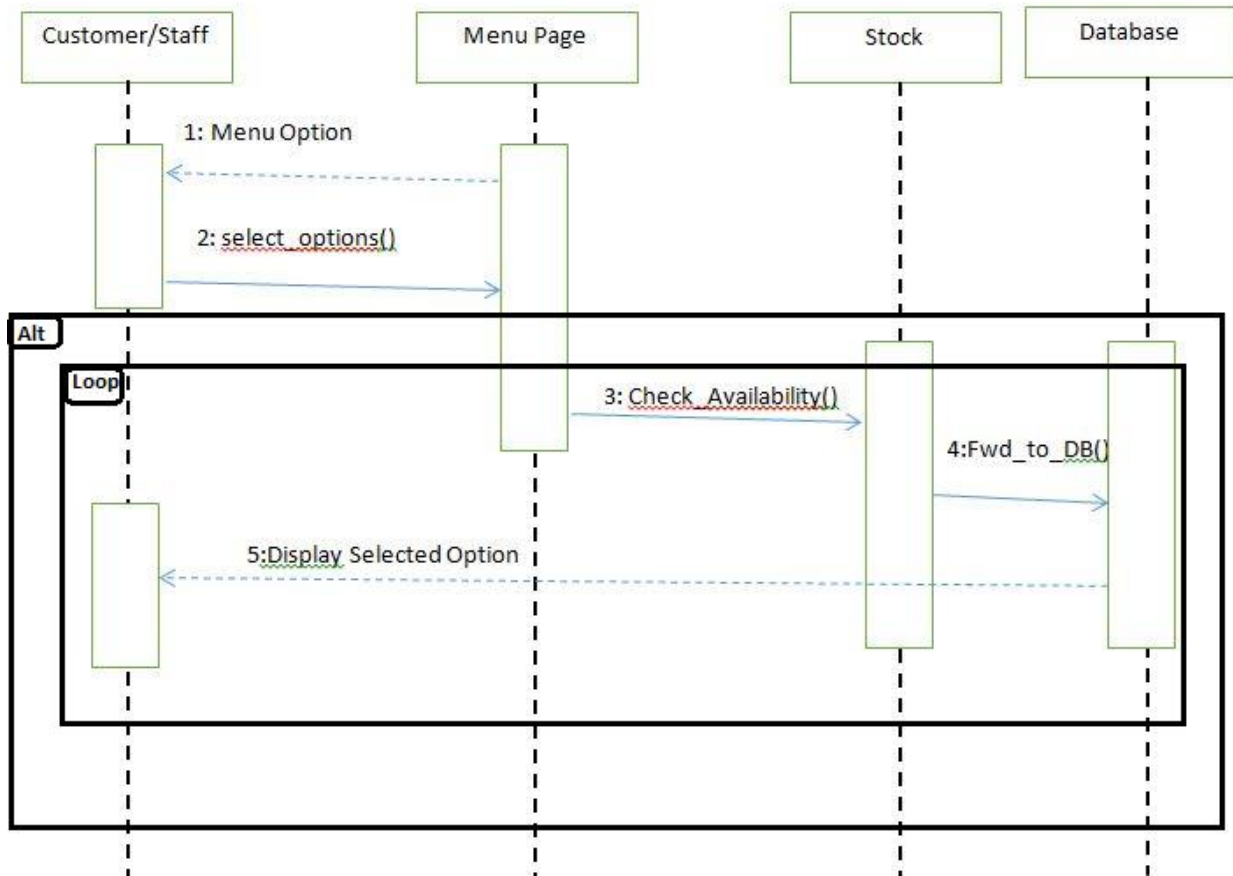


Fig (4.5.2)

Menu Options



Fig(4.5.3)

Bill

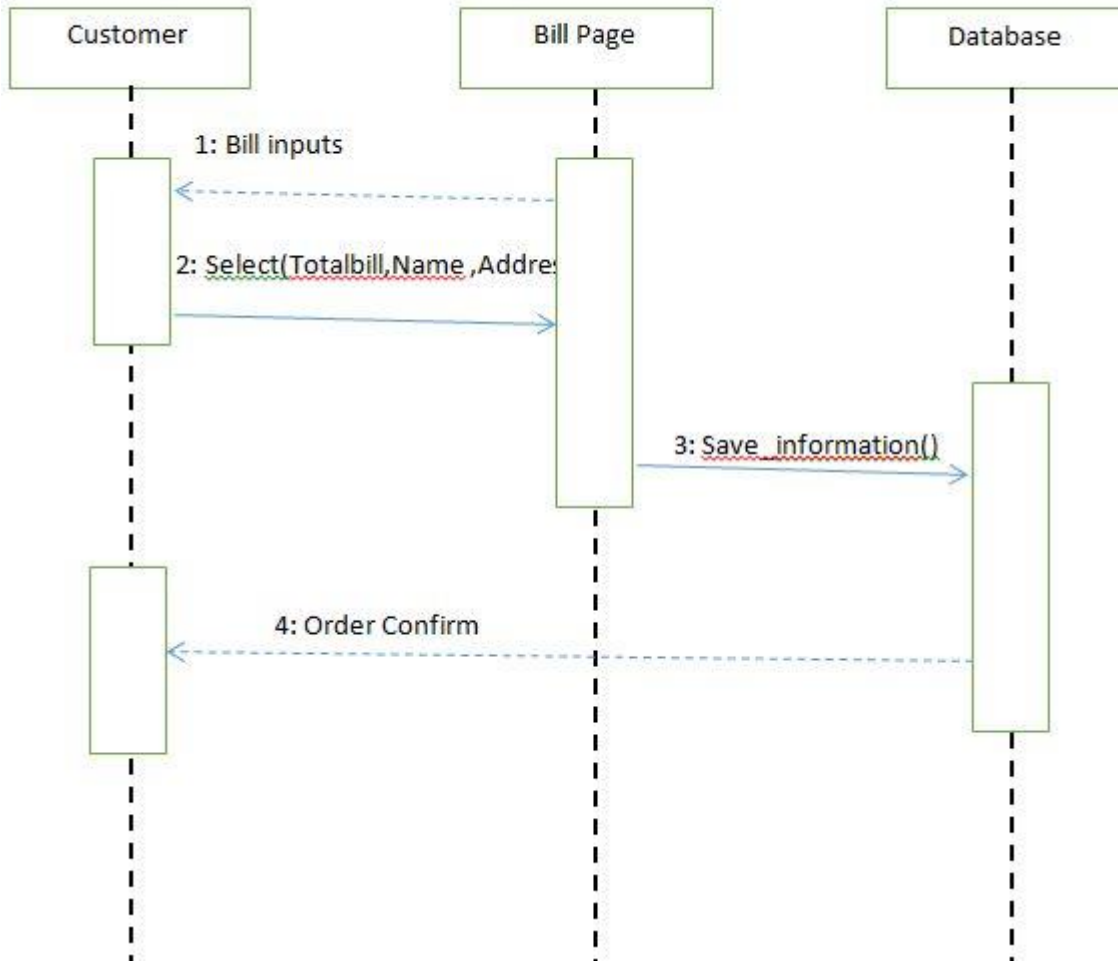


Fig (4.5.4)

Give Feedback

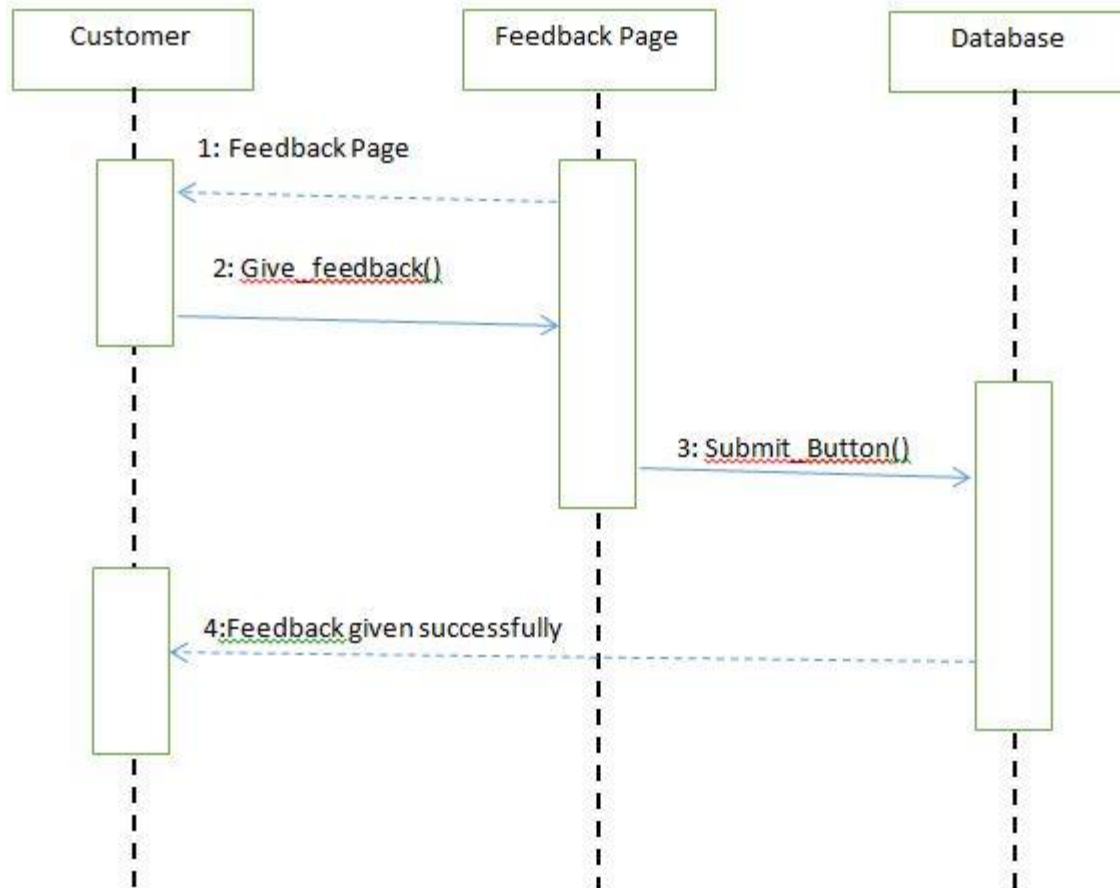


Fig (4.5.5)

Update Menu:

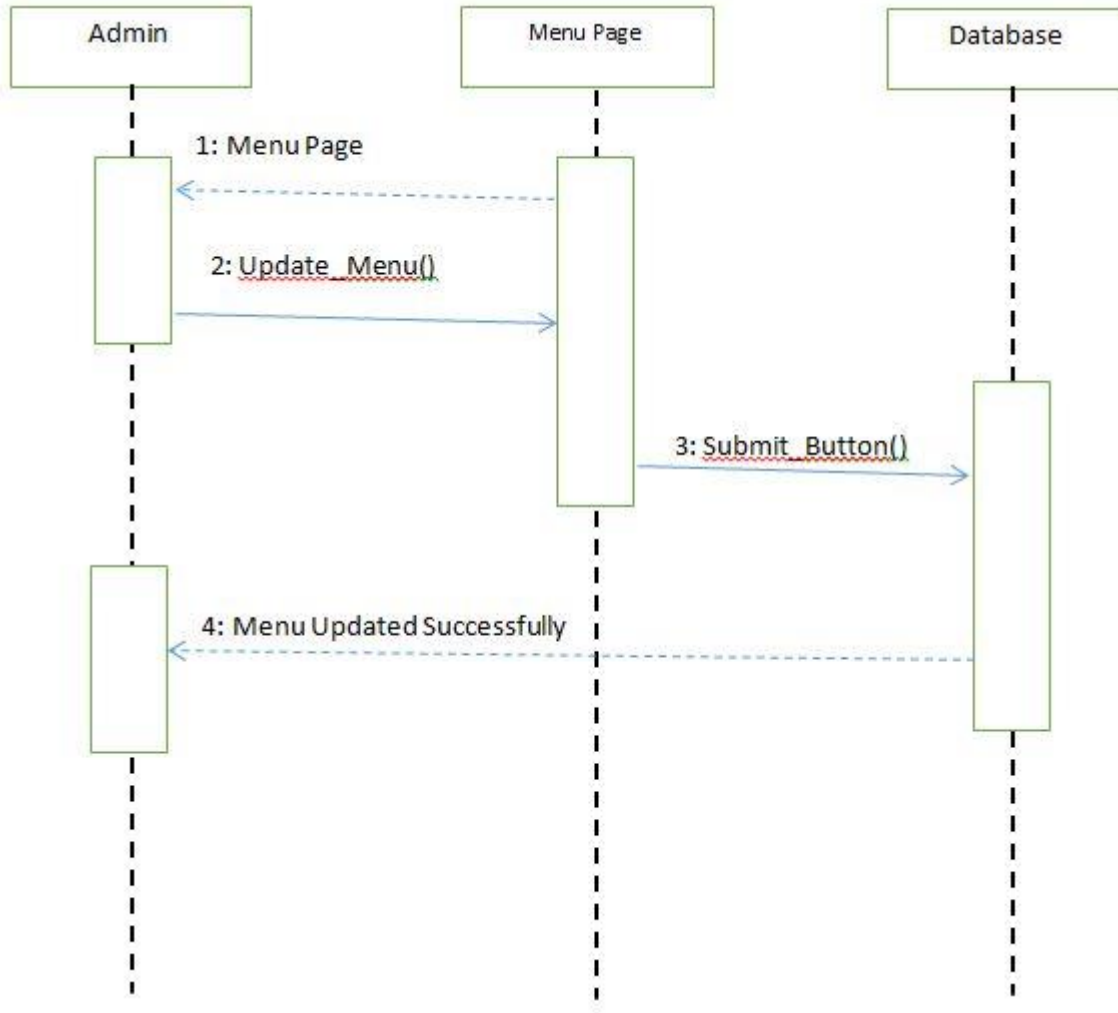


Fig (4.5.6)

Insert Menu

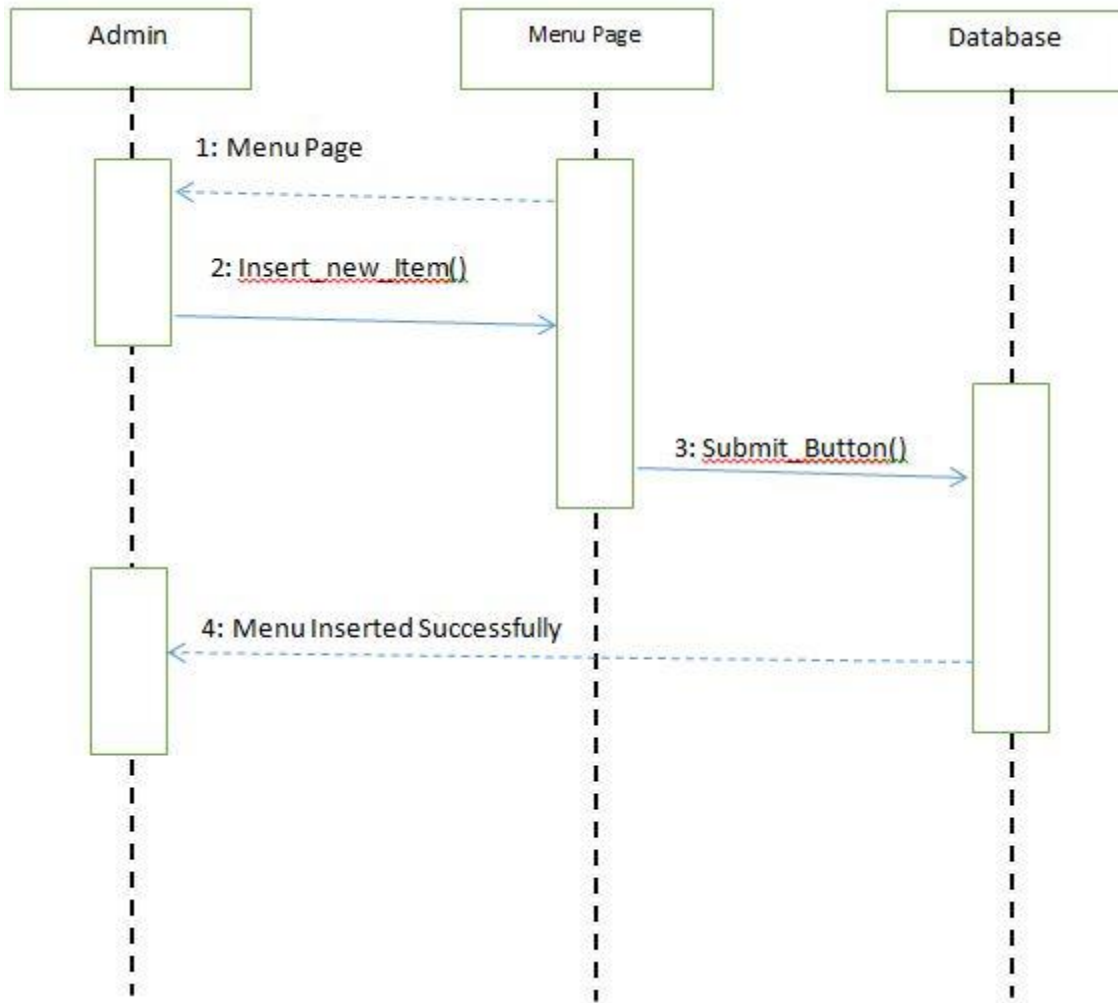


Fig (4.5.7)

Delete Menu

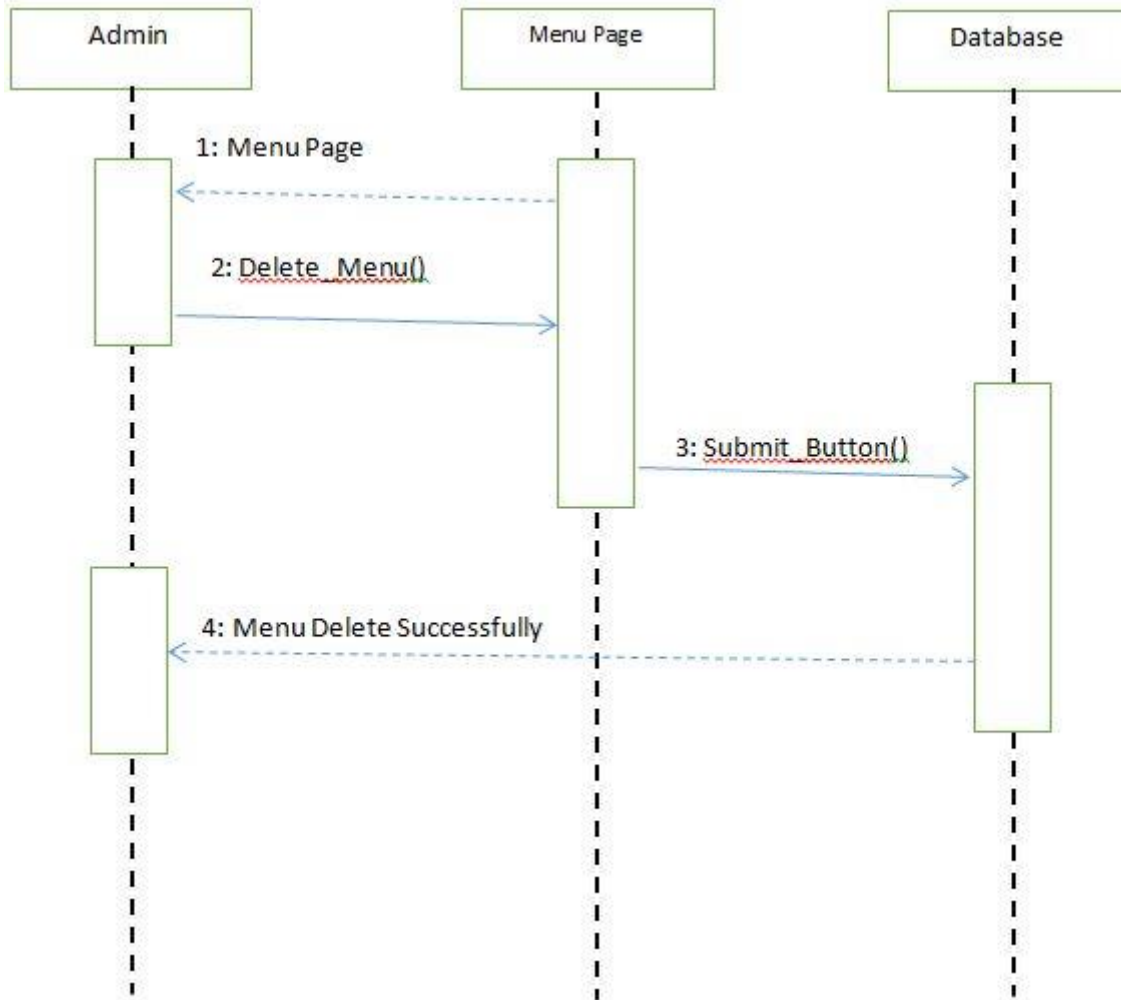


Fig (4.5.8)

SALE REPORT

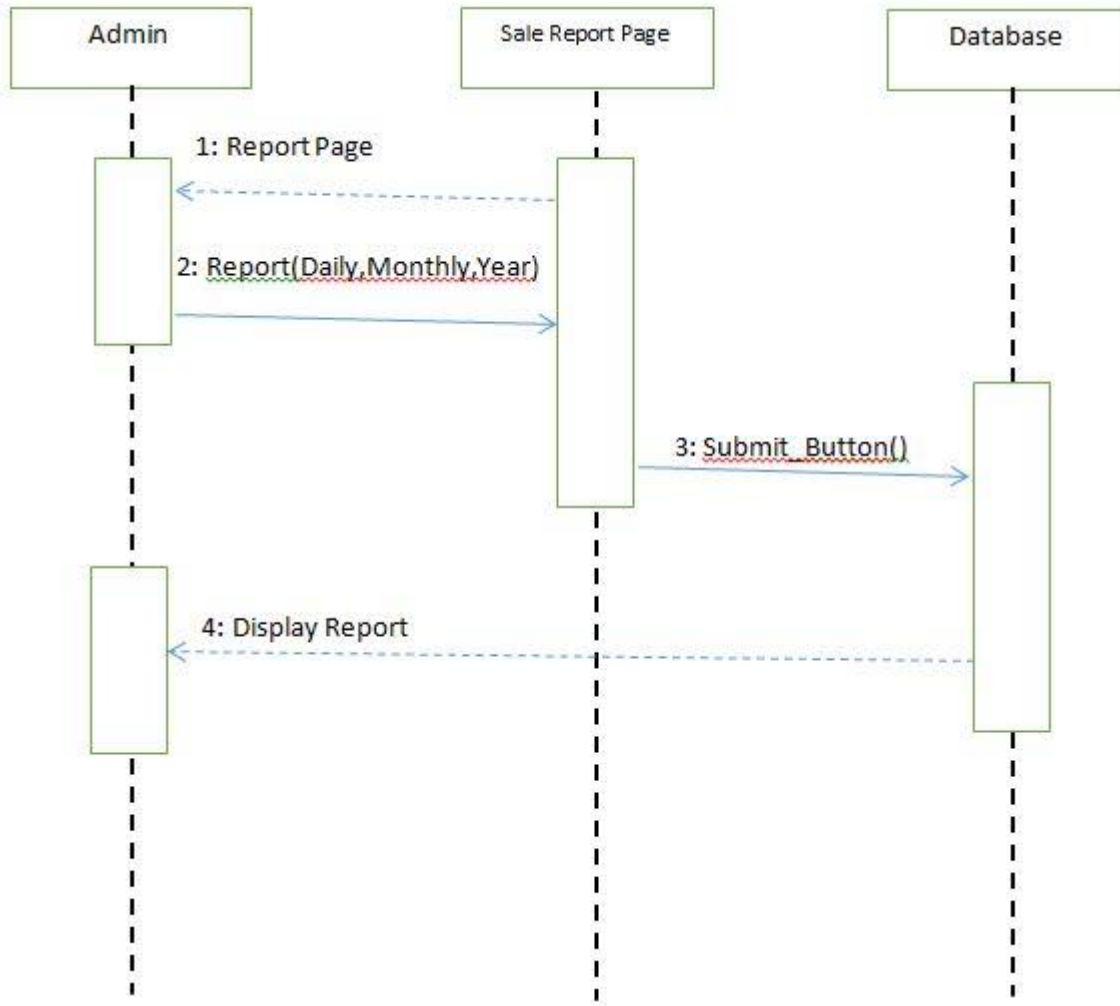


Fig (4.5.9)

LOGOUT

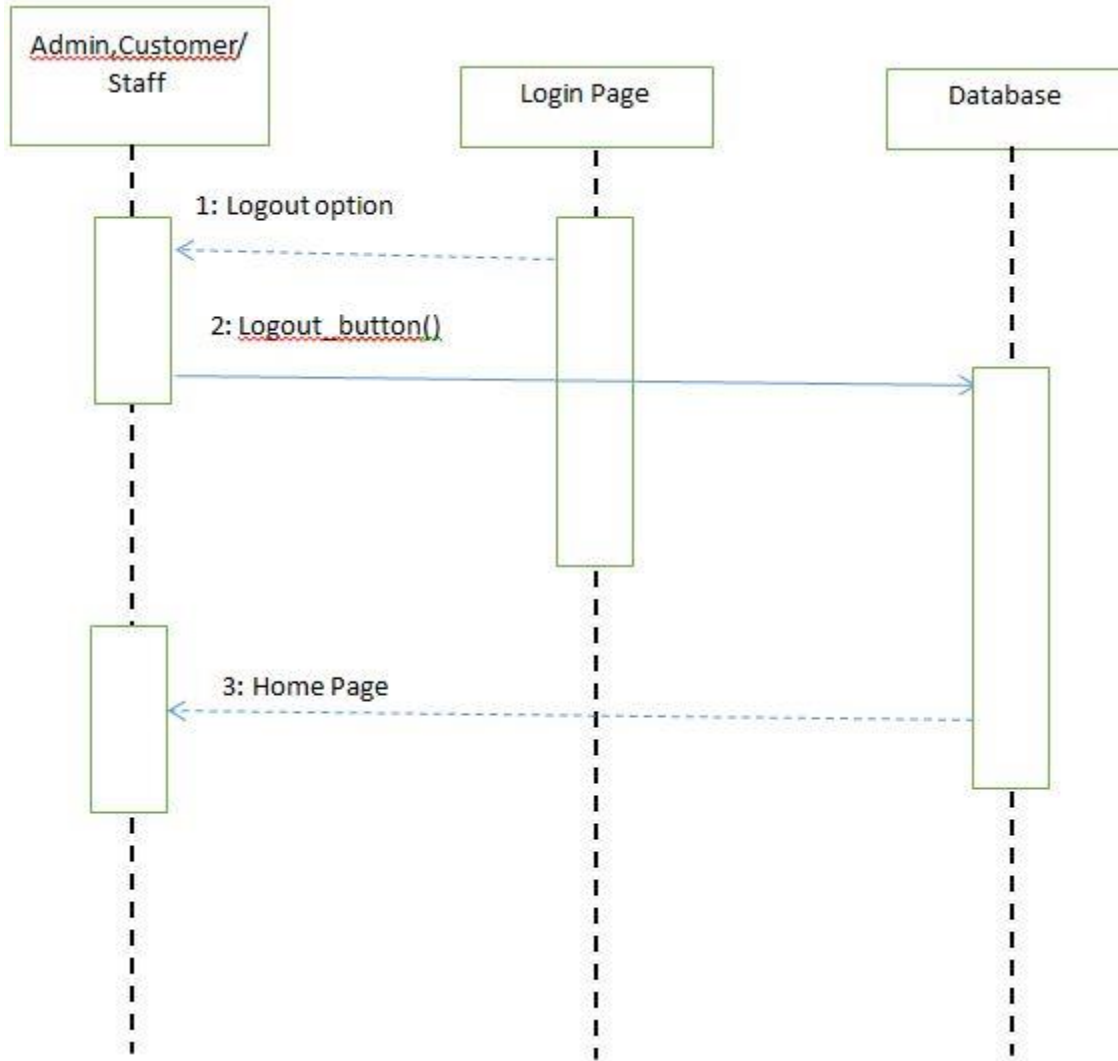


Fig (4.5.10)

4.6. Operation Contract

Operation:-Register();

Cross Reference

Use Case:-Registration

Scenario:-Registration

Pre-Condition:-

none

Post-Condition:-

user.name=name;

user.email=email;

user.password=password;

Operation:-login();

Cross Reference

Use Case:-login

Scenario:-login

Pre-Condition:-

have register on website

Post-Condition:-

user.email=email;

user.password=password;

Operation:-TakeOrder();

Cross Reference

Use Case:-take order

Scenario:- take order

Pre-Condition:-

1. Login on website
2. The customer selects 'Order' from the menu
3. Check stock availabilit

Post-Condition:-

Order.items=item;

Operation:-giveorder();

Cross Reference

Use Case:-give order

Scenario:-give order

Pre-Condition:-

1. Must Login on website
2. The customer selects 'Order' from the menu
3. Check stock availaility

Post-Condition:-

1. Order.item=item;

Operation:-Bill();

Cross Reference

Use Case:-bill

Scenario:-bill

Pre-Condition:-

1. Must Login on website
2. Must select order from menu.

Post-Condition:-

 bill.price=price;
 bill.tax=totalGST;
bill.Total=grandTotal;

Operation:-GenerateReport();

Cross Reference

Use Case:-Generate Report

Scenario:-Generate Report

Pre-Condition:-

1. Must Login on website
2. Must have sale record

Post-Condition:-

Report.totalIncome=totalIncome;

Operation:-UpdateMenu();

Cross Reference

Use Case:- Update Menu

Scenario:- Update Menu

Pre-Condition:-

1. Admin Must be login.
2. Item must be available

Post-Condition:-

Manu.item.ID=item.Price;

Operation:-Stock();

Cross Reference

Use Case:- Stock

Scenario:- Stock

Pre-Condition:-

1. Admin login
2. Item Available

Post-Condition:-

Stock.check.id=stock.qty;

Operation:-FeedBack();

Cross Reference

Use Case:- Feedback

Scenario:- Feedback

Pre-Condition:-

1. Login on website
2. Order must be taken

Post-Condition:-

feedback.item=item.discription;

Operation:-LogOut();

Cross Reference

Use Case:- Logout

Scenario:- Logout

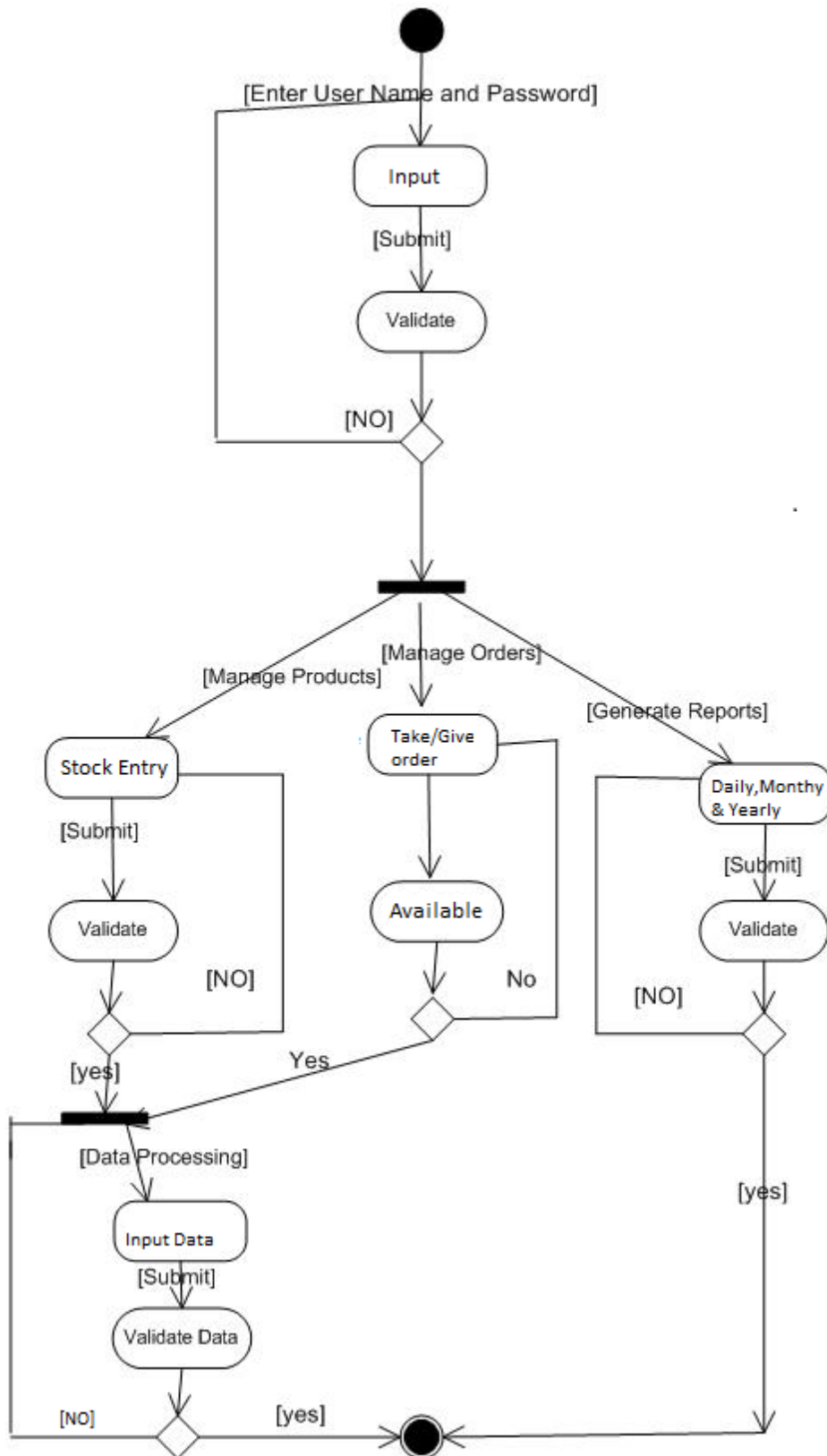
Pre-Condition:-

User must be login

Post-Condition:-

User.id=logout();

4.7. Activity Diagram



State Transition Diagram

State diagram in below Fig 4.7.1

Registration:

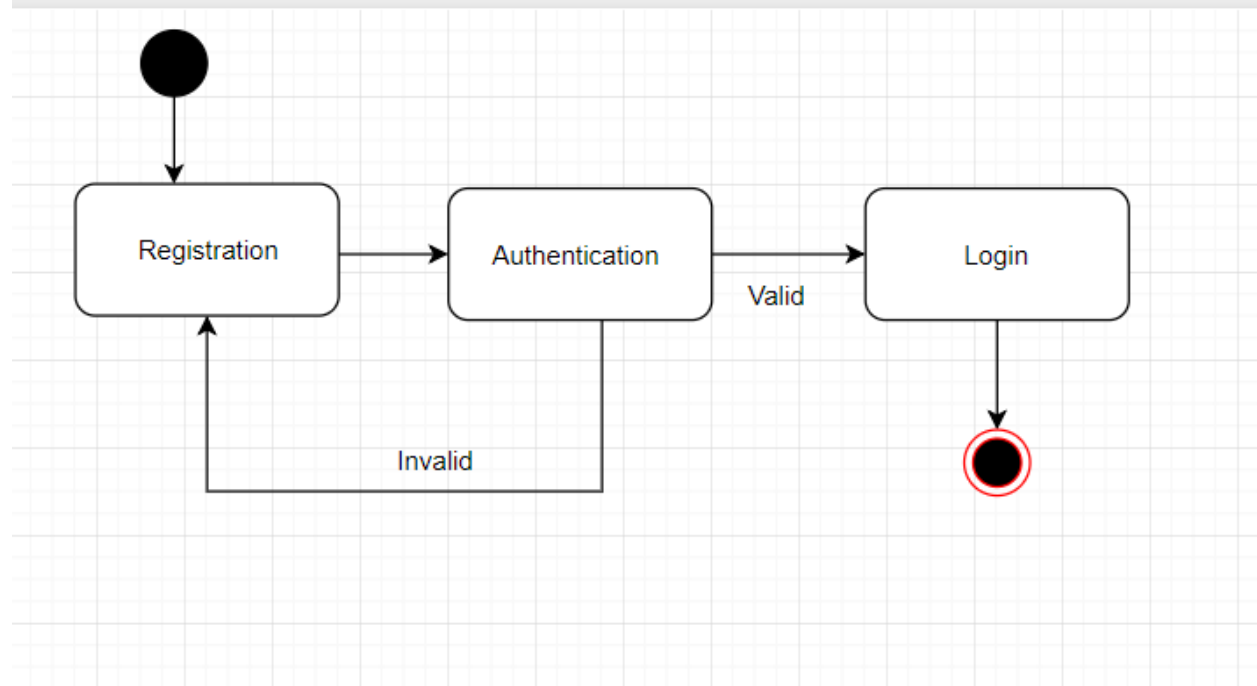


Fig 4.7.1

Login:

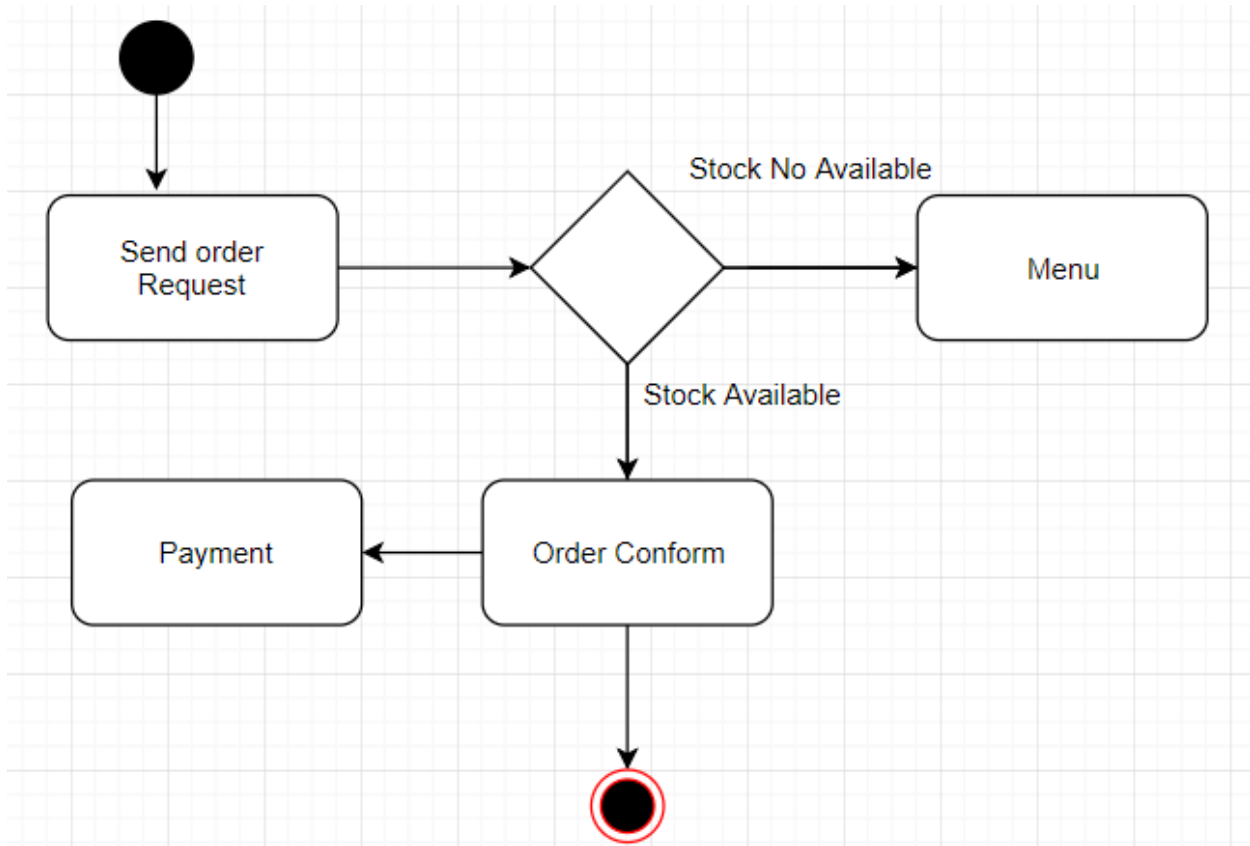


Fig 4.7.2

Payment

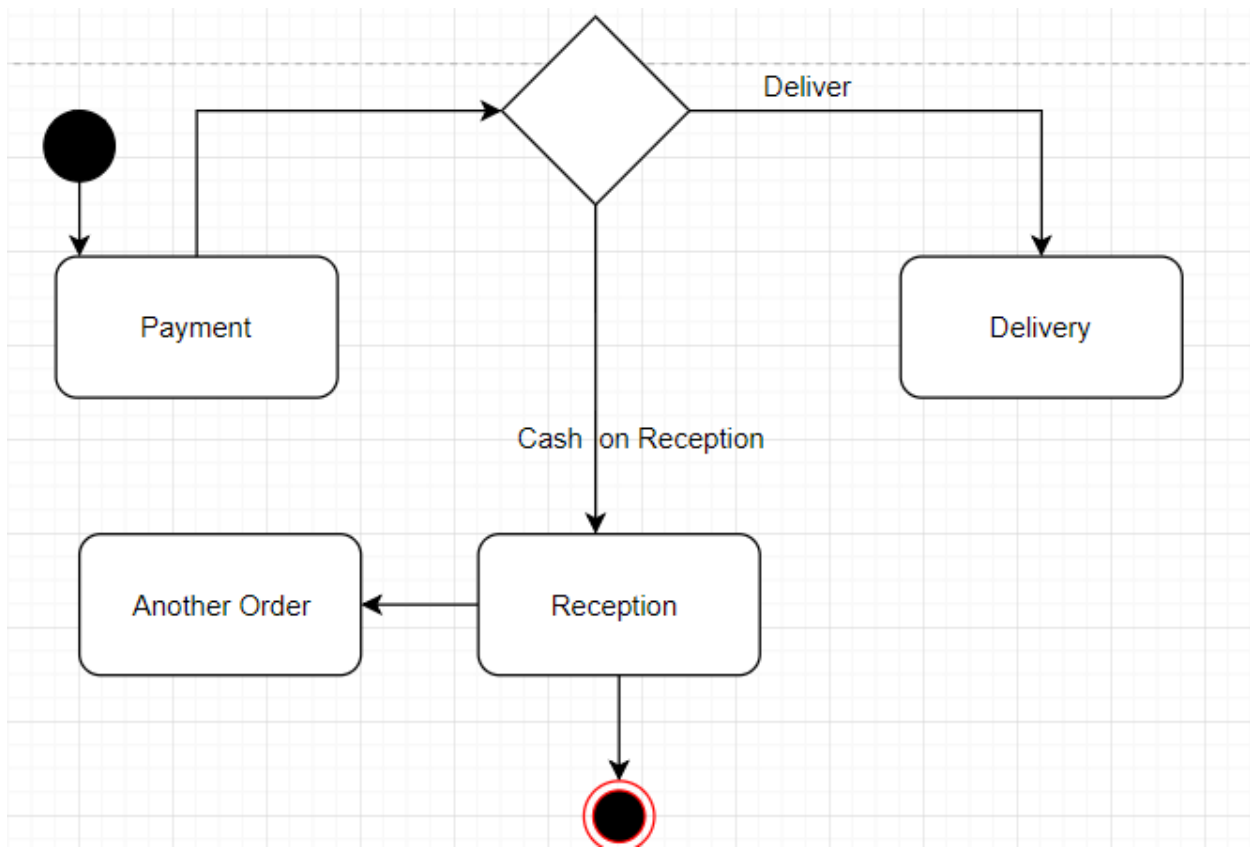


Fig 4.7.3

Item

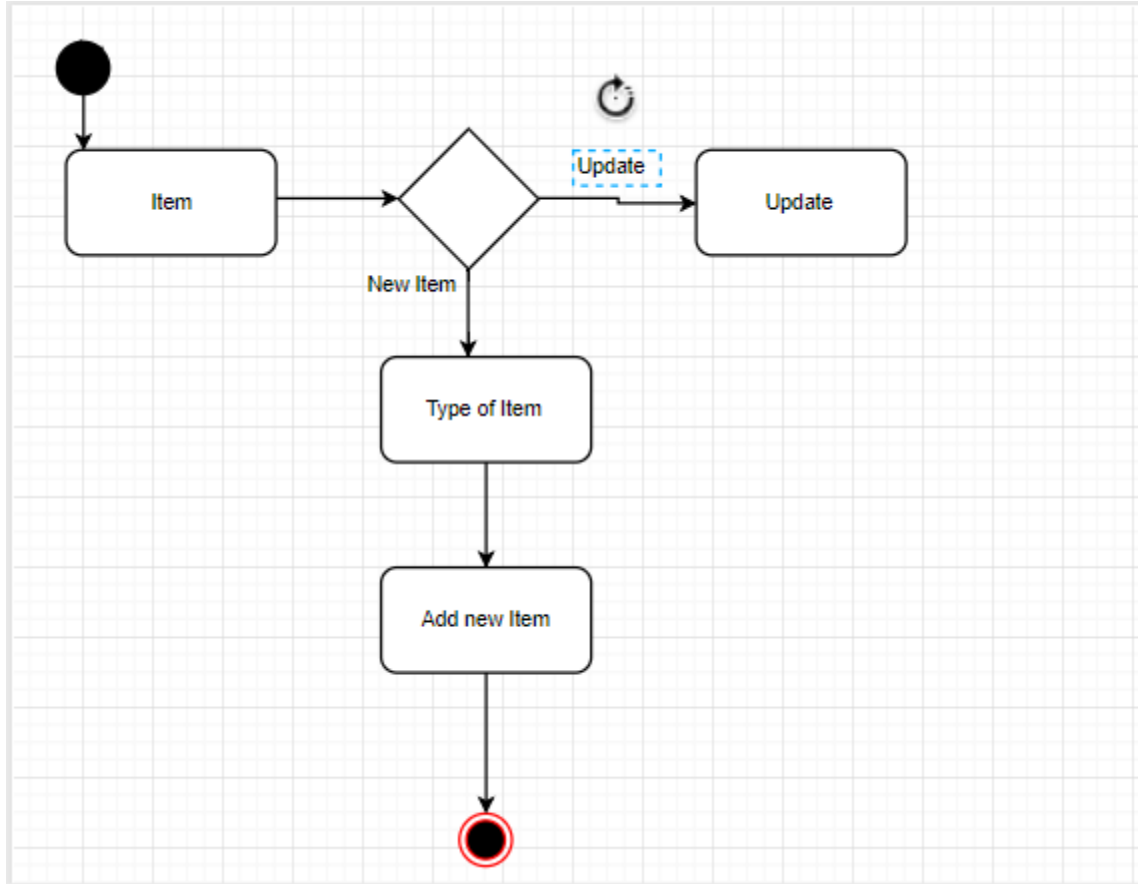


Fig 4.7.4

Delivery Conformation

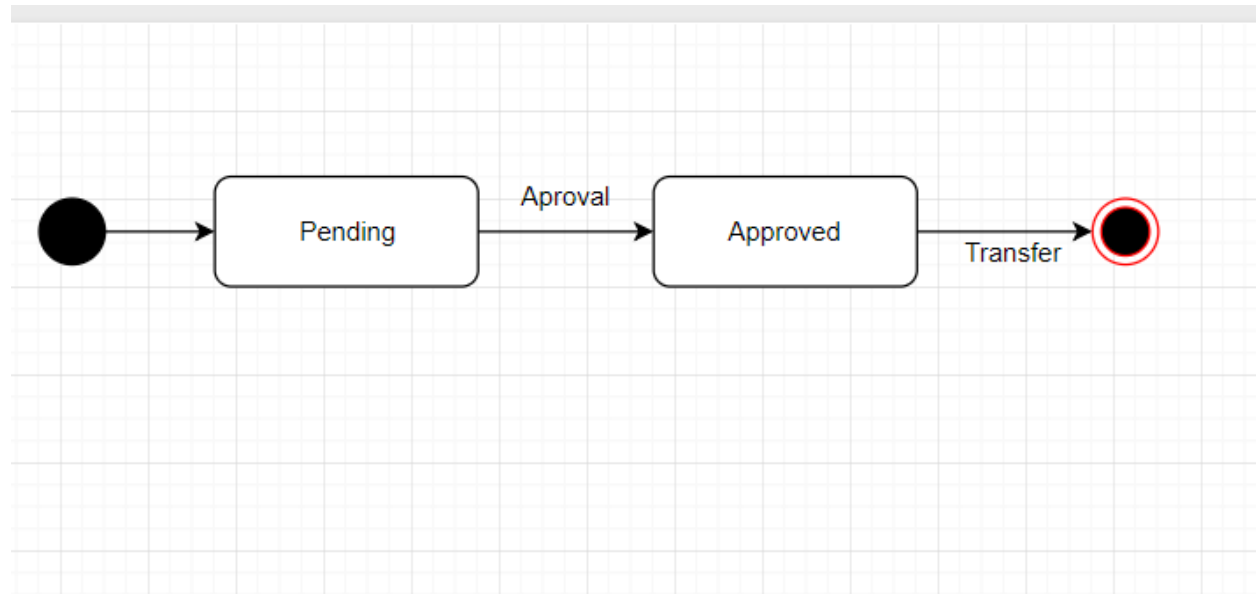


Fig 4.7.5

4.8. Deployment Diagram

Deployment Diagram is below in fig 4.8.1

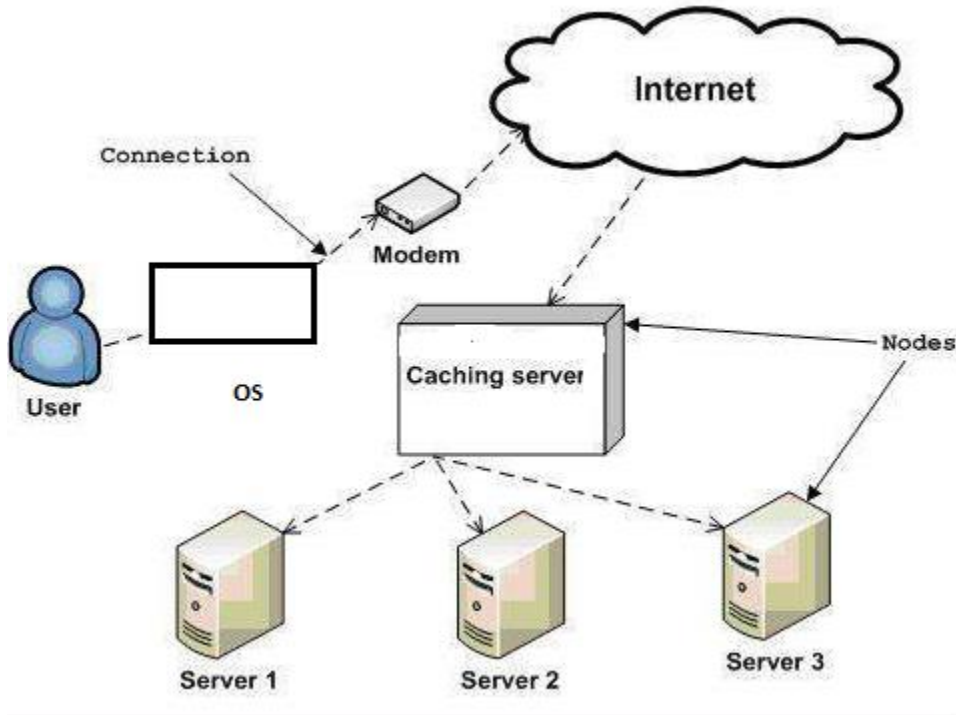
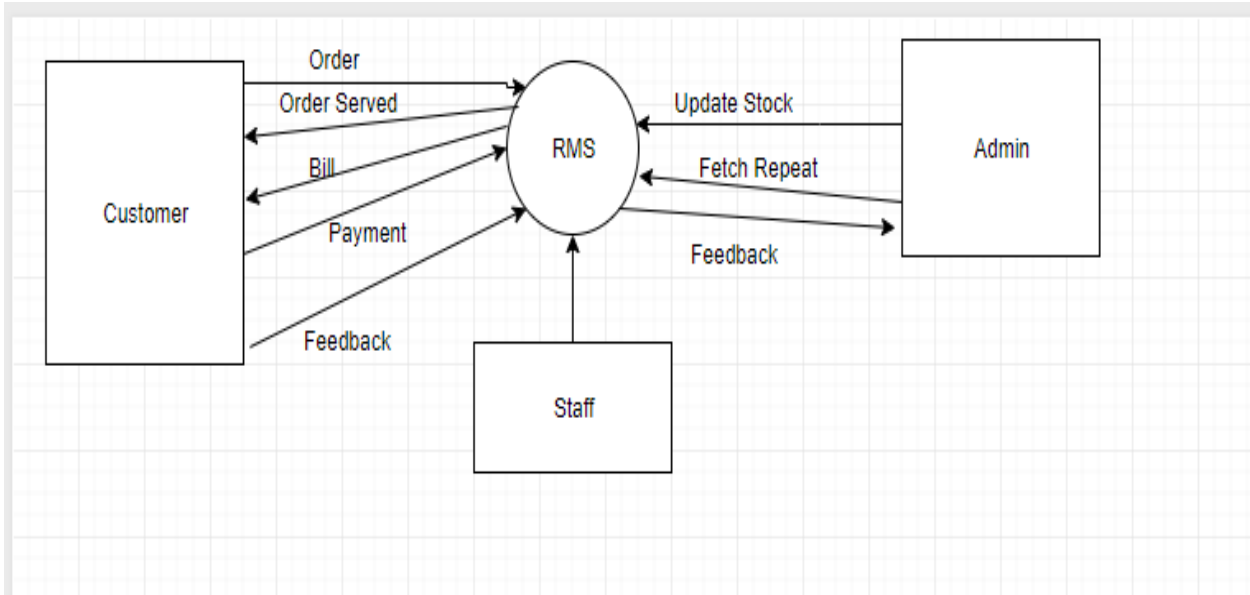


Fig (4.8.1)

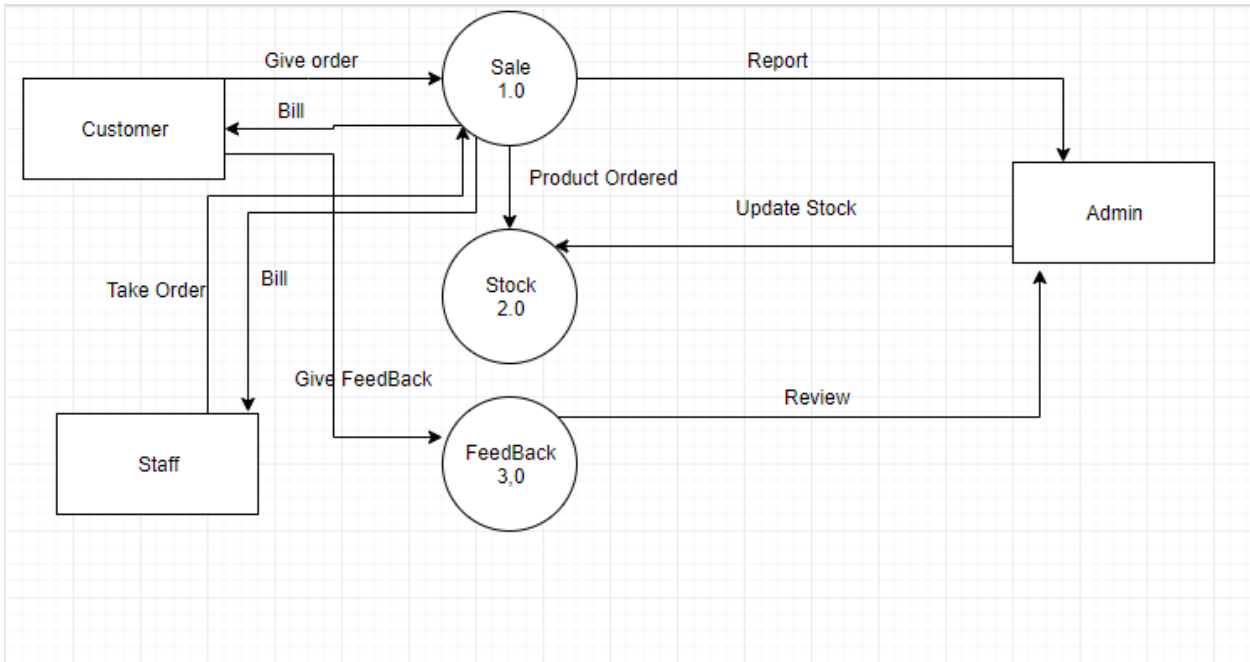
4.9. Data Flow diagram

Context Level



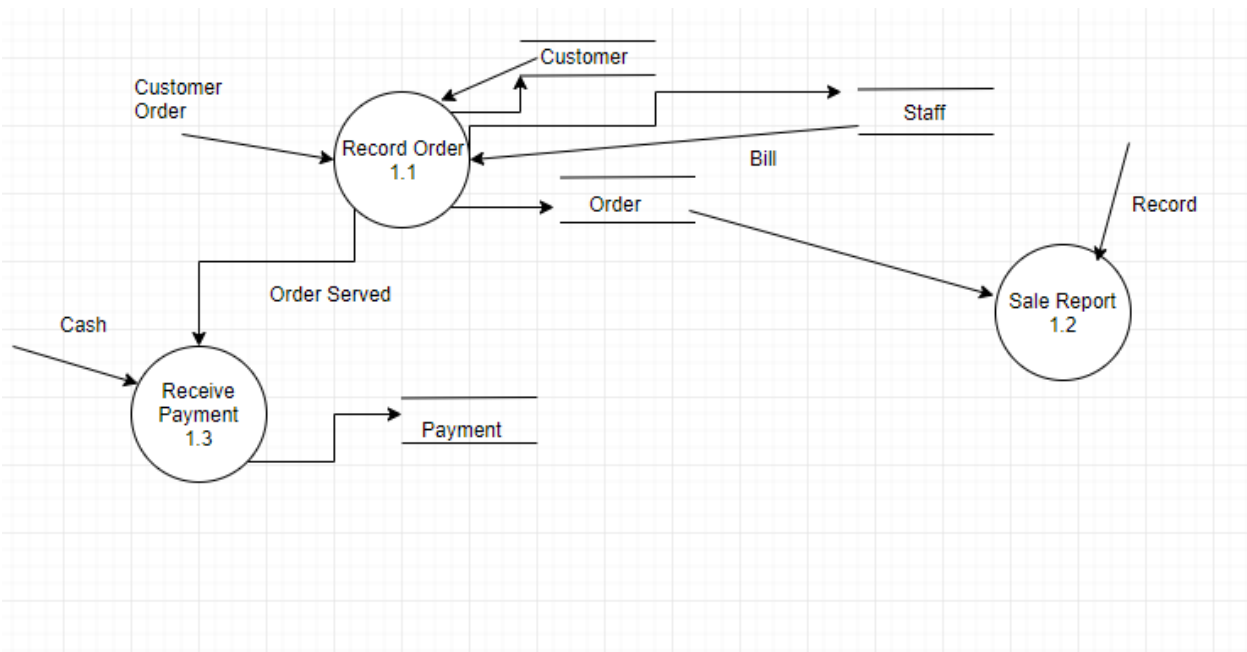
(Fig 4.9.1)

Level 0



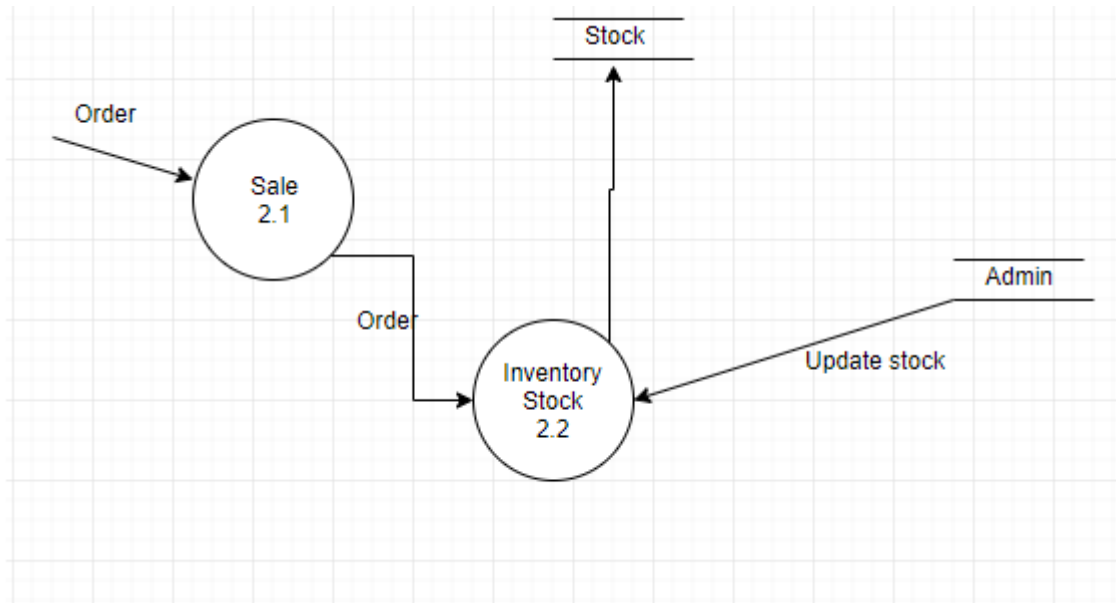
(Fig 4.9.2)

Level 1



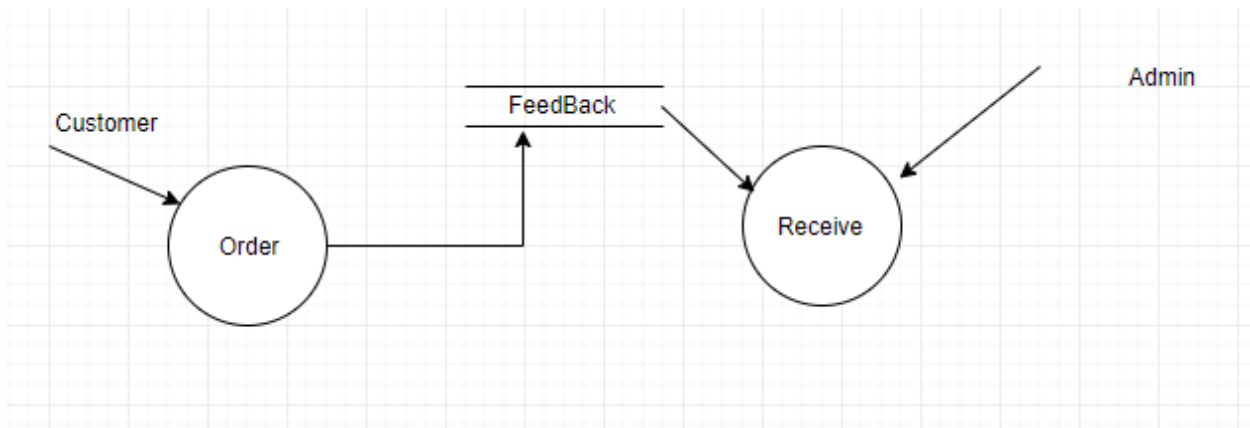
(Fig 4.9.3)

Level 1



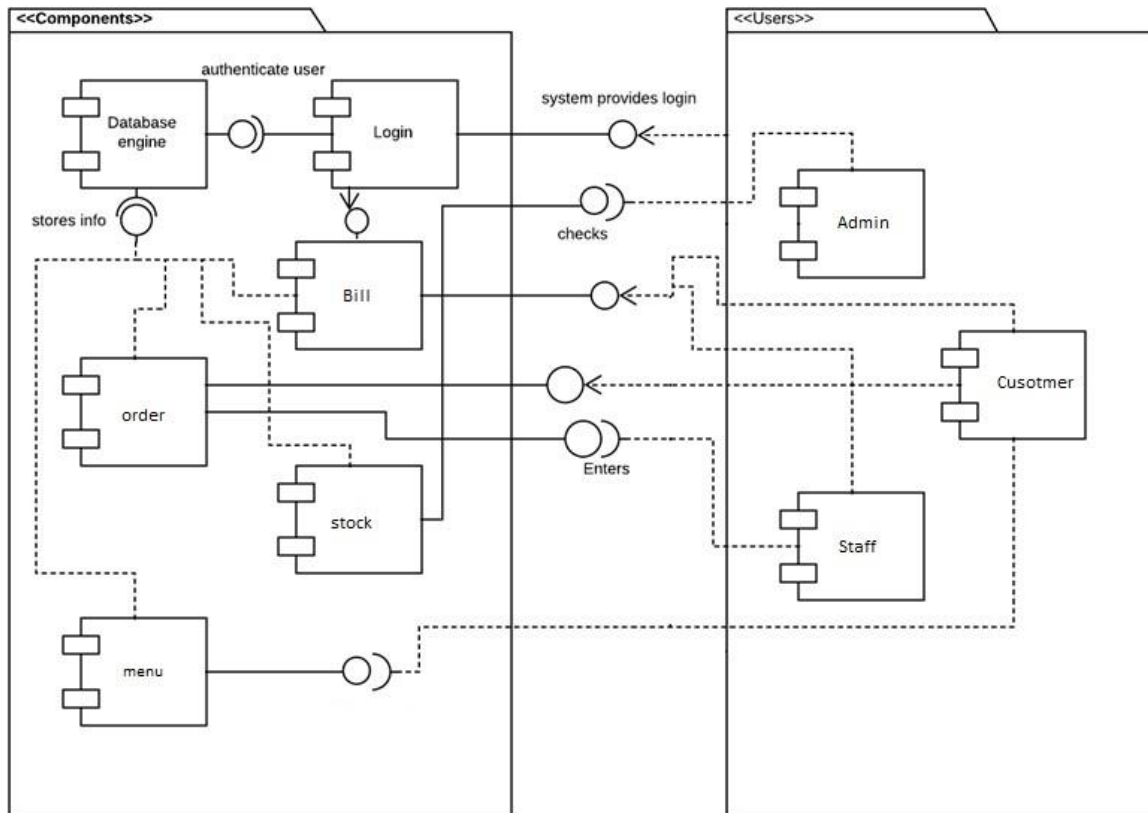
(Fig 4.9.4)

Level 1



(Fig 4.9.5)

4.10. Component Diagram



(Fig 4.10.1)

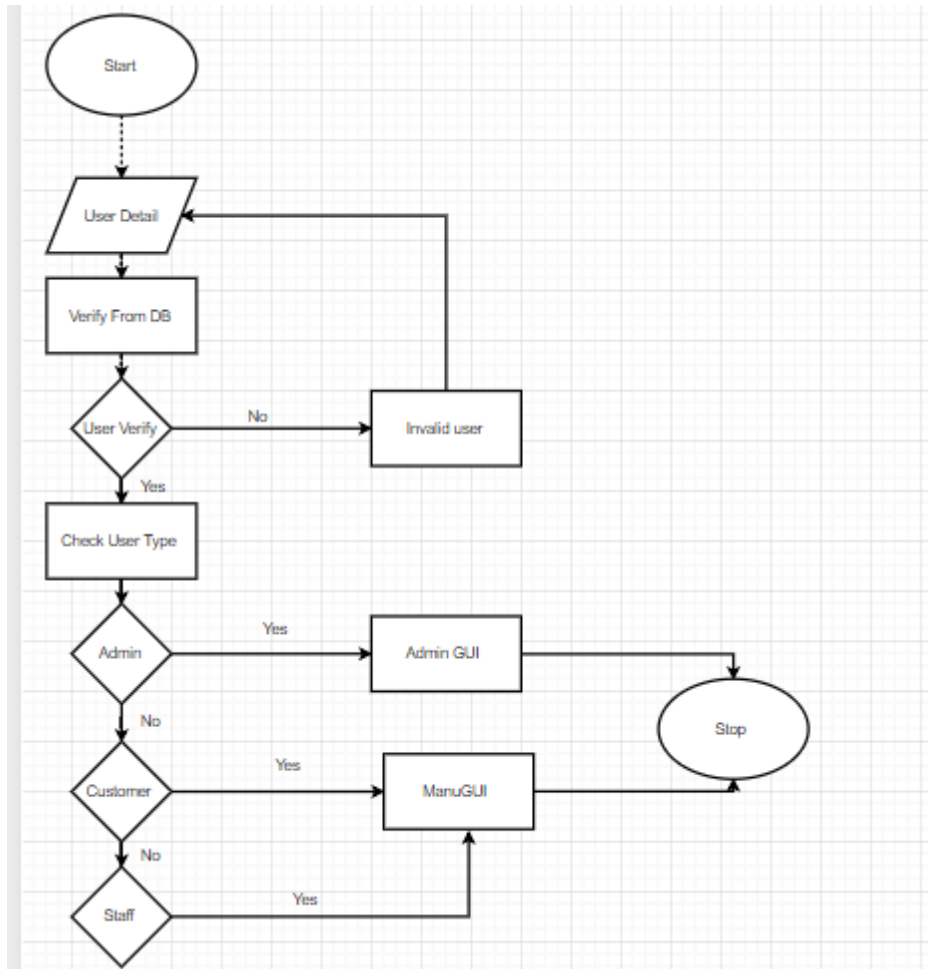
Chapter 5

Implementation

Chapter 5: Implementation

5.1. Important Flow Control/Pseudo codes

This project is used an agile methodology classed extreme Programming. This type of Methodology Utilized an alterative Process involving customer communication and feedback



5.2. Components, Libraries, Web Services and stubs

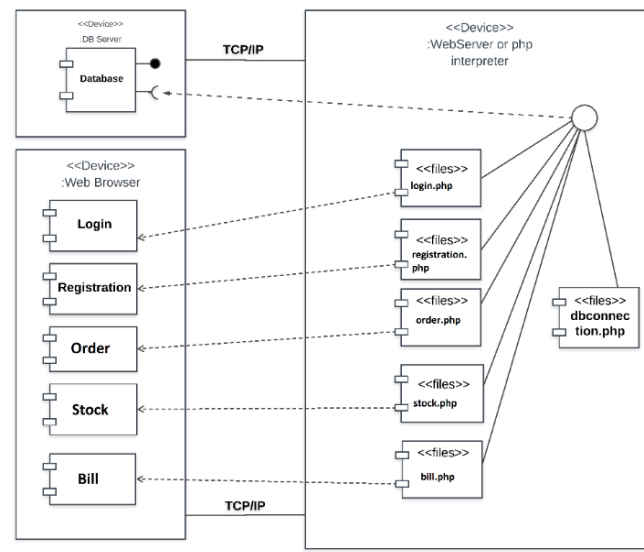
“The Web and Android service is a software system, whose public interfaces and bindings are defined and described using HTML & XML. Its definition can be discovered by other software systems. These systems may then interact with the Web service in a manner prescribed by its definition, using HTML & XML based messages conveyed by internet protocols.”

Libraries:

- Laravel Composer
- RDBMS
- JAVA
- PHP library
- Bootstrap
- JS

Stubs:

A method stub or simply stub in software development is a piece of code used to stand in for some other programming functionality. A stub may simulate the behavior of existing code (such as a procedure on a remote machine, such methods are often called mocks) or be a temporary substitute for yet-to-be-developed code.

5.3. Deployment Environment

5.4. Tools and Techniques

Tools:

Laravel, XAMPP,MYSQL, Sublime Text,

Technique:

Agile Method

5.5. Best Practices / Coding Standards

Coding standards should be best code should be simple and divided into different chunks that will make ease to modify code. Factorization of code is good habit it make your Application & website light and easily access able.

5.6 Version Control

Task #	Deliverable	Activity #	Duration (# of Days)	Responsible Team Member(s) & Role(s)
1	Web Designing & Coding	1	1.5 Months	2- Front End Designing & coding, Abdul Ahad, Haris Ahsan, Waleed Muaz
2	Web Database	2	1 Month	-Manage whole Database, Nouman Abdul Ahad, Haris Ahsan ,Waleed Muaz
3	Application Designing & Coding	3	1 Month	1-App Front end Designing & coding, Abdul Ahad, Haris Ahsan ,Waleed Muaz

Chapter 6

Testing and Evaluation

Chapter 6: Testing and Evaluation

In this chapter we have added the results of the testing and evaluation of our system. This chapter discusses the brief testing procedures that have been carried out throughout the testing process to test the system functionalities. Testing is performed on localhost and by using different manual and automated tools. Anyhow errors and variation may affect the actual results.

6.1. Use Case Testing

Case ID	Objective	Steps	Test Data	Expected Result
TC101	To verify valid login details	1. Launch restaurant management application using URL as in test data. 2. Login to the application using username and password as in test data.	User: {haris} Password: {Haris5122}	User should login to the application.
TC102	To Update the main menu	1. Must login. 2. Click add new menu on dashboard.. 3. Click add Menu. 4. Enter tthe name of the Menu.	Menu Name:: Karahi	System Should update the main menu list.
TC103	Enter new entry in the sub menu	1. Click the main menu list for you want enter. 2. Click on add More menu name.. 3. Enter the sub item name price	Sub Item Name: Chicken Karahi Price:1200 Quantity: 80	System Should update the sub menu item with its inventory that admins enter.

		quantity. 3. Click on submit		
TC104	For Give or take order	1 Click on order new 2 Select product name whom wants from brookdown list or seen from the menu that shown on the right side. 3 Select quantity. 4 Click on add.	Product Name: Chicken Karahi Quantity: 1	System should put the record in the order database
TC105	For take another order for a same customer	1. Click on add new from below of order list 2 Select product name whom wants from brookdown list or seen from the menu that shown on the right side. 3 Select quantity. 4 Click on add.	Product Name: Chicken Karahi Quantity: 1	The new order of a customer is update in a order table.
TC106	For generate bill from staff .	1. When staff complete to take order then if its click on the Paid Button. Then its mean customer Paid the bill.	Click on the Paid.	System generate the bill according to his order and thier status is paid.
TC107	For generate bill from customer	1. When Customer complete to give order then the Paid Button is not shown to customer.	Click on submit.	System generate the bill according to his order and thier status is unpaid.
TC108	For update the customer status unpaid to paid when customer paid his cash on delivery.	1. Click on dashboard. 2. Check the name or detail of the customer on	Click on Paid	System should update the status of customer.

		Latest Orders. 3. Update the status of customer from unpaid to paid when cash receive.s		
TC109	To verify that the order gets cancelled after click on Cancel order number link	Click the delete button on the order list on order page.	Click on Delete	System should delete the order from the order database.
TC110	To Verify Title of every Page reflects what the page objective is. For example Title of Order Page page should have "Order"	Verify that title of each page is the same as the page objective Click on Order and verify whether application directs to order page.	User: {haris} Password: {Haris5122}	Title of each page should reflect its objective and the buttons should redirect as specified, to the relevant page.

6.2. Equivalence partitioning

It is a software testing technique that divides the input test data of the application under test into each partition at least once of equivalent data from which test cases can be derived.

Username for Login with email

INVALID	VALID	INVALID
Superior university	Superioruni@gmail.com	Superior university
Result Status	Pass	Fail

While creating an account password should be more than 6 characters

INVALID	VALID	INVALID
Full4	Superior1234	12344
Result Status	Pass	Fail

Taking order from only available stock

INVALID	VALID	INVALID
Beer	Pizza	Beer
Result Status	Pass	Fail

Only admin can delete or add things to System

INVALID	VALID	INVALID
A staff trying to add an item	Admin added Pizza to items	A user trying to add an item
Result status	Pass	Fail

Bill generate

INVALID	VALID	INVALID
Without order not generate bill	Must be order then generate bill	Without order not generate bill
Result Status	Pass	Fail

6.3. Boundary value analysis

Boundary testing is the process of testing between extreme ends or boundaries between partitions of the input values.

- Let's consider the behavior of Order Pizza .
- Pizza values 1 to 10 is considered valid. A success message is shown.
- While value 11 to 99 are considered invalid for order and an error message will appear, "**Order Out Of stock**"

6.4. Data flow testing

Data Flow Testing Results		
Data Unit Name	Data Flow Method	Test Results
Registration Form	Interface-->Database, Database-->Interface	Pass

Login Form	Interface-->Database, Database-->Interface	Pass
Account Recovery	Interface-->Database, Database-->Interface	Pass
Dashboard	Interface-->Database, Database-->Interface	Pass
Order	Interface-->Database, Database-->Interface	Pass
Add New Menu	Interface-->Database, Database-->Interface	Pass
Add Item	Interface-->Database, Database-->Interface	Pass
Update And Delete Item	Interface-->Database, Database-->Interface	Pass
Add Staff	Interface-->Database, Database-->Interface	Pass
Change Staff	Interface-->Database, Database-->Interface	Pass
Bill Generate	Interface-->Database, Database-->Interface	Pass
Logout	Interface-->Database, Database-->Interface	Pass

6.5. Unit Testing

Unit Testing of POS Software Units		
Software Unit Name	Testing Method	Test Results
Registration Form	Manual testing on localhost	Pass
Login Form	Manual testing on localhost	Pass
Dashboard	Manual testing on localhost	Pass
Order	Manual testing on localhost	Pass
Add New Menu	Manual testing on localhost	Pass
Add Item	Manual testing on localhost	Pass
Update And Delete Item	Manual testing on localhost	Pass
Add Staff	Manual testing on localhost	Pass
Change Staff	Manual testing on localhost	Pass
Bill Generate	Manual testing on localhost	Pass
Logout	Manual testing on localhost	Pass

6.6. Performance testing

Integration testing of POS units			
Software Unit Name	Test with Unit	Testing Method	Test Results
Registration Form	POS Interface and relevant feature	Manual testing on localhost	Pass
Login Form	POS Interface and relevant feature	Manual testing on localhost	Pass
Dashboard	POS Interface and relevant feature	Manual testing on localhost	Pass
Order	POS Interface and relevant feature	Manual testing on localhost	Pass
Add New Menu	POS Interface and relevant feature	Manual testing on localhost	Pass
Add Item	POS Interface and relevant feature	Manual testing on localhost	Pass
Update And Delete Item	POS Interface and relevant feature	Manual testing on localhost	Pass
Add Staff	POS Interface and relevant feature	Manual testing on localhost	Pass
Change Staff	POS Interface and relevant feature	Manual testing on localhost	Pass

Bill Generate	POS Interface and relevant feature	Manual testing on localhost	Pass
Logout	POS Interface and relevant feature	Manual testing on localhost	Pass

Chapter 7

Summary, Conclusion and Future Enhancements

Chapter 7: Summary, Conclusion & Future Enhancements

7.1. Project Summary

In this chapter we discuss the proper description, analysis, solution of the system. We are developing a system of restaurant. The restaurant management systems help the restaurant manager to manage the restaurant more effectively and efficiently by computerizing meal ordering, billing, and inventory control. It will be all web based that handle all the transaction between customer and sales department and take all the record of stock and create list of the weekly, monthly or yearly sales. In addition to this we run the app on an Android based tablet & not on an iOS based tablet which is more expensive alternative.

7.2. Achievements and Improvements

In the system planning stage, the project objectives that the system must achieve are clearly stated and provide with clear description.

The first objective is to reduce the human work force in the restaurant. By using the new Restaurant Management System, some of the manual works is being computerized. Therefore, the needs of the larger workforce to complete customer registration and registration are no longer needed.

The second objective to achieve is to reduce human mistake. Now, by using the system implemented to carry out the daily transaction, such as order processing, the waiter just need to enter the customer order into the system and the system will process the order and display on the kitchen monitor. The misinterpretation of order taken due to handwriting issues is now eliminated and also the chances of losing of the order sheet.

The third objective is to simplify the food ordering process as now the system allows the dine-in customer to make their own food ordering through our developed restaurant mobile application.

The staff in-charge just need to key in the details through our system and our system will provide information maintenance to be made in a simple and easy way . In a nutshell, the project objectives that stated in the System Planning phase are successfully implemented and achieved. The Restaurant Management System is now preparing to provide better service to the restaurant and customers

7.3. Lessons Learnt

Following are the learning outcome that we have learned during the development of the project:

- Learn how to apply software engineering techniques on a project.
- Learn about the importance and role of documentation in a software project.
- Learning of new technology and tools.
- Learn to meet deadlines.
- Learn to be flexible while developing a software project.
- Team building skill and responsibilities.
- Learn to manage the project effectively.
- Learn how to tackle the changes to the software scope during development.
- Learn how to develop use full and live working project.
- Learn communication skill for better project communication management.
- Learn new testing and development tools.
- Learn to develop better documentation.

7.4. Future Enhancements/Recommendations

After carrying a series of project evaluation, it found out that the project is still able to be improved by the following suggestion:

Password encryption

The system can be more secure to its information by further improve the security of the login module by providing encrypted password. By implementing password encryption, it will increase the security level of the system towards the system user. It is difficult for the unauthorized person to obtain the login credential even though they illegally access to the database.



Forum Discussion

Although now the system is designed to have instant messaging between the restaurant member and the restaurant staff, however, it is best to allow the restaurant member to communicate with another restaurant member in exchanging information and feedback of visiting the restaurant. It can be served as a platform for sharing feedback and comment which is open to public.

Ability in inventory control and managing more than one restaurant

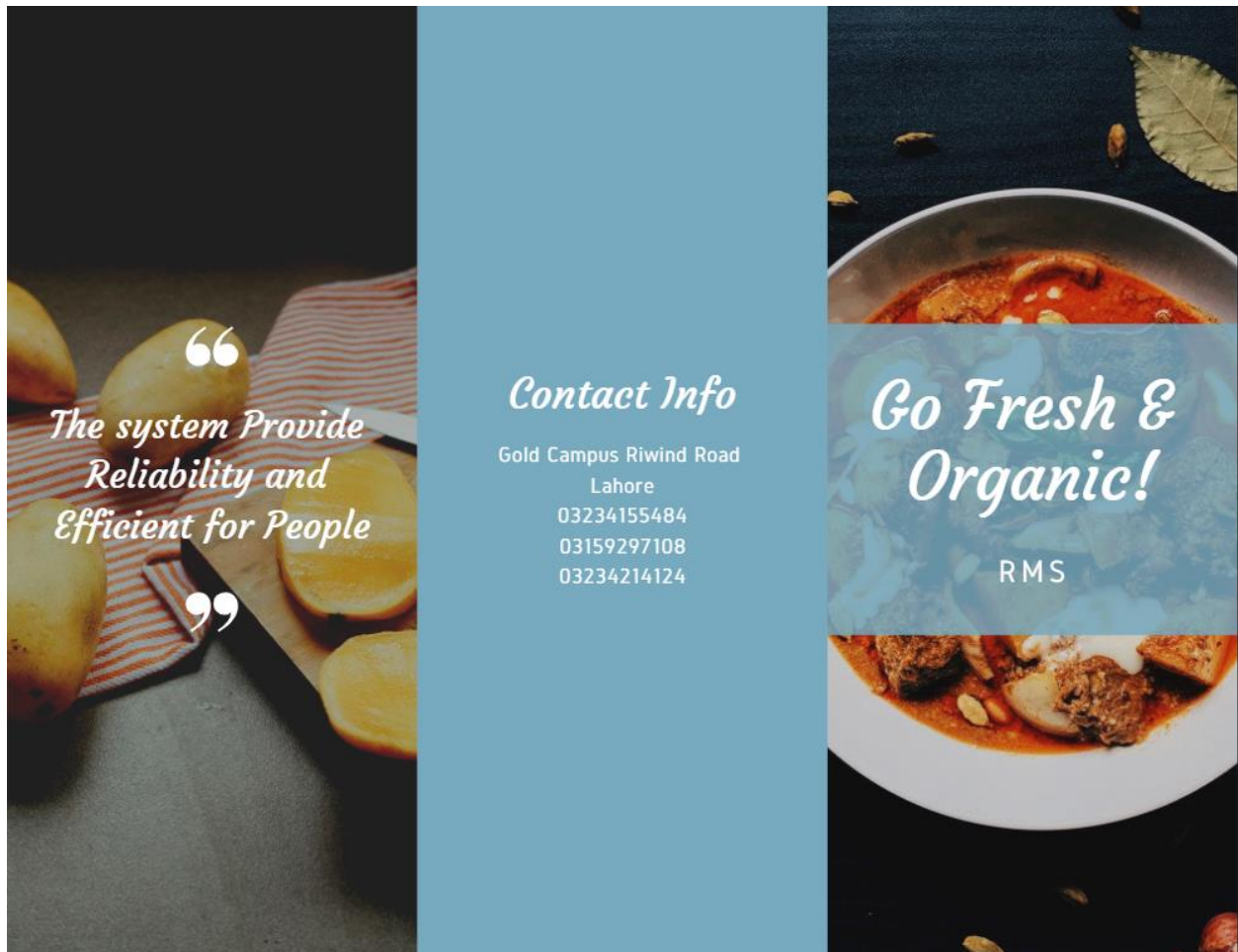
The system implemented does not provide inventory control facilities which can further be improved. Every restaurant is guaranteed to involve in stock managing, if inventory control is implemented, then the system can be said as a more complete system. Besides, the system can also be implemented to support the restaurant business grows as there is a possibility that the restaurant will expand their business with several branches.

Appendices

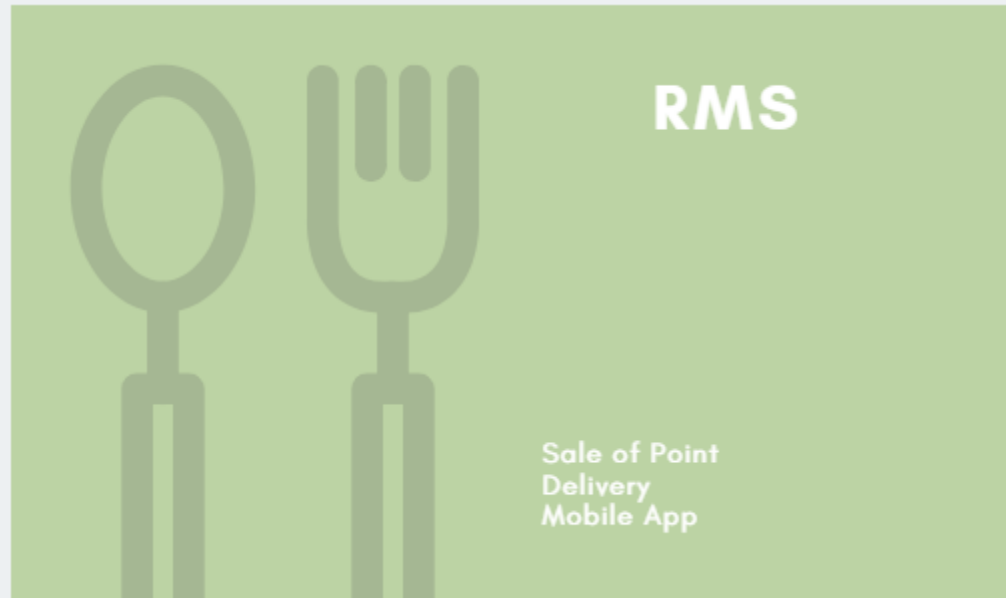
Appendix A: Information / Promotional Material

Marketing material is of great value in promoting the business/any product you are willing to sell. It helps you in catching the client's attention towards your product. Following are the promotional materials that we will utilize to promote our project in the market

A1: Broacher



A2: Visiting Card



A:3 Flayer



Every hour is happy hour!

RMS

**Our Mission is to provide
Pakistan With the Quality
Experience**

Features :
Point of Sale
Mobile App

Superior Gold Campus Riwind Road Lahore
03234155484
03159297108
03234214124

A4: Standee

Your Best Manager

RMS

Unique Values:

- *Point of Sale*
- *Control ed Production Cost*
- *Automatic Billing*
- *Easy to Use*

SUPERIOR UNIVERSITY LAHORE

CMACED
CHAUDHRY MUHAMMAD AKRAM
CENTER FOR ENTREPRENEURSHIP
DEVELOPMENT

STUDENT JOBS.PK

Reference and Bibliography

Reference and Bibliography

- C. Larman, APPLYING UML AND PATTERNS An Introduction to Object-Oriented Analysis and Design and Iterative Development, 3rd ed., Massachusetts: Pearson Education, 2005.
- *Morochove, Richard (26 February 2008). "Simpler Project-Managing Apps Get the Job Done". PCWorld.* Retrieved 13 May 2012
- <https://www.slideshare.net/Aurnob0071/software-requirements-specification-for-restaurant-management-system>
- ChiselAppChangeOver Complete - CHISEL FOSSIL SCM HOSTING". Andreas Kupries. 30 April 2013. Retrieved 1 May 2013
- *S.Ruttloff, Computer Hard- & Software Software- und IT-Dienstleistungen. 21 March 2013.* Retrieved 21 March 2013.
- "What's New — Project KickStart Pro 4 Key Features — Document Management". Experience In Software, Inc. Archived from the original on 11 April 2012. Retrieved 11 April 2012
-
- D. Carrington, CSSE3002 Course Notes, School of ITEE University of Queensland, 2008 .
- IEEE Recommended Practice for Software Requirements Specifications, IEEE Standard 830, 1998.
- Java Developer Tutorials and Online Training. 2015. [ONLINE] Available at: <http://www.oracle.com/technetwork/java/index-jsp-135888.html>
- Apache Tomcat - Apache Tomcat 7 Downloads. 2015. [ONLINE] Available at: <https://tomcat.apache.org/download-70.cgi>
- Daniweb (http://www.daniweb.com/)
- Bushnell, Mona (11 October 2017). "5 Restaurant Employee Scams That Will Cost Your Business Big". Business.com. Retrieved 12 October 2017.

- ["Penguin MAX & Penguin MIN - Restaurant Management Software"](#). *Penguin App*. Prompt Softech. 2017. Retrieved 12 October 2017.
- Bortone, Brad (20 June 2017). ["Why Scalability Is Priority 1 For Restaurant Management Systems"](#). *Restaurant Insider*. Upserve. Retrieved 12 October 2017.