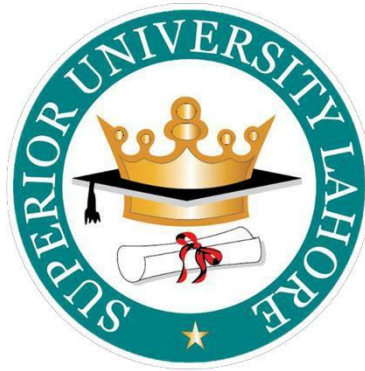


# SUPERIOR UNIVERSITY LAHORE



Faculty of Computer Science & IT

## FINAL YEAR PROJECT      FINAL YEAR PROJECT PROJECT PROPOSAL & PLAN

### Meet Online

#### Project Team

Student Name	Student ID	Program	Contact Number	Email Address
Hamza Hafeez(Leader)	BCSM-F15-296	BSCS	03040464045	hamzahafeez64@gmail.com
Shameekh Zafar	BCSM-F15-122	BSCS	03204795778	shameekhz@gmail.com
Waqar Iftikhar	BCSM-F15-335	BSCS	03034159697	waqar.chauhan1@gmail.com

**Mahrukh Batool**  
**(Senior Lecturer)**

# Project Report

## MEAT ONLINE

### Change Record

Author(s)	Version	Date	Notes	Supervisor's Signature
	1.0		<Original Draft>	
			<Changes Based on Feedback from Supervisor>	
			<Changes Based on Feedback From Faculty>	
			<Added Project Plan>	
			<Changes Based on Feedback from Supervisor>	

## APPROVAL

---

### PROJECT SUPERVISOR

Comments: \_\_\_\_\_

---

Name: \_\_\_\_\_

Date: \_\_\_\_\_ Signature: \_\_\_\_\_

---

### PROJECT MANAGER

Comments: \_\_\_\_\_

---

Date: \_\_\_\_\_ Signature: \_\_\_\_\_

### HEAD OF THE DEPARTMENT

Comments: \_\_\_\_\_

---

Date: \_\_\_\_\_ Signature: \_\_\_\_\_

## **Dedication**

We dedicate this project to our parents, teacher and to all those who support, encourage us in the entire challenging situation which has made this possible for us to be true. Especially our supervisor has played a very important role in it. He provides us knowledge and awareness about a lot of things related to our project.

## **Acknowledgements**

All indication of recall is to All-powerful Allah, Who gave us the extent, training and mettle to finish this project and task. Working on this project was indeed a challenging task that demanded immense efforts. We truly acknowledge the cooperation and help made by Ms. Ms. Mahrukh and Mr. Iphraseab at Superior University Lahore. They have been a constant source of guidance throughout the course of this project. His support led us to complete our project. His continuous encouragement motivated us a lot to complete our project with great attention

## Executive Summary

The growing use of Internet in Pakistan provides a developing prospect for online shopping. If E-marketers know the factors affecting online Pakistan behavior, and the relationships between these factors and the type of online buyers, then they can further develop their marketing strategies to convert potential customers into active ones, while retaining existing online customers.

This project which we are made is based on E-marketing. We are providing the platform to the butcher to sell their meat online. We are going to online the butchers, because nowadays E-marketing is growing in Pakistan. Everyone buy and sell their product online. Because you directly buy the product from company not distributors, that's why online purchasing is cheaper than the market.

Customers don't want to go the outside of their homes. They want everything in home by sitting their on bed.

We are going to online the butchers, like Uber and Careem. The customers will book the butchers on Eid for qurbani.

This is the need of market no one offers this kind of services.

## Table of Contents

Dedication .....	iv
Acknowledgements .....	v
Executive Summary .....	vi
Table of Contents .....	vii
List of Figures .....	ix
Chapter 1 .....	1
Introduction .....	1
.1. Background .....	2
.2. Motivations and Challenges .....	2
.3. Goals and Objectives .....	2
.4. Literature Review/Existing Solutions .....	2
.5. Gap Analysis .....	3
.6. Proposed Solution .....	3
.1. Project Plan .....	3
.1.1. Work Breakdown Structure .....	5
.1.2. Roles & Responsibility Matrix .....	5
.1.1. Gantt chart .....	6
Chapter 2 .....	7
Software Requirement Specifications .....	7
.1. Introduction .....	8
.1.1. Purpose .....	8
People have difficulty to book the butchers on Eid days and also face issue to book fresh and halaal meat. So this is the purpose to build this application to provide the facility to the customers to book the butchers and the halaal and fresh meat just one click. We are facilitating the customers to save their time because people have too much short time to visit the market and buy the meat. ....	8
.1.2. Document Conventions .....	8
.1.1. Intended Audience and Reading Suggestions .....	8
.1.1. Product Scope .....	9
.1. Overall Description .....	10
.1.1. Product Perspective .....	10
.1.2. Product Functions .....	10
.1.3. User Classes and Characteristics .....	10
.1.4. Operating Environment .....	10
.1.5. Design and Implementation Constraints.....	10
.1.1. User Documentation .....	11
.1.2. Assumptions and Dependencies .....	11

.2.	External Interface Requirements .....	11
.2.1.	User Interfaces .....	11
.1.1.	Hardware Interfaces .....	12
.1.2.	Software Interfaces .....	12
.1.3.	Communications Interfaces .....	12
.2.	System Features.....	12
.2.1.1.	Stimulus/Response Sequences .....	13
.2.1.2.	Functional Requirements .....	13
.2.2.	System Feature 2 .....	14
.2.2.1.	Stimulus/Response Sequences .....	14
.2.2.2.	Functional Requirements .....	14
.3.	Other Nonfunctional Requirements .....	15
.3.1.	Performance Requirements.....	15
.3.2.	Safety Requirements .....	15
.1.1.	Security Requirements .....	16
.1.1.	Software Quality Attributes .....	16
.1.2.	Business Rules .....	17
.1.	Other Requirements.....	17
Chapter 3	.....	18
Use Case Analysis.....		18
3.1.	Use Case Model .....	20
3.2.	Fully Dressed Use Cases .....	21
Chapter 4	.....	32
System Design .....		32
4.1.	Architecture Diagram .....	33
4.2.	Domain Model.....	34
4.3.	Entity Relationship Diagram with data dictionary .....	35
4.4.	Class Diagram .....	36
4.5.	Sequence / Collaboration Diagram .....	37
4.6.	Operation contracts .....	38
4.7.	Activity Diagram .....	41
4.8.	State Transition Diagram .....	43
4.9.	Component Diagram .....	44
4.10.	Deployment Diagram .....	45
Chapter 5	.....	46
Implementation .....		46
5.1.	Important Flow Control/Pseudo codes .....	47
5.2.	Components, Libraries, Web Services and stubs .....	47
5.3.	Deployment Environment .....	48
5.4.	Tools and Techniques.....	48
5.5.	Best Practices / Coding Standards .....	48
5.6.	Version Control .....	49
Chapter 6	.....	
Testing and Evaluation .....		
6.1.	Use Case Testing.....	
6.2.	Equivalence partitioning .....	
6.3.	Boundary value analysis.....	
6.4.	Data flow testing .....	
6.5.	Unit testing .....	
6.6.	Integration testing.....	50

6.7.	Performance testing.....	
6.8.	Stress Testing .....	
	Chapter 7.....	
	Summary, Conclusion and Future Enhancements .....	
7.1.	Project Summary .....	
7.2.	Achievements and Improvements .....	
7.3.	Critical Review.....	<b>Error! Bookmark not defined</b>
7.4.	Lessons Learnt.....	
7.5.	Future Enhancements/Recommendations .....	
	Appendices .....	
	Appendix A: Information / Promotional Material .....	51
	Reference and Bibliography .....	52

## List of Figures

Figure 1 DTD .....	4
Figure 2 WBS .....	5
Figure 3 USE CASE .....	20
Figure 4 ARCHITECTURE DIAGRAM.....	33
Figure 5 DOMAIN MODEL .....	34
Figure 6 ERD .....	35
Figure 7 CLASS DIAGRAM.....	36
Figure 8 SEQUENCE DIAGRAM .....	37
Figure 9 ACTIVITY DIAGRAM .....	41
Figure 10 ACTIVITY DIAGRAM .....	42
Figure 11 STATE TRANSITION DIAGRAM.....	43
Figure 12 STATE TRANSITION DIAGRA 1.1 .....	43
Figure 13 COMPONENT DIAGRAM .....	44
Figure 14 DEPLOYEENT DIAGRAM .....	45

# Chapter 1

## **Introduction**

# Chapter 1: Introduction

1. We are making an application which is "Meat Online".
2. We are providing a feature in this app which is online meat order. This feature keeps help those customers who are not able to go outside and buy the meat.
3. We are providing the platform to the butchers to sell their meat online.
4. It also helps to find the butchers near to your location for bakraeid.
5. In this app, there are registered professional butchers.

## .1. Background

Our project background is related to the meat online. We are facilitating the customer to book the meat and the butcher online. Because people have problem on normal days to go outside and buy the meat and problem also on eid days to book the butchers.

## .2. Motivations and Challenges

We are facilitating the customer to book the meat and the butcher online. Because people have problem on normal days to go outside and buy the meat and problem also on Eid das to book the butchers.

## .3. Goals and Objectives

- We are facilitating the customer to booked the meat and the butcher online
- The main purpose of our system is to provide better online system for the Online meat and butchers
- This will save the visitors' time and Cost which is not able to visit the market.

## .4. Literature Review/Existing Solutions

Yes there is a lot of existing systems in Pakistan,

Like Zeenat, Careeb, Meat one and others. All companies are sell their meat. So our unique point is we are going online the butchers to sell their meat online and customers will book the butchers on Eid for qurbani. This application wills works like Zeenat and others companies app. The payment method is cash on delivery. If you don't get good healthy meat then you will

return the meat and get your cash back. And you can also book you butcher on Eid. The butcher will be on time and affordable fares. No one butcher will unprofessional. All the butchers are professionals. We are also provide their experiences which will helps you to book the butchers online for Eid. This platform will help in E-marketing in Pakistan. We will provides the all kind of meat in this app like camel, beef, chicken, mutton etc.

## **.5. Gap Analysis**

Others companies are offering online meat but no one offering the online butchers.

## **.6. Proposed Solution**

- The purpose of this system is we are going to online the butchers.
- It focuses on the online sales and purchasing of the meat and the details that how system will fulfil the needs of seller, buyers and end-users.
- The main purpose of our system is to provide better online system for the online meat and butchers. This will save the visitors' time and Cost which is not able to visit the market.
- We are making an Application which is Meat Online that helps to find the good quality of meat and the butchers near you.
- All Qasai are registered and professionals.
- This app helps you to buy online a good quality of meat which kind of meat you want.
- This is also helps you to find your nearest butchers for Eid.

## **.1. Project Plan**

A project plan is a formal document designed to guide the control and execution of a project. A project plan is the key to a successful project and is the most important document that needs to be created when starting any business project. These are following steps for completing project plan.

- **Requirement Gathering**
- **Analyses of Requirements**
- **Designing**

- **Development Testing& Deployment**

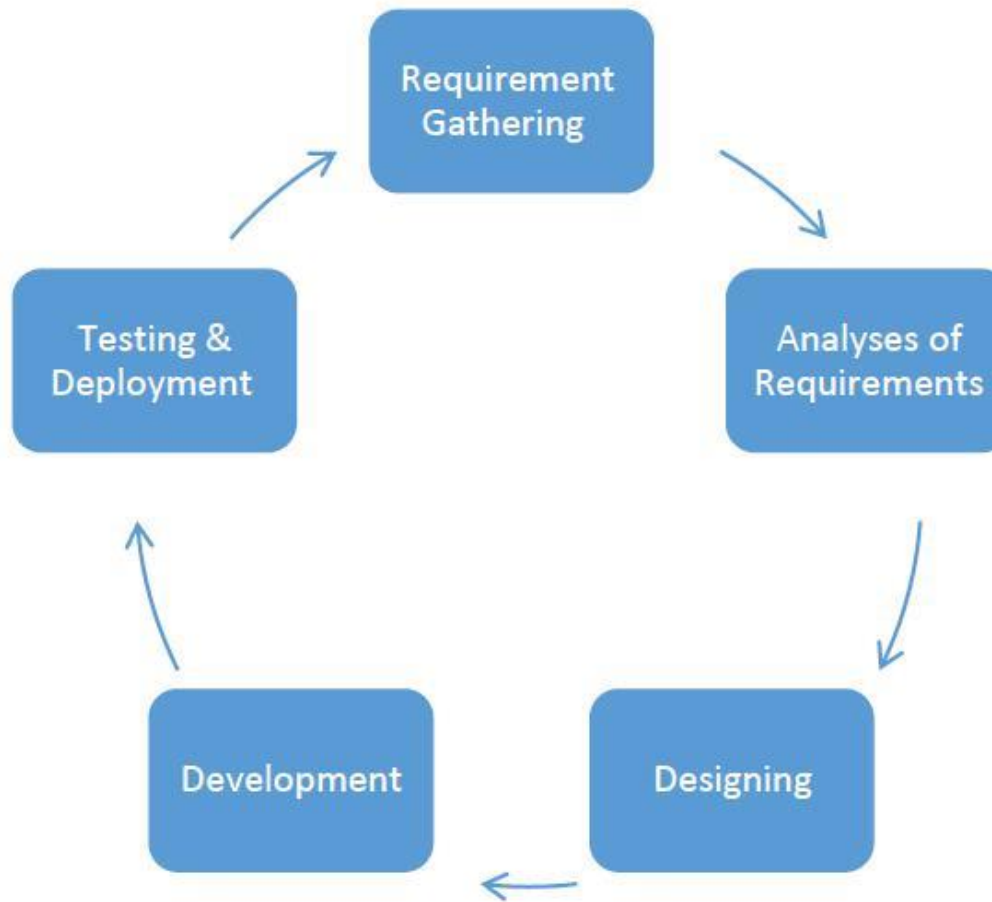


Figure 1 DTD

### .1.1. Work Breakdown Structure

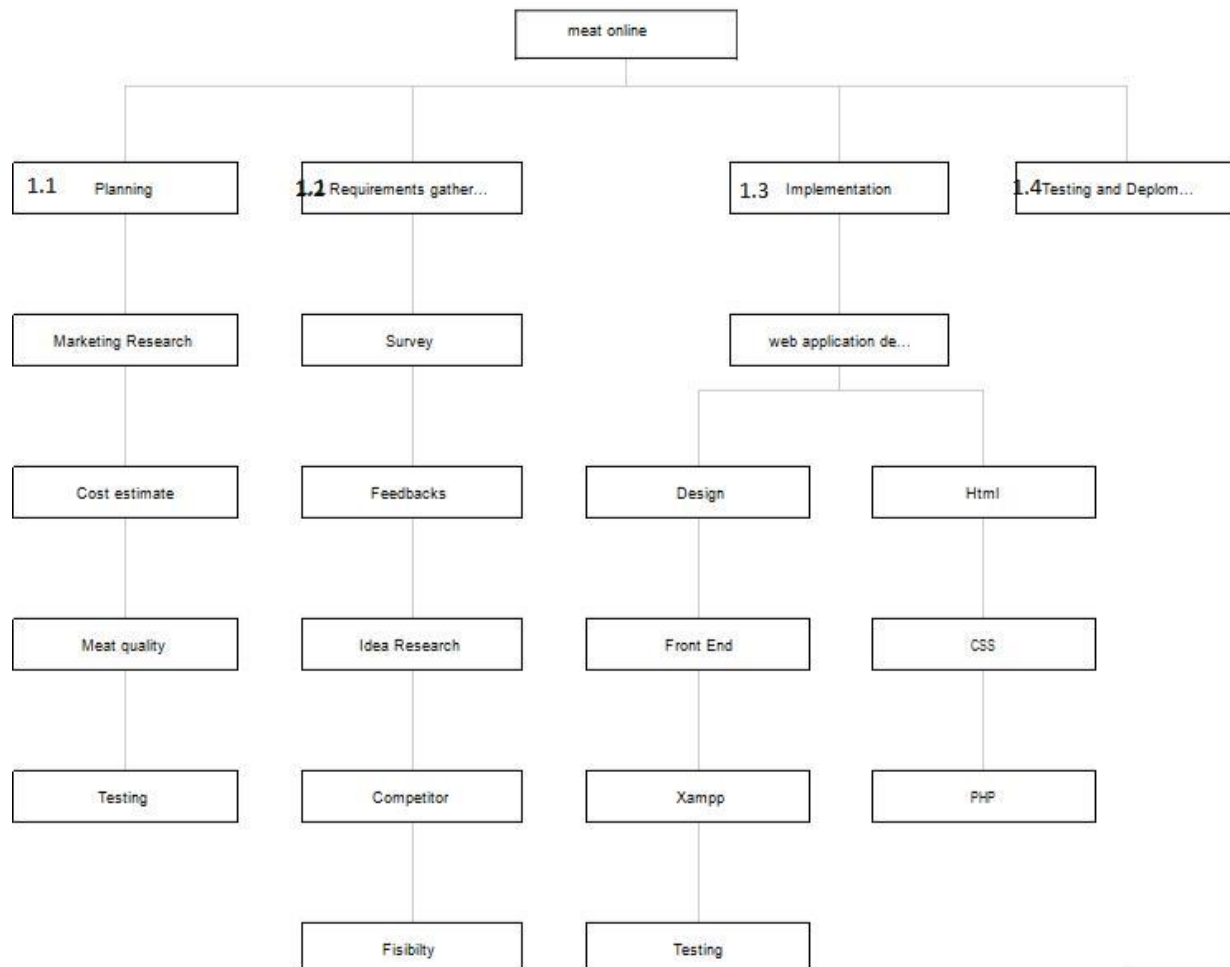


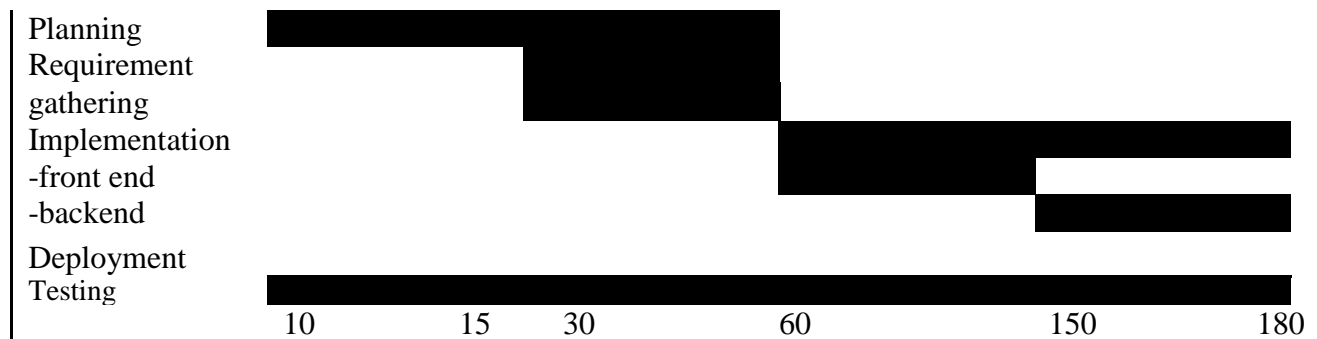
Figure 2 WBS

### .1.2. 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	Meat Online	Requirement Gathering	All working of Requirement	10-15	WaqarIftikhar Kashan Ahad
2	Meat Online	Analyses of Requirement	All working of Requirement	15-30	WaqarIftikhar Kashan Ahad
3	Meat Online	Designing	Front End Working of Project	30-60	Kashan WaqarIftikhar
4	Meat Online	Development	Website back-end Working	60-120	Ahad
5	Meat Online	Testing	Web, App	120-150	Kashan
6	Meat Online	Deployment	Web, App	150-180	WaqarIftikhar

### .1.1. Gantt chart



# Chapter 2

## **Software Requirement Specifications**

## Chapter 2: Software Requirement Specifications

### .1. Introduction

#### .1.1. Purpose

People have difficulty to book the butchers on Eid days and also face issue to book fresh and halaal meat. So this is the purpose to build this application to provide the facility to the customers to book the butchers and the halaal and fresh meat just one click. We are facilitating the customers to save their time because people have too much short time to visit the market and buy the meat.

#### .1.2. Document Conventions

- The document covers the conventions as described by the template given to us by the management

Font style	Calibri
Text size	12
Line spacing	1.5
Paragraph	Justified
URL	Colour-blue and italic

#### .1.1. Intended Audience and Reading Suggestions

- Intended audience of this report are :

1. Developer
2. Project tester
3. Customer

- Reading Suggestion: The System includes the following reading suggestions.
  - 1) Overall Description: It defines the general factors that affect the product and its requirements.
  - 2) System Features: It specify overall features available in the system.
  - 3) External Interface Requirements: It defines the hardware, software, user as well as communication interface of the system.
  - 4) Other Non-functional Requirements: It describes the non-functional requirements like Performance requirements, software quality requirements, and safety and security requirements

### **.1.1. Product Scope**

- The accuracy rate during inserting records and during processing, the system will be reliable.
- In manual system many inherit problems were existing. So that it is required to replace the manual system with a computerized system.
- A project vision is meant to be changeable as the understanding of requirements, architecture, plans and technology evolves. However, it should be changing slowly and normally throughout the earlier portion of the lifecycle. Our proposed system is very strong in implementation and will be user friendly. This system will be so easy to use that after training users can run this software easily.
- Our purpose is to provide the system with minimum level of errors, maximum level of efficiency and user friendly. Meat online is going to be very beneficial for all types of users like that seller buyer and end-user etc.

## **.1. Overall Description**

### **.1.1. Product Perspective**

All the systems which is currently working in the market they cannot offering the online butchers with the different type of meat. so this product is very useful and helpful for the customers.

### **.1.2. Product Functions**

The following list and description of the main feature and functionality of meat online.in this application customer will register itself. Customer searches the butchers. Which is nearest of them?

- Register it self
- View the meat list which customer want
- Customer book the butchers
- Customer show the profile of the butchers like experience name number and location
- Customer will book the meat

### **.1.3. User Classes and Characteristics**

Our application is user friendly. Each classes and functions are user friendly. The Technical and non-technical person will also use this product easily.

### **.1.4. Operating Environment**

- **IDE** Our software will be developed in visual paradigm community addition.
- The language which will we coded isHTML , PHP , JAVASCRIPT , CSS
- This is web based software system. This will be run in different web browsers.

### **.1.5. Design and Implementation Constraints**

- This product is initially being developed for a non-profit organization with a limited budget, and therefore is constrained to low-cost methods of implementing the system. Since the system will be implemented, the development team will be locked into using visual paradigm community addition do the majority of the implementation work.

- The team has selected JavaScript, php, css, html, language to implement the Code Behind functionality of the application. The team will be using the visual paradigm community addition Integrated Development Environment to edit and create the application.
- The customer's organization will be solely responsible for maintaining the software after final product has been delivered. Since the organization will not have a large budget for maintenance, strict and thorough documentation needs to be generated to provide a third party with a method of understanding and maintaining the internal aspects of the system.
- *Customer*

### **.1.1. User Documentation**

As stated in a previous section, extensive online help will be available at all times when using the system. This online help will guide the user.

### **.1.2. Assumptions and Dependencies**

- It assumed that this application will be developed in visual paradigm community addition.
- It is assumed that this application will helped to the customer to book the butchers for bakraeid.
- It is assumed that this application will helped the customer to find a good healthy meat
- It assumed that this application willintract with the SQL server.

## **.2. External Interface Requirements**

### **.2.1. User Interfaces**

- On the first the user will see 2 input fields. One will be username. And the other will be password. Both will be provided to the user. After entering the data in the fields the user will be able to access the features. With the help of application use can get notification of

all the update regarding. After creation and login into the application the user will see the drop down menu to see the type of meat. And after choosing the meat type then is fill the fields. And other button will show the butchers list.

### **.1.1. Hardware Interfaces**

- LAN
- SWITCHES
- ROUTER WIRES
- INTERNET
- DEVICES
- LAPTOP

### **.1.2. Software Interfaces**

Our application is web based application. When the customer want to access the application he must be interact with the browser like chrome and others. We use the SQL queries for database in this project. The tool which we will use for this application is visual paradigm community addition. Our application have two main purposes, one of the purpose of this application is, book the butchers and different type of meat just on click.

### **.1.3. Communications Interfaces**

The customer will interact this application with using different browser. And the customers will contact the butchers and the delivery boy with the help of text messaging and customers and the company will contact with the help of e-mail.

## **.2. System Features**

Following are the main system features of the Kasai online system and also for the mobile application.

- **User Registration, Login and Logout**

- **Description and Priority:**

This system feature describes user registration, login and log out successfully.

- Identifier SF-001
- Title User registration, login/logout

- Dependencies Req-SF-1, Req-SF-2, Req-SF-3
- Priority High

### .2.1.1. Stimulus/Response Sequences

Actor Action	System Response
User clicks on the search button to search Kasai.	The system will generate a list of searched kasai.
Customer click on send Select to send the order.	The system will submit an order to the kasai.
Customer click on kasai details to check kasai's detail.	System will generate kasai detail page.

### .2.1.2. Functional Requirements

Function requirements are following:

- System shall verify the user.
- System shall de-activate user account.
- System shall add new Product.
- System shall update Product.
- System shall add Kasai.
- System shall add payment.
- System shall remove Place order.
- System shall remove booked Kasai.
- System shall add booked Kasai.
- System shall show select Item.
- System shall show product history record
- System shall show Kasai details

## .2.2. System Feature 2

- **Manager panel**

- **Description and Priority:**

This system feature describes about working of the Manager in the system like verify user information, add, delete and update system.

- Identifier SF-002
- Title Manager Panel
- Dependencies REQ-SF-1, REQ-SF-2, REQ-SF-3, REQ-SF-4, REQ-SF-5,
- Priority High

### .2.2.1. Stimulus/Response Sequences

Actor Action	System Response
Manager login in admin panel.	The system will check data and generate the login page.
Manager clicks on kasai detail to check kasai details.	The system will generate kasai adds page.
Manager click on users to check user detail.	The system will display user's information page.

### .2.2.2. Functional Requirements

Function requirements are following:

- System shall verify the user.
- System shall de-activate user account.
- System shall add new Product.
- System shall update Product.
- System shall add Kasai.
- System shall add payment.
- System shall remove Place order.
- System shall remove booked Kasai.
- System shall add booked Kasai.

- System shall show select Item.
- System shall show product history record
- System shall show Kasai details

### **.3. Other Non-functional Requirements**

Non-functional requirements (NFA) are also important as functional as functional requirement. They are also called quality attributes. Following are the non-functional requirement of Kasai online system.

#### **Performance Requirements**

- This system isn't break down any reason.
- The improvement of the product will be founded on the protest arranged model.
- The execution of each module must be in the better way.
- The System should be able to service multiple terminal connections simultaneously.
- Maximum output can be achieved with availability.
- No Confinement is connected to the number of questions performed to be a customer.
- Clients are served in First-Come First-served module.
- Product sending to the manager. This module working without breakdown any issues.

#### **.3.1. Performance Requirements**

Performance of every function of this application will high. The server of this web application will very fast to increase the speed of browsing.

#### **.3.2. Safety Requirements**

- In the system, backup database is present for user data automatically.
- Safety disclaimers will be given to both manager and customer.
- If bad language is used by manager and customer, then the account will be removed from the system.

### **.1.1. Security Requirements**

- The user of the system can only access their own data by their credentials.
- Any other user should not be able to see and access to other user data and product.
- Customer's history can only view by manager which kasai selected.
- User password will be encrypted in the database of the system.
- The doctor online system user SSL connection to transfer private data of the user easily.
- In this software login and logout will be secure

### **.1.1. Software Quality Attributes**

- **Usability**

The Manager online system interface will be user-friendly by providing the user instruction on the spot and easy to use.

All website pages colours and fonts will be consistent.

- **Availability**

The system will be available 24 hours a day and 7 days a week with the help of internet connection.

- **Reliability**

The database of the system will be able to handle maximum 10000 users without delay time and failure.

The system will perform corrective action on user demand.

- **Portability**

The user login on the system will be in windows.

The user code is host 100% independent.

The system can be used on a different device like mobile, laptops etc.

- **Maintainability**

The server and client side scripting will be well maintained according to the user requirement.

## **.1.2. Business Rules**

- The system only works when an internet connection is available.
- Manager has only authority to add Kasai and customer.
- Manager can update all modules according to environmental requirement.
- Customer and Kasai will not be able to add other users. They can only suggest and invite other on the platform.
- For the misunderstanding between kasai and user website is not responsible

## **.1. Other Requirements**

Following are the other requirements for this SRS

- Technical
- Economical
- Motivational

# Chapter 3

## Use Case Analysis

## **Chapter 3: System Analysis**

This chapter is all about the use case diagrams which tells that how the system will react and flow of data in software. These diagrams are a complete structural flow and working of our web application. Fully address diagram tells in the detail about each function.

### 3.1. Use Case Model

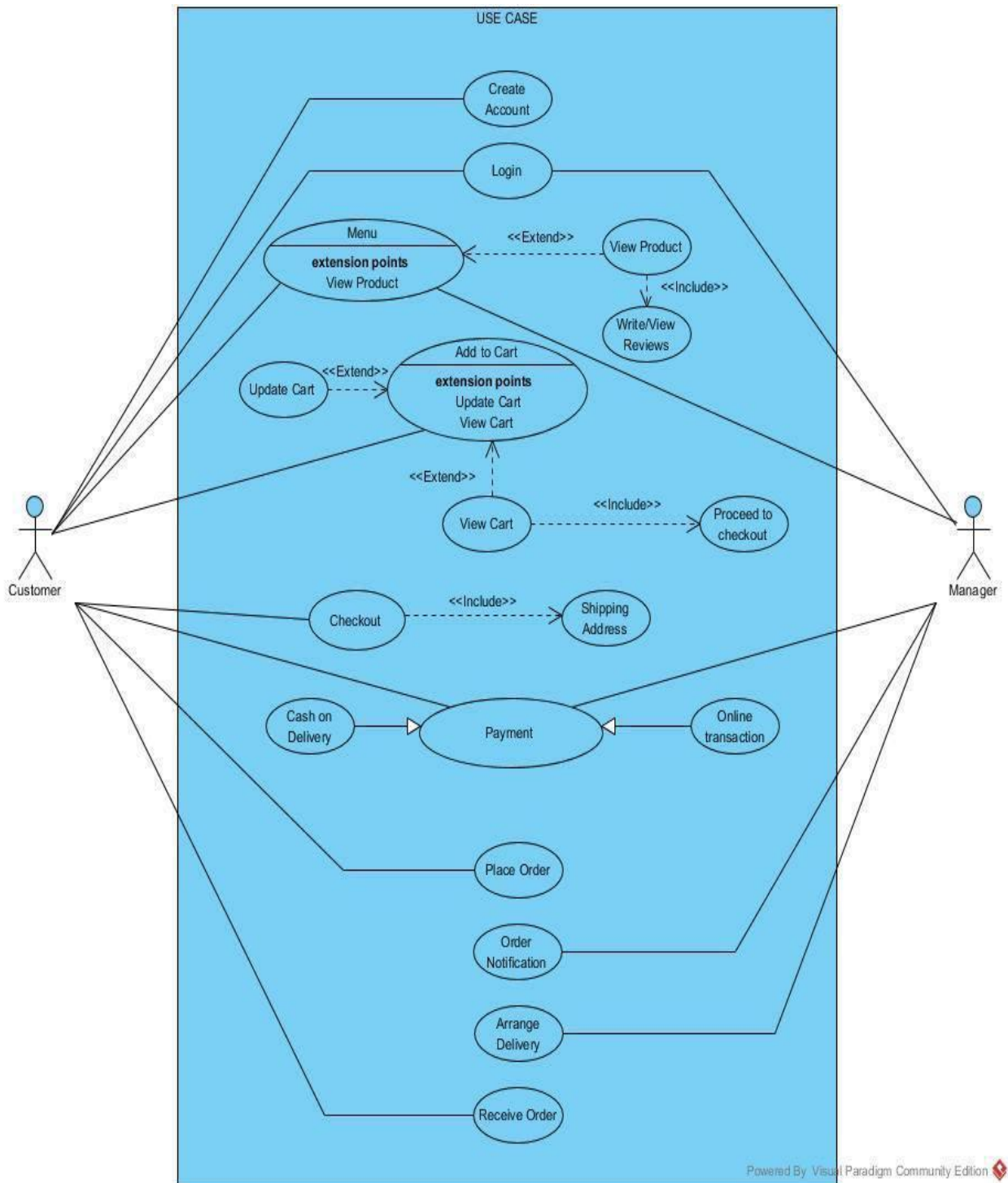


Figure 3 USE CASE

## 3.2. Fully Dressed Use Cases

### Login

<b>Use case ID</b>	UC_01
<b>Use case name</b>	Login
<b>Description</b>	This user case takes the new information (name, Password) in a form field and after validation it provides the access the user
<b>Scope</b>	E-commerce site
<b>Level</b>	User Goal
<b>Primary actor</b>	Customer
<b>Stake holder</b>	Customer: want to get access to the web site
<b>Pre-condition</b>	<ul style="list-style-type: none"> <li>• User must have the internet access.</li> </ul>
<b>Post-condition</b>	<ul style="list-style-type: none"> <li>• User successfully login and got access to purchase product</li> <li>• Information provides by the user is invalid and system show the error</li> </ul>
<b>Main success scenario</b>	<ul style="list-style-type: none"> <li>• The guest user clicks on "Create new account".</li> <li>• System displays the input field on the screen.</li> <li>• User provides his information (name, address, postal code, phone number, email id, credit card information) to form.</li> <li>• The user clicks on submit and system validates the user information.</li> <li>• System displays new account confirmation message on display.</li> </ul>
<b>Special requirements</b>	User must select an appropriate password string.

## Product

<b>Use case ID</b>	UC_02
<b>Use case name</b>	View product
<b>Description</b>	Use case will displays the price, screen shots, deals and discounts on the product.
<b>Scope</b>	E-commerce site
<b>Level</b>	User goal
<b>Primary actor</b>	Customer
<b>Stake holder and Interests</b>	<ul style="list-style-type: none"> <li>• Guest user: wants to know about the products.</li> <li>• Customer: want to see product details.</li> <li>• Admin: want to check the product details.</li> <li>• Owner: want to check the product details.</li> </ul>
<b>Pre-conditions</b>	<ul style="list-style-type: none"> <li>• User must have working internet.</li> <li>• User must search the product properly.</li> </ul>
<b>Post-conditions</b>	<ul style="list-style-type: none"> <li>• Users successfully learnt about the product details (price, screen shots, deals and discounts).</li> <li>• User purchases some product.</li> </ul>
<b>Basic flow</b>	<ul style="list-style-type: none"> <li>• User clicks on the product or enters the name in the search bar at the top of page and press goes.</li> <li>• System displays all matching products regarding user's search.</li> <li>• User clicks his desired product.</li> <li>• System displays the all details (prices, remaining stock, and varieties) about product.</li> </ul>
<b>Alternate flow</b>	<ul style="list-style-type: none"> <li>• If connection lost user will not able to view products <ul style="list-style-type: none"> <li>➤ Reset internet connection and try again.</li> </ul> </li> <li>• If searched product is not in the stock than system will show message "item not found". <ul style="list-style-type: none"> <li>➤ Correct the spelling mistakes or go through suggestions.</li> </ul> </li> </ul>
<b>Special requirements</b>	Highlight the hot deal items.
<b>Frequency of occurrence</b>	Depends upon the user.
<b>Open issues</b>	Do guest users allow to view details?

## Manager

<b>Use case ID</b>	UC_03
<b>Use case name</b>	Manage cart
<b>Description</b>	This use case will allow user to add products into cart, to delete products form cart or to purchase cart items.
<b>Scope</b>	E-commerce site
<b>Level</b>	User goal
<b>Primary actor</b>	Customer
<b>Stake holder and Interests</b>	<ul style="list-style-type: none"> <li>• Customer: wants to add products, delete products or remove products from cart.</li> <li>• Admin: wants to manage the cart.</li> </ul>
<b>Pre-conditions</b>	<ul style="list-style-type: none"> <li>• User must have signed in to manage the cart items</li> <li>• There must be some items in the cart to delete or purchase it.</li> </ul>
<b>Post-conditions</b>	<ul style="list-style-type: none"> <li>• Cart products purchased successfully.</li> <li>• Cart products deleted successfully.</li> <li>• Cart products updated successfully.</li> <li>• Error occurred</li> </ul>
<b>Basic flow</b>	<ul style="list-style-type: none"> <li>• User clicks on “Manage cart” and system displays all cart items.</li> <li>• User selects “Check out/delete/add” and system displays a form about payment/product or a confirmation message to delete item.</li> <li>• User provides his billing information/search product to add and clicks on submit/delete button.</li> <li>• System shows a notification message about confirmation.</li> </ul>
<b>Alternate flow</b>	<ul style="list-style-type: none"> <li>• If connection lost user will not able to manage cart. <ul style="list-style-type: none"> <li>➤ Reset your internet connection and try again.</li> </ul> </li> <li>• If user provide invalid information than purchase will fail. <ul style="list-style-type: none"> <li>➤ Recheck your data and try again.</li> </ul> </li> <li>• If Authentication System cannot validate the credit card than user will not able to purchase cart. <ul style="list-style-type: none"> <li>➤ Try to use a valid credit card.</li> </ul> </li> </ul>

<b>Special requirements</b>	Auto fills the field data from the user profile that are available while purchasing.
<b>Frequency of occurrence</b>	Only once.
<b>Open issues</b>	None

## Update Cart

<b>Use case ID</b>	UC_04
<b>Use case name</b>	Update Cart
<b>Description</b>	This use case will allow the registered user to update his card information such as name, password, postal code, credit card information.
<b>Scope</b>	E-commerce site
<b>Level</b>	User goal
<b>Primary actor</b>	Customer
<b>Stake holder and Interests</b>	<ul style="list-style-type: none"> <li>• Customer: wants to update his card information.</li> <li>• Admin: wants to take control over the card of the user.</li> </ul>
<b>Pre-conditions</b>	<ul style="list-style-type: none"> <li>• User must have logged in before update his card.</li> </ul>
<b>Post-conditions</b>	<ul style="list-style-type: none"> <li>• User card updated successfully and displays the latest details.</li> <li>• Updating error</li> </ul>
<b>Basic flow</b>	<ul style="list-style-type: none"> <li>• The user clicks on “My account” on the display and system shows the information about user.</li> <li>• User clicks on “Update card” and provides latest information (name, password, address, postal code, phone number, email id, credit card information) to form.</li> <li>• The user clicks on submit and system validates the user new information.</li> <li>• System displays update confirmation message on screen.</li> </ul>
<b>Alternate flow</b>	<ul style="list-style-type: none"> <li>• If connection lost user will not able to update his card. <ul style="list-style-type: none"> <li>➤ Reset your account connection and try again.</li> </ul> </li> <li>• If user does not have a registered account, then user will not able to change his information. <ul style="list-style-type: none"> <li>➤ Create a new account.</li> </ul> </li> <li>• If user provides wrong information than system will show message about invalid information. <ul style="list-style-type: none"> <li>➤ Check your information and try again.</li> </ul> </li> </ul>
<b>Special requirements</b>	Ask old password while updating password.
<b>Frequency of occurrence</b>	Depends upon the user.
<b>Open issues</b>	Does user needs to place his DP.

## Place order

<b>Use Case ID</b>	UC_05
<b>Use case name</b>	Place order
<b>Description</b>	This use case will allow user to place his order to purchase products.
<b>Scope</b>	E-commerce site
<b>Level</b>	User goal
<b>Primary actor</b>	Customer
<b>Stake holder and Interests</b>	Customer: wants to place his order online.
<b>Pre-conditions</b>	<ul style="list-style-type: none"> <li>• User must have logged in before placing order.</li> <li>• User must provide valid information.</li> </ul>
<b>Post-conditions</b>	<ul style="list-style-type: none"> <li>• Order placed successfully and system displays the order number.</li> <li>• Error occurred.</li> </ul>
<b>Basic flow</b>	<ul style="list-style-type: none"> <li>• User selects product and click on “Check out”.</li> <li>• System displays a form field about payment.</li> <li>• User provides his information and clicks on submit button.</li> <li>• Authentication System validates the information of the user.</li> <li>• System shows a notification message about confirmation.</li> </ul>
<b>Alternate flow</b>	<ul style="list-style-type: none"> <li>• If connection lost user will not able to checkout. <ul style="list-style-type: none"> <li>➤ Reset your internet connection and try again.</li> </ul> </li> <li>• If user provide invalid information than purchase will fail. <ul style="list-style-type: none"> <li>➤ Recheck your information provided and try again.</li> </ul> </li> <li>• If Authentication System cannot validate the credit card than user will not able to purchase cart. <ul style="list-style-type: none"> <li>➤ Use a valid credit card.</li> </ul> </li> </ul>
<b>Special requirements</b>	Auto fills the field data from the user profile that are available while purchasing.
<b>Frequency of occurrence</b>	<ul style="list-style-type: none"> <li>• Depends upon the user purchasing’s</li> <li>• Block user after 5 invalid attempts</li> </ul>
<b>Open issues</b>	None

## Cancel order

<b>Use case ID</b>	UC_06
<b>Use case name</b>	Cancel order
<b>Description</b>	This use case will allow user to cancel his placed order.
<b>Scope</b>	E-commerce site
<b>Level</b>	User goal
<b>Primary actor</b>	Customer
<b>Stake holder and Interests</b>	Customer: wants to cancel his placed order.
<b>Pre-conditions</b>	<ul style="list-style-type: none"> <li>• User must have logged in before placing order.</li> <li>• User must have placed an order to cancel.</li> </ul>
<b>Post-conditions</b>	<ul style="list-style-type: none"> <li>• Order canceled successfully and system displays the message.</li> <li>• No order found to be canceled.</li> </ul>
<b>Basic flow</b>	<ul style="list-style-type: none"> <li>• User clicks on “My orders”.</li> <li>• System displays a list of placed order.</li> <li>• User selects order to delete.</li> <li>• System shows a confirmation message.</li> <li>• User clicks on confirm and order deleted successfully.</li> </ul>
<b>Alternate flow</b>	<ul style="list-style-type: none"> <li>• If connection lost user will not able to cancel order. <ul style="list-style-type: none"> <li>➤ Reset your internet connection and try again.</li> </ul> </li> <li>• If user did not place any order, then he cannot be able to cancel. <ul style="list-style-type: none"> <li>➤ Users have to place at least on order.</li> </ul> </li> </ul>
<b>Special requirements</b>	None
<b>Frequency of occurrence</b>	<ul style="list-style-type: none"> <li>• Depends upon the order placed.</li> </ul>
<b>Open issues</b>	After how much time user will not allowed to cancel his order?

## Feedback

<b>Use case ID</b>	UC_07
<b>Use case name</b>	Feedback
<b>Description</b>	This use case will allow the user to delete/reply the customer feedbacks.
<b>Scope</b>	E-commerce site
<b>Level</b>	User goal
<b>Primary actor</b>	Admin
<b>Stake holder and Interests</b>	<ul style="list-style-type: none"> <li>• Admin: wants to delete customer feedbacks.</li> <li>• Admin: wants to reply the customer feedbacks.</li> </ul>
<b>Pre-conditions</b>	<ul style="list-style-type: none"> <li>• Admin must have to logged in before deleting feedbacks</li> </ul>
<b>Post-conditions</b>	<ul style="list-style-type: none"> <li>• System displays a message “feedback deleted successfully”.</li> <li>• Response generated successfully.</li> <li>• Error occurred</li> </ul>
<b>Basic flow</b>	<ul style="list-style-type: none"> <li>• User clicks on “feedbacks”.</li> <li>• System displays all feedbacks regarding to products.</li> <li>• User selects a feedback and clicks on delete/reply feedback.</li> <li>• System responses to the user choice.</li> </ul>
<b>Alternate flow</b>	<ul style="list-style-type: none"> <li>• If connection lost user will not able to feedbacks. <ul style="list-style-type: none"> <li>➤ Reset internet and try again</li> </ul> </li> <li>• If user do not have login than he cannot manage feedbacks. <ul style="list-style-type: none"> <li>➤ Admin must log in before manage feedbacks.</li> </ul> </li> <li>• There is no feedback to manage <ul style="list-style-type: none"> <li>➤ System displays a message “No feedback found”.</li> </ul> </li> </ul>
<b>Special requirements</b>	None
<b>Frequency of occurrence</b>	Depends upon the number of feedbacks.
<b>Open issues</b>	Does user allowed to delete his feedback?

## ADD Product

<b>Use Case ID</b>	UC_8
<b>Use case name</b>	Add product
<b>Description</b>	This use case will allow user to add new products to the page for sell.
<b>Scope</b>	E-commerce site
<b>Level</b>	User goal
<b>Primary actor</b>	Admin
<b>Stake holder and Interests</b>	<ul style="list-style-type: none"> <li>• Admin: wants to add new products.</li> <li>• Owner: wants to add new products.</li> </ul>
<b>Pre-conditions</b>	<ul style="list-style-type: none"> <li>• Admin must have to logged in.</li> <li>• New product should be available in the stock.</li> </ul>
<b>Post-conditions</b>	<ul style="list-style-type: none"> <li>• Product added successfully and system displays the product on screen.</li> <li>• Product not added successfully an error occurred.</li> </ul>
<b>Basic flow</b>	<ul style="list-style-type: none"> <li>• Manager opens “Inventory menu” and clicks on “Add product” option.</li> <li>• Systems show a new page about adding product.</li> <li>• Manager provides information (details, price, category) about new product and uploads the product pictures.</li> <li>• Manager clicks on “Add product”.</li> <li>• System shows a notification message about confirmation.</li> </ul>
<b>Alternate flow</b>	<ul style="list-style-type: none"> <li>• If connection lost manager will not able to add product. <ul style="list-style-type: none"> <li>➤ Reset your internet connection and try again.</li> </ul> </li> <li>• If admin is not signed in than he is not able to add. <ul style="list-style-type: none"> <li>➤ Sign in your account as an admin to add a product.</li> </ul> </li> <li>• If product item repeats than user cannot add new product. <ul style="list-style-type: none"> <li>➤ Add unique products.</li> </ul> </li> </ul>
<b>Special requirements</b>	Add 360 view of product
<b>Frequency of occurrence</b>	Depends upon the number products to be added
<b>Open issues</b>	None

## Delete product

Use Case ID	UC_9
<b>Use case name</b>	delete product
<b>Description</b>	This use case will allow user to delete products from the site.
<b>Scope</b>	E-commerce site
<b>Level</b>	User goal
<b>Primary actor</b>	Admin
<b>Stake holder and Interests</b>	<ul style="list-style-type: none"> <li>• Admin: wants to delete products.</li> <li>• Owner: wants to delete products.</li> </ul>
<b>Pre-conditions</b>	<ul style="list-style-type: none"> <li>• Admin must have to logged in.</li> <li>• There must be at least one product to be deleted.</li> </ul>
<b>Post-conditions</b>	<ul style="list-style-type: none"> <li>• Product deleted successfully and system displays the confirmation message on screen.</li> <li>• Product not deleted successfully an error occurred.</li> <li>• No product found to be deleted.</li> </ul>
<b>Basic flow</b>	<ul style="list-style-type: none"> <li>• Admin opens “Inventory menu” and clicks on “Manage product” option.</li> <li>• Systems shows new page about product management.</li> <li>• Manager searches the product to be deleted and clicks on the product.</li> <li>• System displays the details about the product on screen.</li> <li>• Manager clicks on “Delete product” and clicks on confirm.</li> <li>• System shows a notification message about confirmation.</li> </ul>

<b>Alternate flow</b>	<ul style="list-style-type: none"><li>• If connection lost manager will not able to delete product.<ul style="list-style-type: none"><li>➤ Reset your internet connection and try again.</li></ul></li><li>• If admin is not signed in than he is not able to delete product.<ul style="list-style-type: none"><li>➤ Sign in your account as an admin to add a product.</li></ul></li><li>• If there are no products in the stock than user cannot delete product.<ul style="list-style-type: none"><li>➤ Add some products to be deleted.</li></ul></li></ul>
<b>Special requirements</b>	None
<b>Frequency of occurrence</b>	Depends upon the number products in the stock
<b>Open issues</b>	None

# Chapter 4

## System Design

## Chapter 4: System Design

Here we describe that how the system behaves on user interactions with the system. In Meat online administrators have the ability to add Kasai. The admin also views doctors' detail and the user also view doctor details and booked appointment easily. The system also describes about the patient's history and previous record about his health. The user also communicates with a doctor by video and audio chat. In following use case which shows how user interacts with Meat online system.

### 4.1. Architecture Diagram

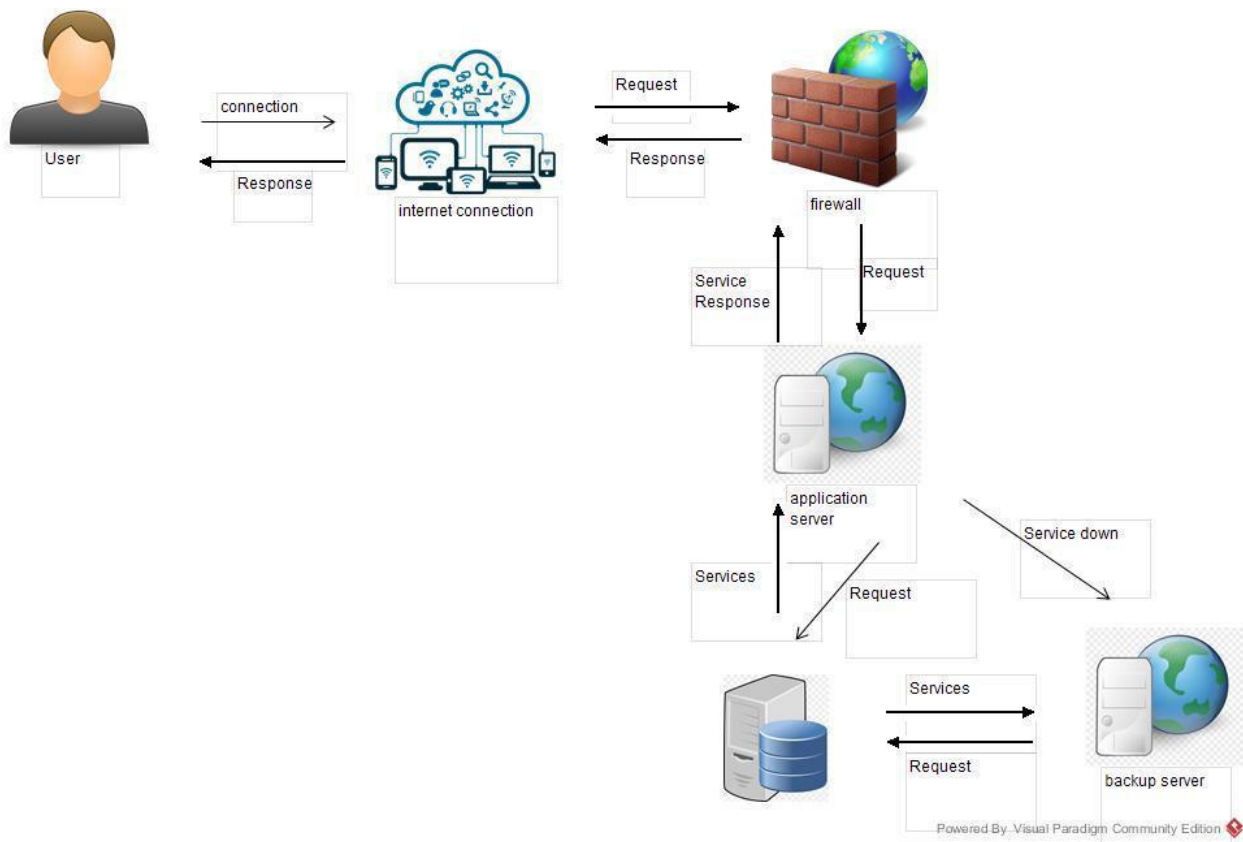


Figure 4 ARCHITECTURE DIAGRAM

## 4.2. Domain Model

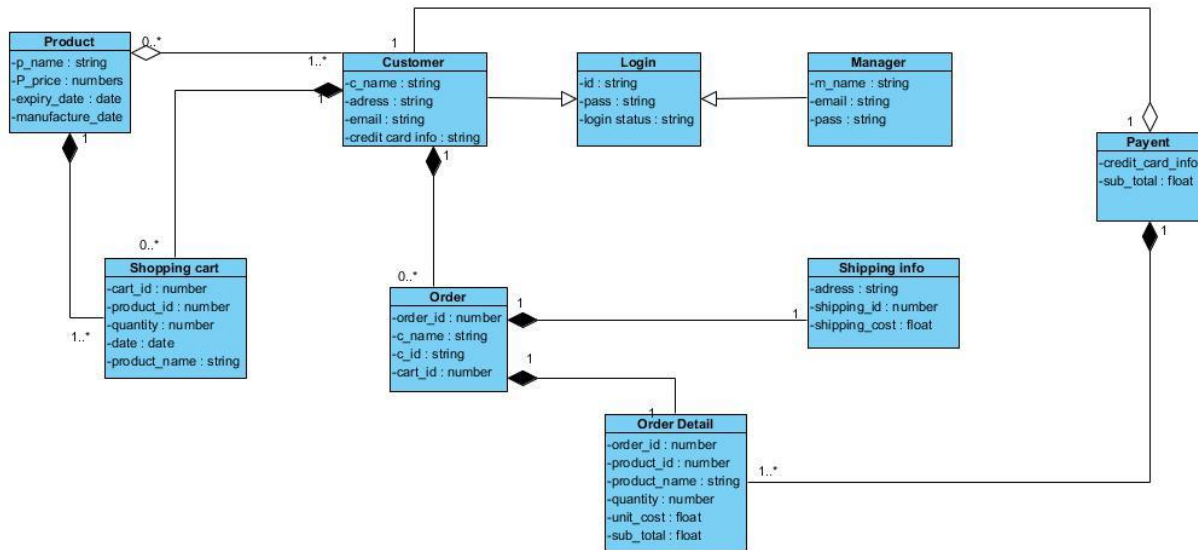


Figure 5 DOMAIN MODEL

### 4.3. Entity Relationship Diagram with data dictionary

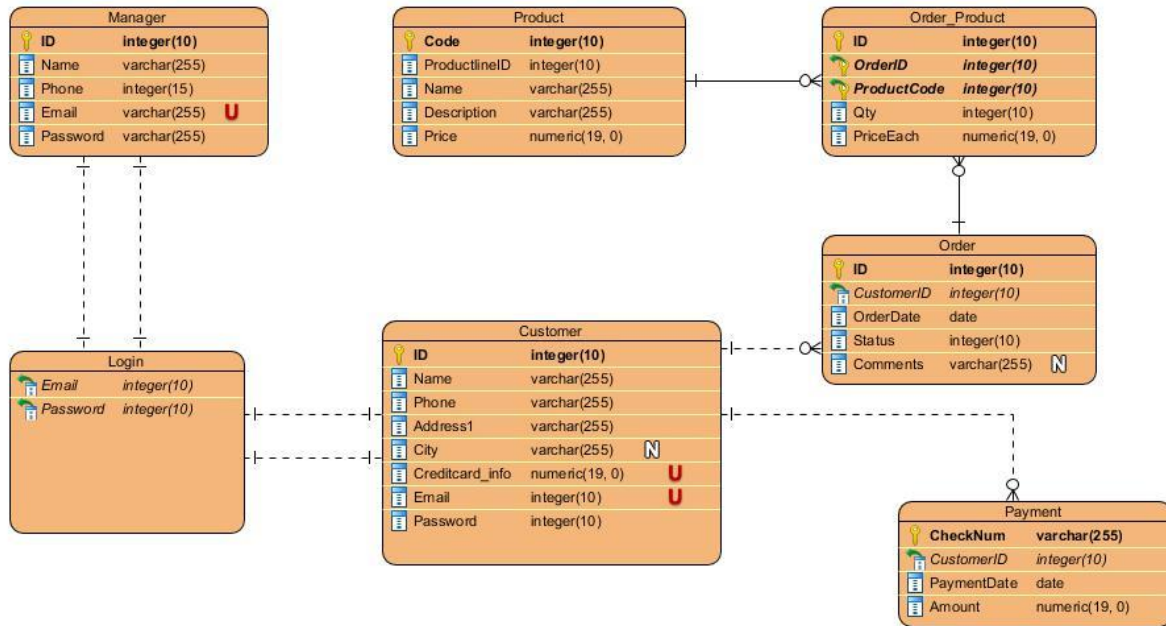


Figure 6 ERD

### 4.4. Class Diagram

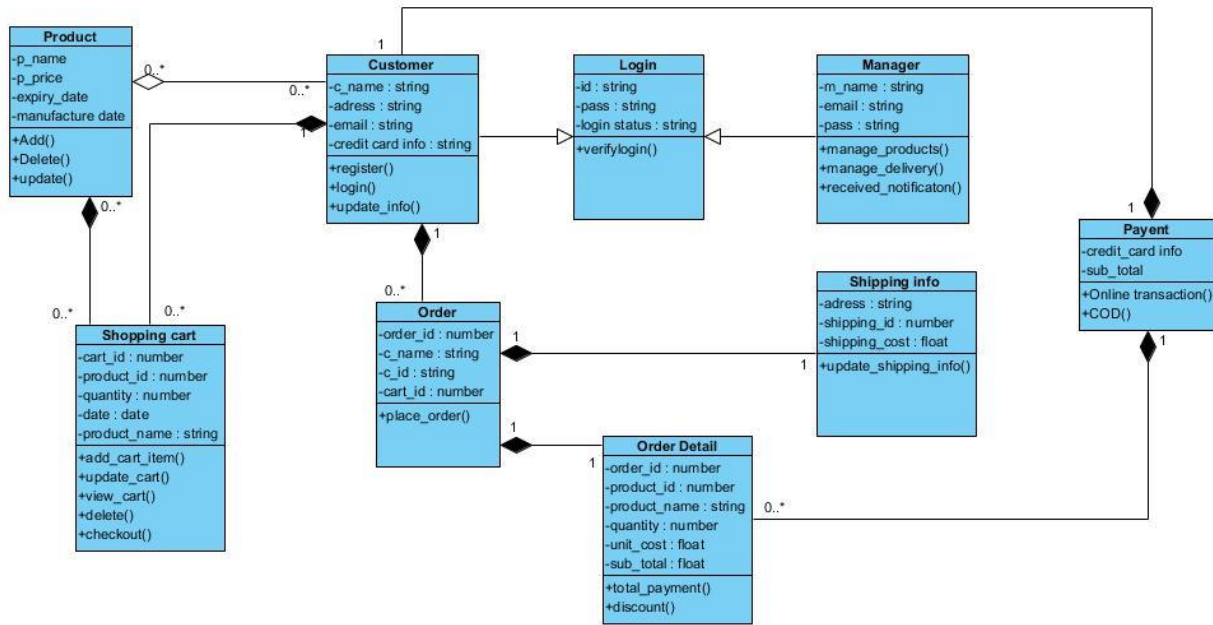


Figure 7 CLASS DIAGRAM

### 4.5. Sequence / Collaboration Diagram

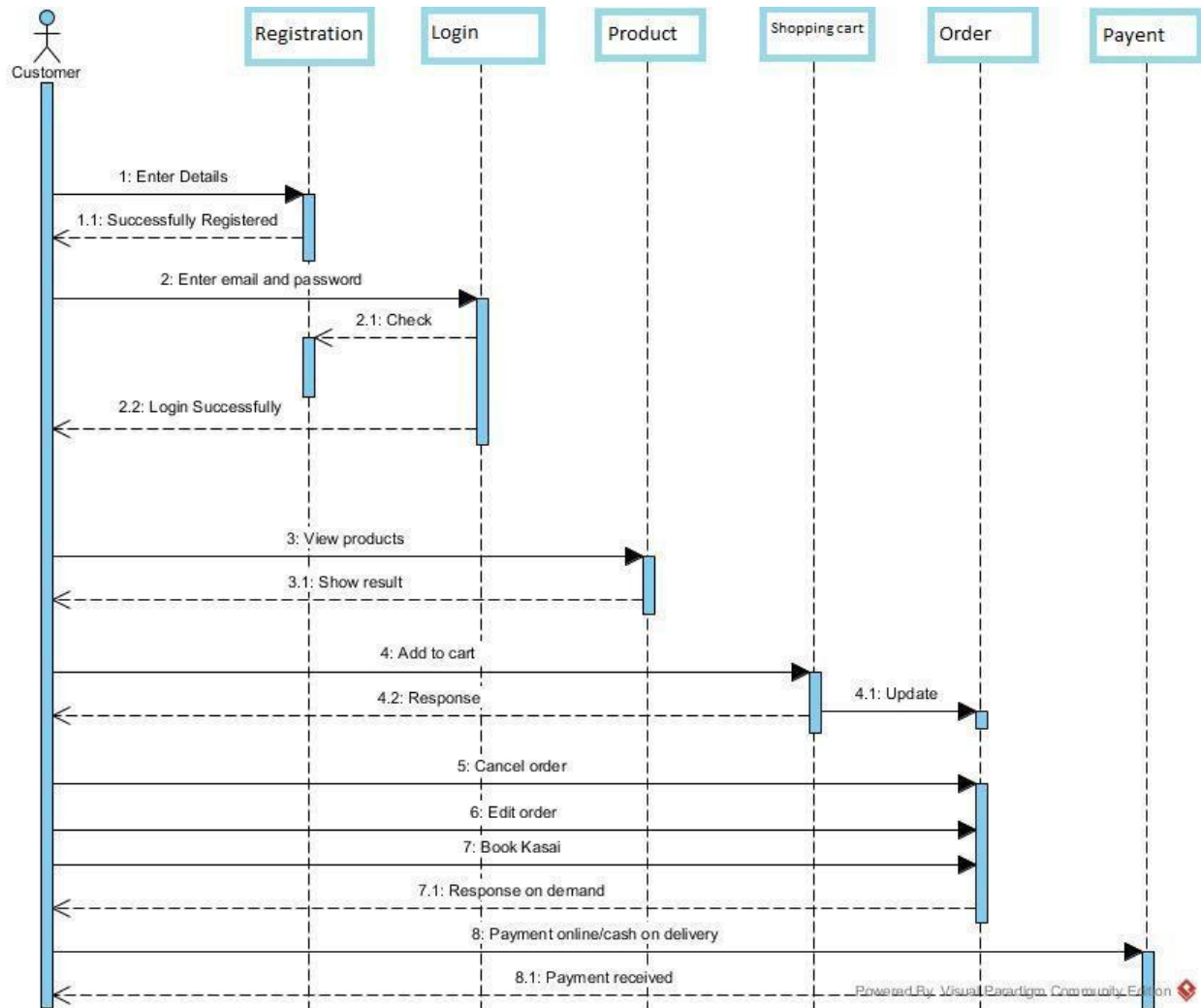


Figure 8 SEQUENCE DIAGRAM

## 4.6. Operation contracts

Following are the operation contracts of Kasai online system.

### **Operation Contract:**

- Name: Register.
- Responsibilities: To register the new user.
- Cross References: Use case: Register.
- Exceptions: None.
- Preconditions: user enter valid data for registration
- Post conditions: system opens successfully

### **Operation Contract:**

- Name: Login.
- Responsibilities: A user is logged in.
- Cross References: Use Case: login.
- Exceptions: invalid username or password.
- Preconditions: the user must have an account.
- Post-conditions: Successfully login to the system.

### **Operation Contract:**

- Name: logout.
- Responsibilities: the user is logged out.
- Cross References: Use Case: Logout.
- Exceptions: None.
- Preconditions: Must be logged in.
- Post-conditions: Log out successfully.

### **Operation Contract:**

- Name: Select item
- Responsibilities: User must log in and select item.
- Cross References: Use Case: Select item.
- Exceptions: None.
- Preconditions: Must be logged in.
- Post-conditions: Select item successfully.

**Operation Contract:**

- Name: View detail.
- Responsibilities: User must View detail of the Kasai.
- Cross References: Use Case: View Kasai Detail.
- Exceptions: None.
- Preconditions: User must log in.
- Post-conditions: View detail successfully.

**Operation Contract:**

- Name: Select item
- Responsibilities: User must log in and select item.
- Cross References: Use Case: Select item.
- Exceptions: None.
- Preconditions: Must be logged in.
- Post-conditions: Select item successfully.

**Operation Contract:**

- Name: Select Kasai.
- Responsibilities: User must select Kasai.
- Cross References: Use Case: Select Kasai.
- Exceptions: None.
- Preconditions: User must login.
- Post conditions: Customer successfully to select Kasai.

**Operation Contract:**

- Name: Booked Kasai.
- Responsibilities: User must booked a Kasai on his required timing.
- Cross References: Use Case: Booked Kasai.
- Exceptions: None.
- Preconditions: User must log in.
- Post-conditions: user booked Kasai successfully.

**Operation Contract:**

- Name: User cancels Kasai.
- Responsibilities: user or customer must cancel his Order.
- Cross References: Use Case: Cancel Order
- Exceptions: None.

- Preconditions: User must login.
- Post conditions: Customer successfully cancel Order

**Operation Contract:**

- Name: Change booking.
- Responsibilities: User must change booking from the system.
- Cross References: Use Case: Change booking.
- Exceptions: None.
- Preconditions: User must login.
- Post conditions: user change booking successfully.

**Operation Contract:**

- Name: Add Product.
- Responsibilities: Manager must add Product to system.
- Cross References: Use Case: Add Product.
- Exceptions: None.
- Preconditions: User must login.
- Post conditions: Add Product successfully.

**Operation Contract:**

- Name: Payment.
- Responsibilities: User must select payment method after complete his requirement from doctors.
- Cross References: Use Case: Payment method.
- Exceptions: None.
- Preconditions: User must login.
- Post-conditions: user successfully select payment method.

**Operation Contract:**

Name: View Product History.

Responsibilities: User must view product's history.

Cross References: Use Case: View Product History

Exceptions: None.

Preconditions: User must login.

Post-conditions: successfully view product's history.

### 4.7. Activity Diagram

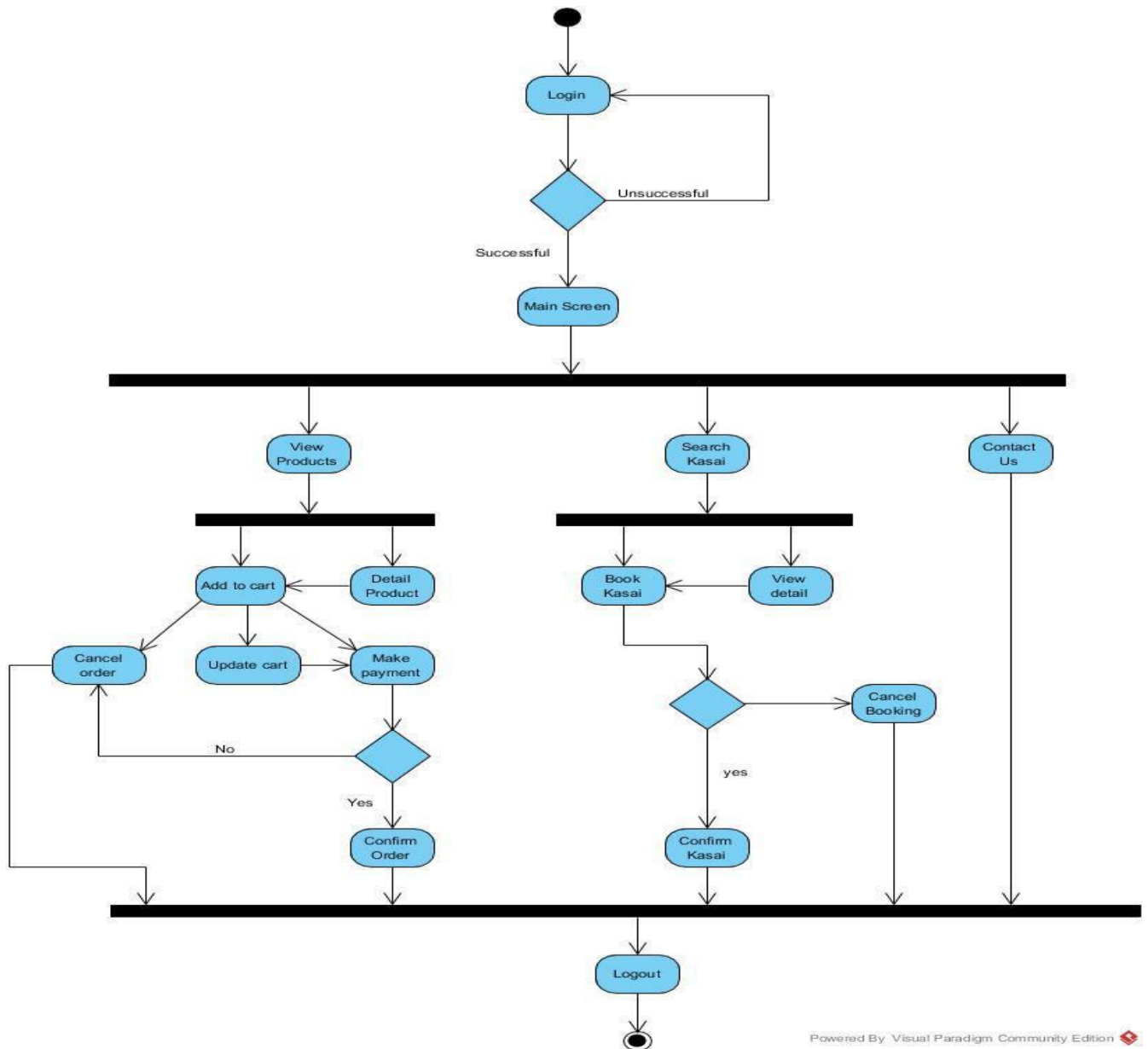
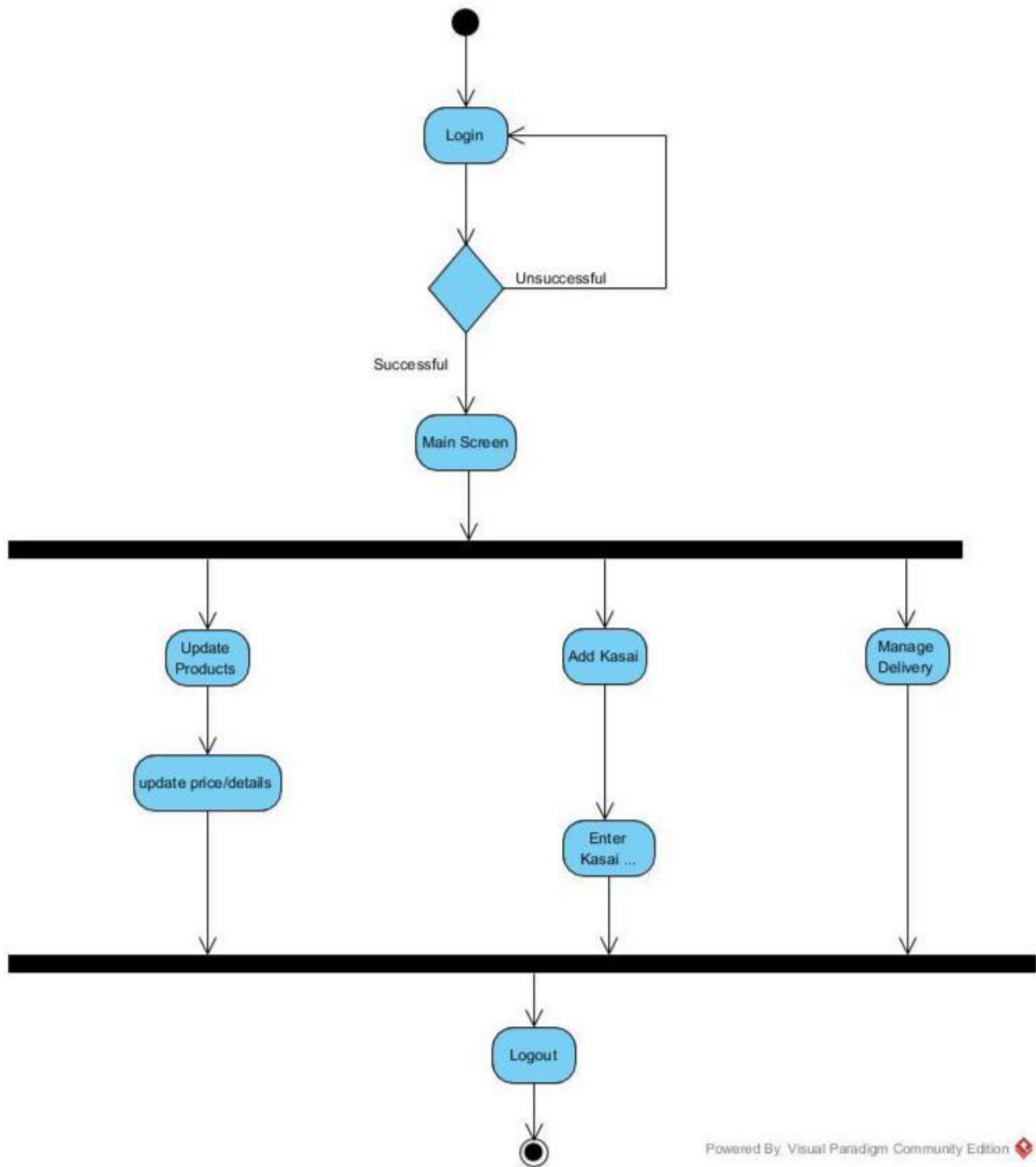


Figure 9 ACTIVITY DIAGRAM

Powered By Visual Paradigm Community Edition



Powered By Visual Paradigm Community Edition

Figure 10 ACTIVITY DIAGRAM

## 4.8. State Transition Diagram

### CUSTOMER:

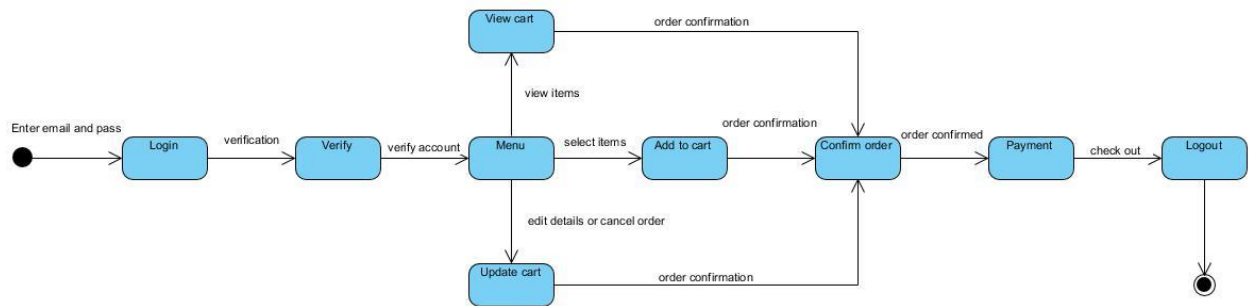


Figure 11 STATE TRANSITION DIAGRAM

### Manager:

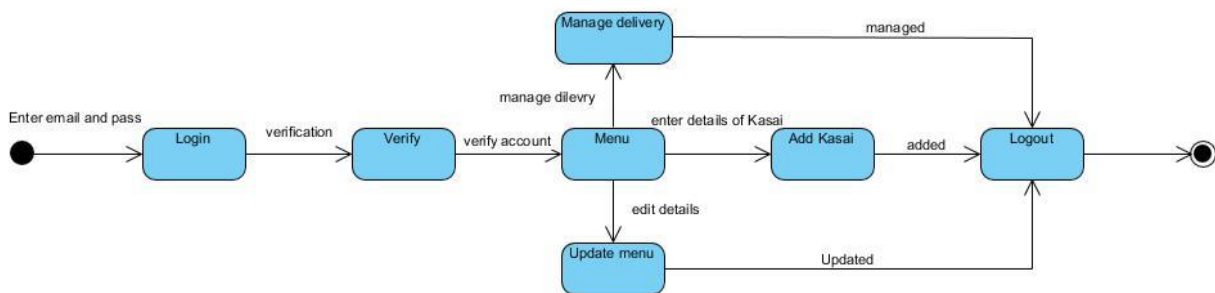


Figure 12 STATE TRANSITION DIAGRA 1.1

## 4.9. Component Diagram

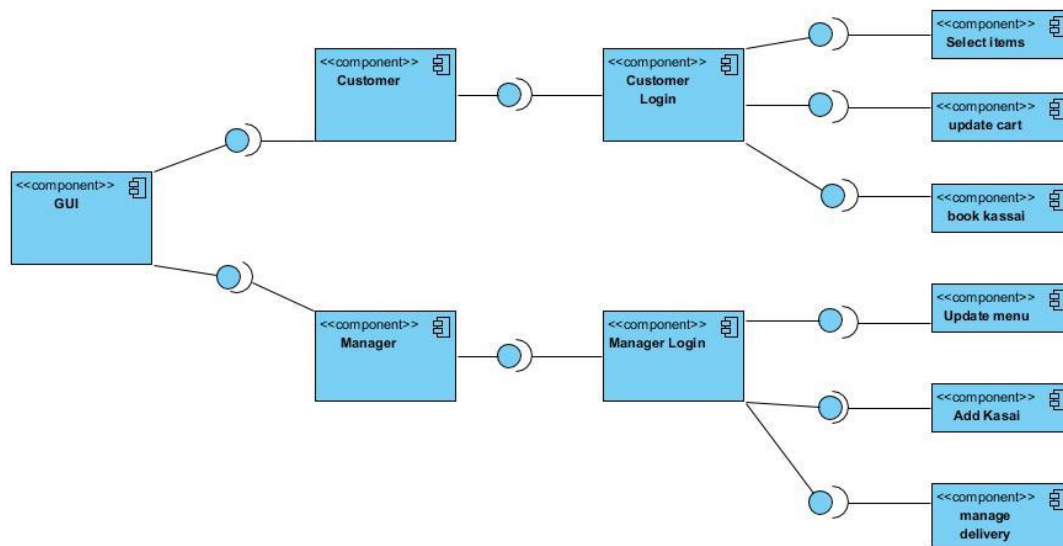


Figure 13 COMPONENT DIAGRAM

### 4.10. Deployment Diagram

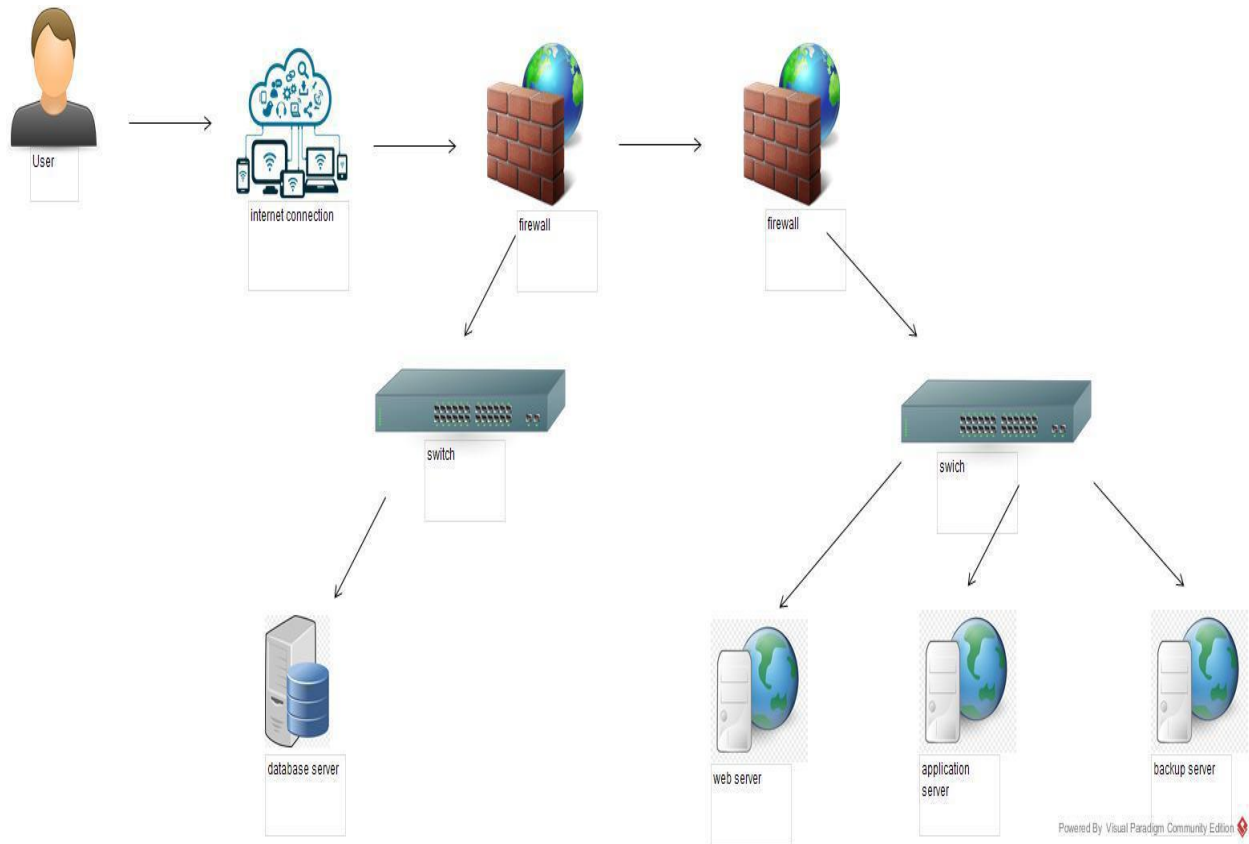


Figure 14 DEPLOYEENT DIAGRAM

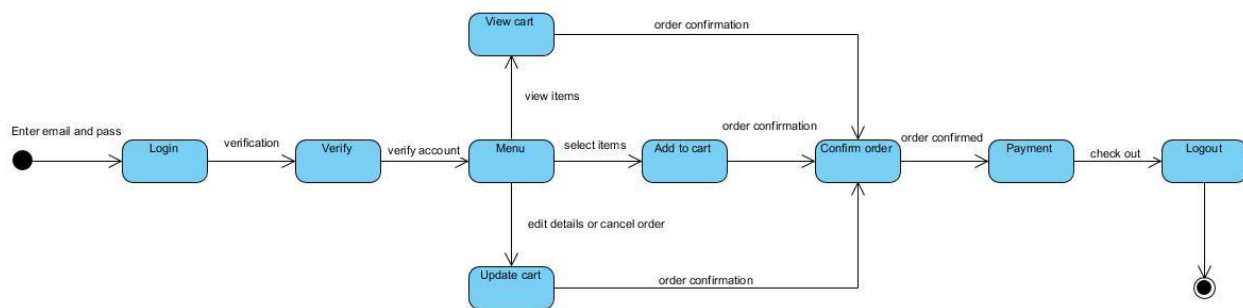
# Chapter 5

## Implementation

## Chapter 5: Implementation

We are going to implement of this application. As our software include Website for Promotion or Telling People about our brand and product and let people aware of our product. We build the system according to user need and check that system actually completes user requirements.

### 5.1. Important Flow Control/Pseudo codes



### 5.2. Components, Libraries, Web Services and stubs

Libraries:

MySQL: this will be implemented for the database

- HTTP: For website
- Language:
- Apache
- HTML CSS
- JavaScript
- PHP

### **5.3. Deployment Environment**

Deployment environment will be on

- Chrome
- Safari
- Opera
- Internet explorer

### **5.4. Tools and Techniques**

- Visual paradigm community addition
- MySQL
- Dreamveiwew
- Visual Studio

### **5.5. Best Practices / Coding Standards**

**JavaScript coding standards.**

- It is highly recommended to follow coding standards.

- Whenever we are writing any components the name of the components should reflect the purpose functionality.

### **Coding Standards.**

- Usually interfaces names are adjectives.
- Should starts with upper case letter and if it contains multiple words every inner word should starts with upper case letter.

### **Coding Standards for interfaces:**

- Usually interfaces names are adjectives.
- Should starts with case and if it contains multiple words every inner words should starts with upper case letter.

### **Coding Standards for methods:**

- Usually method names are either verbs noun combination.
- Should starts with lowercase character and if it contains multiple words every inner word
- Should starts with upper case letter.

## **5.6. Version Control**

The version of Visual paradigm community addition will be 15.2. Every version of the browser will be run this application.

# Chapter 6

## Testing and Evaluation

# Chapter 6: Testing and Evaluation

In this chapter we will test the whole system in parts. And we also test the whole system

## 6.1. Use Case Testing

Scenario ID	Test case ID	Test case	Test case description	Test step	Test step description	Expected Result	Actual Result	Status	Comment
UC_01	UC_01	Login	This user case takes the new information (name, Password) in a form field and after validation it provides the access the user	Pre-condition	Customer: want to get access to the web site User wants to login				
				Step 1	User launch the website	Website launched	Website launched successfully	Pass	
				Step 2	Navigate the login page	Display the login page and login input field	Login page loaded	Pass	
				Step 3	Enter wrong id and password	Input field should be editable form	Give the message the id and password is wrong	Pass	
				Step 4	Enter the true id and password	Input field should be editable form	Give the message the id and password is true	Pass	
				Step 5	Click login button	Button works properly	Button is send the input	Pass	

Scenario ID	Test case ID	Test case	Test case description	Test step	Test step description	Expected Result	Actual Result	Status	Comment
UC_02	UC_02	View product	Use case will displays the price, screen shots, deals and discounts on the product.	Pre-condition	User must search the product properly.  User view product properly				
				Step 1	User launch the website	Website launched	Website launched successfully	Pass	
				Step 2	Navigate the product page	Display the product page	Product page loaded	Pass	
				Step 3	All screen shoots of product visible	Product screen shoot are visible	Product screen shoot are visible	Pass	

Scenario ID	Test case ID	Test case	Test case description	Test step	Test step description	Expected Result	Actual Result	Status	Comment
UC_03	UC_03	Place order	This use case will allow user to place his order to purchase products. .	Pre-condition	User must have logged in before placing order.  User will place the order				
				Step 1	User launch the website	Website launched	Website launched successfully	Pass	
				Step 2	Navigate the product page	Display the product page	Product page loaded	Pass	
				Step 3	All screen shoots of product visible	Product screen shoot are visible	Product screen shoot are visible	Pass	
				Step 4	Go to the login page and login the account	Login the account	Login the account	Pass	
				Step 5	Place the order	Oder is book	Order is book successfully	Pass	

Scenario ID	Test case ID	Test case	Test case description	Test step	Test step description	Expected Result	Actual Result	Status	Comment
UC_04	UC_04	Place order	This use case will allow user to cancel the order.	Pre-condition	User must have logged in before placing order.  User will place the order After place the order he will cancel the order				
				Step 1	User launch the website	Website launched	Website launched successfully	Pass	
				Step 2	Navigate the product page	Display the product page	Product page loaded	Pass	
				Step 3	All screen shoots of product visible	Product screen shoot are visible	Product screen shoot are visible	Pass	
				Step 4	Go to the login page and login the account	Login the account	Login the account	Pass	
				Step 5	Place the order	Oder is book	Order is book successfully	Pass	
				Step 6	Cancel the order	Order cancel	Order cancel successfully	Pass	

## 6.1. Equivalence partitioning

Scenario ID	Test case ID	Test case	Test case description	Test step	Test step description	Expected Result	Actual Result	Status	Comment
T001	T001	Password length test	In this testing we check the boundary of input fields	Pre-condition	user must enter 6-8 length password and it must be digit value				
				Step 1	User launch the website	Website launched	Website launched successfully	Pass	
				Step 2	Navigate the login page	Display the login page	login page loaded	Pass	

				Step 3	Enter the password 6-10 digit	Show msg: your password must be within 6-8 digit	Login unsuccessful	Pass	
				Step 4	Enter the password character value	your password must be within 6-8 digit	Login unsuccessful	Pass	
				Step 5	Enter the password 6-10 digit	Login successful	Login successful	Pass	

## 6.2. Boundary value analysis

Scenario ID	Test case ID	Test case	Test case description	Test step	Test step description	Expected Result	Actual Result	Status	Comment
T002	T002	Login id field length test	In this testing we check the boundary of input fields	Pre-condition	user must enter 6-10 length character in username field				
				Step 1	User launch the website	Website launched	Website launched successfully	Pass	
				Step 2	Navigate the login page	Display the login page	login page loaded	Pass	
				Step 3	Enter the 6-10 character in username field	Login successful	Login successful	Pass	
				Step 4	Enter the 6-12 character value	your username must be within 6-10 character	Login unsuccessful	Pass	
				Step 5	Enter the 6-10 digit value	Login unsuccessful	Login unsuccessful	Pass	

### 6.3. Data flow testing

Scenario ID	Test case ID	Test case	Test case description	Test step	Test step description	Expected Result	Actual Result	Status	Comment
T003	T003	Data flow testing	How data flow from login page to database	Pre-condition	User must have logged in.				
				Step 1	User launch the website	Website launched	Website launched successfully	Pass	
				Step 2	Navigate the login page	Display the login page	Login page loaded	Pass	
				Step 3	Enter the id and password	Account will login	Account login successfully	Pass	
				Step 4	Data will pass from form to database	Data passes into database	Data pass into database successfully	Pass	
				Step 5	Print message data is pass successfully	Print the message	Print message data is pass successfully	Pass	

### 6.4. Unit testing

Scenario ID	Test case ID	Test case	Test case description	Test step	Test step description	Expected Result	Actual Result	Status	Comment
T004	T004	Login	This user case takes the new information (name, Password) in a form field and after validation it provides the access the user	Pre-condition	Customer: want to get access to the web site User wants to login				
				Step 1	User launch the website	Website launched	Website launched successfully	Pass	
				Step 2	Navigate the login page	Display the login page and login input field	Login page loaded	Pass	

				Step 3	Enter wrong id and password	Input field should be editable form	Give the message the id and password is wrong	Pass	
				Step 4	Enter the true id and password	Input field should be editable form	Give the message the id and password is true	Pass	
				Step 5	Click login button	Button works properly	Button is send the input	Pass	

Scenario ID	Test case ID	Test case	Test case description	Test step	Test step description	Expected Result	Actual Result	Status	Comment
T005	T005	View product	Use case will displays the price, screen shots, deals and discounts on the product.	Pre-condition	User must search the product properly.  User view product properly				
				Step 1	User launch the website	Website launched	Website launched successfully	Pass	
				Step 2	Navigate the product page	Display the product page	Product page loaded	Pass	
				Step 3	All screen shoots of product visible	Product screen shoot are visible	Product screen shoot are visible	Pass	
				Step 4					
				Step 5					

Scenario ID	Test case ID	Test case	Test case description	Test step	Test step description	Expected Result	Actual Result	Status	Comment
T006	T006	Place order	This use case will allow user to place his order to purchase products. .	Pre-condition	User must have logged in before placing order.  User will place the order				
				Step 1	User launch the website	Website launched	Website launched successfully	Pass	
				Step 2	Navigate the product page	Display the product page	Product page loaded	Pass	
				Step 3	All screen shoots of product visible	Product screen shoot are visible	Product screen shoot are visible	Pass	
				Step 4	Go to the login page and login the account	Login the account	Login the account	Pass	
				Step 5	Place the order	Oder is book	Order is book successfully	Pass	

## 6.5. Integration testing

Scenario ID	Test case ID	Test case	Test case description	Test step	Test step description	Expected Result	Actual Result	Status	Comment
T007	T007	Test the whole system	Test the whole system is the system is working properly	Pre-condition	User will explore the all feature of website				
				Step 1	User launch the website	Website launched	Website launched successfully	Pass	
				Step 2	Navigate the home page	Display the home page	home page loaded	Pass	
				Step 3	View all products	Product screen shoot are visible	Product screen shoot are visible	Pass	

				Step 4	Go to the login page and login the account	Login the account	Login the account	Pass	
				Step 5	Place the order	Oder is book	Order is book successfully	Pass	

## 6.6. Performance testing

Scenario ID	Test case ID	Test case	Test case description	Test step	Test step description	Expected Result	Actual Result	Status	Comment
T008	T008	Performance testing	The website response time will good.	Pre-condition	The website will launched in 3 second				
				Step 1	User launch the website	Website launched within the 3 second	Website launched In 5 second	Fail	

## 6.7. Stress Testing

Scenario ID	Test case ID	Test case	Test case description	Test step	Test step description	Expected Result	Actual Result	Status	Comment
T009	T009	Stress testing	We will give the load on website with heavy traffic	Pre-condition	The website will not crash				
				Step 1	User launch the website	Website launched	Website launched successfully	Pass	
				Step 2	Navigate the product page	Display the product page	Product page loaded	Pass	

				Step 3	All screen shoots of product visible	Product screen shoot are visible	Product screen shoot are visible	Pass	
				Step 4	Go to the login page and login the account	Login the account	Login the account	Pass	
				Step 5	Place the order	Oder is book	Order is book successfully	Pass	
				Step 6	The website will crash	Website will not crash	Website is crash	fail	

# Chapter 7

## Summary, Conclusion and Future Enhancements

# Chapter 7: Summary, Conclusion & Future Enhancements

## 7.1. Project Summary

- We are providing the online meat and kasai to the customers.
- We describe the time period in which we will make this project through Gant chart.
- We describe the functional and non functional requirements of our project.
- We describe the whole system component, how the component will work in the project through the use case diagram.
- We describe the design with the help of prototype.
- We describe the system how it will work through its sequence diagram.
- We describe the database through the class diagram.

## 7.2. Achievements and Improvements

We are facilitating the customers to book their meat and kasai on Eid and normal days, no one offering the book online the kasai with meat in their system, only we are offering this services.

## 7.3. Lessons Learnt

- Team work
- Time management
- Communication skills

## 7.4. Future Enhancements/Recommendations

We will work the efficiency of this project and also its security , no one will use our data.

# Reference and Bibliography

## Reference and Bibliography

- [1] Book by Craig Lorman, Applying UML and Patterns.
- [2] [https://en.wikipedia.org/wiki/Unified\\_Modeling\\_Language](https://en.wikipedia.org/wiki/Unified_Modeling_Language)
- [3] [https://en.wikipedia.org/wiki/Class\\_diagram](https://en.wikipedia.org/wiki/Class_diagram)
- [4] <https://www.lucidchart.com/pages/uml-sequence-diagram>
- [5] <https://reqtest.com/tutorials/how-to-develop-a-template-for-test-cases/>
- [6] <http://centreurope.info/sequence-diagram-template/sequence-diagram-template-sequence-diagram-templates-to-instantly-view-object-interactions-free/>
- [7] <https://www.edrawsoft.com/template-uml-class-diag>