

FOOD HUB

Final Year Project

Session 2017-2021

A project submitted in partial fulfillment of the degree of

BS in Information Technology



Department of Information Technology

Faculty of Computer Science & Information Technology

The Superior College Lahore, Pakistan

Spring 2021

FOOD HUB	[☒] Development [☐] Research [☐] R&D			
Area of specialization	Mobile Application Development			
Project ID	FYP-BITM-F20-005			
Project Group Members				
Sr.#	Reg. #	Student Name	Email ID	*Signature
(i)	Bitm-f17-030	Mian Rashid	Bitm-f17-030@superior.edu.pk	
(ii)	Bitm-f17-017	Muhammad Raza	Bitm-f17-017@superior.edu.pk	
(iii)	Bitm-f17-044	Muhammad Waleed	Bitm-f17-044@superior.edu.pk	

Plagiarism Free Certificate

This is to certify that, I **Muhammad Raza** Son of **Ashiq Hussain** group leader of FYP under registration no **Bitm-f17-017** at Information Technology Department, The Superior University, Lahore. I declare that my FYP proposal is checked by my supervisor and the similarity index is 10 that is less than 20%, an acceptable limit by HEC. Report is attached herewith as Appendix D.

Date _____ Name of Group Leader: **Muhammad Raza** Signature: _____

Name of Supervisor: Javaid Iqbal

Designation: Lecturer

Signature: _____

HoD: Dr. Syed Asad Ali Naqvi

Signature: _____

FOOD HUB

Change Record

Author(s)	Version	Date	Notes	Supervisor's Signature
Rashid ,raza , waleed	1.0		Template 1	
Rashid ,raza , waleed	1.1	28-11-2020	documentation	

APPROVAL

PROJECT SUPERVISOR

Comments: _____

Name: _____

Date: _____ Signature: _____

PROJECT MANAGER

Comments: _____

Date: _____ Signature: _____

HEAD OF THE DEPARTMENT

Comments: _____

Date: _____ Signature: _____

Dedication

“We dedicated our work to the most respectful personality of the world and beloved PROPHET (PBUH) of Allah then to our parents, teachers, friends and all those persons who prayed and helped us in our project.”

Acknowledgements

I am really thankful to my supervisor who has guided us to make a wonderful website. Who teaches us how we manage our work and tell us pros and cons of our ideas. He is deal us very politely and give us suggestions. He has a best communication level. He gave me and my group more or more motivation and confidence. He did not refuse our idea but he enhances that idea.

Executive Summary

The “**Food Hub**” is a project introduced for the mess system with the combination of android application and web application (for admin panel). Our project is based on Mess Food system which will provide easy facilities to people. This system will help the organization in such a way that it will save time for the members and minimize human errors in currently out-dated manual file based mess management system.

Table of Contents

Final Year Project	i
Session 2017-2021	i
BS in Information Technology	i
Plagiarism Free Certificate	ii
Dedication	v
Acknowledgements	vi
Executive Summary	vii
Table of Contents	viii
List of Figures	xi
List of Tables	xii
Chapter 1	1
Introduction	1
1.1. Background	2
1.2. Motivations and Challenges	2
1.3. Goals and Objectives	3
1.4. Literature Review/Existing Solutions	3
1.5. Gap Analysis	3
1.6. Proposed Solution	4
1.7. Project Plan	4
1.7.1. Work Breakdown Structure	5
1.7.2. Roles & Responsibility Matrix	6
1.7.3. Gantt Chart	7
1.8. Report Outline	7
Chapter 2	8
Software Requirement Specifications	8
2.1. Introduction	9
2.1.1. Purpose	9
2.1.2. Document Conventions	9
2.1.3. Intended Audience and Reading Suggestions	9
2.1.4. Product Scope	9
2.2. Overall Description	10
2.2.1. Product Perspective	10
2.2.2. User Classes and Characteristics	10
2.2.3. Operating Environment	10
2.2.4. Design and Implementation Constraints	10
2.2.5. Assumptions and Dependencies	11
2.3. External Interface Requirements	11
2.3.1. User Interfaces	11
2.3.2. Hardware Interfaces	11
2.3.3. Software Interfaces	12
2.3.4. Communications Interfaces	12
2.4. System Features	12
2.4.1. System Feature 1	12
2.4.1.1. Description and Priority	12

2.4.1.2.	Stimulus/Response Sequences	12
2.4.1.3.	Functional Requirements	13
2.4.2.	System Feature 2	13
2.4.2.1.	Description and Priority	13
2.4.2.2.	Stimulus/Response Sequences	13
2.4.2.3.	Functional Requirements	13
2.4.3.	System Feature 3 (and so on)	14
2.5.	Nonfunctional Requirements.....	14
2.5.1.	Performance Requirements.....	14
2.5.2.	Safety Requirements	14
2.5.3.	Security Requirements.....	15
2.5.4.	Usability Requirements	15
2.5.5.	Maintainability/Supportability Requirements	15
2.5.6.	Efficiency Requirements	15
2.6.	Domain Requirements.....	15
Chapter 3.....		16
Use Case Analysis.....		16
3.1.	Use Case Model	17
3.2.	Use Cases Description.....	18
	Fully Dressed Use Case	18
Chapter 4.....		21
System Design		21
4.1.	Architecture Diagram	24
4.2.	Domain Model.....	25
4.3.	Entity Relationship Diagram with data dictionary	26
4.4.	Class Diagram	28
4.5.	Sequence / Collaboration Diagram	29
4.6.	Activity Diagram.....	31
4.7.	State Transition Diagram	32
4.8.	Component Diagram	32
4.9.	Deployment Diagram	33
4.10.	Data Flow diagram	33
Chapter 5.....		35
Implementation		35
5.1.	Important Flow Control/Pseudo codes.....	36
5.2.	Components, Libraries, Web Services and stubs	36
5.3.	Deployment Environment	36
5.4.	Tools and Techniques.....	37
5.5.	Best Practices / Coding Standards.....	37
Chapter 6.....		38
Testing and Evaluation		38
6.1.	Use Case Testing.....	39
6.2.	Equivalence partitioning	39
6.3.	Boundary value analysis.....	40
6.4.	Data flow testing	40
6.5.	Unit testing	40

6.6. Performance testing.....	40
6.7. Stress Testing	41
Chapter 7	42
Summary, Conclusion and Future Enhancements	42
7.1. Project Summary	43
7.2. Achievements and Improvements	43
7.3. Critical Review.....	43
7.4. Lessons Learnt.....	44
7.5. Future Enhancements/Recommendations	44
Appendices.....	45
Appendix A: User Manual	46
Appendix B: Administrator Manual	48
Appendix C: Information / Promotional Material	48
References.....	50
References:.....	51

List of Figures

1.1	Work breakdown structure	5
1.2	Gantt chart	7
3.1	Use case model	14
2.2	Caption of second figure of second chapter	22
2.3	Caption of third figure of second chapter	26
5.1	Caption of first figure of fifth chapter	49
5.2	Caption of second figure of fifth chapter	49

List of Tables

1.1	Roles and responsibility Matrix	6
3.1	Use case Description	18
2.0	label of second table of second chapter	22
2.1	label of third table of second chapter	26
5.1	label of first table of fifth chapter	49
5.2	label of second table of fifth chapter	49

Chapter 1

Introduction

Chapter 1: Introduction

In today's age everyone try to prepare food and deliver them quickly. Now recently all the orders are placed over the phone or by going and waiting in the queue to wait for your turn, but there are many disadvantages to this system including the inconvenience of customers and mess staff both. The costumers need to wait a lot before they could receive their order or to start eating food. The data operators in the mess staff has to receive and record money for each and every item that a customer orders. This task is quiet a Sisyphean. In addition to it the costumers lack the visual confirmation of the orders they place which could result in unintended orders placed. To solve all these challenging issues of staff and employees, here it is. The main advantage of an online ordering system is that it greatly simplifies the process of ordering for costumers and mess staff both. When a costumer visit the ordering portal, they are presented with an interactive and regularly updated menu for breakfast, lunch, snacks and dinner completely with all available options and adjusting prices after selecting the quantities for each food item from menu. Within the application all items in order are displayed along with their corresponding options and other details in a concise and easy to read manner. At the time of a successful signup of the employee, he/she will be allotted a unique password along with their personal numbers to help them in placing orders.

1.1. Background

If we have a look on file base manual entries of any kind of transaction, what do we first assume? Yes, that it might include mathematical error which we could neglect; still knowing that they exist in our system. In addition to these errors, people also are facing difficulties while food ordering as they have to go to the mess, order food and wait outside for their meal to be cooked and packed. People nowadays demand automation and mobile facilities which they could grab while they are in their offices, homes and other institutes as well.

1.2. Motivations and Challenges

Our main Motivation is that we are providing people a facilitation to get their Mess food by ordering it through a Mobile Application or through a Website. Besides waiting for the food order, People can order it through Application.

The Main Challenge we have is to make it popular amount people so that they can get such facilitation.

1.3. Goals and Objectives

Our goal is to make an Application (Phone and Web Based) by which user can easily choose between the cooked food in the Mess and Ordering according to their requirements.

The purpose of this study was to identify the benefits and barriers to implement an online mess management system embedded into the billing system of the entire organization discussed in the literature. In addition to it, the reason was to help employees save their time during their regular office tasks and to minimize the errors in the manual file based system.

1.4. Literature Review/Existing Solutions

If we have a look on file base manual entries of any kind of transaction, what do we first assume? Yes, that it might include mathematical error which we could neglect; still knowing that they exist in our system. In addition to these errors, people also are facing difficulties while food ordering as they have to go to the mess, order food and wait outside for their meal to be cooked and packed. People nowadays demand automation and mobile facilities which they could grab while they are in their offices, homes and other institutes as well.

- **Coderbotics** has developed only a desktop based mess management system for which there is no android application is built.
- Another similar project called **Mess Manager** is built but in this project, only the Web base is developed which deals with the database itself but it is applicable to very small organizations only.

1.5. Gap Analysis

There is no Online web and android based system in which staff can be facilitated with these sources .This app should helpfull in order to consume to much time of staff.

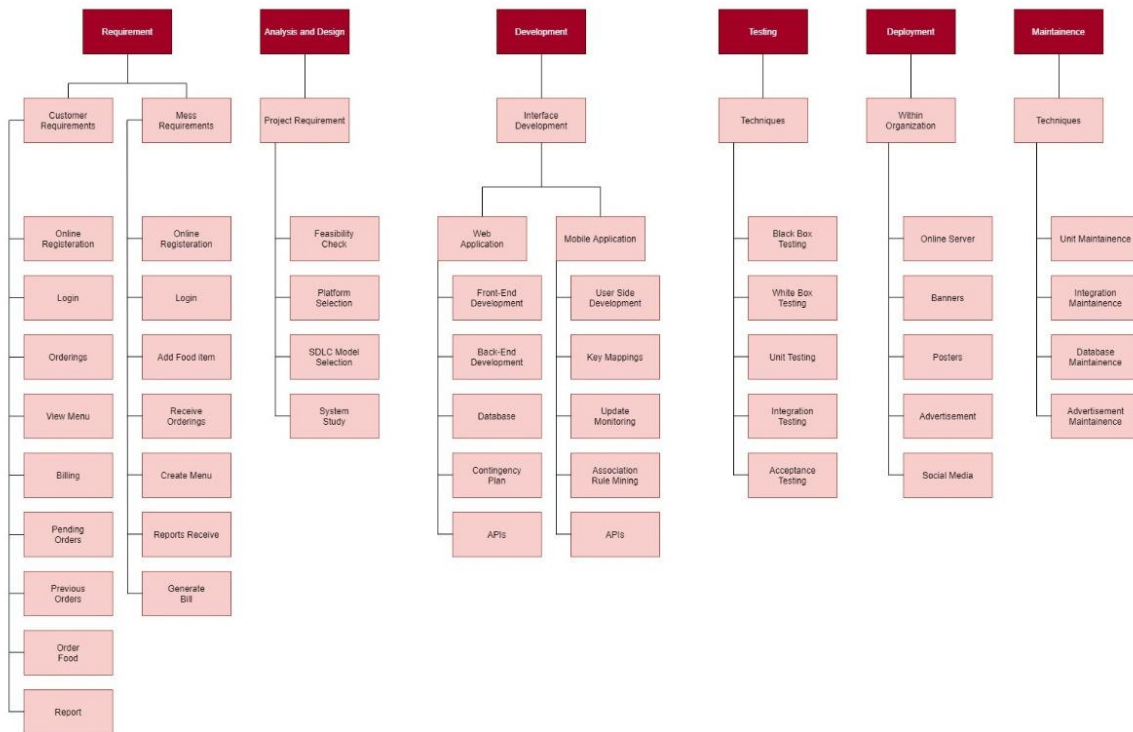
1.6. Proposed Solution

- Minimize the faults in the billing systems by human-errors made by either the data entry operators or the personal details by costumers.
- Build a multi-function desktop application for the admin panel connected to a real-time database that stores each and every transaction and details of the customers. This platform will provide all the read and write access of database to the admin
- Deploy a fully automated and convenient mess system with the combination web application in such a massive organization.
- Build a “Food Suggestion” block which includes the costumer’s favourite food items that he/she orders the most by using the machine learning approach.

1.7. Project Plan

We gathered all the requirements which were necessary for the development of Food Hub project in the planning phase of Agile Model in each iteration, with following responsibilities matrix ,WBS and Gantt chart.

1.7.1. Work Breakdown Structure



1.7.2. Roles & Responsibility Matrix

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

WBS #	WBS Deliverable	Activity #	Activity to Complete the Deliverable	Duration (# of Days)	Responsible Team Member(s) & Role(s)
1	Initiation	001	Documentation	10	Mian Rashid
2	Taking Requirements	002	Documentation	10	Mian Rashid
3	Documentation	003	Documentation	10	M.Raza,Rashid
4	Analysis & Design	004	Documentation	10	M.Raza
5	User Interface Design	005	Html,css, javascript, Bootstrap	10	M.Waleed,rashid,raza
6	Back-end coding	006	PHP Coding	30	Mian Rashid,waleed,raza
7	Database	007	My Sql Database	30	Waleed,raza,waleed
8	Testing	008		10	Mian Rashid,raza
9	Deployment	009	Documentation(User Manual)	10	M.Raza
10	Maintenance	010		10	Waleed,rashid,raza

1.7.3. Gantt Chart



1.8. Report Outline

- The requirement was gathered in about 20days.
- Documentation Continue....

Chapter 2

Software Requirement Specifications

Chapter 2: Software Requirement Specifications

2.1. Introduction

2.1.1. Purpose

This Application describes and define the functions and specifications of our website is the primary goal of this software requirement specification. This SRS illustrates in clear terms, the system's primary uses and required functionality as specified by our customer. Our purpose is to create an Application **FoodHub**.

2.1.2. Document Conventions

A document is primarily prepared for a project, software or any kind of website. SRS is a document that explain the nature of a project, website and software. It is a manual document of a project that provide you immediately search from website. This document is known as SRS report or software document.

2.1.3. Intended Audience and Reading Suggestions

Intended audience of this website is a project manager which use SRS to evaluate the site in all phases SDLC either project process going according to system requirement.

- This Application is also used by the designer to make perfect design flow of the product according to SRS.
- SRS is used by the developers to check the implementation fulfill the functionalities and features of the website that defined in it.

2.1.4. Product Scope

Today the use of computer and internet has become necessary for everyone.

- This application provides online all types of dishes in a Mess with their brief characteristics.
- Information of food are provided here on Daily Basis.
- Customer visit this website and easily check the availability of the Food and can place an Order.
- Customer can Review the food quality.

2.2. Overall Description

2.2.1. Product Perspective

The Application can easily be accessed by using every operating system (Web Based) and Mobile application of Android OS.

2.2.2. User Classes and Characteristics

Management & Customer are the types of users that interacted with this system. Each of these two types of users has different use of the Application so each of them has their own requirements. Management can upload about food variety, notices and prices of the food. Customer can place an order.

2.2.3. Operating Environment

Website

- HTML and CSS is used for front end in our project.
- PHP is use for back end and JAVA is also used in it.
- Local host is used for server in our website.
- MYSQL is used for connectivity for Database.

Mobile application

- XML and Java is used for developing the Application.
- Firebase is used for Database

2.2.4. Design and Implementation Constraints

- The invalid data should be ignored and error messages should be given.
- The **website** also runs on IOS, it can easily accessed by all PC.
- The Application (Mobile and Web based) cannot run without Internet.

2.2.5. Assumptions and Dependencies

- Management and customers are connected with each other at same platform through internet.
- Information accessed from the database.
- The website is for all type of PCs and easily accessible.
- The Application is only for Android OS currently (Will work on IOS later on)

2.3. External Interface Requirements

2.3.1. User Interfaces

Management & Customer are the types of users that interacted with this system. Each of these two types of users has different use of the Application so each of them has their own requirements. Management can upload about food variety, notices and prices of the food. Customer view food types and can place an order.

2.3.2. Hardware Interfaces

Web Based:

- All series of Core I processor Pc
- 1 GB RAM of system
- 500GB Hard disk
- Laptops, PC, tablet and mobile devices

Mobile Based:

- Android version not less than 4.1 Jelly Beans.
- 1 GB RAM of Mobile
- Upto 50MB required Storage Space
- Minimum of Dual core Processor

2.3.3. Software Interfaces

Web Based

- Visual studio
- Photoshop
- Windows XP to Window 10
- Web domain .com

Mobile Application

- Android Studio
- FireBase
- Photoshop

2.3.4. Communications Interfaces

- Database SQL & Firebase
- Emails
- Domain DNS

2.4. System Features

2.4.1. System Feature 1

Required information should be shown at same platform that can easily be vied by both admin and user.

2.4.1.1. Description and Priority

The user can get information of Food available in the Mess from the features. The application cannot work without this priority feature.

2.4.1.2. Stimulus/Response Sequences

Every time the data is saved and retrieved from the database easily.

2.4.1.3. Functional Requirements

Functional requirements describe a function of Application or its components. A function explains the behavior, a set of inputs processing and output.

For Customer

- REQ-SF1-1: Mess Management can upload pictures.
- REQ-SF1-2: Mess Management can add details.
- REQ-SF1-3: Customer Search or view the product and can place an order.

2.4.2. System Feature 2

Functional requirements describe a function of Application. A function explains the behavior, a set of inputs processing and output.

2.4.2.1. Description and Priority

It is a best priority project. The customer can get information from the features and retrieve the data through the connected Database. Checking availability, choosing and then ordering. The application cannot work without this priority feature.

2.4.2.2. Stimulus/Response Sequences

Every time the data is saved and retrieved from the database easily.

2.4.2.3. Functional Requirements

Functional requirements describe a function of website. A function explains the behavior, a set of inputs processing and output.

For Management of Mess

- REQ-SF2-1: Allow the management to add new Food lists every day or keeping the existing one
- REQ-SF2-2: Allow the management to delete the data
- REQ-SF2-3: Allow the management to update the description of everything
- REQ-SF2-4: Allow the management to upload different varieties of Food.
- REQ-SF2-5: Allow the customer to achieve these schemes.

- REQ-SF2-6: Allow the customer to place an order.

2.4.3. System Feature 3 (and so on)

Functional requirements describe a function of the application. A function is the behavior, a set of inputs processing and output.

For Administrator:

- Mess can add/drop the services
- Mess can view/upload the services
- Customer can track their order
- Mess can view the order of customer

2.5. Nonfunctional Requirements

Non-functional requirements are those requirements that the user has no direct control over but are part of the system nonetheless.

1. Equity and Constraints to prevent invalid data
2. Reasonable performance.
3. Quality assurance (notification of faults).
4. Documentation (optional extra).
5. Discoverability.
6. Interactive and Elegant GUI.
7. Feedback and Toleration in actions.

2.5.1. Performance Requirements

The application should be able for the customer that they can easily view the available food in the Mess through internet. This project also provides Food information for customer.

2.5.2. Safety Requirements

Safety requirement are used to save the information and no one can use this information for wrong purpose. So that is why we make sure anyone who access this application should be a valid purpose and must giving their username and password to access.

2.5.3. Security Requirements

Administrator is only a person who has access on the application and he is able to review the information. Every user has their own login and password.

2.5.4. Usability Requirements

For the Application, It can be accessed by the help of internet and is very user friendly. The user simply enters their registered mess name and can get access to the available food over there, later the customer can place a desired order.

2.5.5. Maintainability/Supportability Requirements

The Admin will be maintaining the application so that it would not get slow.

2.5.6. Efficiency Requirements

In order for efficiency of the application, the Database would have to be regularly maintained so that the data which is to be stored or fetched should be done quickly.

Some points for fast usage of the application

- More System memory
- Fast processor for processing
- High speed internet

2.6. Domain Requirements

The Website will be placed over the domain name **FOODHUB.COM**. The domain will be purchased from a good domain provider such as **GoDaddy.com** or **Namecheap.com**.

The mobile application will be linked to the domain.

Chapter 3

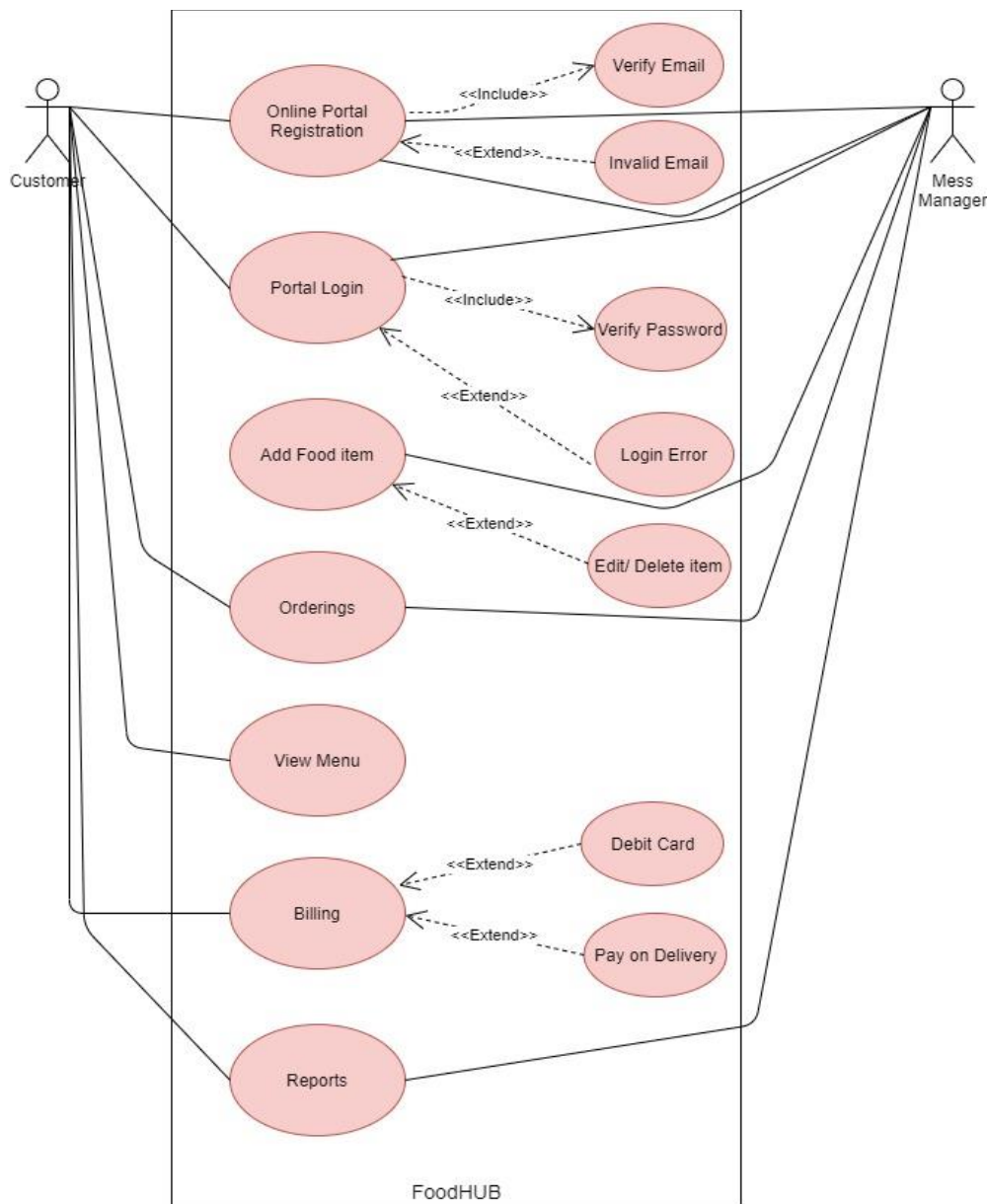
Use Case Analysis

Chapter 3: Use Case Analysis

A use case diagram at its simplest is a representation of a user's interaction with the system that shows the relationship between the user and the different use cases in which the user is involved.

The Below USE CASE describes how our system will work.

3.1. Use Case Model



3.2. Use Cases Description

Fully Dressed Use Case

Use case name: Portal Registration
Primary Actors: The Customer and Mess will register their new account.
Precondition: The user should not have their current account. Triggers: Need to create an account.
Scenario: - Go to the main page, click on Sign up and fill the registration form. - Click Create account button. - Account is created only if there are no errors in the form.
Use case name: Portal Login
Primary Actors: The Customer and Mess will Login their new account.
Precondition: - The user should already have an account on the application - With having an account, the user should also have their information in the account
Scenario: Go to the main page, click on Login and enter your Login details.
Use case name: Add Food Item
Primary Actors: The Mess will add the food items in the Food menu.
Precondition: The Mess should have not Added their existing food item
Scenario: Go to the Mess Menu page and view the Food Items.

Use case name: Orderings	
Primary Actors: The Customer will place an order, The Mess will receive the orders	
Precondition: • The Customer should not have placed an Order earlier to the new order. • Mess should have not received any of the customers orders	
Scenario: • Go to the Mess Menu page and view the Food Items then Select the item and place an order.	
Use case name: View Menu	
Primary Actors: The Customer will view the Menu provided by the Mess.	
Precondition: • The Mess should have Added their existing food item	
Scenario: • Go to the Mess Menu page and view the Food Items.	
Use case name: Billing	
Primary Actors: The Customer will pay the bill through Debit Card or through Cash on Delivery to the Mess.	
Precondition: • The Customer should not have Received the Bill amount • The customer should have placed an order • The Mess should have Generated the Bill amount	
Scenario: • While placing an order on the application, Customer will receive the Bill amount in the Cart section.	

Use case name: Report
Primary Actors: The Customer can report the Mess about the Food Quality and the Mess will receive it
Precondition: • The Customer should not have Received the Food Item • The customer should have placed an order • The Mess should have delivered the food right on time
Scenario: • After Receiving the delivery of food item, the customer can report to the Mess about the food quality by going through My orders and clicking on Report.

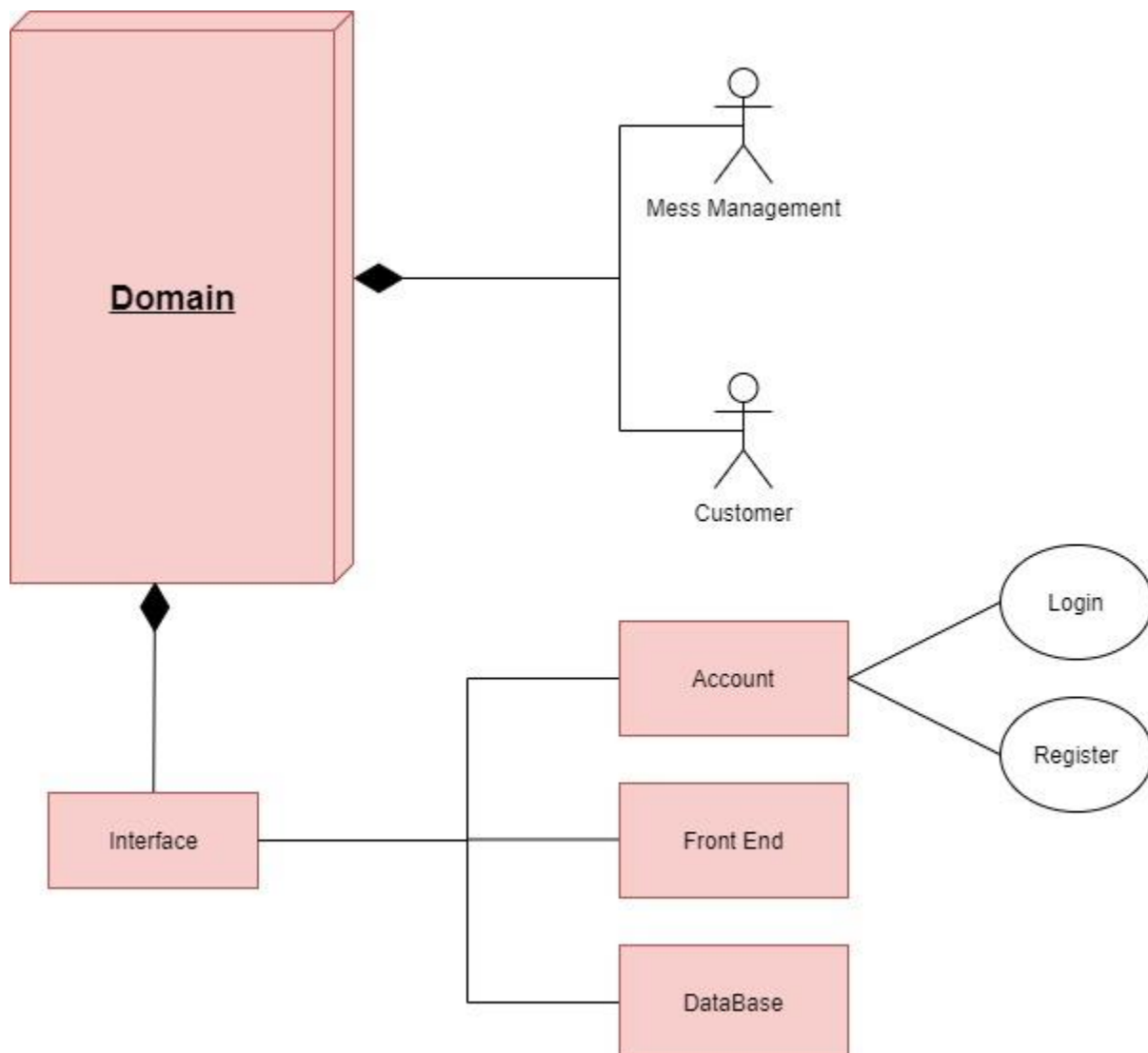
Chapter 4

System Design

Chapter 4: System Design

System diagram is a visual model of a system which shows its components, and their interactions with each other. Below is the System Design of how **FoodHub** system will work and interact with each other.

The Interface of the Application (Web Based and Mobile) will be connected to a custom Domain and the users of the application will access the application through the domain, by connecting to it.

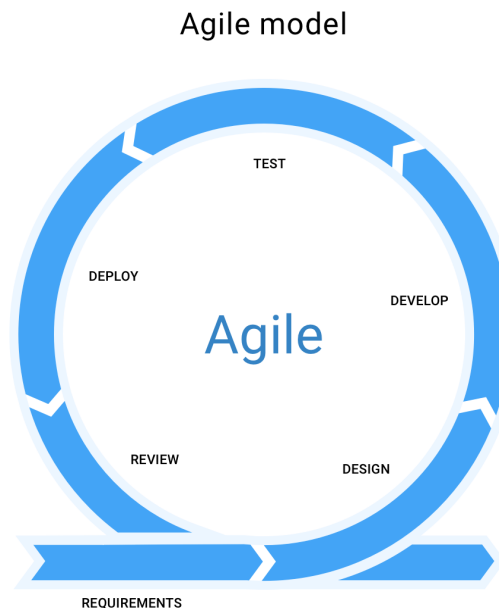


Agile Model is for modeling and documenting software systems based on best practices. It is a collection of values and principles, that can be applied on a software development project.

You can gather requirements and then create a design. After the Design phase, you move towards developing then testing, deployment and Reviewing.

The benefit of the Agile model is that you can review your all tasks performed in creating the system and you are able to correct your mistakes in order to make the system perfectly working.

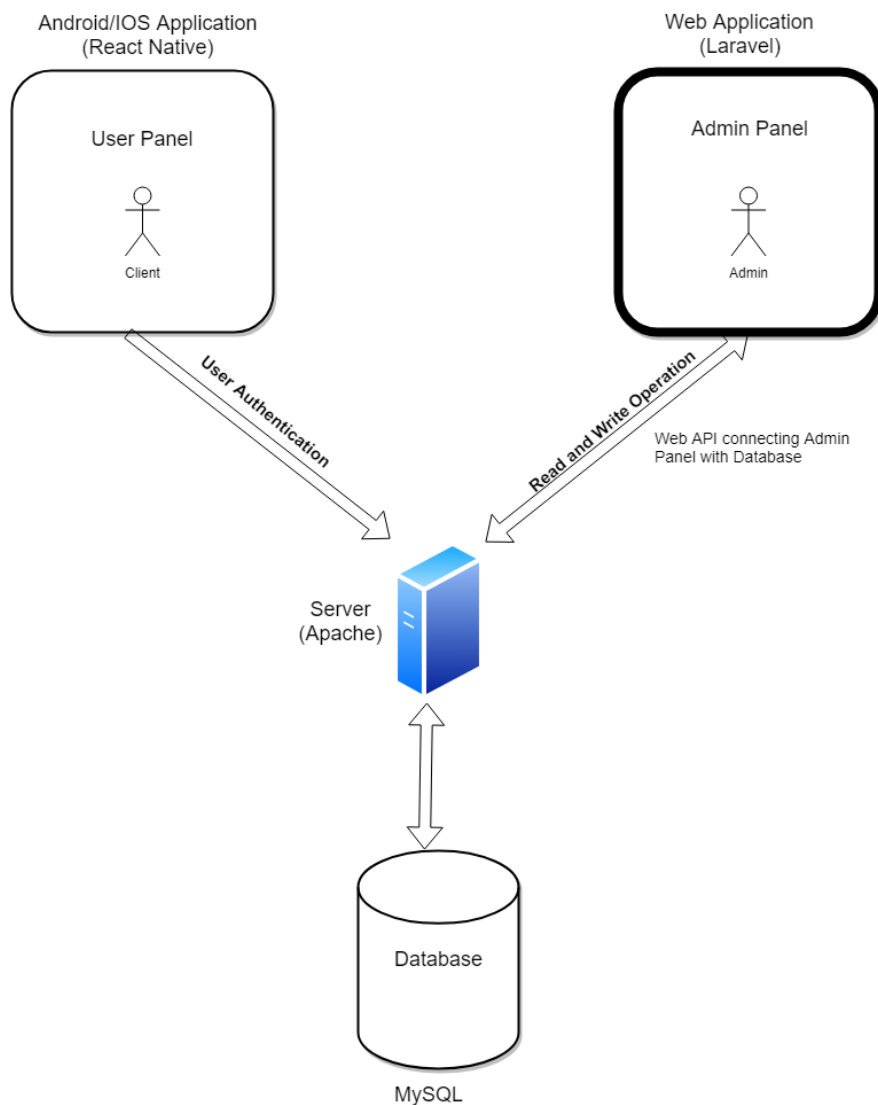
- Comparatively to other models, in the agile model avoidance of risk is high due to iterations one by one and testing in all the iterations so, it's suitable for our project.
- Additional Functionalities can be added at a later date due to continuous iterations.
- This model is used for big projects.



4.1. Architecture Diagram

Architectural diagram is a diagram of a system that is used to abstract the whole outline of the software system with the relationships of the system with each other, constraints of the system with each other, and boundaries between components.

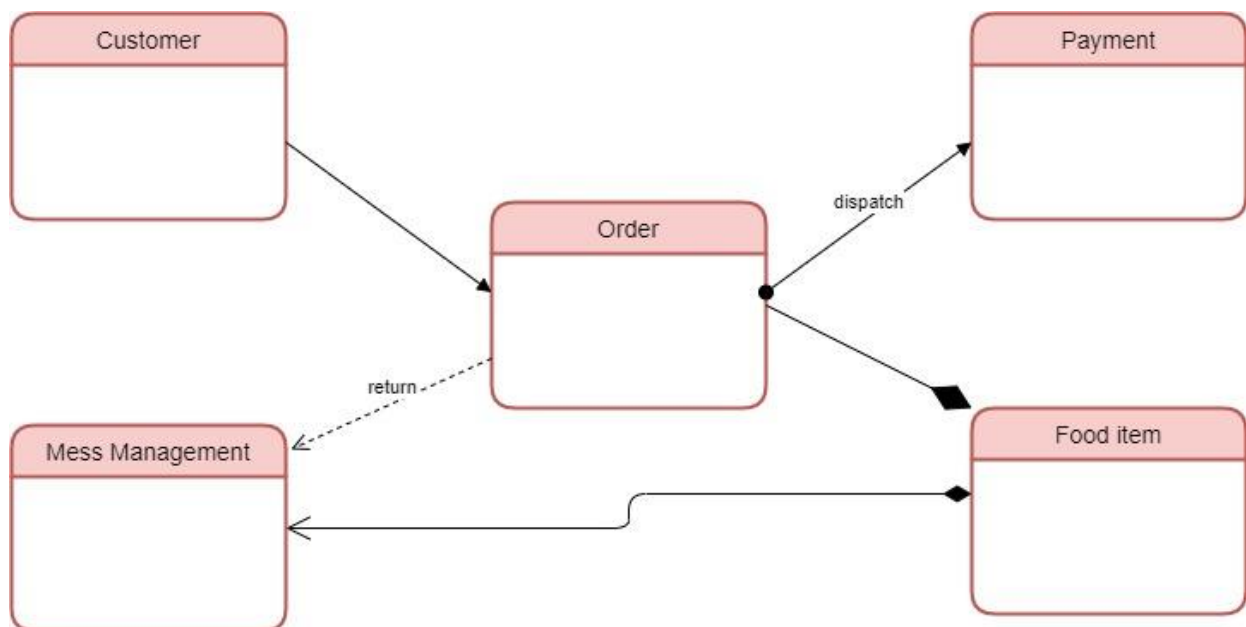
It explains how a system will be working with all of its components. It explains the physical deployment of the system.



4.2. Domain Model

Domain model is a model of the domain that incorporates with the both behaviour and the data. Domain model is a representation of a domain with the concepts of the system, roles of the system, datatypes of the system, individuals of the system, and rules of the system.

Here, The customer places an order by selecting the Food item and then the order is directly returned to the Mess Management. The management is responsible for adding food items in the Menu. After the Food Orderings, a payment receipt will be generated to the Customer.

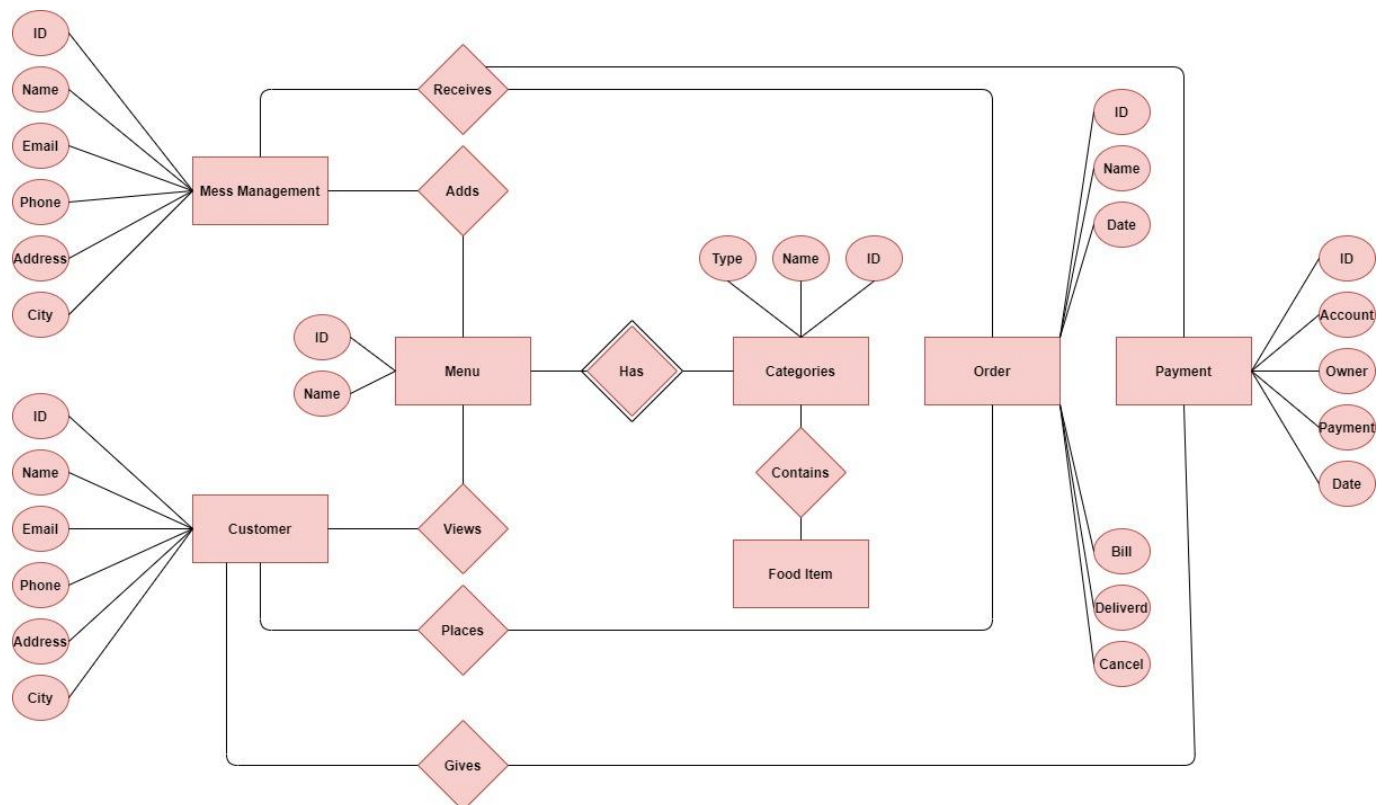


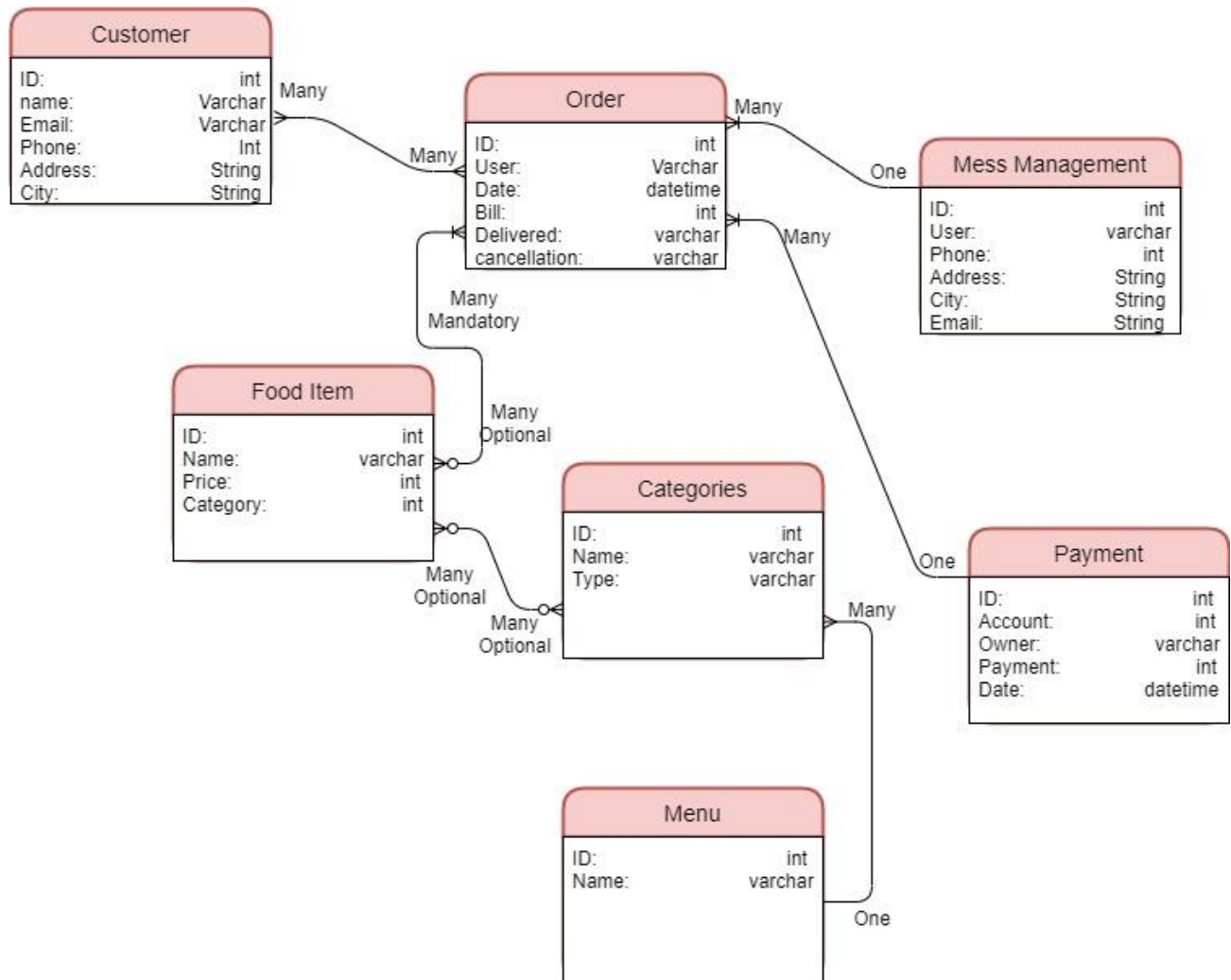
4.3. Entity Relationship Diagram with data dictionary

An entity relationship diagram (ERD) shows the relationships of entities stored in a database. By defining the attributes of the system, with their entities, and showing the relationships between them, an Entity Relationship diagram shows the logical structure of databases.

In our system,

1. Many customers can order many orders.
2. One Mess Management can receive many orders respectively.
3. In an Order there can consists of Many options of Food items
4. A Menu can have one or many Categories
5. Categories can have many options
6. Different orders can have one or many payment types (Online payment, Cash on delivery)

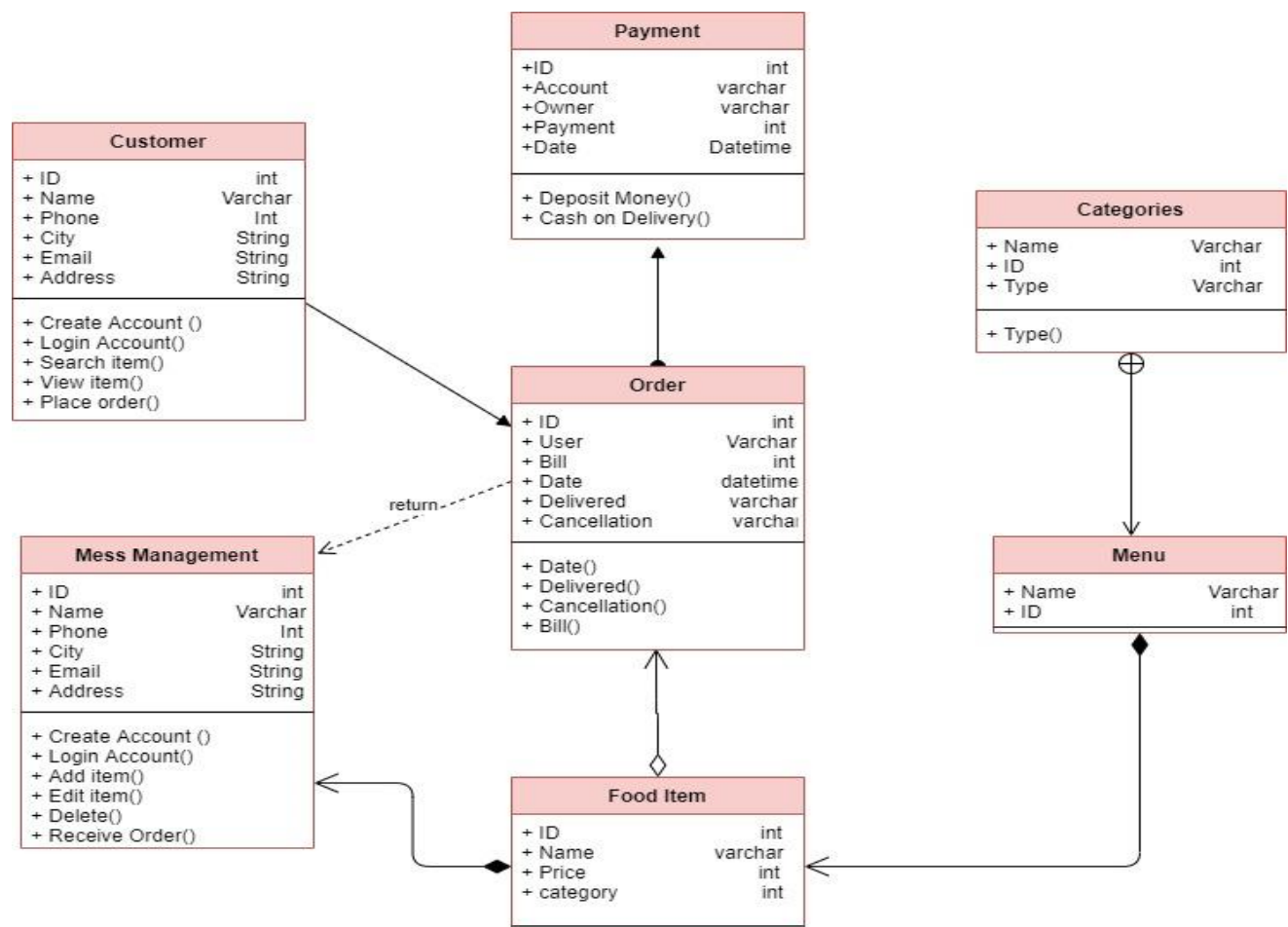




4.4. Class Diagram

Class Diagram is used for general modeling of the complete structure of the application. For the detailed modeling of the system, translating the models into programming code is used. Class diagrams can also be used for data modeling.

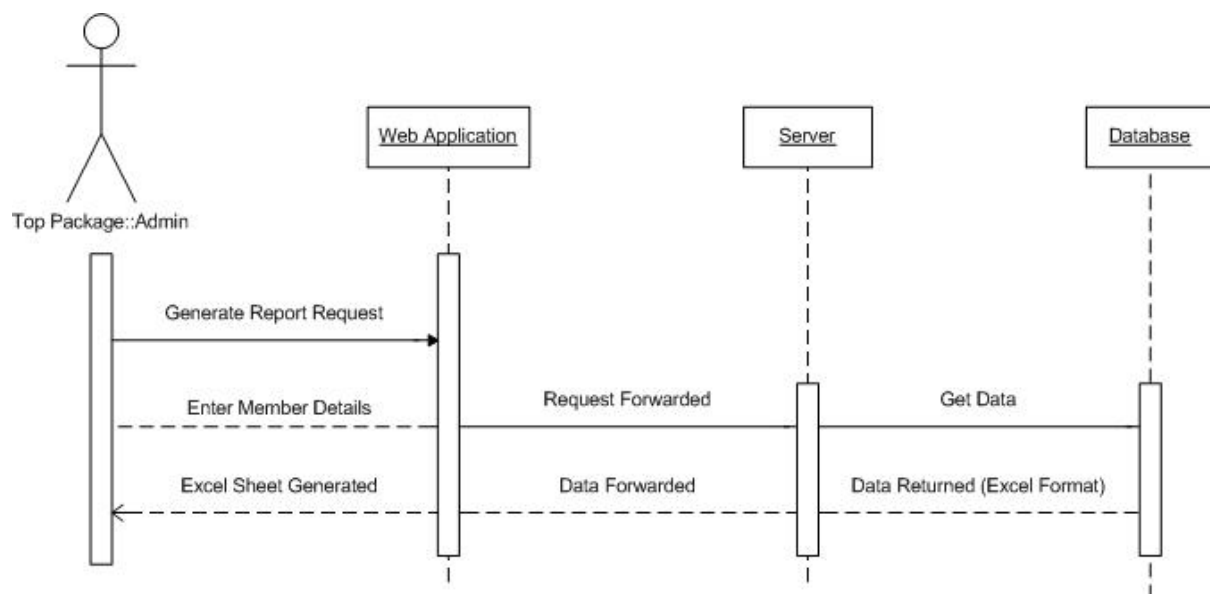
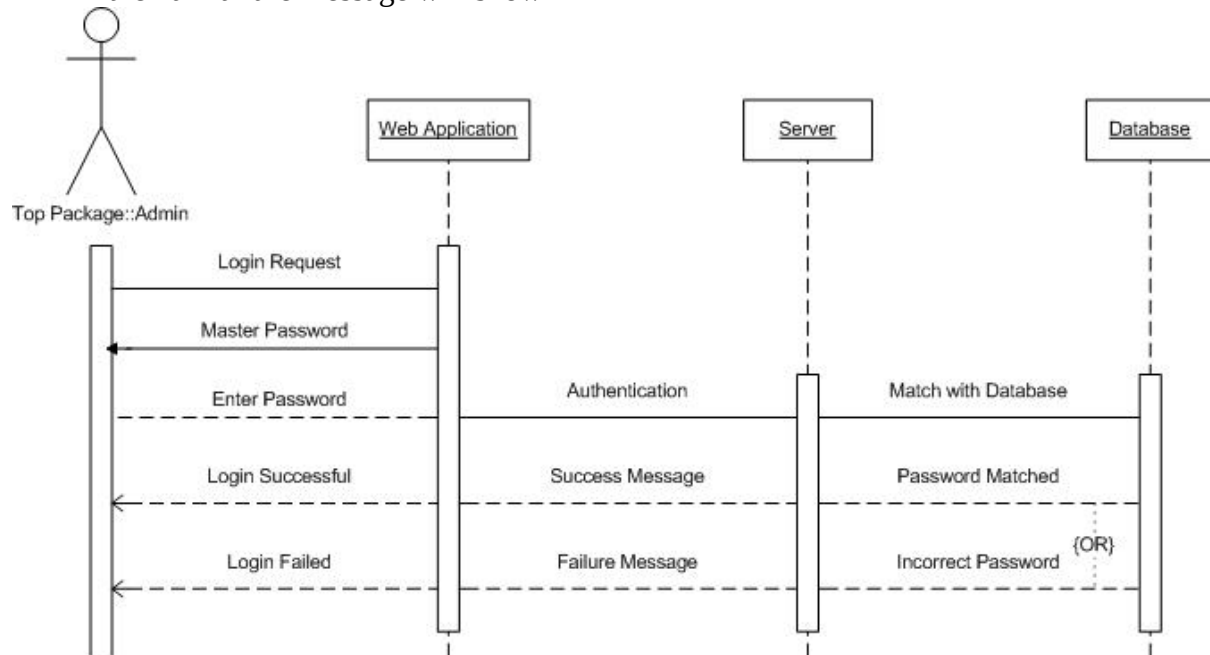
1. The customer can Create, Login, search View and Place orders. One customer can have different orders.
2. The Mess Management can Create, Login, add items to the Menu, and can Edit and Delete the items. The Mess Receives the orders placed by the customers. One Mess can Receive different orders.
3. An order consists of its ID number, The User name, Bill amount, Email of the Customer and Delivery. One Order can have many Food items.
4. The Food items consists of the Name, ID, Price and Category.
5. One Customer can pay one payment amount which will be driven from the orders.
6. There will be two payment opens, either pay through online or choose Cash on Delivery.

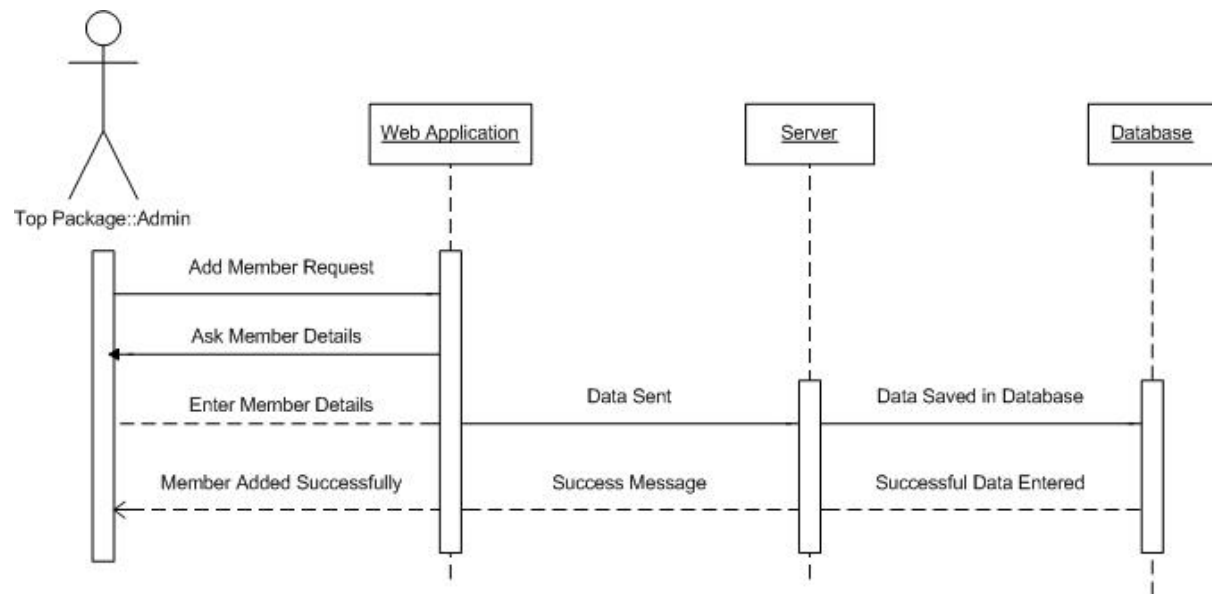


4.5. Sequence / Collaboration Diagram

Sequence Diagram shows the view of how the User will interact to the system and through what sequence He will use the system.

1. The User can Login to their account by entering the password by accessing through the Web Application.
2. The web application will be redirected to the server for authentication of the users account. For the authentication, the server will look forward in the stored database.
3. If the user account exists, the user will be logged in and if the account does not exists then an failure message will show

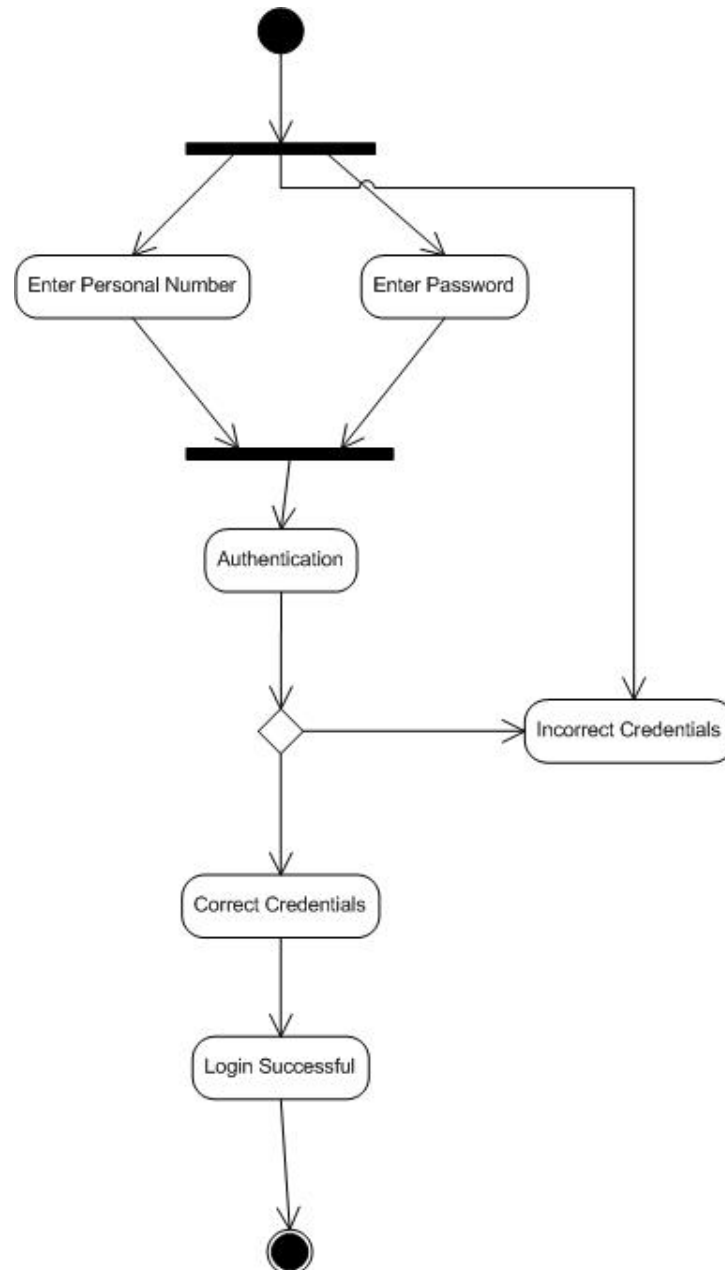




4.6. Activity Diagram

An Activity diagram shows the activity states of a system about how the system will be performing its activities.

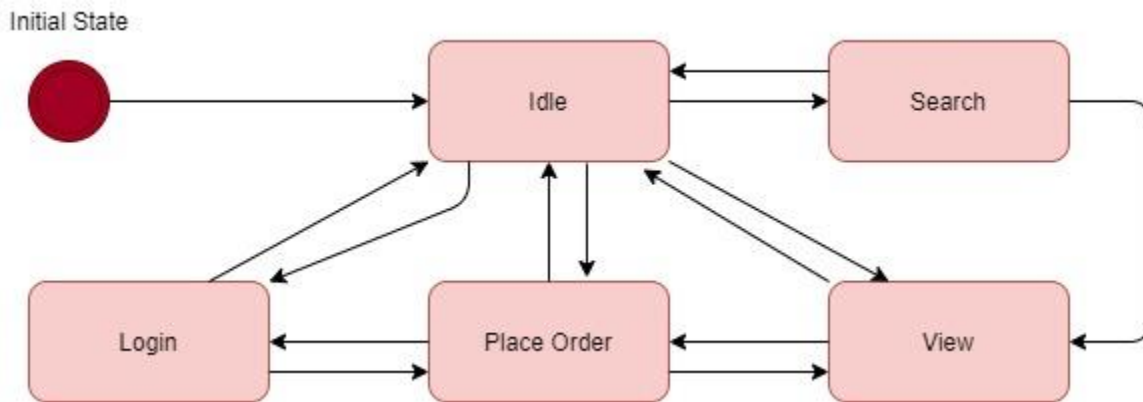
1. The system will provide a option whether login through Phone number or through a Password while logging in.
2. After entering details, the system will take a break and will authenticate the entered details from the database server.
3. If the details are correct, the system will Log in. However, if the details are wrong, the system will show a Incorrect message and will go back to the start state to the Log in Page.



4.7. State Transition Diagram

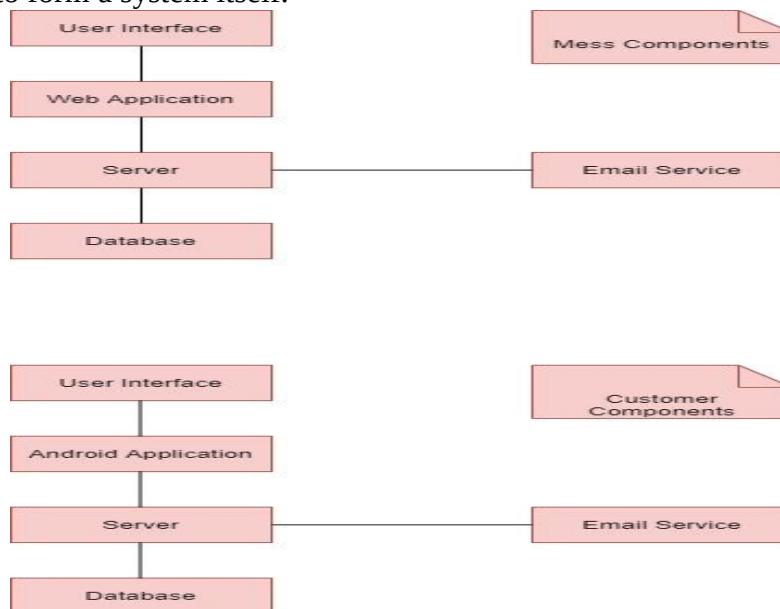
State Transition Diagram shows a representation of how the system will be performing.

1. From the initial state, the system remains idle as no work is being done.
2. When the user searches for an item, the system moves from its idle state to Search State.
3. When the user searches, the system shows the items which were being searched.
4. The system can move to Login State after its idle state, and after login state the system can go back to idle state or can go to the placing order state.



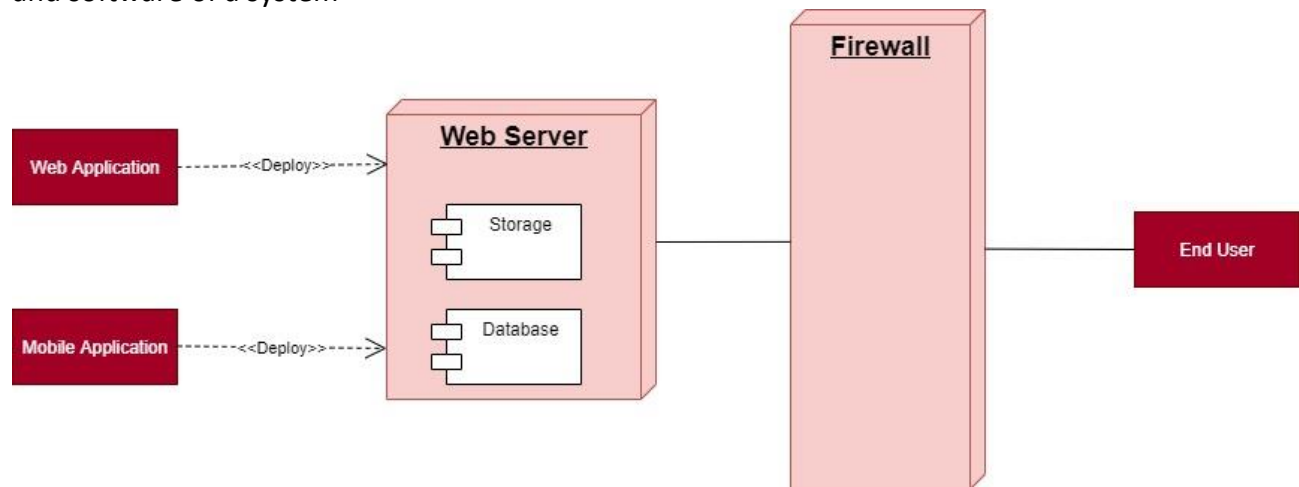
4.8. Component Diagram

Component diagram shows the components include in the system in order for its to be working. The component diagram is an important diagram as it shows every component of the system which combines to form a system itself.



4.9. Deployment Diagram

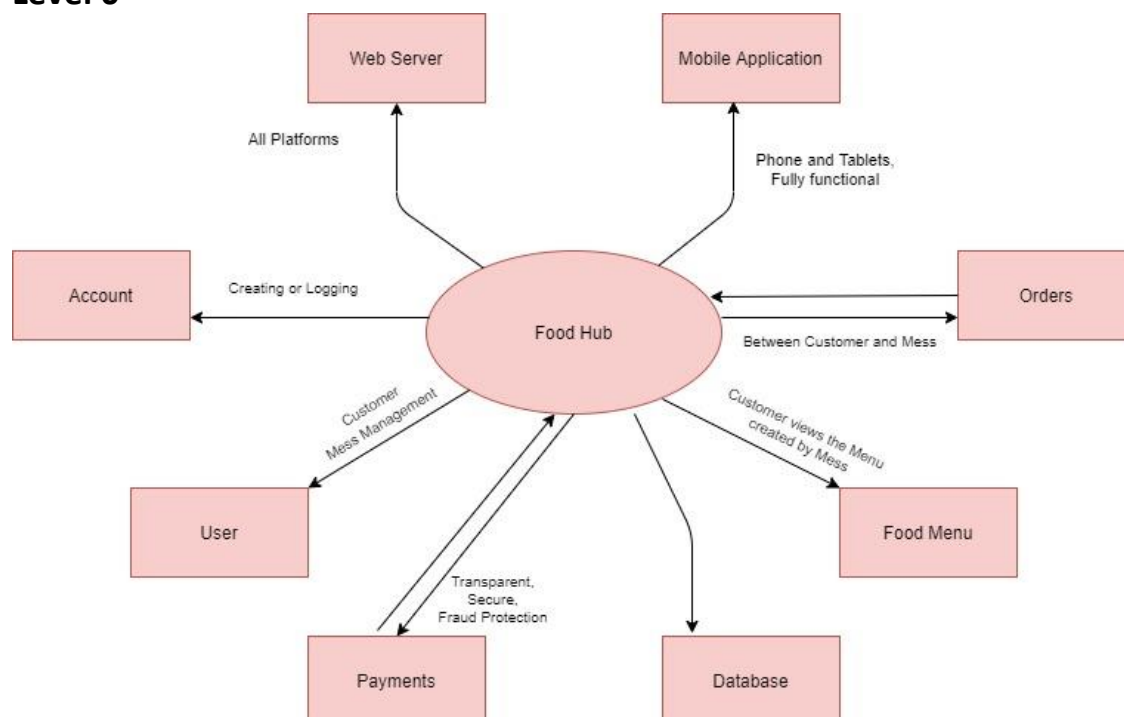
Deployment Diagram is of how the system will be deployed and how the user will be using it. including nodes such as hardware or software execution environments, and the middleware connecting them. Deployment diagrams are typically used to visualize the physical hardware and software of a system



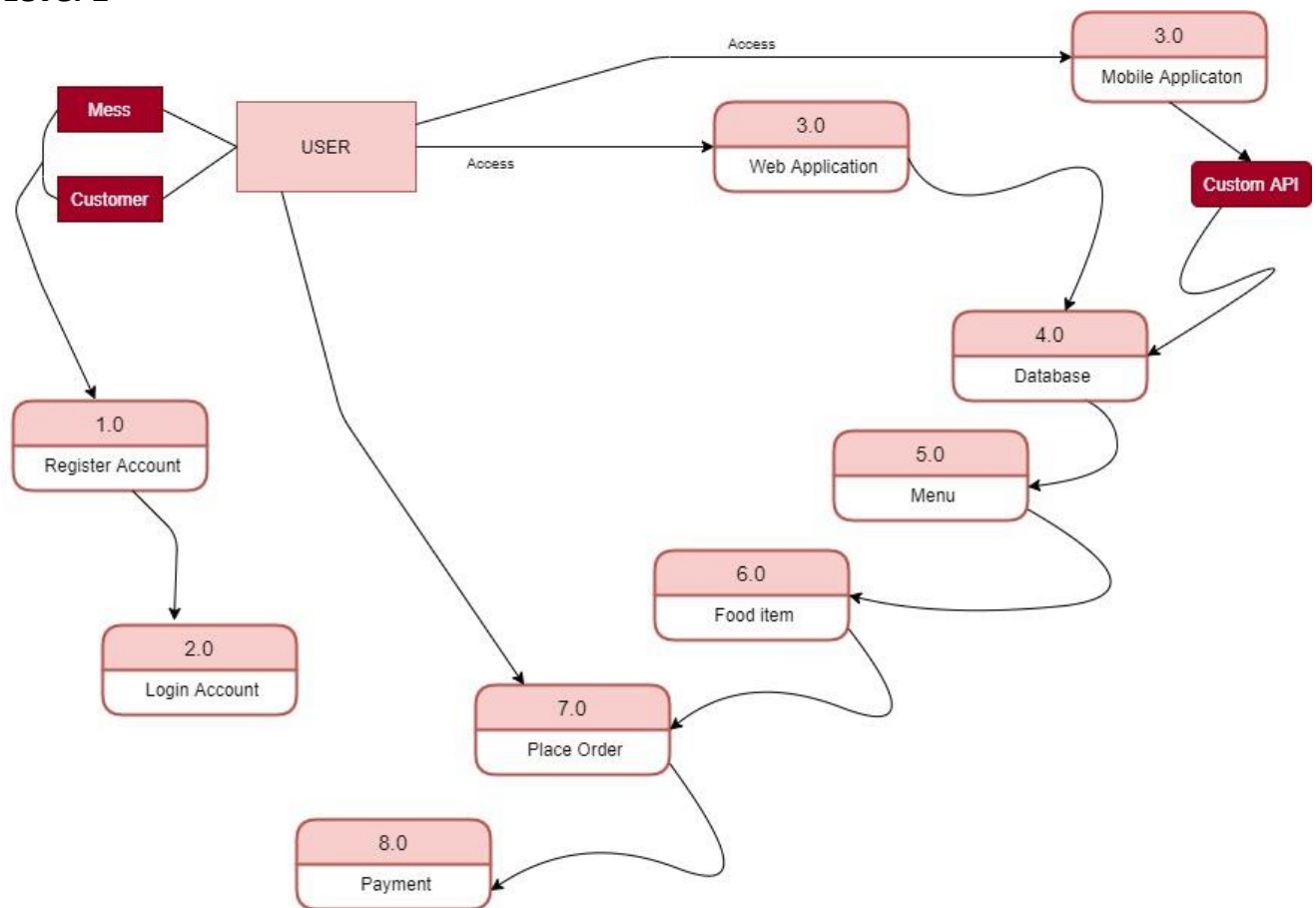
4.10. Data Flow diagram

A data-flow diagram is a way of representing a flow of data through a process or a system. The DFD also provides information about the outputs and inputs of each entity and the process itself. A data-flow diagram has no control flow, there are no decision rules and no loops.

Level 0



Level 1



Chapter 5

Implementation

Chapter 5: Implementation

This is a task with the goal to build up an essential site where a user is given a platform where customer can use our services within a city. This report will talk about every one of the fundamental advances to make and represent a web site and Application tools and techniques.

5.1. Important Flow Control/Pseudo codes

- Services that this application will provide are the following:
- There are two types of users.
- 1.Mass Management can create there account
- And can make food listing menu and there prices
- 2.customer who wants to order
- Can view list and make order through app

5.2. Components, Libraries, Web Services and stubs

Properly Object Oriented Website:

Food Hub is easy to use. If a user wants to use its services then he/she must be login on our application before their order.

Mobile Friendly Experience:

Today, the world of technology is evolving quickly; more and more people have been using their mobile devices to get any information and using any customers related services. Our Food Hub website and Application is mobile friendly. This website mobile navigation is easy to use.

5.3. Deployment Environment

In Deployment environment we work on android studio for build our android based application

5.4. Tools and Techniques

We use this editor/tool for our project

- Visual studio

We use these technologies for front end Development

- HTML
- CSS
- Javascript
- Bootstrap

We use these technologies for back end development and Database.

- PHP
- SQL database

5.5. Best Practices / Coding Standards

- No coding replication
- Function and classes are always commented showing expected input and output.

Chapter 6

Testing and Evaluation

Chapter 6: Testing and Evaluation

In this phase we will test the iteration produces in each cycle. We will be testing the iterations by integration testing and regressing testing. This is done to make sure that the iterations work the way we intended them to work when producing final result.

This model also provides the facility of adding additional functionalities at a later date due to iterative layout of the model.

6.1. Use Case Testing

To verify that by clicking Signup button username, password, email, phone# and address store in users detail in database.

Use Case Testing is a functional black box testing technique that helps testers to identify test scenarios that exercise the whole system on each transaction basis from start to finish.

6.2. Equivalence partitioning

1. User name is alphabetic. (valid)
2. User name is not alphabetic. (invalid)
3. Password is equal to 6 or greater than 8 characters in length. (valid)
4. Password is 2 to 15 characters in length. (invalid)
5. Password include one uppercase letter & one lowercase letter. (valid)
6. Password include '_'. (invalid)
7. Email without '@' and '.com'. (invalid)
8. Email with '@' and '.com'. (valid)
9. Phone# only be number. (valid)
10. Phone# with alphabetic. (invalid)
11. Search product with alphabetic. (valid)
12. Search product with special character (/ , % , ^ , # , @). (invalid)
13. Empty field (invalid)

6.3. Boundary value analysis

Sr.		Partition 1	Partition 2	Partition 3
1.	Password	Less than 8 character	1 – 8 character	9 – 12
2.	Phone	<=0	1 - 11	9 – 12

6.4. Data flow testing

The relationship between one entity and another while performing a specific task in during data flow. Such as between the sign up and registration process etc

6.5. Unit testing

In unit testing we have testified our different panel codes individually by performing different tests and by executing them individually, separately on different computers and they were successfully executed and they performed well.

6.6. Performance testing

System response time of all functionality testing

Performance testing (System response time testing)

Test result

Required performance

The required response time of the system is less than 7 seconds.

All functionality of system response accordingly.

System performance

The actual performance response time of system is 5 seconds.

Test passed

6.7. Stress Testing

Stress testing is a process of measuring the stability and reliability of the system.

Stress testing used to check the ability of system to handling the bugs and errors.

Stress testing used to test the system's ability of error handling during extremely heavy load.

To ensure that our system behaves normal during the stress we overload our system and examine the system's behavior in extreme circumstances. The Super cab behaves normally in unfavorable conditions and all of its features and functions perform their functionality.

Through this testing, we monitor our system and make sure our system

- System should save all data before crashing
- Make sure the connection from server
- If server no response, wait on login current page
- Make sure the saving of data and then show success message after operation done
- System should do deleting and updating of records correctly.

Chapter 7

Summary, Conclusion and Future Enhancements

Chapter 7: Summary, Conclusion & Future Enhancements

7.1. Project Summary

In this documentation, we explained our project with details and diagrams. Every step was more challenging than the previous one but it helped us in clearing our obscurity. Despite of the fact there are many platforms that are failed to provide these facilities; our system will provide all those facilities which a member needs in a food management system. This system will help the organization in such a way that it will save time for the members and minimize human errors in currently out-dated manual file based mess management system.

7.2. Achievements and Improvements

The biggest achievements here are that we were able to compile all we learned in four years of studying BSIT and apply it to this project. We learned software architecture design techniques, android studio, UML modeling, project management, software quality assurance, testing and much more, and were able to apply it all in this project. The next big achievement is the things we learnt during this project. New languages, frameworks, libraries, data handling techniques and API. All that will be useful for us in our futures. We were able to connect all our products via API. Change in data in one place will result in change in all of our products

7.3. Critical Review

The critical element of our system is: Is our system is fast moving, give user friendly interface, not very costly, fulfill the specific requirements of the users at a runtime, has all the features which any organizations can require. To manage all these aspects it took lot of time and hard work

7.4. Lessons Learnt

- ✓ Use case description took a lot of time but we were determined to our goal.
- ✓ We have never worked on collaboration diagram before, so we have to understand it from tutorials and that took a lot of time too.
- ✓ We also learned Microsoft Visio and Microsoft Project. In addition to these we learned various online tools like 'Draw.io' and 'Diagramata'.
- ✓ We implemented our conceptual understanding of subjects like Software Requirement Engineering and Introduction to Software Engineering.

7.5. Future Enhancements/Recommendations

This project can be implemented in all the other organizations which have a mess system, just with a few modifications. More and more features can be added into these applications for the custom needs of an organization like complaint section, party bookings, mass ordering etc. This project can also be enhanced on a hardware level as well by including the vehicle registration plate recognition, voice and face recognition devices at the takeaway lot.

Appendices

Appendix A: User Manual

In this appendix section we describe the different phases of user interface and also describe how user can use our project

Appendix A: Food Hub

A.1. Registration :

Sign up and registration

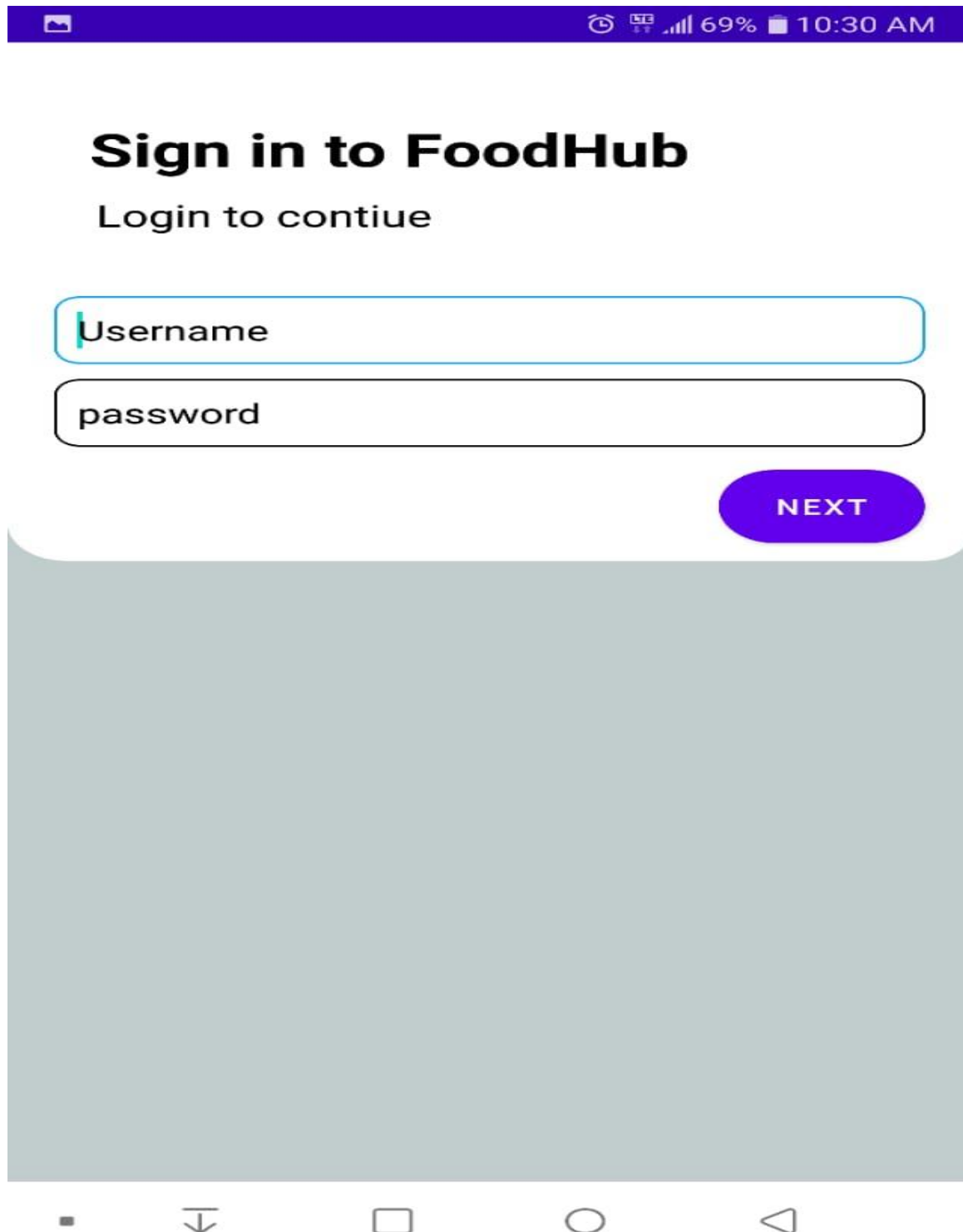
Sign up to FoodHub
Signup to continue

Enter Your Name

Enter Your Mail

Password

NEXT

A.1.1. LOGIN:

The image shows a mobile application interface for logging into 'FoodHub'. At the top, there is a purple status bar with icons for a home screen, alarm, LTE signal, 69% battery, and the time 10:30 AM. Below the status bar, the title 'Sign in to FoodHub' is displayed in a large, bold, black font. Underneath the title, the text 'Login to continue' (note the typo) is shown in a smaller black font. There are two input fields: the first is labeled 'Username' and the second is labeled 'password'. Both fields have a light blue border. To the right of the 'password' field is a purple button with the text 'NEXT' in white. Below the input fields and the 'NEXT' button is a large, solid light gray rectangular area. At the very bottom of the screen is an Android-style navigation bar with five icons: a square, a downward arrow, a square, a circle, and a triangle.

Appendix B: Administrator Manual

In this appendix section we describe the different phases of admin interface and also describe how admin can manage the system.

Appendix C: Information / Promotional Material

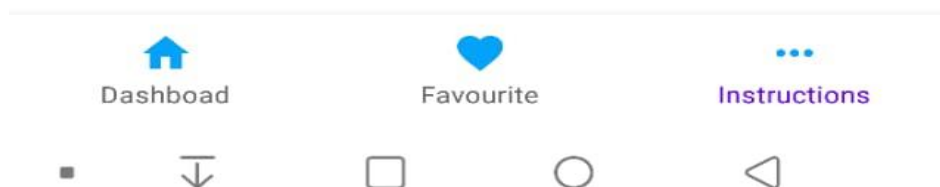
In Appendix we add promotional material about our application and application. We add these for promote our application.



Instructions:

App role is growing every year as smartphone penetration in the world increases by the minute, consumers have taken an acute preference for their mobile devices – using it for almost everything, including ordering their food!

App role is growing every year as smartphone penetration in the world increases by the minute, consumers have taken an acute preference for their mobile devices – using it for almost everything, including ordering their food!



References

References:

1. Doug H Channel <https://goo.gl/wN3c3p>
2. https://www.visual-paradigm.com/support/documents/vpuserguide/94/2577/7025_drawingseque.html
3. Web application <https://www.draw.io/>
4. courses.cs.washington.edu/courses/cse403/11sp/lectures/lecture08-uml1.pdf written by Martin Fowler, Kendall Scott - 2004
5. Agile Model https://www.tutorialspoint.com/sdlc/images/sdlc_agile_model.jpg
6. SDLC https://www.tutorialspoint.com/sdlc/sdlc_agile_model.htm
7. Agile Modelling in SDLC https://en.wikipedia.org/wiki/Agile_modeling
8. Report Template <https://sites.google.com/a/ciitlahore.edu.pk/cs-fyp/home>
9. <https://docs.google.com/viewer?a=v&pid=sites&srcid=Y2lpdGxhaG9yZS5lZHUucGt8Y3MtZnlwfGd4OjY1MzlmMjYwZjRhNzdmMjM>
10. <https://online.visual-paradigm.com/diagrams/features/uml-tool/>
11. https://www.lucidchart.com/pages/examples/uml_diagram_tool
12. <https://www.engro.com/>
13. <https://www.engrofertilizers.com/>
14. <https://blog.cimpl.com/billing-errors-are-you-safe>
15. <https://expertsystem.com/machine-learning-definition/>
16. [https://en.wikipedia.org/wiki/Scope_\(project_management\)](https://en.wikipedia.org/wiki/Scope_(project_management))
17. <https://reqtest.com/requirements-blog/requirements-analysis/>
18. <https://www.investopedia.com/terms/s/stakeholder.asp>

19. https://en.wikipedia.org/wiki/Functional_requirement
20. https://en.wikipedia.org/wiki/Non-functional_requirement
21. <https://www.guru99.com/traceability-matrix.html>
22. <https://www.usability.gov/how-to-and-tools/methods/use-cases.html>
23. https://en.wikipedia.org/wiki/Use_case_diagram
24. https://en.wikipedia.org/wiki/Work_breakdown_structure
25. <https://www.workbreakdownstructure.com/>
26. <https://www.geeksforgeeks.org/unified-modeling-language-uml-activity-diagrams/>
27. <https://developer.ibm.com/articles/the-sequence-diagram/>
28. <https://www.quora.com/What-are-the-best-ways-to-diagram-software-architecture>
29. <https://www.projectmanager.com/gantt-chart>
30. <https://searchsoftwarequality.techtarget.com/definition/collaboration-diagram>
31. <http://softwaretestingfundamentals.com/system-testing/>
32. <https://www.guru99.com/test-case.html>