

ITE 2952 Programming Group Project

Web based Food Order Management System for Cheezy Blast Pvt Ltd

Group: Code Blast

JRS Sudharaka E2041378

Amarasena E2041305

HMP Laksahan E2041348

Bachelor of Information Technology (External Degree)

Faculty of Information Technology

University of Moratuwa

2022 December

Abstract

The purpose of Online Food Ordering System is to automate the existing manual system by the help of computerized equipment's and full-fledged computer software, fulfilling their requirements, so that their valuable data/information can be stored for a longer period with easy accessing and manipulation of the same. This system will allow hotels and restaurants to increase scope of business by reducing the labor cost involved. The system also allows to quickly and easily manage an online menu which customers can browse and use to place orders with just few clicks. Restaurant employees then use these orders through an easy to navigate graphical interface for efficient processing. The required software and hardware are easily available and easy to work with. Online Food Ordering System, as described above, can lead to error free, secure, reliable and fast management system. It can assist the user to concentrate on their other activities rather to concentrate on the record keeping. Thus, it will help organization in better utilization of resources. The organization can maintain computerized records without redundant entries. That means that one need not be distracted by information that is not relevant, while being able to reach the information. The aim is to automate its existing manual system by the help of computerized equipment's and full-fledged computer software, fulfilling their requirements, so that their valuable data/information can be stored for a longer period with easy accessing and manipulation of the same. This system allows the user to select the desired food items from the displayed menu. The user orders the food items. The payment can be made online or pay-on-delivery system. The user's details are maintained confidential because it maintains a separate account for each user. An id and password are provided for each user. Therefore, it provides a more secured ordering. It is known globally that, in today's market, it is extremely difficult to start a new small-scale business and live-through the competition from the well-established and settled owners. In fast paced time of today, when everyone is squeezed for time, the majority of people are finicky when it comes to placing a food order. The customers of today are not only attracted because placing an order online is very convenient but also because they have visibility into the items offered, price and extremely simplified navigation for the order. Basically, the project describes how to manage for good performance and better services for the clients.

Key word – customer, employees, Food, menu, delivery

Table of Content

Table of Content.....	ii
List of Tables	v
List of Figures	vi
List of Abbreviations	vii
1 Chapter 01 – Introduction	8
1.1 Overview	8
1.2 Background and Motivation.....	8
1.3 Aim and Objectives.....	9
1.4 Aim	9
1.5 Objective.....	9
1.6 Summary.....	10
2 Chapter 02 – Related Work.....	11
2.1 Introduction.....	11
2.2 Manual System.....	11
2.3 Similar System 01- Domino’s Pizza	12
2.4 Similar System 02 – Sponge	13
2.5 Similar System 03 – KFC.....	13
2.6 Similar System 04 - Pizza Hut	14
2.7 Similar System 05 - Taco Bell	15
2.8 Comparison Table.....	16
2.9 Summary.....	17
3 Chapter 03 – System Analysis and Design	18
3.1 Introduction.....	18
3.2 Requirement Analysis	18
3.2.1 Requirement Gathering Techniques	18
3.3 Use Case Diagram	20
3.3.1 Use case diagram for Manual System.....	20
3.3.2 Use case Diagram for Staff Management.....	21
3.3.3 Use case Diagram for Customer Registration.....	23
3.3.4 Use case Diagram for Order Management	25
3.3.5 Use case Diagram for Category Management	27

3.3.6	Use case Diagram for Inquiry	29
3.4	Class Diagram.....	31
3.5	Activity Diagram	32
3.5.1	Activity Diagram for Admin Login.....	32
3.5.2	Activity Diagram for Order	33
3.5.3	Activity Diagram for Registration	34
3.6	Sequence Diagram.....	35
3.6.1	Sequence Diagram for Admin.....	35
3.6.2	Sequence Diagram for Order.....	36
3.6.3	Sequence Diagram for Registration	37
3.7	EER Diagram.....	38
3.7.1	Relational Schema	39
3.8	Summary.....	40
4	Chapter 04- Technologies Adapted.....	41
4.1	Introduction.....	41
4.2	Software Requirements	41
4.2.1	Star UML.....	41
4.2.2	phpMyAdmin.....	41
4.2.3	Operating System	41
4.2.4	Visual Code	42
4.2.5	XAMPP.....	42
4.3	Hardware Requirements	42
4.3.1	RAM.....	42
4.3.2	Processor	42
4.4	Technologies	42
4.4.1	PHP.....	42
4.4.2	CSS.....	43
4.4.3	HTML	43
4.4.4	JavaScript	43
4.4.5	MySQL.....	43
4.5	Other Requirements (N/A).....	44
4.6	Summary.....	44
5	Chapter 05-Implementation.....	45
5.1	Introduction.....	45
5.2	User Interfaces.....	45

5.2.1	Font End User interfaces	45
5.2.2	Back end user interfaces.....	48
5.3	Code Segments	51
5.3.1	Front End.....	51
5.3.2	Backend code segments	73
5.4	Summary.....	84
6	Chapter 06 – Testing and Evaluation	85
6.1	Introduction.....	85
6.2	Types of Software Testing	85
6.2.1	Unit Testing	85
6.2.2	Back Box Testing	85
6.2.3	Security Testing	86
6.2.4	Integration Testing.....	86
6.2.5	System Testing.....	86
6.3	Test Cases.....	87
6.4	Summary.....	90
7	Chapter 07 – Conclusion and Future work.....	91
7.1	Introduction.....	91
7.2	Conclusion	91
7.3	Future work.....	92
7.4	Challenges faced.....	92
7.5	Summary.....	92
	Appendix A.....	93
	Appendix B.....	97
	Appendix C.....	100
	Appendix D.....	103
	Appendix E	122
	Reference.....	124

List of Tables

Table 1-Comparison Table	16
Table 2- Narrative Table for staff registration	22
Table 3- Narrative Table for customer management.....	24
Table 4- Narrative Table for Order Management.....	26
Table 5- Narrative Table of Category Management.....	28
Table 6- Narrative Table of Inquiry Management.....	30
Table 7-Narrative Table for Feedback.....	94
Table 8-Narrative Table for Cart.....	96

List of Figures

Figure 1-Use case diagram for Manual System.....	20
Figure 2-Use case Diagram for Staff Management.....	21
Figure 3-Use case Diagram for Customer Registration	23
Figure 4-Use case Diagram for Order Management.....	25
Figure 5-Use case Diagram for Category Management.....	27
Figure 6-U se case Diagram for Inquiry.....	29
Figure 7-Class Diagram for Propose System.....	31
Figure 8-Activity Diagram for Admin Login	32
Figure 9-Activity Diagram for Order.....	33
Figure 10-Activity Diagram for Registration	34
Figure 11-Sequence Diagram for Admin	35
Figure 12-Sequence Diagram for Order.....	36
Figure 13-Sequence Diagram for Registration	37
Figure 14-EER Diagram for Propose System.....	38
Figure 15-Interface of Customer Registration	46
Figure 16-Interface of About Us.....	47
Figure 17-Interface of Home	48
Figure 18-Interface of Admin Login	49
Figure 19-Interface of Admin Change Password.....	50
Figure 20-Interface of Manage Admin	50
Figure 21-Test Case of Cart.....	87
Figure 22-Test Case of Login.....	88
Figure 23-Test case of Search.....	89
Figure 24-Use case of feedback.....	93
Figure 25-Use case of Cart.....	95
Figure 26-Database of Admin Table	97
Figure 27-Database of Category Table	98
Figure 28-Database of Customer Table	98
Figure 29-Database of Food Table	99
Figure 30-Database of Order Table	99
Figure 31-Interface of Update Category	100
Figure 32-Interface of Category.....	101
Figure 33-Interface of Admin	102
Figure 34-Test case of category	122
Figure 35-Test case of inquiry	123

List of Abbreviations

HTML = Hyper Text Markup Language

PHP = Hypertext Preprocessor

SQL = Structured Query Language

EER = Enhanced Entity-Relationship

WWW = World Wide Web

CSS=Cascading Style Sheet

1 Chapter 01 – Introduction

1.1 Overview

Due to the COVID-19 pandemic that has been going on for over two years now, therefore people have had to practice social distancing. So, if there is a manual system, there will be a large queue to get a number to get an order or get a reservation. With such a long queue, it is difficult to maintain social distance. When using a manual system, large files have to be maintained, so it is very difficult to enter new data into the system, to change the existing data, and to get the most important information for management from the data. So, as a solution for that case, we are preparing an online Food Ordering System.

We are developing our project Food Ordering System for the Cheezy Blast Pvt. This system requirements specification is mainly developed for the project development team. In this team, there are the project manager, project developer, quality assurance, designer, scrum team, tester and the users of the project also. They are the people who check whether customer requirements are there or not. So, they have to get the functional and non-functional requirements from the system requirements specification. With this software requirement specification, the people mentioned above can have a clearer idea about the project.

1.2 Background and Motivation

A web-based database system resides on an Internet server. The database can be accessed through a web browser. A distributed system is a system consisting of a collection of autonomous machines connected by communication networks and equipped with software systems designed to produce an integrated and consistent computing environment. Distributed systems help let the users cooperate all the activities more effectively and efficiently.

The key purpose of the distributed systems is represented by resource sharing, openness, concurrency, scalability, fault tolerance, and transparency. Web-services provide a standard means of interoperating between different software applications running on a variety of platforms and/or frameworks. Web applications use web documents written in a standard format such as HTML and JavaScript, which are supported by a variety of web browsers.

The advantages of web-based distributed databases are easy maintenance and updating, reusability and modularity, distribution of data update, and security. The architecture used for the web-based distributed database is the Client/Server model. In this model, the client sends a request to the webserver. The request is then transferred to the database server. The results are sent back to the web browser on the client-side after the database server processes the requests by the clients. In this Food Ordering System, the primary focus is on ordering food through the online method from Cheezy Blast to the customers.

1.3 Aim and Objectives

1.4 Aim

The aim of this project is to design and develop web-based food ordering system for Cheezy Blast Company with help of PHP and MYSQL technologies.

1.5 Objective

- To analyze most demanding item of the company.
- To analyze least demanding item of the company.
- To analyze profits and sales for a particular month.
- To reduce the inconvenience of ordering food items.
- To facilitating contact with the restaurant for those who find it difficult to come to the restaurant.
- To facilitate the day-to-day business activities of the company.
- To provide better service using this system and build an image of this system among society.

1.6 Summary

Our issue, the project's goal and objectives, as well as all the background information pertaining to the manual system, were all clearly identified in the first chapter.

We are going to identify a system that is similar to our project in the following chapter. To find the similarities and the functional and non-functional requirements of the chosen systems, four similar systems from the Sri Lankan domain and four from the foreign domain are chosen.

2 Chapter 02 – Related Work

2.1 Introduction

The "Online Food Ordering System" has been developed to override the problems prevailing in the practicing manual system. This software is supported to eliminate and, in some cases, reduce the hardships faced by this existing system. Moreover, this system is designed for the particular need of the company to carry out operations in a smooth and effective manner.

The application is reduced as much as possible to avoid errors while entering the data. It also provides error message while entering invalid data. No formal knowledge is needed for the user to use this system. Thus, by this all it proves it is user-friendly. Online Food Ordering System as described above, can lead to error free, secure, reliable and fast management system. It can assist the user to concentrate on their other activities rather than concentrate on the record keeping. Thus, it will help organization in better utilization of resources.

Every organization, whether big or small, has challenges to overcome and managing the information of Category, Food Item, Order, Payment, Confirm Order. Every Online Food Ordering System has different Food Item needs; therefore, we design exclusive employee management systems that are adapted to your managerial requirements. This is designed to assist in strategic planning, and will help you ensure that your organization is equipped with the right level of information and details for your future goals. Also, for those busy executive who are always on the go, our systems come with remote access features, which will allow you to manage your workforce anytime, at all times. These systems will ultimately allow you to better manage resources.

2.2 Manual System

The manual system of the Chez Blast was maintaining the staff details, customer details, food categories, and food ordering management details with the help of the staff members according to a manual file-based system. Staff members include an administrator, managers, and devious boys. In the manual system, the administrator managed the overall management of the restaurant. The admin handled the customer details, ordering, and category management details by writing down

every detail in papers, and she managed the staff member details and food categories by keeping handwritten files for each section. The bills were created manually by the administrator. The admin manually checks customer inquiries, feedback, and food availability.

2.3 Similar System 01- Domino's Pizza

Domino's Pizza is a well-known Sri Lankan e-commerce platform that offers thousands of Sri Lankans and Sri Lankan expats throughout the world a first-rate online food ordering experience. Our services and goods are renowned for their excellence, novelty, and freshness. In order to become the most dependable and well-liked e-commerce brand in Sri Lanka, we are safeguarding the magic while updating the rest. We are really proud of our consistently excellent customer satisfaction rating since dependability and quality are our key principles in both our business models and the way we serve our clients.

When comparing this website with the proposed website, there are similarities and differences in both functional and non-functional requirements. Both the Domino's Pizza website and the proposed website have equal functionalities in order management, where they provide the functionalities for customer registration before the first order. Both websites have equal functionality in the customer confirmation process, where they ask the customers to fill out a customer detail form to register them before ordering. In the proposed system, the registration process is done for both staff members and customers of the Cheezy Blast, while the Domino's Pizza website allows customers to register for their system and make orders. The proposed system has create, edit, add, and update functionalities for the users of the system. The Domino's Pizza website has language selection functionality, which is primarily aimed at foreigners, whereas the proposed system only has main language. The Domino's Pizza website navigation bar includes a contact us page, fact sheet and brochure functionality, where the fact sheet and brochure are linked only for pdf formatted documents to get an idea about the restaurant, whereas the proposed website has linked the contact us and about us for different web pages where customers can directly go into the linked web pages, implying the non-functionalities of both websites as usability a problem.

[1]

2.4 Similar System 02 – Sponge

The sponge was from a pastry shop on Galle Road in Colombo 3. They have provided dining, takeaway, delivery, and online ordering facilities. They had a category management system. The home page was categorized as Cakes, Savories, Catering, Online Menu, Delivery Areas, about, and Contact. With the exception of about, contact, and catering, these primary categories were handled by subcategories. These subcategories are where all of the items are controlled. Customers may place orders online using the online menu, which is in a different category. They offered pricing and item specifications for each of their offerings. Customers were given information on the item's size, price, amount, and servings, as well as a brief description. They have implemented order management effectively. Their website provided user-friendly features for the customers. Other than those features, they have provided customer management, inquiry handling, feedback handling, delivery management, and online payment handling.

Both the sponge website and the proposed website provide the same functionalities in order management. Customers may place orders online using the online menu, which is in a different category. Both websites have equal functionality in the order confirmation process, where they ask the customers to fill out a customer detail form to register them before ordering. In the proposed system, the registration process is done for both staff members and customers of the Cheezy Blast, while the sponge website allows customers to register for their system and make orders. The proposed system has create, edit, add, and update functionalities for the administrators (users) of the system. [2]

2.5 Similar System 03 – KFC

The KFC is a popular restaurant in Sri Lanka and has several branches around the country. They have provided dining, takeaway, delivery, and online ordering facilities. They had a category management system. These subcategories are where all of the items are controlled. Customers may place orders online using the online menu, which is in a different category. They offered pricing and item specifications for each of their offerings. They have implemented order management effectively. Their website provided user-friendly features for the customers. Other than those features, they have provided customer management, inquiry handling, feedback handling, delivery

management, and online payment handling. They have provided coupons for the customers. They have provided security features for online payment handling. Furthermore, they have customer verification in their ordering system. This system is very user-friendly as it uses meaningful icons, and customers can easily understand the product by looking at the images.

When comparing this website with the proposed website, there are similarities and differences in both functional and non-functional requirements. Both the KFC website and the proposed website have equal functionalities in order management, where they provide functionalities for customer registration and staff registration before the first order. Both websites have equal functionality in the customer confirmation process, where they ask the customers to fill out a customer detail form to register them before ordering. In the proposed system, the registration process is done for both staff members and customers of the Cheezy Blast, while the KFC website allows customers to register for their system and make orders. The proposed system has create, edit, add, and update functionalities for the users of the system. The KFC website has language selection functionality, which is primarily aimed at foreigners, whereas the proposed system only has one main language. [3]

2.6 Similar System 04 - Pizza Hut

Pizza Hut, a subsidiary of Yum! Brands, is the world's largest pizza company and the home of Pan Pizza. Pizza Hut began 60 years ago in Wichita, Kansas, and today is an iconic global brand that delivers more pizza, pasta, and wings than any other restaurant in the world. Pizza Hut is a well-known Sri Lankan e-commerce platform that offers thousands of Sri Lankans and Sri Lankan expats throughout the world a first-rate online food ordering experience. Our services and goods are renowned for their excellence, novelty, and freshness. In order to become the most dependable and well-liked e-commerce brand in Sri Lanka, we are safeguarding the magic while updating the rest. We are really proud of our consistently excellent customer satisfaction rating since dependability and quality are our key principles in both our business models and the way we server our clients.

When comparing this website with the proposed website, there are similarities and differences in both functional and non-functional requirements. Both the Pizza Hut website and the proposed website have equal functionalities in order management, where they provide the functionalities for customer registration before the first order. Both websites have equal functionality in the customer confirmation process, where they ask the customers to fill out a customer detail form to register them before ordering. In the proposed system, the registration process is done for both staff members and customers of the Cheezy Blast, while the Pizza Hut website allows customers to register for their system and make orders. The proposed system has create, edit, add, and update functionalities for the users of the system. [4]

2.7 Similar System 05 - Taco Bell

At Taco Bell, we've had innovation on our mind since Glen Bell started serving tacos at the first location in 1962 in Downey, California. Since then, they've grown to be a culture-centric, lifestyle brand that provides craveable, affordable Mexican-inspired food with bold flavors. Not only do they provide breakthrough value, they offer quality ingredients and are the first quick-service restaurant to offer American Vegetarian Association (AVA)-certified menu items. Taco Bell and our more than 350 franchise organizations operate over 7,000 restaurants that serve more than 40 million customers each week in the U.S. Internationally, the brand is growing, with nearly 500 restaurants across almost 30 countries across the globe.

When comparing this website with the proposed website, there are similarities and differences in both functional and non-functional requirements. Both the Taco Bell website and the proposed website have equal functionalities in order management, where they provide the functionalities for customer registration before the first order. Both websites have equal functionality in the customer confirmation process, where they ask the customers to fill out a customer detail form to register them before ordering. In the proposed system, the registration process is done for both staff members and customers of the Cheezy Blast, while the Taco Bell website allows customers to register for their system and make orders. The proposed system has create, edit, add, and update functionalities for the users of the system. [5]

2.8 Comparison Table

Table 1-Comparison Table

	Foreign Domain					Local Domain				
Functional Requirements	<u>Domin</u> <u>o's</u> <u>Pizza</u>	<u>Pizza</u> <u>Hut</u>	<u>KFC</u>	<u>McDon</u> <u>ald's</u>	<u>Burg</u> <u>er</u> <u>King</u>	<u>Taco</u> <u>Bell</u>	<u>spon</u> <u>ge</u>	<u>Barracud</u> <u>a</u>	Manu al Syste m	Propose d System
Customer Registration	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	No	Yes
Staff Registration	No	No	No	No	No	No	No	No	No	Yes
Product Management	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Category Management	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Order Management	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	No	Yes
Order Cancellation	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	No	Yes
Delivery Management	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Inquiry	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	No	Yes
Feedback	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	No	Yes
News & Notification Management	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes

2.9 Summary

This chapter lists existing systems that are comparable to those used by food businesses and compares them to the web-based alternative we propose. A few things were the same, but a crucial difference is that our system provides reports for management decision-making.

The diagrams of the suggested system are implied in the following chapter, which also provides a clear understanding of the system's database. Under the UML diagrams, EER diagram, and relational schema, this chapter also includes use case diagrams, class diagrams, activity diagrams, and sequence diagrams to illustrate the proposed system's database.

3 Chapter 03 – System Analysis and Design

3.1 Introduction

It mentions a comparison of related systems from the local and international domains in the second chapter. An outline of the system's capabilities and limitations is given in this chapter. Based on the comparison, we discovered eleven functions.

The proposed system is illustrated in this chapter, along with details on the system's database. This chapter includes use case diagrams, class diagrams, activity diagrams, sequence diagrams, an EER diagram, and a relational schema to show the proposed system's database behind the UML diagrams.

3.2 Requirement Analysis

3.2.1 Requirement Gathering Techniques

3.2.1.1 Similar Systems

We can find the differences between our project and similar projects by referring to similar systems. The proposed system and the selected similar systems can be compared to find similarities and differences. Recognizing the difference between existing systems and the one being proposed allows us to tailor our project to the main requirements of the client. In addition, we can pinpoint the technologies in use and gain a clear understanding of how to design and put the suggested system into practice. We compared the similar system and the suggested system to get a better idea of the booking management, customer management, cancellation, facility and room details, contact us, and about us functionalities.

3.2.1.2 Manual System Documentation

Using the manual system documentation as a guide, we can learn about some of the issues they ran into. Workers were under pressure from the manual system to produce the monthly reports and order summaries. The manual system is ineffective for generating revenue and maintaining the facilities. Because of the manual system, neither the staff nor the patrons can accurately describe the restaurant to their satisfaction for the price they are paying. When profit calculations are made, the manual system does not give precise information. Maintaining customer information manually is challenging, and as the number of files rises, so does the security of the data. In order to address the aforementioned issues, we introduce new requirements after consulting this manual system documentation.

3.2.1.3 Questioners

Questionnaires are special-purpose documents that enable for the collection of information from a large number of people while maintaining some control over their responses. No other fact-finding technique can tabulate the same data as efficiently when working with a big number of listeners or audience. [6]

3.2.1.4 Interviews

The most popular and efficient method of gathering information is through interviews. Face-to-face interviews are a viable option for gathering data from people. A variety of objectives can be accomplished through interviews, including gathering information, validating that information, clarifying information that has already been released, igniting interest, involving the end user, defining requirements, and eliciting opinions and thoughts. There are two approaches to interviews. The first is open-ended, while the second is closed-ended. Here, we employed the open-ended interview method. This meant that we questioned actual businesspeople. [6]

3.3 Use Case Diagram

A graphical representation of a user's potential interactions with a system is called a use case diagram. A use case diagram will frequently be accompanied by other types of diagrams and will display the various use cases and user types the system has. Either circles or ellipses are used to represent the use cases. [7]

3.3.1 Use case diagram for Manual System

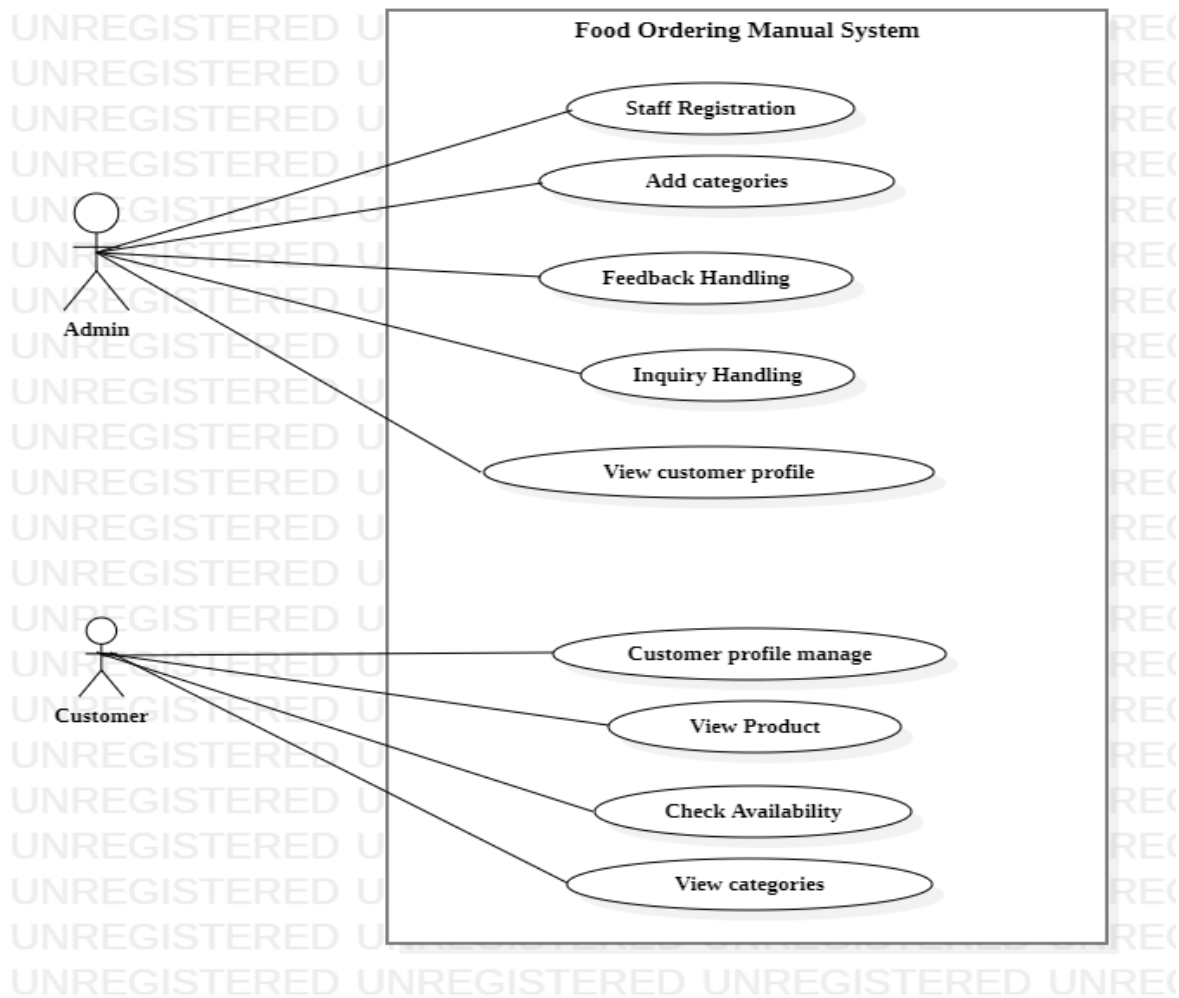


Figure 1-Use case diagram for Manual System

3.3.2 Use case Diagram for Staff Management

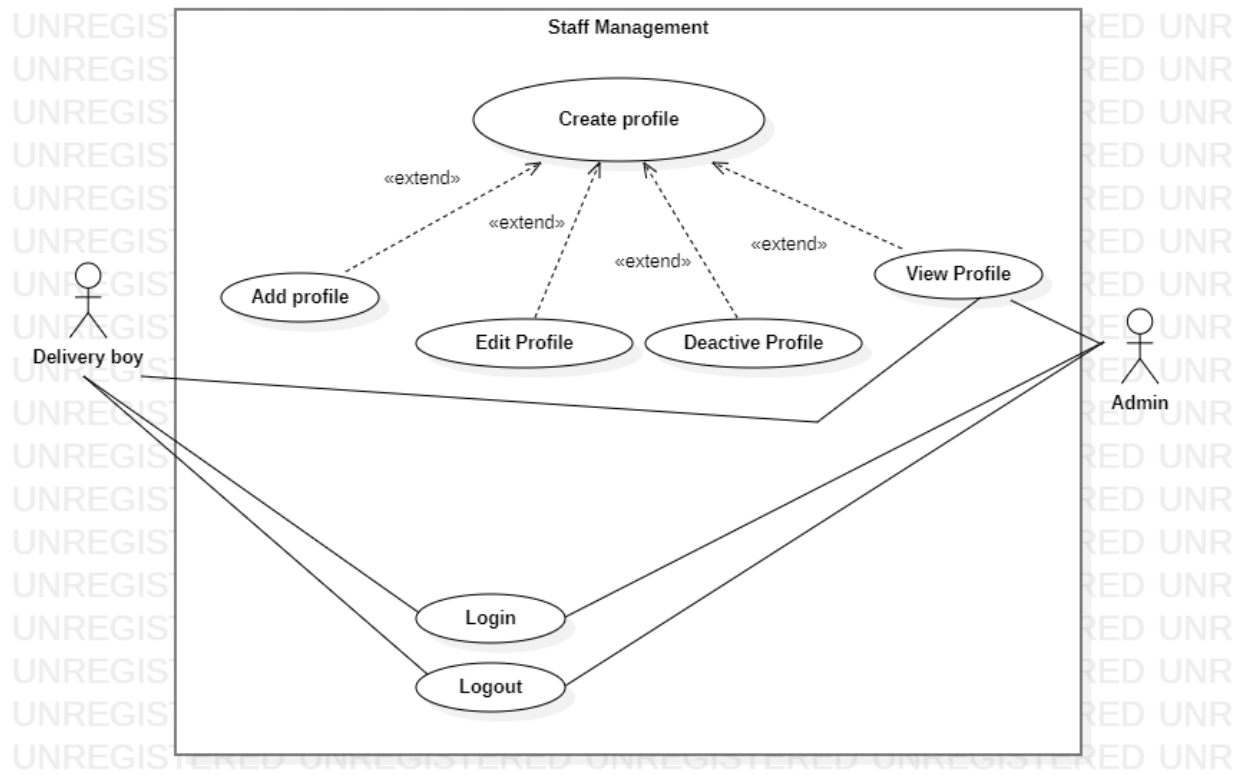


Figure 2-Use case Diagram for Staff Management

Narrative Table for staff registration

Author: Pubudu

Date: 2022/12/25

Version: 3.0

Table 2- Narrative Table for staff registration

Use-Case Name:	Staff registration	Use-Case Type - Business Requirements
Use-Case ID:	FO-01	
Priority:	High Priority	
Source:	Online Form	
Primary Business Actor:	Admin	
Other Participating Actors:	HR management	
Other Interested Stakeholders:		
Description:	<div>1. Create profile</div> <div>2. Login</div> <div>3. View profile</div> <div>4. Edit Profile</div> <div>5. Deactivate profile</div> <div>6. Add profile</div> <div>7. Logout</div>	

3.3.3 Use case Diagram for Customer Registration

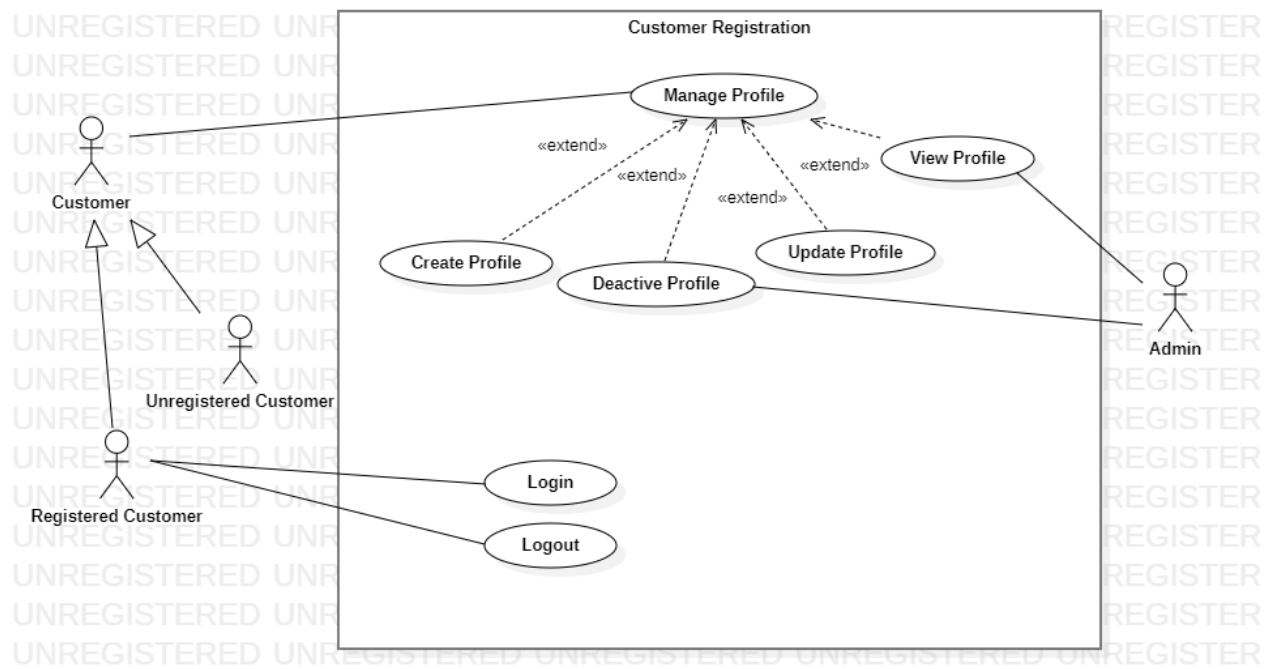


Figure 3-Use case Diagram for Customer Registration

Narrative Table for customer management

Author: Pubudu

Date: 2022/12/25

Version: 3.0

Table 3- Narrative Table for customer management

Use-Case Name:	Customer Management	Use-Case Type - Business Requirements
Use-Case ID:	FO-02	
Priority:	High Priority	
Source:	Online Form	
Primary Business Actor:	Admin	
Other Participating Actors:	Admin	
Other Interested Stakeholders:	Customer	
Description:	1. Customer Registration 2. Customer Login 3. Customer Edit Profile 4. Customer Delete Profile 5. Customer Logout	

3.3.4 Use case Diagram for Order Management

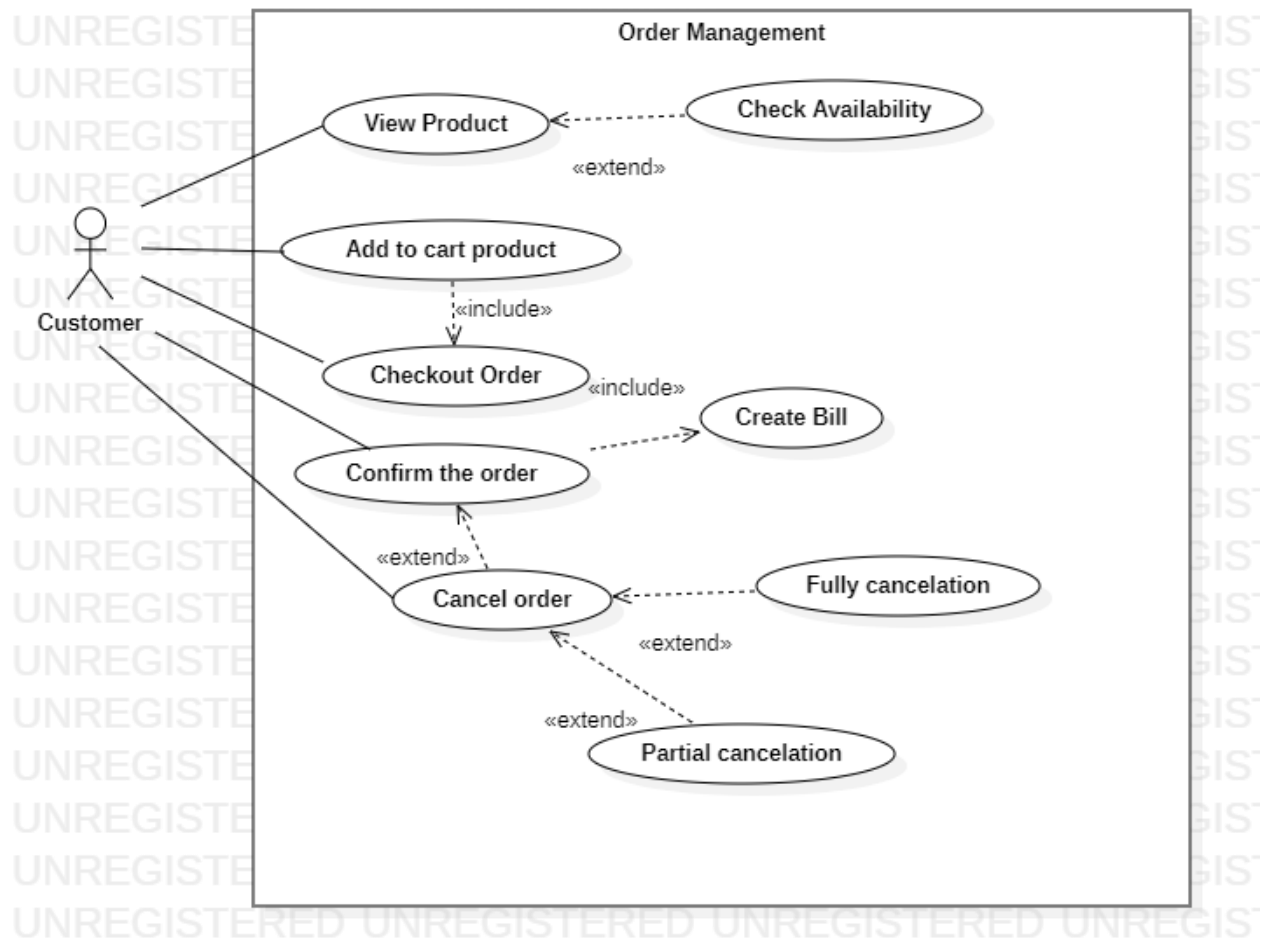


Figure 4-Use case Diagram for Order Management

Narrative Table for Order Management

Author: Pubudu

Date: 2022/12/25

Version: 3.0

Table 4- Narrative Table for Order Management

Use-Case Name:	Order management	Use-Case Type - Business Requirements
Use-Case ID:	FO-03	
Priority:	High Priority	
Source:	Online Form	
Primary Business Actor:	Customer	
Other Participating Actors:		
Other Interested Stakeholders:		
Description:	1. View product 2. Add to cart 3. Checkout order 4. Confirm order	

3.3.5 Use case Diagram for Category Management

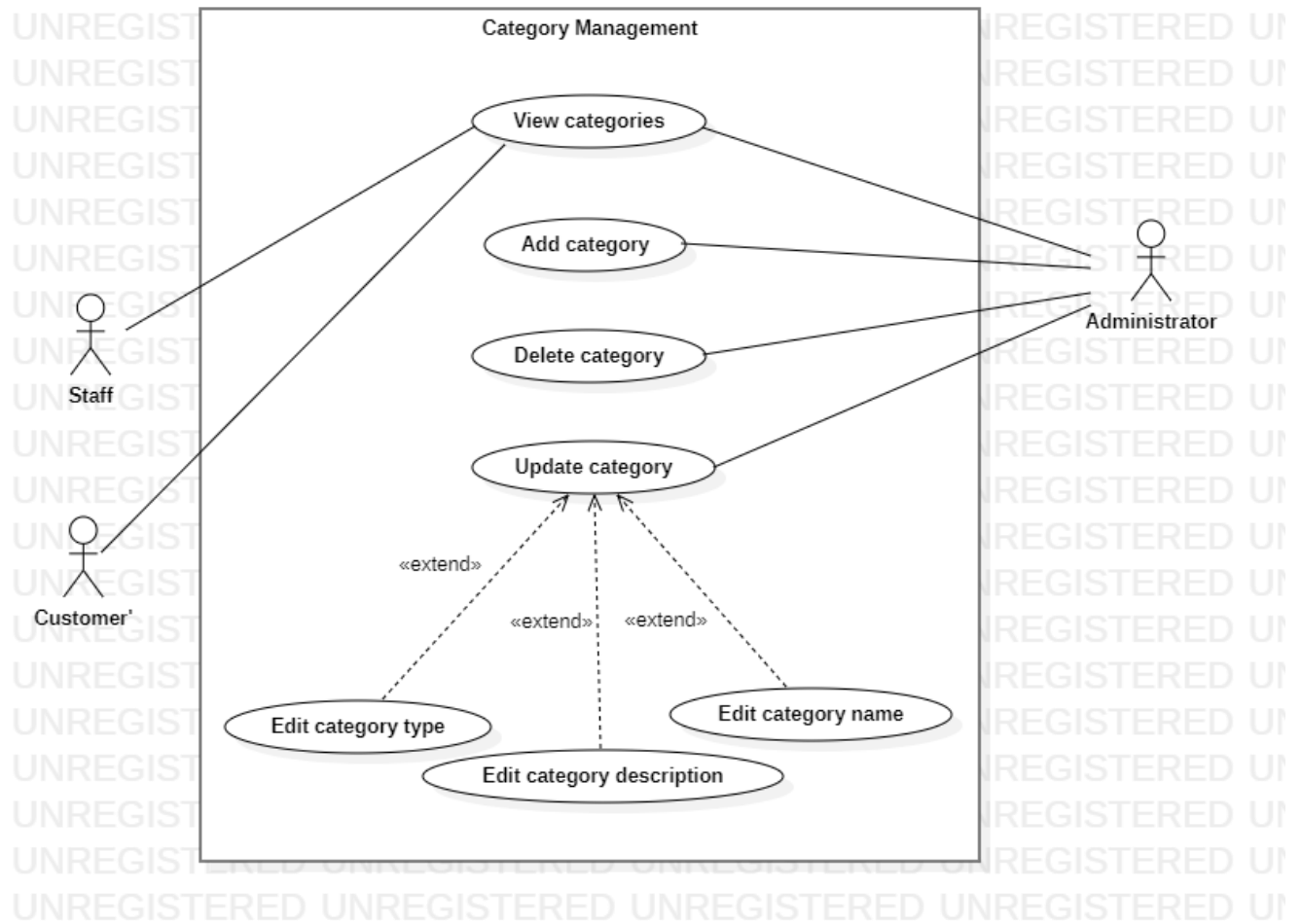


Figure 5-Use case Diagram for Category Management

Narrative Table of Category Management

Author: Pubudu

Date: 2022/12/25

Version: 3.0

Table 5- Narrative Table of Category Management

Use-Case Name:	Category management	Use-Case Type - Business Requirements
Use-Case ID:	FO-04	
Priority:	High Priority	
Source:	Online Form	
Primary Business Actor:	Administer	
Other Participating Actors:	Staff	
Other Interested Stakeholders:	Customer	
Description:	1. View category 2. Add category 3. Delete category 4. Update category	

3.3.6 Use case Diagram for Inquiry

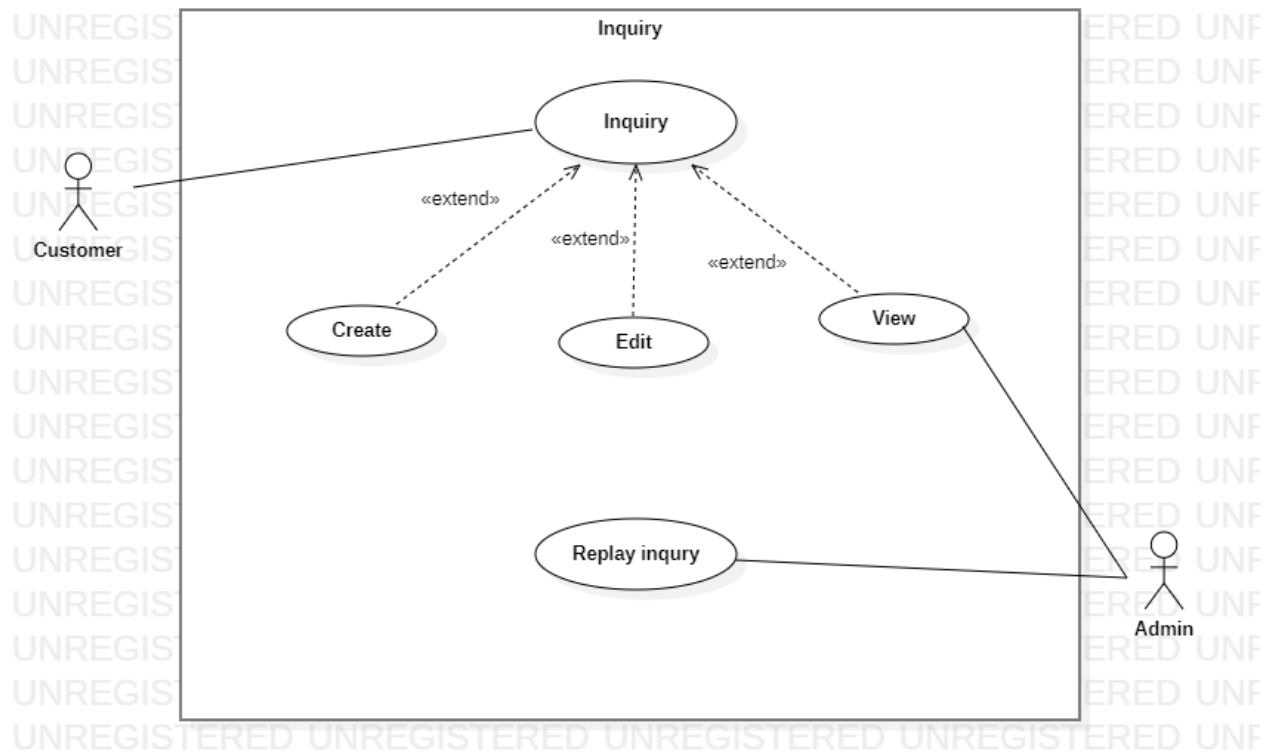


Figure 6-Use case Diagram for Inquiry

Narrative Table of Inquiry Management

Author: Pubudu

Date: 2022/12/25

Version: 3.0

Table 6- Narrative Table of Inquiry Management

Use-Case Name:	Inquiry management	Use-Case Type - Business Requirements
Use-Case ID:	FO-05	
Priority:	High Priority	
Source:	Online Form	
Primary Business Actor:	Administer	
Other Participating Actors:	Customer	
Other Interested Stakeholders:	Customer	
Description:	1. View inquiry 2. Add inquiry 3. Delete inquiry 4. Update inquiry 5. Replay inquiry	

For rest of the use case diagrams and narratives refer Appendix A

3.4 Class Diagram

A class diagram in software engineering is a kind of static structure diagram that reveals the classes, attributes, operations, and relationships between objects in a system to describe the system's structure. [8]

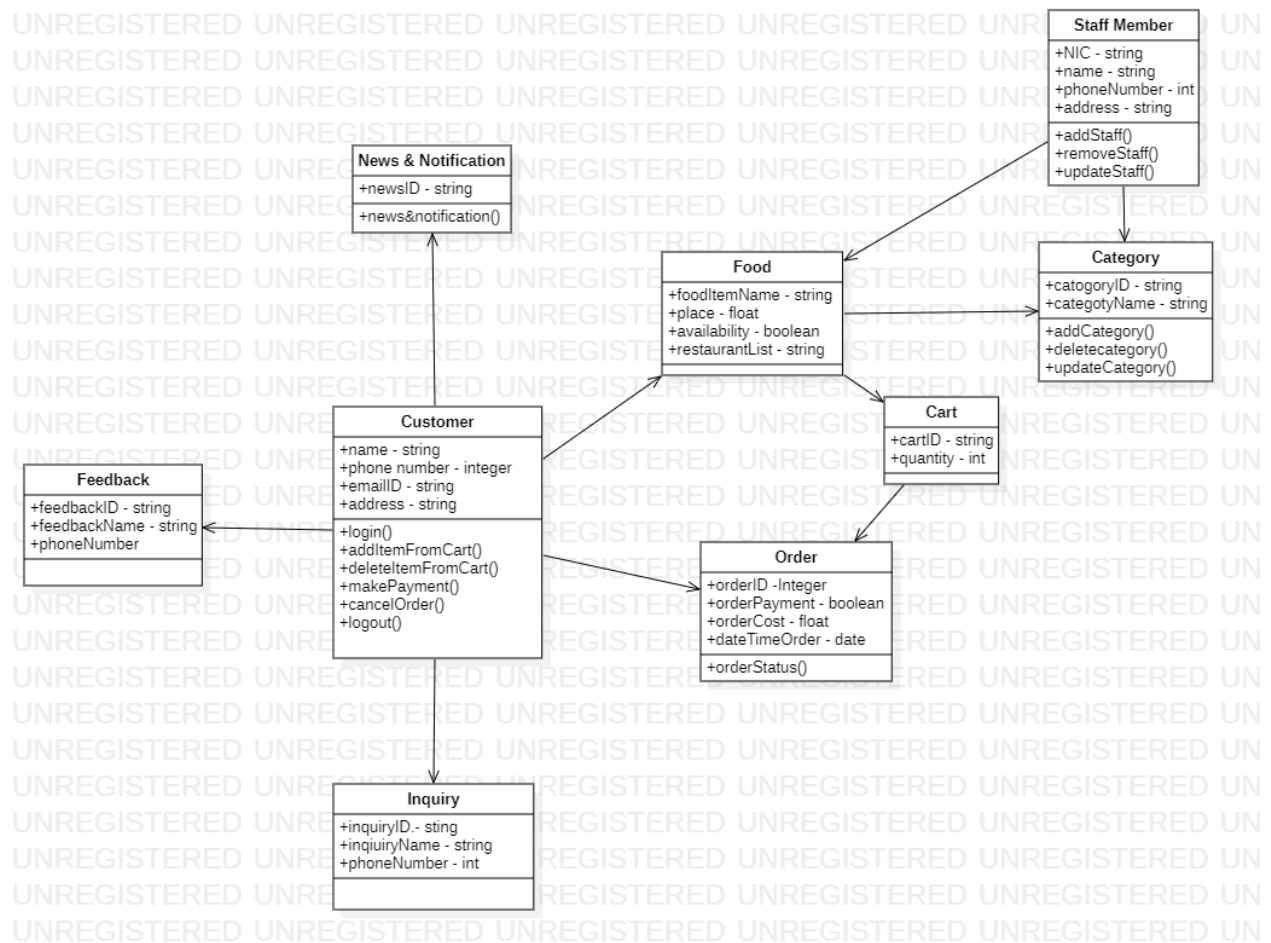


Figure 7-Class Diagram for Propose System

3.5 Activity Diagram

A class diagram in software engineering is a kind of static structure diagram that reveals the classes, attributes, operations, and relationships between objects in a system to describe the system's structure. [9]

3.5.1 Activity Diagram for Admin Login

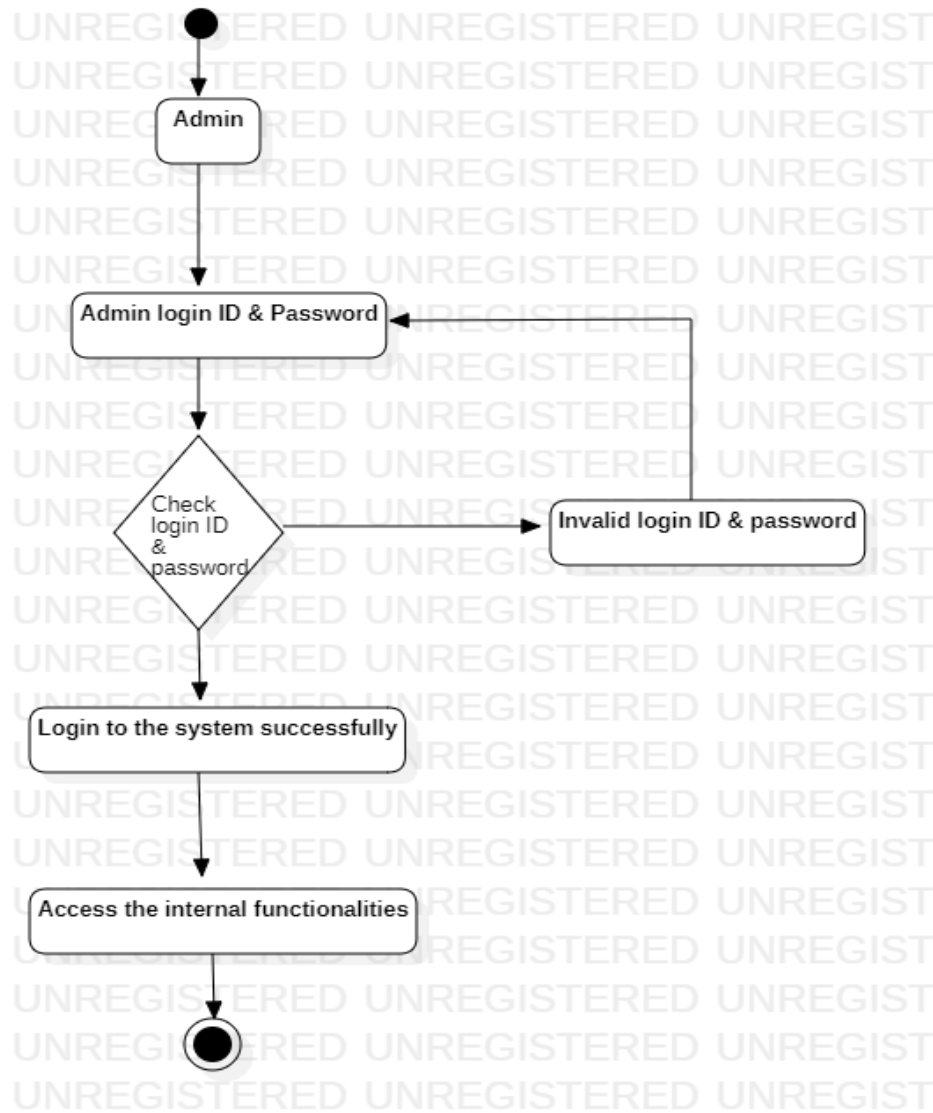


Figure 8-Activity Diagram for Admin Login

3.5.2 Activity Diagram for Order

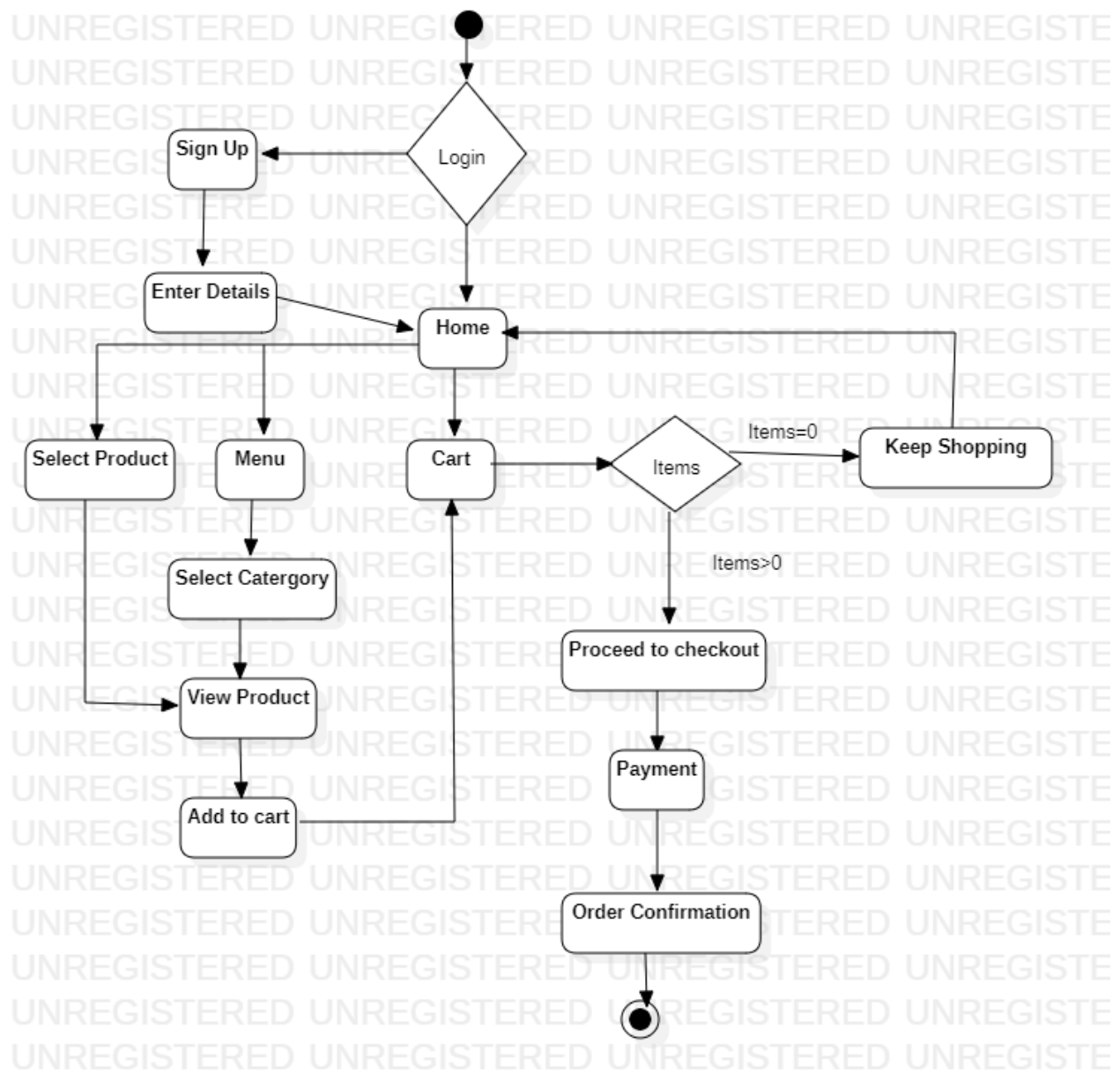


Figure 9-Activity Diagram for Order

3.5.3 Activity Diagram for Registration

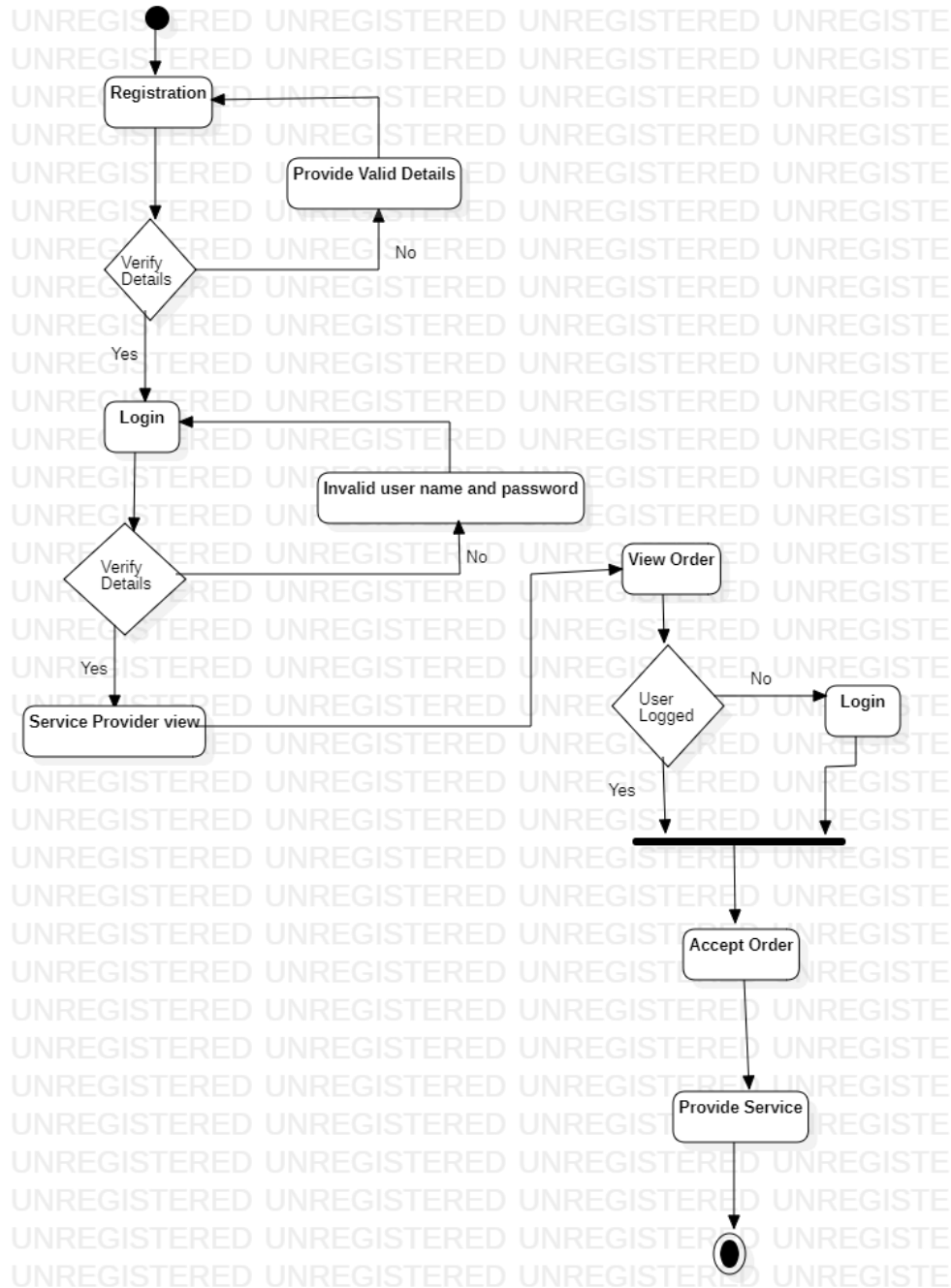


Figure 10-Activity Diagram for Registration

3.6 Sequence Diagram

In the field of software engineering, a sequence diagram, also known as a system sequence diagram, displays process interactions arranged in time sequence. It illustrates the processes involved and the order in which messages must be exchanged for the processes to perform the functionality. [10]

3.6.1 Sequence Diagram for Admin

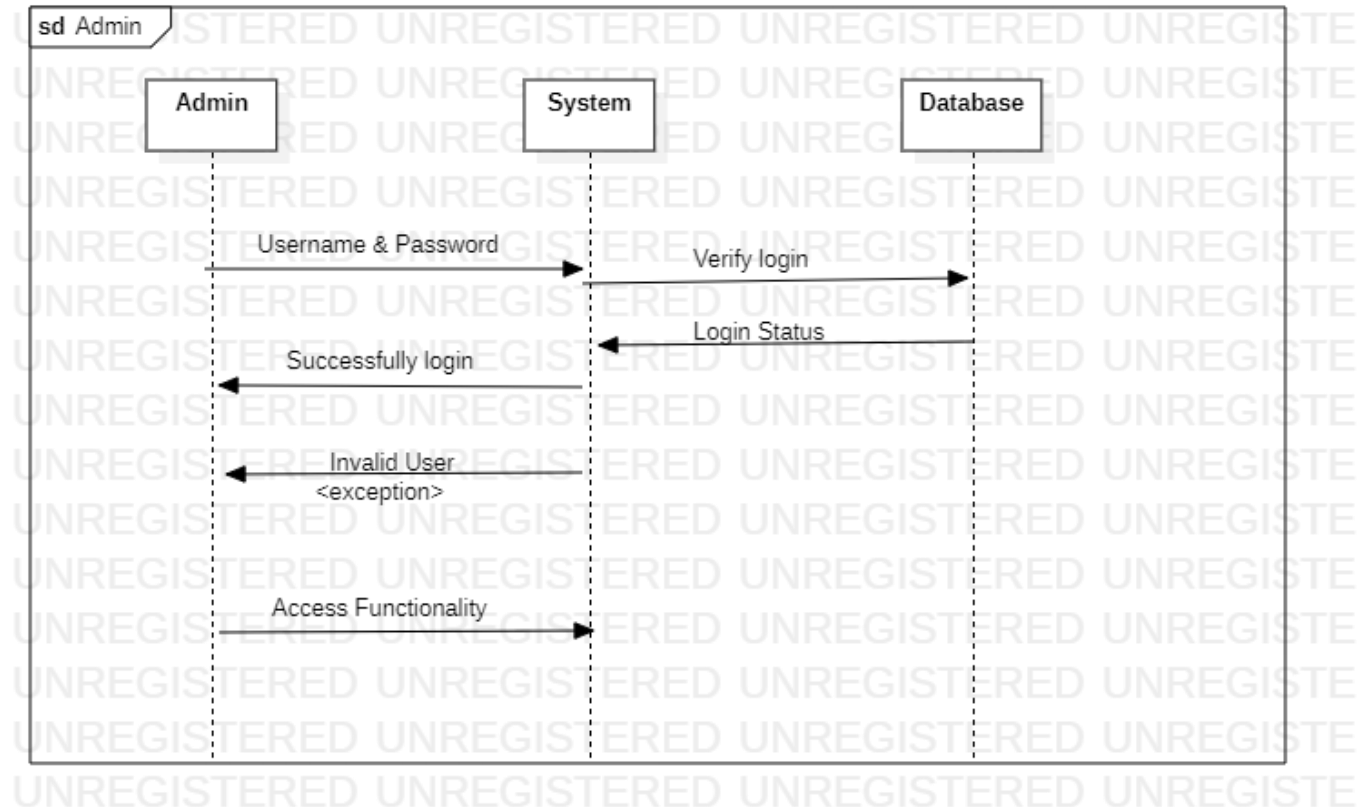


Figure 11-Sequence Diagram for Admin

3.6.2 Sequence Diagram for Order

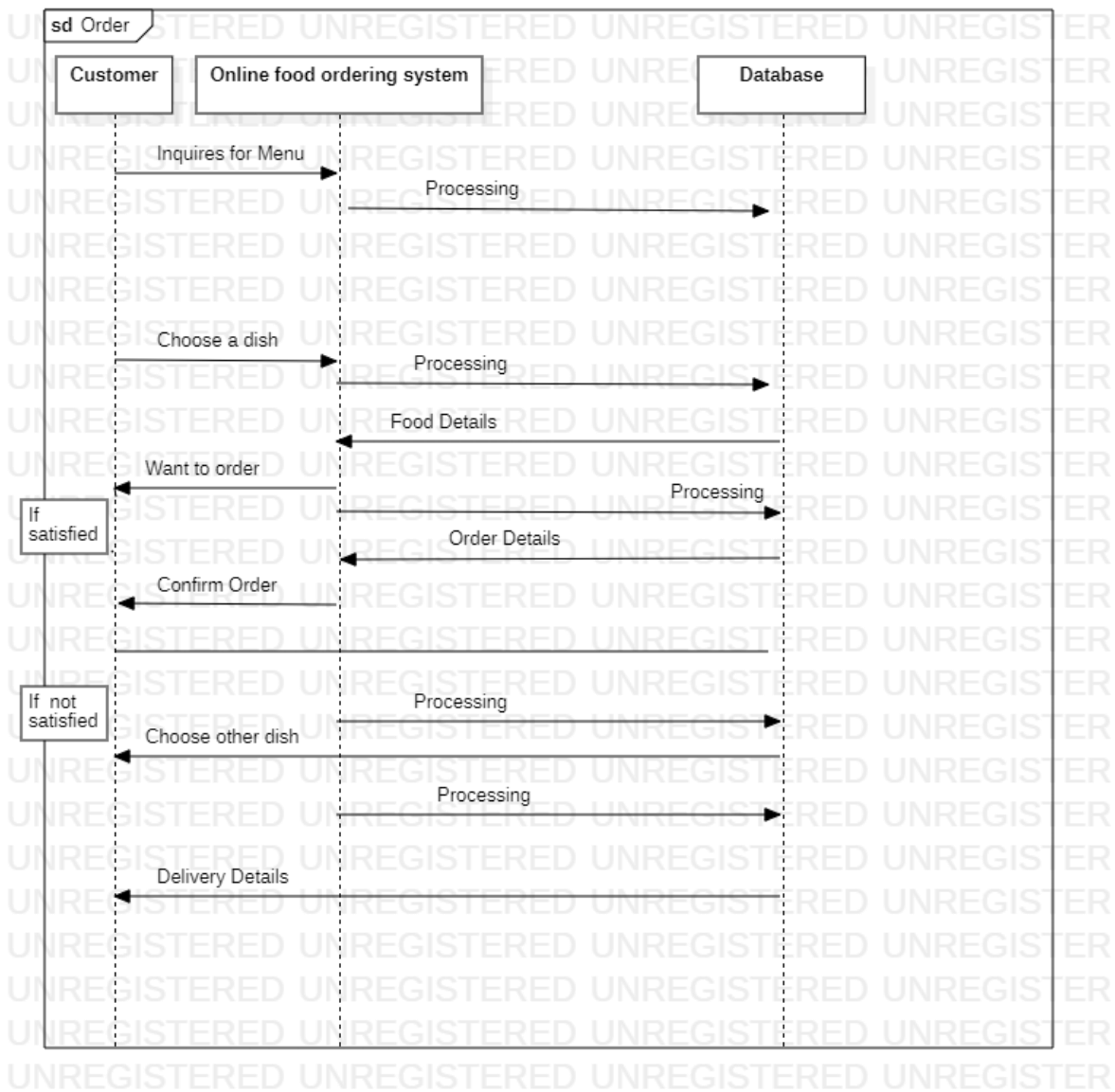


Figure 12-Sequence Diagram for Order

3.6.3 Sequence Diagram for Registration

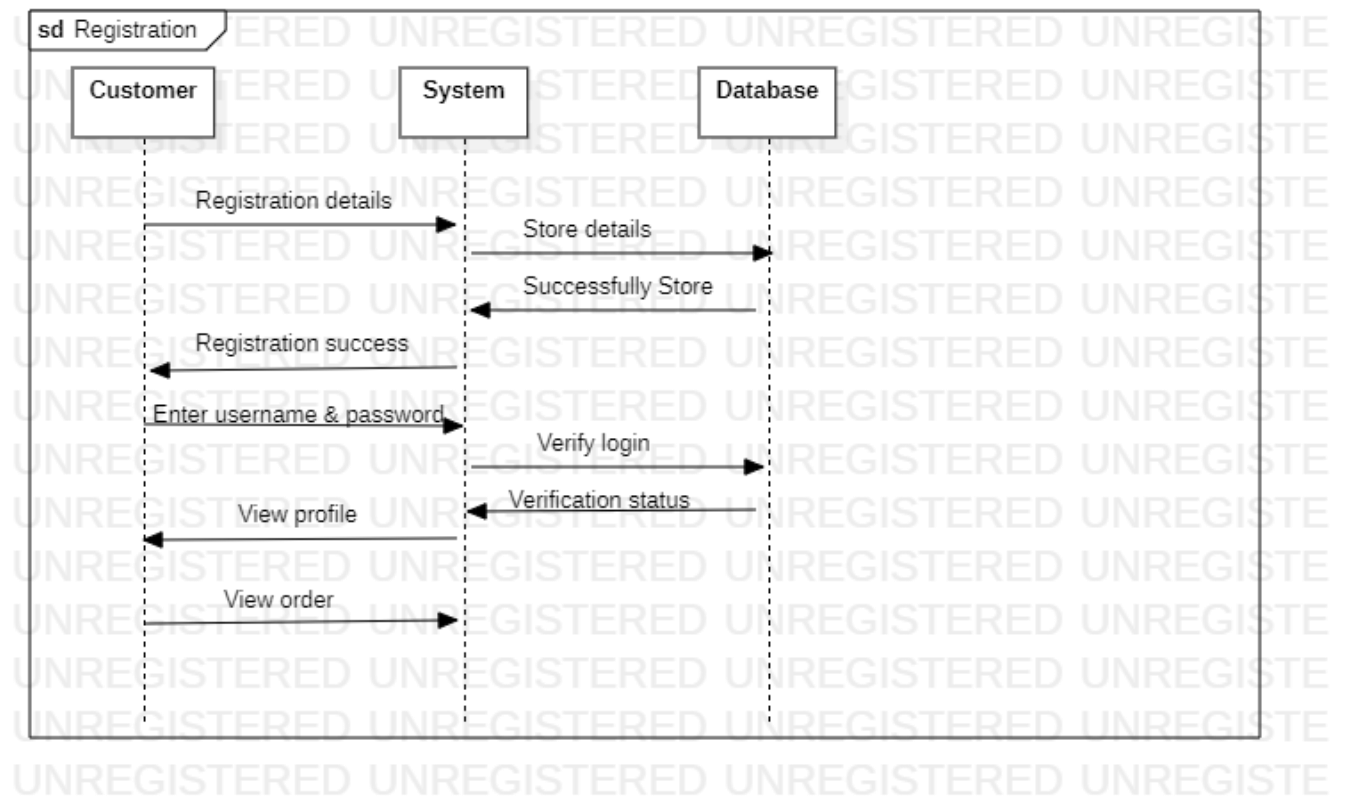


Figure 13-Sequence Diagram for Registration

3.7 EER Diagram

Through high-level models and tools, EER Diagram, also known as "Enhanced Entity-Relationship Diagram," assists us in building and maintaining comprehensive databases. Additionally, they are an extended form of the fundamental ER diagrams and were created from them. [11]

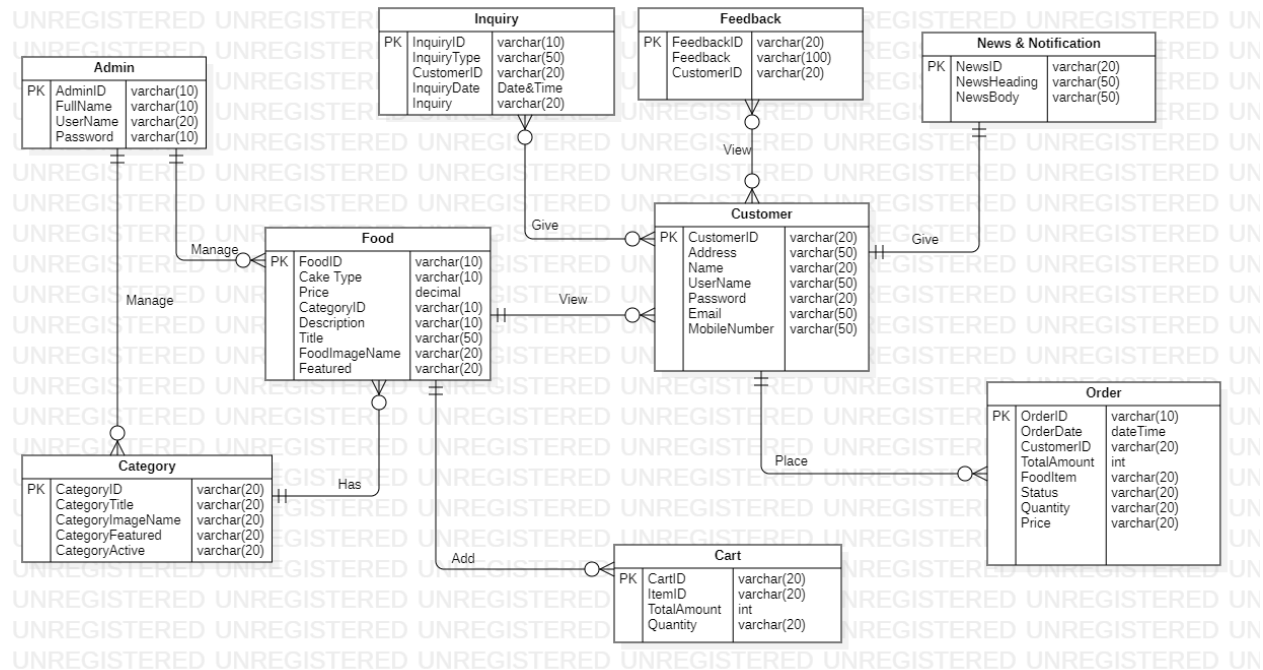


Figure 14-EER Diagram for Propose System

EER Diagram Assumptions

- Specific Food items have only one category and one subcategories.
- The Customer ID of each table identifies each customer, order, Customer Request, inquiry, category, payment, Order, Food and Customer Request.
- Customers can order Food items and they can customize Food then they can customize their Food.
- The customer can add inquiries about their Food items.
- Customers can pay their payment after the purchasing and delivering.

3.7.1 Relational Schema

Customer Table

CustomerID	CustomerName	CustomerEmail	CustomerAddress	CustomerPhoneNo	UserName	Password
------------	--------------	---------------	-----------------	-----------------	----------	----------

Admin Table

AdminID	AdminFullName	UserName	Password
---------	---------------	----------	----------

Category Table

CategoryID	CategoryTitle	CategoryImageName	CategoryFeatured	CategoryActive
------------	---------------	-------------------	------------------	----------------

Food Table

FoodID	Title	Description	Price	FoodImageName	CategoryID	Featured
--------	-------	-------------	-------	---------------	------------	----------

Order Table

OrderID	FoodItems	OrderDate	CustomerID	Price	Qty	Total	Status
---------	-----------	-----------	------------	-------	-----	-------	--------

Cart Table

CartID	ItemID	Quantity	TotalAmount
--------	--------	----------	-------------

Inquiry Table

InquiryID	InquiryDate	InquiryType	Inquiry	CustomerID
-----------	-------------	-------------	---------	------------

Feedback Table

FeedbackID	Feedback	CustomerID
------------	----------	------------

For database design refer appendix B

3.8 Summary

This chapter covered the topic of creating diagrams. There are several schematics for designing. Use case diagrams, Activity diagrams, Sequence diagrams, and Class diagrams are all located beneath the UML diagrams. EER diagram and Relational Schema are used in database design. In order to better comprehend the system, use case diagrams and use case narratives are generated for each module. For further details on the diagrams, see Appendices A and B. The next report will contain the remaining chapter.

4 Chapter 04- Technologies Adapted

4.1 Introduction

The above chapter covered the topic of creating diagrams. There are several schematics for designing. Use case diagrams, Activity diagrams, Sequence diagrams, and Class diagrams are all located beneath the UML diagrams. EER diagram and Relational Schema are used in database design. In order to better comprehend the system, use case diagrams and use case narratives are generated for each module.

This chapter covered the topic of technologies adapted. There are requirement in this chapter, Software requirement, Hardware requirement, Technologies, and Other requirement.

4.2 Software Requirements

4.2.1 Star UML

Star UML is an open source software modeling tool that supports the UML (Unified Modeling Language) framework for system and software modeling. It is based on UML version 5.0.2, provides eleven different types of diagram and it accepts UML 2.0 notation. It actively supports the MDA (Model Driven Architecture) approach by supporting the UML profile concept and allowing to generate code for multiple languages. Version 5.0.2 is used for draw diagrams. [12]

4.2.2 phpMyAdmin

phpMyAdmin is a free software tool written in PHP, intended to handle the administration of MySQL over the Web. phpMyAdmin supports a wide range of operations on MySQL and MariaDB. Frequently used operations (managing databases, tables, columns, relations, indexes, users, permissions, etc.) can be performed via the user interface, while you still have the ability to directly execute any SQL statement. [13]

4.2.3 Operating System

The operating system (OS) controls all of the computer's software and hardware. Windows 10 has an official end of support date of October, 2025, with Windows 11 as its successor. This computer's operating system Version is Windows 10 pro. [14]

4.2.4 Visual Code

An easier-to-use code editor called Visual Studio Code supports debugging and version control as well as other development-related tasks. It aims to provide only the tools necessary for a quick cycle of code-build-debugging, leaving more complex workflows to fully functional IDEs like Visual Studio IDE. Version 1.74.2 is used for implementation. [15]

4.2.5 XAMPP

MariaDB, PHP, and Perl are all included in the totally free and easy-to-install Apache distribution known as XAMPP. Installing and using the XAMPP open-source software is incredibly straightforward. We used XAMPP Control panel V3.3.0. [16]

4.3 Hardware Requirements

4.3.1 RAM

The RAM (random access memory) in the CPU is where data, programs, and program output are kept. During machine operation, data is stored in this read/write memory. When a machine is turned off, all data is erased. 4GB RAM is used for this. [17]

4.3.2 Processor

A processor is the piece of logic hardware that receives and processes the fundamental commands that run a computer (CPU). Since it interprets the majority of the computer's commands, the CPU is regarded as the main and most significant integrated circuit (IC) chip in a computer. In there, the processor is Intel(R) Core(TM) i3-2348M CPU @ 2.30GHz 2.30GHz. [18]

4.4 Technologies

4.4.1 PHP

With the help of the programming language PHP, web designers can create dynamic content that communicates with databases. Most web-based software applications are made with PHP. [13]

4.4.2 CSS

A simple and straightforward method for managing the style of web content is CSS. The term "Cascading Style Sheet" is CSS. Cascading Style Sheets, also known as CSS, is a straightforward design language created to streamline the process of making web pages aesthetically pleasing. [19]

4.4.3 HTML

Hyper Text Markup Language, also known as HTML, is the most popular language for creating web pages on the Internet. [20]

4.4.4 JavaScript

JavaScript is a simple, interpreted programming language. The creation of network-centric applications is its intended use. It enhances and combines with Java. JavaScript is comparatively easy to implement because it is integrated with HTML. It is open and cross-platform. [21]

4.4.5 MySQL

The relational database management system MySQL is open source and free. In tables made up of rows and columns, MySQL stores data in a similar manner to other relational databases. Users can define, manipulate, control, and query data using SQL, or Structured Query Language. The most well-liked open-source database system, MySQL, is a strong and versatile tool. [22]

4.5 Other Requirements (N/A)

4.6 Summary

This chapter discusses the adapted technologies described in the previous chapter. There are software requirements, hardware requirements, and technologies under "technologies adapted." In this chapter, we describe the software that we use to create the suggested system under the software requirements section. In this chapter, desired hardware and technologies like HTML, CSS, JavaScript, PHP, and MySQL are listed under hardware requirements.

The proposed system's implementation section is shown in the following chapter. It shows a variety of frontend and backend interfaces for the suggested web application. It also shows the pertinent code segments.

5 Chapter 05-Implementation

5.1 Introduction

The fourth chapter describes the technologies needed to create the proposed system, as well as the hardware and software requirements. This will give users a better idea of how to meet their needs before a web-based application is implemented.

The user interfaces of the suggested system are described in this chapter. The frontend and backend interfaces are shown in this chapter along with the corresponding code segments.

5.2 User Interfaces

5.2.1 Font End User interfaces

The layer above the back end is the front end and it includes all software or hardware that is part of a user interface. Human or digital users interact directly with various aspects of the front end of a program, including user-entered data, buttons, programs, websites and other features. Most of these features are designed by user experience professionals to be accessible, pleasant and easy to use.

5.2.1.1 Customer Registration

This represents the set of customers, which are the clients who will be using this application. The customers are for whom the system is being designed.

Its attribute set includes:

- Name: This is the name of the customer, searching or purchasing the products. When signing up to the website the name of customer is stored, this is done for the future referencing and maintaining the user's data record. It is the composite attribute which contains two more attributes that is First_Name and Last_Name. That contains user's first name and last name.
- User name: This is the identity number assigned by the admin to the users so as to identify them uniquely in future. This identity number is helpful in fetching data of individual user from a big set. This is mainly to manage the huge database system where the entire data is

being stored. It is a permanent identity number given by the admin to the customer to maintain customer history.

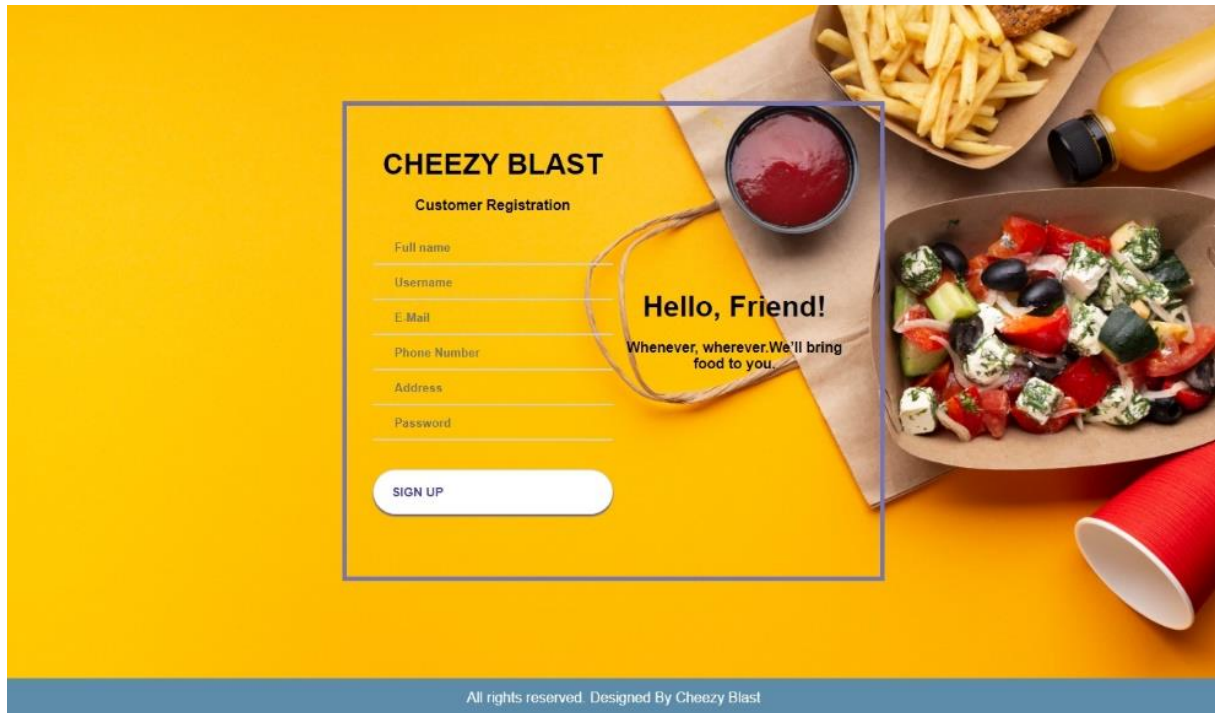


Figure 15-Interface of Customer Registration

5.2.1.2 About Us

This feature will provide the information about the owner, team members or partners and developers, information like ranking of the website, average daily page visits, and journey (like when the system was launched, from where does the motivation came from) will be included

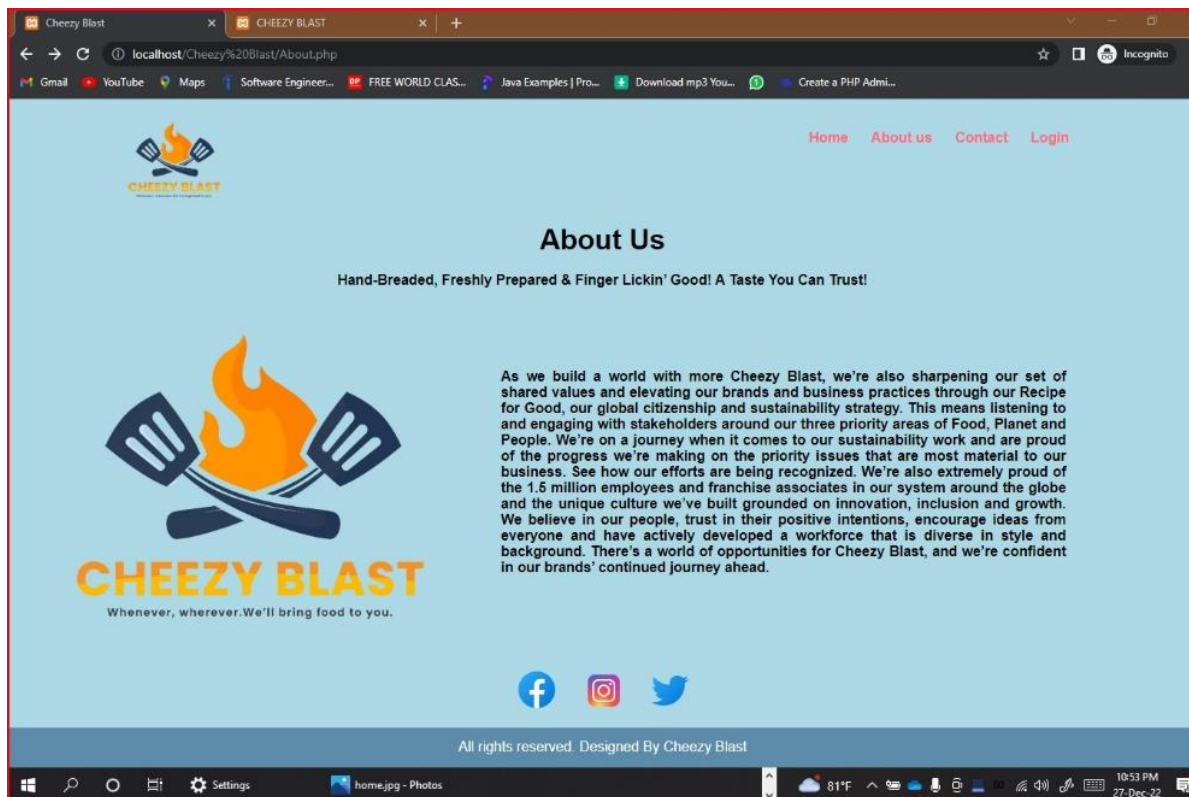


Figure 16-Interface of About Us

5.2.1.3 Home

This is one of the main task of the developer to design a graphical user interface that user attracts to and can use easily, in one word it should be user friendly. So for this you should have better understanding of customers likes and dislikes and the features that are in trend and mesmerize the public easily, initially we need to locate the targeting people that what kind of application do they need. After getting all this information we should start to design the application.

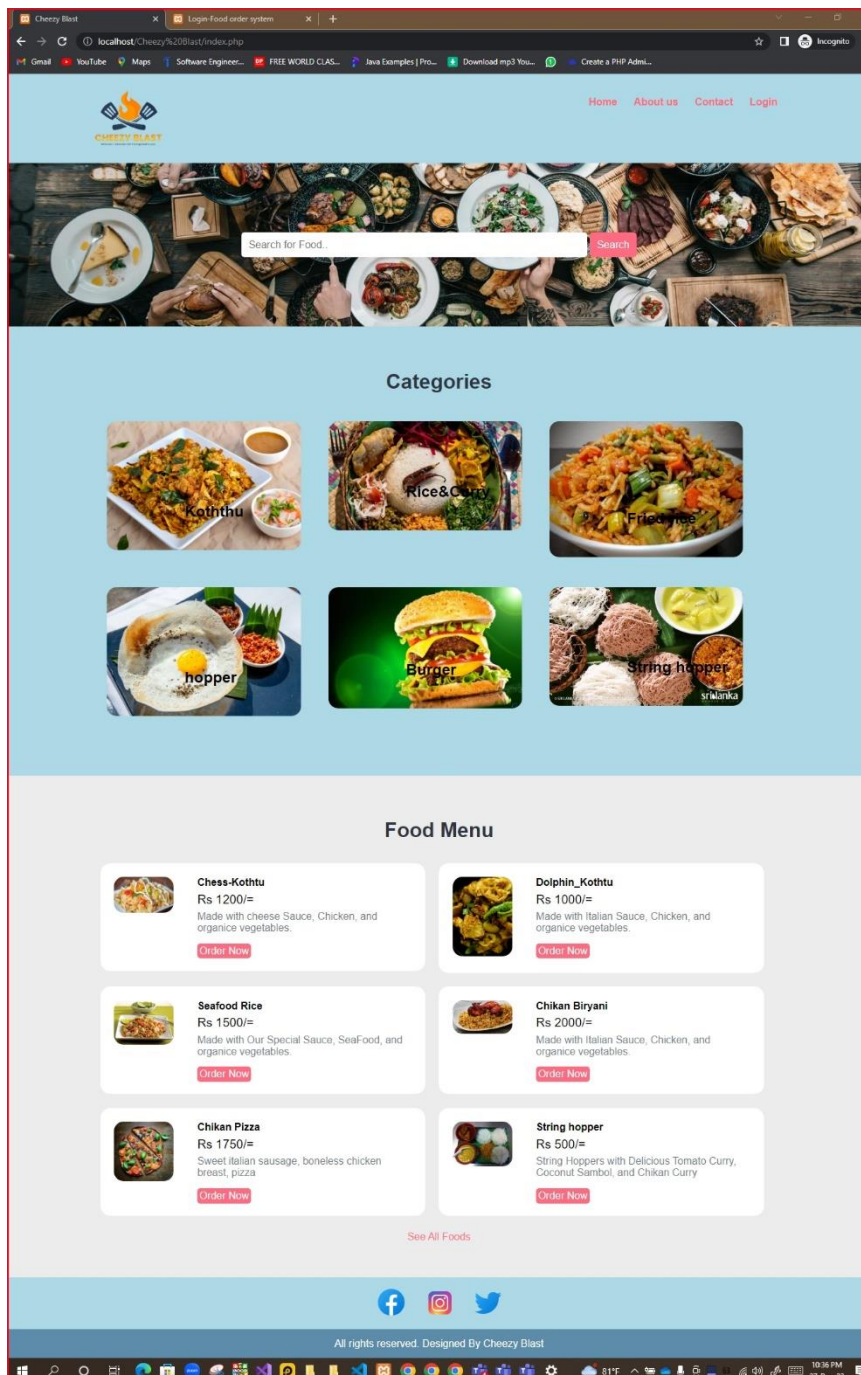


Figure 17-Interface of Home

5.2.2 Back end user interfaces

The back end refers to parts of a computer application or a program's code that allow it to operate and that cannot be accessed by a user. Most data and operating syntax are stored and accessed in

the back end of a computer system. Typically the code is comprised of one or more programming languages. The back end is also called the data access layer of software or hardware and includes any functionality that needs to be accessed and navigated to by digital means.

The following figure interfaces which belong to administrator and it is use to add new administrator , delete an existing administrator and edit the current administrator information. After every deletion, or updating a feedback is given to user whether the modification updates the database or not.

5.2.2.1 Admin Login

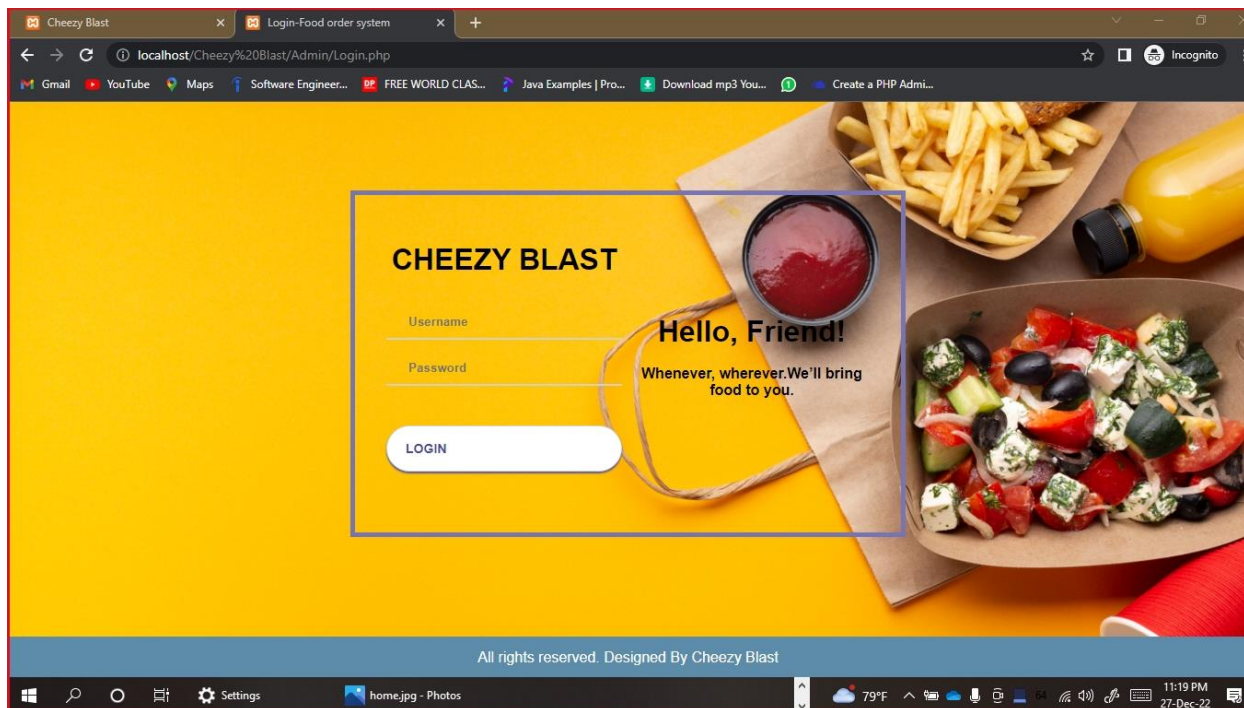


Figure 18-Interface of Admin Login

5.2.2.2 Admin Change Password

Menu Admin Category Food Order Logout

Change Password

117

Current Password :

New Password :

Confirm Password :

[Change Password](#)

All rights reserved. Designed By Cheezy Blast

Figure 19-Interface of Admin Change Password

5.2.2.3 Manage Admin

Menu Admin Category Food Order Logout

Admin

New record Update successfully.

ID	Full_Name	Username	Actions
117	Sandeepa Sudharaka	Sandeepa@admin	Change Password Update Admin Delete Admin
118	Heshan Praneeth	Heshan@admin	Change Password Update Admin Delete Admin
119	Pubudu Lakshan	Pubudu@admin	Change Password Update Admin Delete Admin

[Add Admin](#)

All rights reserved. Designed By Cheezy Blast

Figure 20-Interface of Manage Admin

For more user interfaces refer Appendix C

5.3 Code Segments

Video provides a powerful way to help you prove your point. When you click Online Video, you can paste in the embed code for the video you want to add. You can also type a keyword to search online for the video that best fits your document.

5.3.1 Front End

5.3.1.1 Customer Registration

```
<html>
```

```
  <head>
```

```
    <title>Login-Food order system</title>
```

```
    <link rel="stylesheet" href="css/style.css">
```

```
  </head>
```

```
<body class="body1">
```

```
  <div class="login">
```

```
    <table>
```

```
      <tr>
```

```
        <td width="50%">
```

```
          <h1 class="text-center" >CHEEZY BLAST</h1><br>
```

```
          <h4 class="text-center" >Customer Registration</h4><br>
```

```
        </td>
```

```
      <td rowspan="8">
```

```
        <h1 class="text-center">Hello, Friend!</h1><br>
```

<p class="text-center">Whenever, wherever.We'll bring food
to you.</p>

</td>

</tr>

<?php

```
if(isset($_SESSION['login']))
```

```
{
```

```
    echo $_SESSION['login'];
```

```
    unset($_SESSION['login']);
```

```
}
```

```
if(isset($_SESSION['No_login_message']))
```

```
{
```

```
    echo $_SESSION['No_login_message'];
```

```
    unset($_SESSION['No_login_message']);
```

```
}
```

?>

<form action="" method="POST" >

```

<tr>

<td>

    <input type="text" placeholder="Full name" name="Cusfullname" class="login__input">

</td>

</tr>

<tr>

<td>

    <input type="text" placeholder="Username" name="Cususer" class="login__input">

</td>

</tr>

<tr>

<td>

    <input type="mail" placeholder="E-Mail" name="Cusemail" class="login__input">

</td>

</tr>

<tr>

<td>

    <input type="number" placeholder="Phone Number" name="CusPhoneNumber"
class="login__input">

</td>

</tr>

```

```

</tr>

<td>

    <input type="text" placeholder="Address" name="CusAddress" class="login__input">

</td>

</tr>


<tr>

<td>

    <input      type="password"      placeholder="Password"      name="Cuspassword"
class="login__input">

</td>

</tr>

<tr>

<td>

                                <input      type="submit"      value="Login"      name="submit"
class="login__submit">

</td>

</tr>

</form>


</table>

</div>

```

```

</body>

</html>

<?php

if(isset($_POST['submit']))

{

    $username=$_POST['user'];

    $password=$_POST['password'];

    $sql="SELECT Username AND Password FROM customer WHERE Username='$username'
AND Password='$password'";

    $count=mysqli_num_rows($conn->query($sql));

    if($count==1)

    {

        $_SESSION['login'] ="<b><p style='color:green'>Login Successful.</p></b>";

        $_SESSION['user']=$username;

        header('location:'.$SITEURL.'index.php/');

    }

    else

```



```

{
    $_SESSION['login'] ="<b><p style='color:red'>Login  Not Successful</p></b>";

    header('location:'.$SITEURL.'Admin/index.php');

}

}

```

?>

<?php include('C:\PHP\htdocs\Cheezy Blast\Partials\Footer.php');?>

<script>

```

function myFunction() {

    var x = document.getElementById("show");

    if (x.type === "password") {

        x.type = "text";

    } else {

        x.type = "password";

    }

}

```

</script>

5.3.1.2 About

```
<!DOCTYPE html>

<html lang="en">

<head>

  <meta charset="UTF-8">

  <!-- Important to make website responsive -->

  <meta name="viewport" content="width=device-width, initial-scale=1.0">

  <title>Cheezy Blast</title>


  <!-- Link our CSS file -->

  <link rel="stylesheet" href="css/style.css">

</head>


<body style="background-color:lightblue;">

  <!-- Navbar Section Starts Here -->

  <section class="navbar">

    <div class="container">

      <div class="logo">

        <a href="#" title="Logo">

        </a>

      </div>
```

```

<div class="menu text-right">

    <ul>

        <li>

            <a href="index.php">Home</a>

        </li>

        <li>

            <a href="About.php">About us</a>

        </li>

        <li>

            <a href="index.html">Contact</a>

        </li>

        <li>

            <a href="CustomerRegistration.php">Login</a>

        </li>

    </ul>

</div>

<div class="clearfix"></div>

</div>

</section>

<!-- Navbar Section Ends Here -->

<table class="tbl-full" align="center">

```

```

<tr>

  <td colspan="2"> <h1 class="text-center" color="blue">About Us</h1>

  <br>

  <h4>Hand-Breaded, Freshly Prepared & Finger Lickin' Good! A Taste You Can Trust!</h4>

</td>

</tr>

<tr>

  <td> </td>

  <td> <div class="wrapper text-justify">

    <P>As we build a world with more Cheezy Blast, we're also sharpening our set of shared
    values and elevating our brands and business practices through our Recipe for Good, our global
    citizenship and sustainability strategy. This means listening to and engaging with stakeholders
    around our three priority areas of Food, Planet and People. We're on a journey when it comes to
    our sustainability work and are proud of the progress we're making on the priority issues that are
    most material to our business. See how our efforts are being recognized.

    We're also extremely proud of the 1.5 million employees and franchise associates in our system
    around the globe and the unique culture we've built grounded on innovation, inclusion and growth.
    We believe in our people, trust in their positive intentions, encourage ideas from everyone and
    have actively developed a workforce that is diverse in style and background.

    There's a world of opportunities for Cheezy Blast, and we're confident in our brands' continued
    journey ahead.</P></div>

  </td>

</tr>

</table>

```

```

<!-- social Section Starts Here -->

<section class="social">

  <div class="container text-center">

    <ul>

      <li>

        <a href="#"></a>

        </li>

        <li>

          <a href="#"></a>

          </li>

          <li>

            <a href="#"></a>

            </li>

          </ul>

        </div>

      </section>

<!-- social Section Ends Here -->

<!-- footer Section Starts Here -->

<section class="footer">

```

```

<div class="container text-center">

    <p>All rights reserved. Designed By Cheezy Blast </p>

</div>

</section>

<!-- footer Section Ends Here -->


</body>

</html>

5.3.1.3 Home
<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <!-- Important to make website responsive -->

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <title>Cheezy Blast</title>


    <!-- Link our CSS file -->

    <link rel="stylesheet" href="css/style.css">

</head>


<body style="background-color:lightblue;">

```

```

<!-- Navbar Section Starts Here -->

<section class="navbar">

  <div class="container">

    <div class="logo">

      <a href="#" title="Logo">

      </a>

    </div>

    <div class="menu text-right">

      <ul>

        <li>

          <a href="index.php">Home</a>

        </li>

        <li>

          <a href="About.php">About us</a>

        </li>

        <li>

          <a href="index.html">Contact</a>

        </li>

        <li>

          <a href="CustomerRegistration.php">Login</a>

        </li>

```

```

        </ul>

    </div>

    <div class="clearfix"></div>

</div>

</section>

<!-- Navbar Section Ends Here -->

<!-- fOOD sEARCH Section Starts Here -->

<section class="food-search text-center">

    <div class="container">

        <form action="food-search.html" method="POST">

            <input type="search" name="search" placeholder="Search for Food.." required>

            <input type="submit" name="submit" value="Search" class="btn btn-primary">

        </form>

    </div>

</section>

<!-- fOOD sEARCH Section Ends Here -->

<!-- CAtegories Section Starts Here -->

<section class="categories">

```



```

<div class="container">

    <h2 class="text-center">Categories</h2>

    <a href="#">

        <div class="box-3 float-container">

            <h3 class="float-text text-white">Koththu</h3>

        </div>

    </a>

    <a href="#">

        <div class="box-3 float-container">

            <h3 class="float-text text-white">Rice&Curry</h3>

        </div>

    </a>

    <a href="#">

        <div class="box-3 float-container">

```

```



<h3 class="float-text text-white">Fried rice </h3>

</div>

</a>

<a href="#">

<div class="box-3 float-container">

  <h3 class="float-text text-white">hopper </h3>

</div>

</a>

<a href="#">

<div class="box-3 float-container">

  <h3 class="float-text text-white">Burger</h3>

</div>

</a>

<a href="#">

<div class="box-3 float-container">

```

```

```

```
<h3 class="float-text text-white">String hopper</h3>
```

```
</div>
```

```
</a>
```

```
<div class="clearfix"></div>
```

```
</div>
```

```
</section>
```

```
<!-- Categories Section Ends Here -->
```

```
<!-- fOOD MEnu Section Starts Here -->
```

```
<section class="food-menu">
```

```
<div class="container">
```

```
<h2 class="text-center">Food Menu</h2>
```

```
<div class="food-menu-box">
```

```
<div class="food-menu-img">
```

```

```

```
</div>
```

```
<div class="food-menu-desc">
```

```
<h4>Chess-Kothtu</h4>
```

```
<p class="food-price">Rs 1200/=</p>
```

```
<p class="food-detail">
```

```
    Made with cheese Sauce, Chicken, and organice vegetables.
```

```
</p>
```

```
<br>
```

```
<a href="order.html" class="btn btn-primary">Order Now</a>
```

```
</div>
```

```
</div>
```

```
<div class="food-menu-box">
```

```
<div class="food-menu-img">
```

```

```

```
</div>
```

```
<div class="food-menu-desc">
```

```
<h4>Dolphin_Kothtu</h4>
```

```
<p class="food-price">Rs 1000/=</p>
```

```
<p class="food-detail">
```

```
    Made with Italian Sauce, Chicken, and organice vegetables.
```

</p>

Order Now

</div>

</div>

<div class="food-menu-box">

<div class="food-menu-img">

</div>

<div class="food-menu-desc">

<h4>Seafood Rice</h4>

<p class="food-price">Rs 1500/=</p>

<p class="food-detail">

Made with Our Special Sauce, SeaFood, and organice vegetables.

</p>

Order Now

</div>

</div>

<div class="food-menu-box">

<div class="food-menu-img">

</div>

<div class="food-menu-desc">

<h4>Chikan Biryani</h4>

<p class="food-price">Rs 2000/=</p>

<p class="food-detail">

Made with Italian Sauce, Chicken, and organice vegetables.

</p>

Order Now

</div>

</div>

<div class="food-menu-box">

<div class="food-menu-img">

```
        
```

```
    </div>
```

```
    <div class="food-menu-desc">
```

```
        <h4>Chikan Pizza</h4>
```

```
        <p class="food-price">Rs 1750/=</p>
```

```
        <p class="food-detail">
```

```
            Sweet italian sausage, boneless chicken breast, pizza
```

```
        </p>
```

```
        <br>
```

```
        <a href="#" class="btn btn-primary">Order Now</a>
```

```
    </div>
```

```
</div>
```

```
<div class="food-menu-box">
```

```
    <div class="food-menu-img">
```

```
        
```

```
    </div>
```

```
    <div class="food-menu-desc">
```

<h4>String hopper</h4>

<p class="food-price">Rs 500/=</p>

<p class="food-detail">

String Hoppers with Delicious Tomato Curry, Coconut Sambol, and Chikan Curry

</p>

Order Now

</div>

</div>

<div class="clearfix"></div>

</div>

<p class="text-center">

See All Foods

</p>

</section>

<!-- fOOD Menu Section Ends Here -->

<!-- social Section Starts Here -->


```

<section class="social">

  <div class="container text-center">

    <ul>

      <li>

        <a href="#"></a>

      </li>

      <li>

        <a href="#"></a>

      </li>

      <li>

        <a href="#"></a>

      </li>

    </ul>

  </div>

</section>

<!-- social Section Ends Here -->


<!-- footer Section Starts Here -->

<section class="footer">

  <div class="container text-center">

    <p>All rights reserved. Designed By Cheezy Blast</p>

```

</div>

</section>

<!-- footer Section Ends Here -->

</body>

</html>

5.3.2 Backend code segments

5.3.2.1 Admin Login

```
<?php include('../Config/Constants.php');
```

```
$SITEURL = 'http://localhost/Cheezy%20Blast/';
```

```
session_destroy();
```

```
header('location:'.$SITEURL.'Admin/Login.php');
```

```
?>
```

5.3.2.2 Admin Change Password

```
<?php include('C:\PHP\htdocs\Cheezy Blast\Partials\Menu.php');?>
```

```
<link rel="stylesheet"href="../css/admin.css">
```

```
<?php $SITEURL = 'http://localhost/Cheezy%20Blast/'; ?>
```

```

<!-- Main section -->

<div class="Main">

    <div class="wrapper">

        <h1>Change Password</h1>

        <br>

        <?php

            if(isset($_GET['id']))

            {

                $id=$_GET['id'];

            }

            $sql="SELECT * FROM admin where ID=$id;";

            // $sql="SELECT * FROM admin where Password=$Current_password;";

            if(mysqli_query($conn,$sql)) {

                echo $id;

                $row=mysqli_fetch_assoc(mysqli_query($conn,$sql));

                $Current_password=$row['Password'];

```

```

    }

    else

    {

header("location:http://localhost/Cheezy%20Blast/Admin/ManageAdmin.php");

    }

?>

<form action="" method="Post">

<table class="tbl-full">

<tr>

<td>Current Password :</td>

<td><input type="password" name="Cpassword "
placeholder="Current Password" value="<?php $Current_password;?>"></td>

</tr>

<tr>

<td>New Password :</td>

<td><input type="password" name="Npassword "
placeholder="New Password"></td>

</tr>

<tr>

```

```

        <td>Comfirm Password :</td>

        <td><input type="password" name="Fpassword"
placeholder="Comfirm Password"></td>

    </tr>

</tr>

        <td colspan="2">

            <input type="hidden" name="ID "
value="<?php echo $id;?>"

            <input type="submit" name="sub m it "
value="Change Password" class="btn-2">

        </td>

    </tr>

</table>

</form>

</div>

</div>

```

```
<?php
```

```
if(isset($_POST['submit']))
```

```
{
```

```
    $id=$_POST['ID'];
```

```
    $Current_password=($_POST['Cpassword']);
```

```
    $New_password=($_POST['Npassword']);
```

```
    $Com_password=($_POST['Fpassword']);
```

```
    $sql="SELECT ID FROM admin WHERE ID='$id' AND Password='$Current_password';
```

```
    $count=mysqli_num_rows($conn->query($sql));
```

```
    // $sql="SELECT * FROM admin WHERE ID='$id';
```

```
    // $sql="SELECT * FROM admin WHERE Password='$Current_password';
```

```
    if($count==1)
```

```
    {
```

```
        if($New_password==$Com_password)
```

```
        {
```

```
            $sql2="UPDATE admin SET Password = '$New_password' WHERE ID='$id';
```

```
            if($conn->query($sql2) === TRUE)
```

```
            {
```

```
                $_SESSION['ChangePwd'] ="<b><p style='color:green>Successfully.</p></b>";
```

```

        header('location:'.$SITEURL.'Admin/ManageAdmin.php');

    }

    else

    {

        $_SESSION['ChangePwd'] ="<b><p style='color:red'>Change password.</p></b>";

        header('location:'.$SITEURL.'Admin/ManageAdmin.php');

    }

}

else

{

    $_SESSION['PasswordNotMatch']      ="<b><p      style='color:red'>Password      not
match.</p></b>";

    header('location:'.$SITEURL.'Admin/ManageAdmin.php');

}

//echo $Current_password;

}

else

{

    $_SESSION['UserNotFound'] ="<b><p style='color:red'>User not found.</p></b>";

    header('location:'.$SITEURL.'Admin/ManageAdmin.php');

}

}

```

?>

```
<?php include('C:\PHP\htdocs\Cheezy Blast\Partials\Footer.php');?>
```

5.3.2.3 Manage Admin

```
<?php include('C:\PHP\htdocs\Cheezy Blast\Partials\Menu.php');?>
```

```
<div class="Main">
```

```
<div class="wrapper">
```

```
<h1>Admin</h1></br>
```

```
<?php
```

```
if(isset($_SESSION['add']))
```

```
{
```

```
    echo $_SESSION['add'];
```

```
    unset($_SESSION['add']);
```

```
}
```

```
if(isset($_SESSION['delete']))
```

```
{
```

```
    echo $_SESSION['delete'];
```



```
unset($_SESSION['delete']);

}

if(isset($_SESSION['update']))
{
    echo $_SESSION['update'];
    unset($_SESSION['update']);
}

if(isset($_SESSION['UserNotFound']))
{
    echo $_SESSION['UserNotFound'];
    unset($_SESSION['UserNotFound']);
}

if(isset($_SESSION['PasswordNotMatch']))
{
    echo $_SESSION['PasswordNotMatch'];
    unset($_SESSION['PasswordNotMatch']);
}
```

```
}
```

```
if(isset($_SESSION['ChangePwd']))
```

```
{
```

```
    echo $_SESSION['ChangePwd'];
```

```
    unset($_SESSION['ChangePwd']);
```

```
}
```

```
?>
```

```
</br>
```

```
<table class="tbl-full">
```

```
    <tr>
```

```
        <th>ID</th>
```

```
        <th>Full_Name</th>
```

```
        <th>Username</th>
```

```
        <th>Actions</th>
```

```
    </tr>
```

```
<?php
```

```
$sql = "SELECT ID, Full_Name, Username FROM admin";
```

```
$result = $conn->query($sql);
```

```

if ($result->num_rows > 0) {

    // output data of each row

    while($row = $result->fetch_assoc()) {

        //$_GET['id'] = $row['ID'];

    }

    ?>

    <tr>

    <td><?php echo $row['ID'] ?></td>

    <td><?php echo $row['Full_Name'] ?></td>

    <td><?php echo $row['Username'] ?></td>

    <td>

        <a
                                href="<?php
                                echo
$SITEURL?>Admin/UpdatePassword.php?id=<?php echo $row['ID']; ?>"class="btn-1">Change
Password</a>

        <a
                                href="<?php
                                echo
$SITEURL?>Admin/UpdateAdmin.php?id=<?php echo $row['ID']; ?>" class="btn-2">Update
Admin</a>

        <a
                                href="<?php
                                echo
$SITEURL?>Admin/Delete.php?id=<?php echo $row['ID']; ?>" class="btn-dlt">Delete
Admin</a>

        <!-- <a href="<?php ?>" class="btn-dlt">Delete Admin</a>

    -->

    </td>

```

```

        </tr>

        <?php
        }

        }else {

            echo "0 results";

        }

        //$conn->close();

        ?>

    </table>

    </br>

    <a href="Add-Admin.php" class="btn-1">Add Admin</a>

</div>

</div>

```

```
<?php include('C:\PHP\htdocs\Cheezy Blast\Partials\Footer.php')?>
```

For more code segments refer Appendix D

5.4 Summary

The system is designed with several interaction cues on each web page that makes up the web application. These cues are well-defined such as to make several functionality that the application exposes to collect, process and output data.

6 Chapter 06 – Testing and Evaluation

6.1 Introduction

Details about the proposed system's implementation process are revealed in the fifth chapter. Several user interfaces for the suggested system are shown. A few of the front-end and back-end interfaces of the suggested system are shown in the aforementioned chapter. Additionally, it shows the system's front end and back end code segments.

This chapter describes the proposed system's testing and evaluation phases. We have selected unit testing, black box testing, security testing, integration testing, and system testing as the testing types. The test cases for the proposed system are revealed in this chapter.

6.2 Types of Software Testing

6.2.1 Unit Testing

A unit test is a way of testing a unit - the smallest piece of code that can be logically isolated in a system. That is a method, a function, a subroutine, or a property in the majority of programming languages. The definition's isolated portion is crucial. Michael Feathers, the author of "Working Effectively with Legacy Code," claims that such tests are not unit tests if they depend on external systems. "If it requires system configuration, can't be run concurrently with any other test, talks to the database, talks across the network, touches the file system. [23]

6.2.2 Back Box Testing

Black box testing is a technique for testing software applications' functionalities without having access to their internal code structure, implementation specifics, or internal paths. Black Box Testing is entirely based on software requirements and specifications and primarily concentrates on the input and output of software applications. Additionally called behavioral testing. [24]

6.2.3 Security Testing

Security testing is a type of software testing that uncovers vulnerabilities of the system and determines whether the data and resources of the system are protected from possible intruders. It ensures that the software system and application are free from any threats or risks that can cause a loss. Security testing of any system is focused on finding all possible loopholes and weaknesses of the system which might result in the loss of information or reputation of the organization. [25]

6.2.4 Integration Testing

Integration testing, also known as integration and testing (I&T), is a type of software testing in which the different units, modules, or components of a software application are tested as a combined entity. However, these modules may be coded by different programmers. [26]

6.2.5 System Testing

System testing is a level of testing that validates the complete and fully integrated software product. The purpose of a system test is to evaluate the end-to-end system specifications. Usually, the software is only one element of a larger computer-based system. Ultimately, the software is interfaced with other software/hardware systems. System testing is defined as a series of different tests whose sole purpose is to exercise the full computer-based system. [27]

6.3 Test Cases

Project Details		Project Name Module Name Creted By Reviewd By Created Date:	Cheezy Blast cart Heshan Sandeepa 26-Dec-22 Reviewed Date 27-Dec-22								
Test Case ID	Test Description/Test case name	Prerequisite	Test Steps	Input Data	Expected Result	Actual Result	Status	Severity	Priority	Executed by	Comments
TC_CART_001	ADD ITEMS TO THE CART	Login to the system as registred user	1.browse cheezyblast.lk 2. click on the logging button 3. SELECT CATEGORY AND sub category 4. CLICK ON ADD TO CART BUTTON	Selete category	VIEW ITEMS ON CART	As expected	PASS			Sandeepa, Tester	
TC_CART_002	DELETE ITEMS TO THE CART	Login to the system as registred user	1.browse cheezyblast.lk 2. click on the logging button 3. SELECT CATEGORY AND sub category 4. CLICK ON ADD TO CART BUTTON 5. DELECT ITEM FROM CART	Selete category	DELETED ITEM NOT SEEN ON CART	As expected	PASS			Sandeepa, Tester	

Figure 21-Test Case of Cart

Project Details		Project Name Module Name Created by Reviewed by Created Date		Cheerzy Blast Login Heshan Sandeepa 25/12/2022		Review Date 26/12/2022					
Test Case ID	Test Description/Test case name	Prerequisite	Test Steps	Input Data	Expected Result	Actual Result	Status	Severity	Priority	Executed by	Comments
TC_Login_001	Enter valid username and valid password	Visit the Cheerzy blast.lk	1. browse cheerzyblast.lk 2. click on the login button 3. enter valid user name and password 4. click on submit button	Valid user name and valid password	Direct to profile of this customer	As expected	Pass			Sandeepa - Test leader	
TC_Login_002	Enter valid username and invalid password	Visit the Cheerzy blast.lk	1. browse cheerzyblast.lk 2. click on the login button 3. enter valid user name and invalid password 4. click on submit button	Valid user name and invalid password	Alert message as "Invalid user name or password"	As expected	Pass			Sandeepa - Test leader	
TC_Login_003	Enter invalid username invalid password	Visit the Cheerzy blast.lk	1. browse cheerzyblast.lk 2. click on the login button 3. enter invalid user name and invalid password 4. click on submit button	Invalid user name and invalid password	Alert message as "Invalid user name or password"	As expected	Pass			Sandeepa - Test leader	
TC_Login_004	Enter invalid username valid password	Visit the Cheerzy blast.lk	1. browse cheerzyblast.lk 2. click on the login button 3. enter invalid user name and valid password 4. click on submit button	Invalid user name and valid password	Alert message as "Invalid user name or password"	As expected	Pass			Sandeepa - Test leader	
TC_Login_005	Enter empty user name and empty password	Visit the Cheerzy blast.lk	1. browse cheerzyblast.lk 2. click on the login button 3. enter empty user name and empty password 4. click on submit button	Empty user name and empty password	Alert message as "give credentials"	As expected	Pass			Sandeepa - Test leader	
TC_Login_006	Enter empty user name and valid password	Visit the Cheerzy blast.lk	1. browse cheerzyblast.lk 2. click on the login button 3. enter empty user name and valid password 4. click on submit button	Empty user name and valid password	Alert message as "Invalid user name or password"	As expected	Pass			Sandeepa - Test leader	
TC_Login_007	Enter empty user name and invalid password	Visit the Cheerzy blast.lk	1. browse cheerzyblast.lk 2. click on the login button 3. enter EMPTY user name and INVALID password 4. click on submit button	Empty user name and invalid password	Alert message as "Invalid user name or password"	As expected	Pass			Sandeepa - Test leader	
TC_Login_008	Enter valid user name and empty password	Visit the Cheerzy blast.lk	1. browse cheerzyblast.lk 2. click on the login button 3. enter valid user name and empty password 4. click on submit button	Valid user name and empty password	Alert message as "Invalid user name or password"	As expected	Pass			Sandeepa - Test leader	
TC_Login_009	Invalid username and empty password	Visit the Cheerzy blast.lk	1. browse cheerzyblast.lk 2. click on the login button 3. enter invalid user name and empty password 4. click on submit button	Invalid user name and empty password	Alert message as "Invalid user name or password"	As expected	Pass			Sandeepa - Test leader	

Figure 22-Test Case of Login

Project Details		Project Name Module Name Creted By Reviewd By Created Date:	cheezyblast.lk search Heshan Sandeepa 26-Dec-22 Reviewed Date	27-Dec-22							
Test Case ID	Test Description/Test case name	Prerequisite	Test Steps	Input Data	Expected Result	Actual Result	Status	Severity	Priority	Executed by	Comments
TC_Serach_Box_001	Enter valied keyword which is related to UoM		1.browse cheezyblast.lk 2. insert valied keyword on search box 3. press enter	Valied Keyword:Pizza	show all related words	As expected	Pass			Sandeepa, Tester	
TC_Serach_Box_002	Enter invalied keyword which is not related to UoM	Vist the cheezyblast.lk	1. browse cheezyblast.lk 2. insert invalied keyword on search box 3. press enter	Invalid Keyword:Dell	No result	As expected	Pass			Sandeepa, Tester	
TC_Search_Box_003	Enter Empty search box	Vist the cheezyblast.lk	1. browse cheezyblast.lk 2. Press enter without inserting any type of keyword 3. press enter		RED COLOR WARNNING BOX	As expected	Pass			Sandeepa, Tester	

Figure 23-Test case of Search

Note-Refer the Exel Sheet

Rest of the test cases refer appendix E

6.4 Summary

This chapter explains how testing and evaluation are carried out using a variety of testing techniques, including unit testing, black box testing, security testing, and system testing. Each testing technique illustrates how the testing is carried out using various project sections and how the final results are presented.

The project's executive summary and plans for implementing the suggested system to further improve the hotel will be covered in the following chapter.

7 Chapter 07 – Conclusion and Future work

7.1 Introduction

Details about the proposed system's testing and evaluation phases in the sixth chapter. We had selected unit testing, black box testing, security testing, integration testing, and system testing as the testing types. The test cases for the proposed system are revealed in above chapter.

This chapter describes the proposed system's conclusion. The website has been developed as a public site and anyone could access it via web. Getting the look and feel according to satisfaction of all the customers was another challenge successfully faced during design of the layouts and themes of the website

7.2 Conclusion

An online food ordering system is developed where the customers can make an order for the food and avoid the hassles of waiting for the order to be taken by the waiter. Using the application, the end users register online, read the E-menu card and select the food from the e-menu card to order food online. Once the customer selects the required food item the chef will be able to see the results on the screen and start processing the food. This application nullifies the need of a waiter or reduces the workload of the waiter. The advantage is that in a crowded restaurant there will be chances that the waiters are overloaded with orders and they are unable to meet the requirements of the customer in a satisfactory manner. Therefore by using this application, the users can directly place the order for food to the chef online.

In conclusion an online food ordering system is proposed which is useful in small family run restaurants as well as in places like college cafeteria, etc. This project can later be expanded on a larger scale. It is developed for restaurants to simplify their routine managerial and operational task and to improve the dining experience of the clients. This also helps the restaurant owners develop healthy customer relationships by providing reasonably good services. The system also enables the restaurant to know the items available in real time and make changes to their food and beverage inventory based on the orders placed and the orders completed.

7.3 Future work

In the future it is expected to further improve the functional and non- functional requirements for satisfying the client.

- Registration Details send for mobiles of the users via SMS.
- Credit Note, Debit Note Handling.
- Online Payments

7.4 Challenges faced

The main problem with the development of the system was the lack of basic knowledge of languages on development tools and languages. Online tutorials, forums and books were used to gain the n required level of knowledge.

7.5 Summary

Knowledge acquired throughout the project was really important. In addition, we were given this experience by giving me a complete software development life-cycle from the feasibility studies to end the project. This project gave me the opportunity to test and implement the most important theories and technologies learned through the BIT degree program. When assigning the project proposal did not have much of an idea on how to carry out this project. A valuable knowledge of how to make a successful professional system development project can be done when step-by-step takes place according to the guidelines provided by the university. We learned to schedule my day-to-day activities according to a set schedule, efficiently managing the time needed for development. Further contributing to the work of the project helped me to develop technical skills and intellectual skills.

This Appendix mainly related to chapter 03 Use case diagrams and their narratives (For more information refer chapter 03).

Use Case Diagram for Feedback

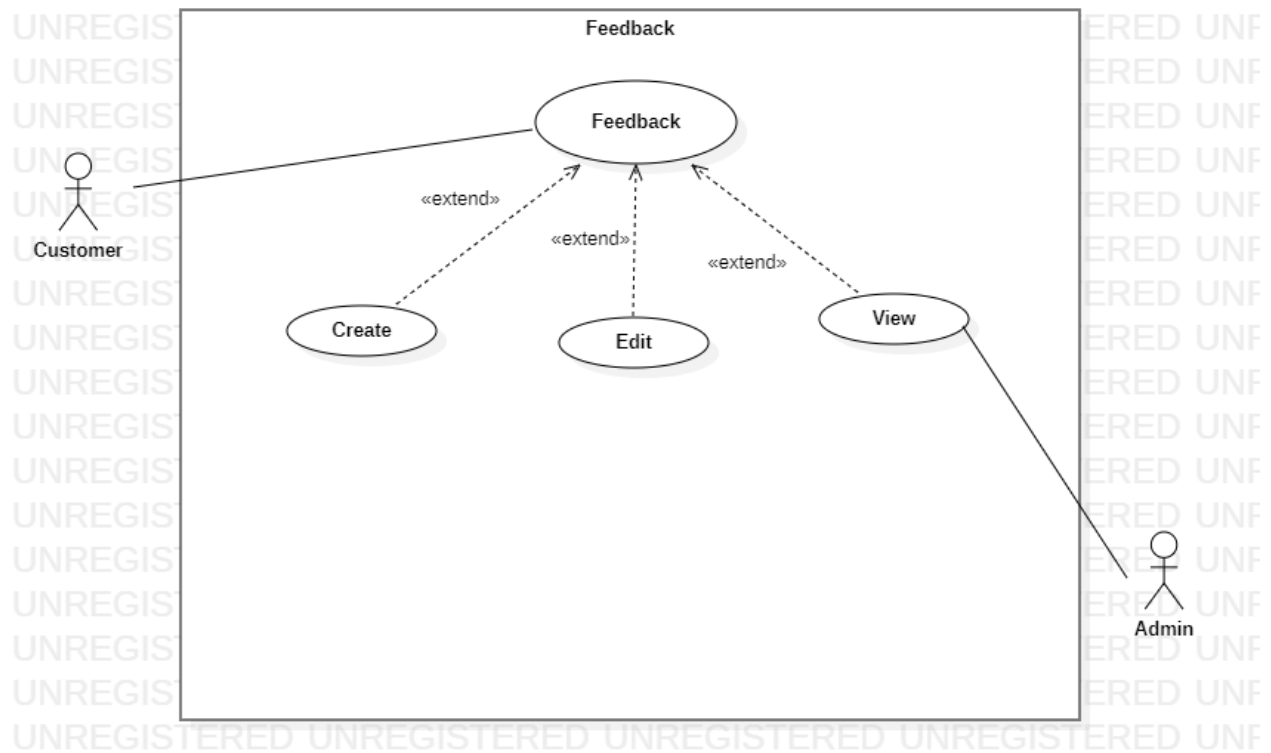


Figure 24-Use case of feedback

Table 7-Narrative Table for Feedback

Use-Case Name:	Feedback	Use-Case Type - Business Requirements
Use-Case ID:	FO-07	
Priority:	High Priority	
Source:	Online Form	
Primary Business Actor:	Admin	
Other Participating Actors:	Customer	
Other Interested Stakeholders:		
Description:	1. Create Feedback 2. Edit Feedback 3. View Feedback	

Use Case Diagram for Cart

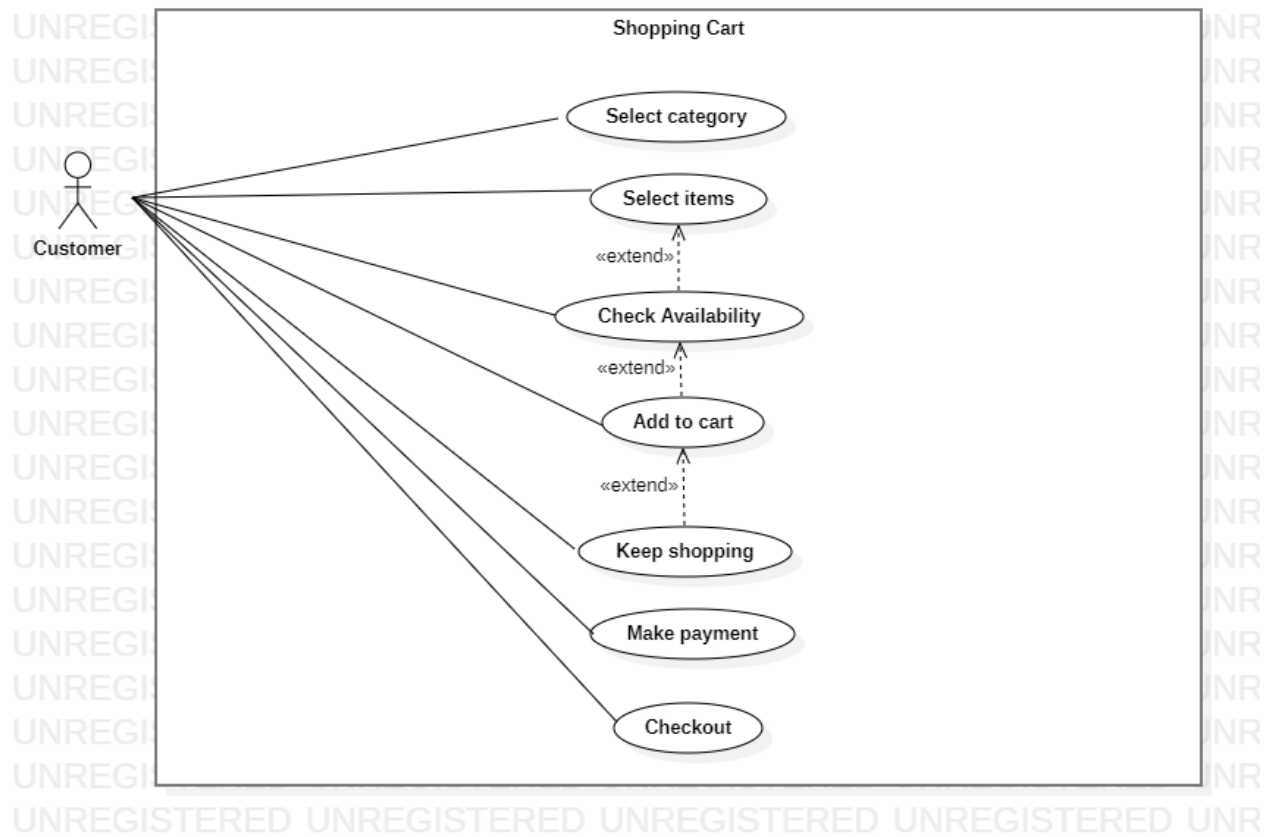


Figure 25-Use case of Cart

Table 8-Narrative Table for Cart

Use-Case Name:	Cart	Use-Case Type - Business Requirements
Use-Case ID:	FO-08	
Priority:	High Priority	
Source:	Online Form	
Primary Business Actor:	Admin	
Other Participating Actors:	Customer	
Other Interested Stakeholders:		
Description:	1. Select category 2. Select item 3. Add to cart 4. Keep shopping 5. Checkout	

Appendix B

This appendix mainly related to database design (for more information please refer Chapter 03).

Admin Table

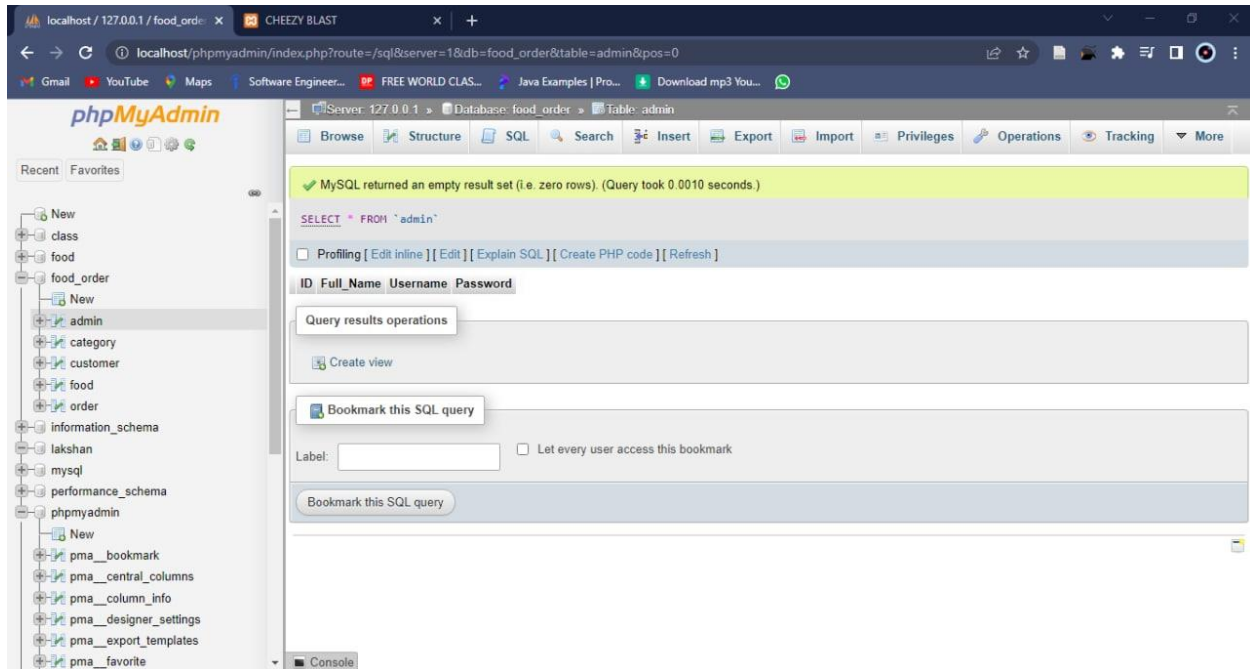


Figure 26-Database of Admin Table

Category Table

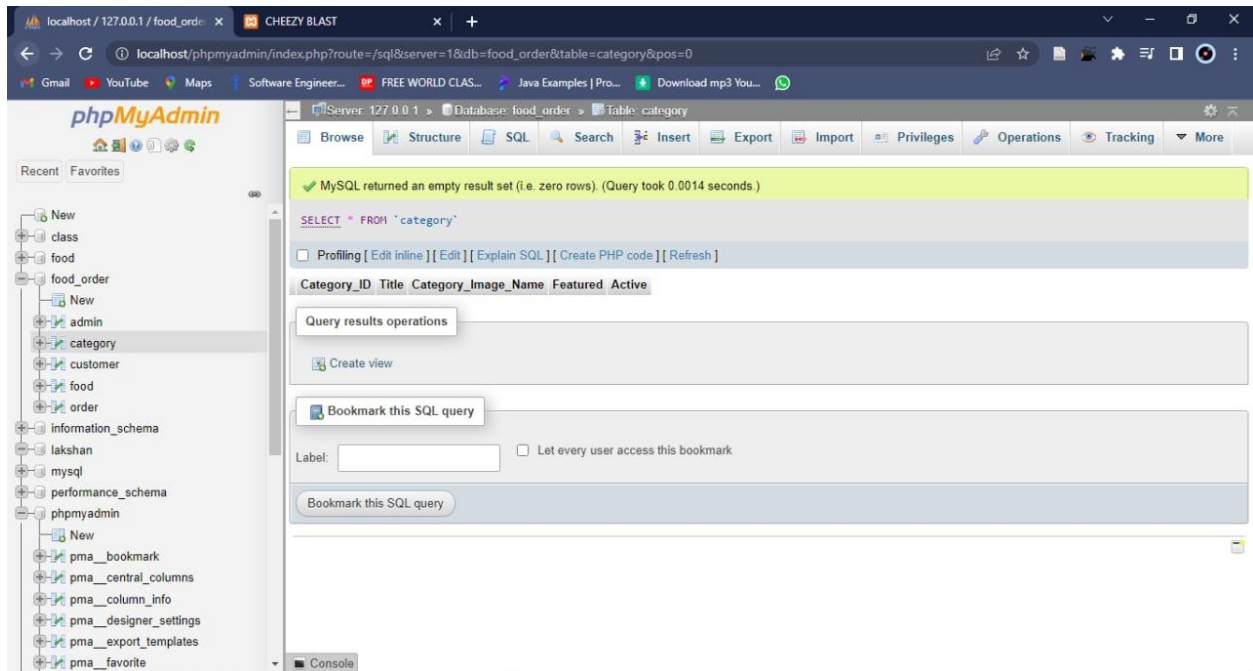


Figure 27-Database of Category Table

Customer Table

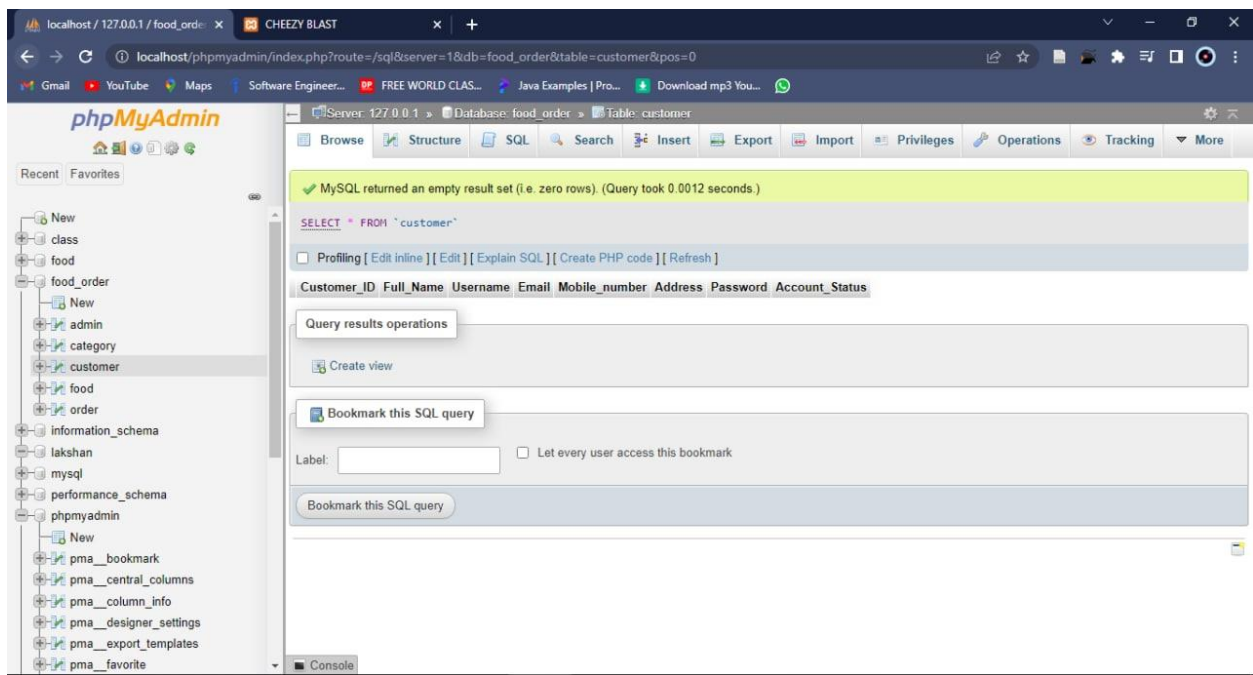


Figure 28-Database of Customer Table

Food Table

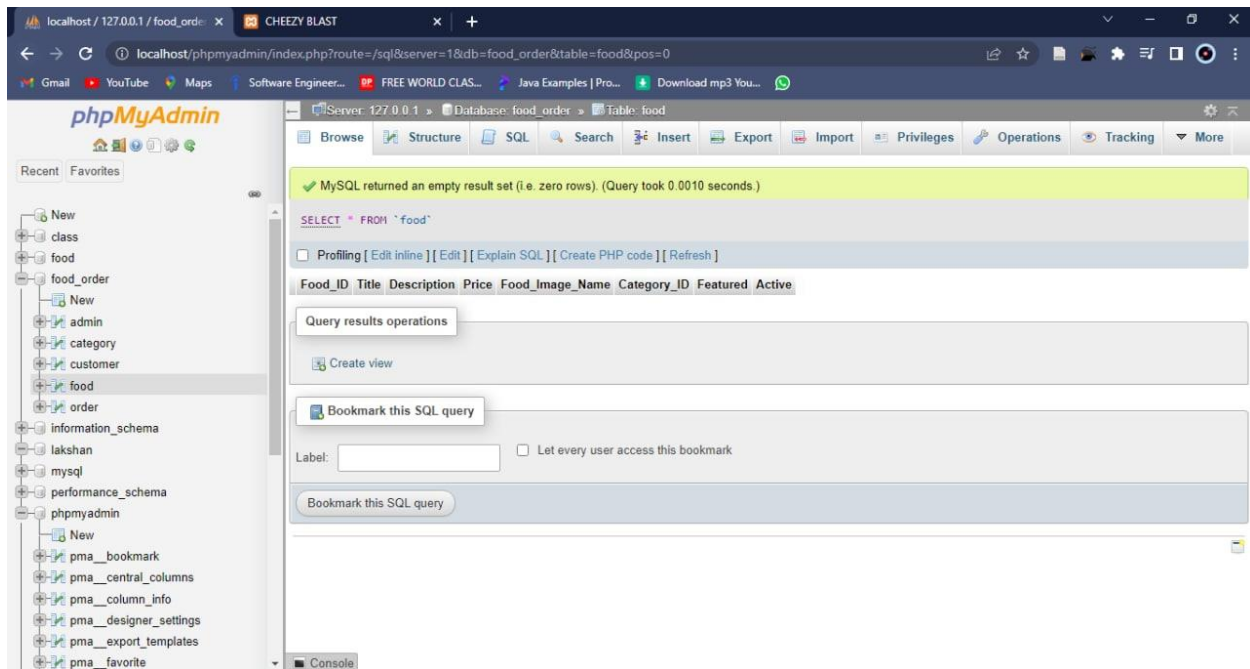


Figure 29-Database of Food Table

Order Table

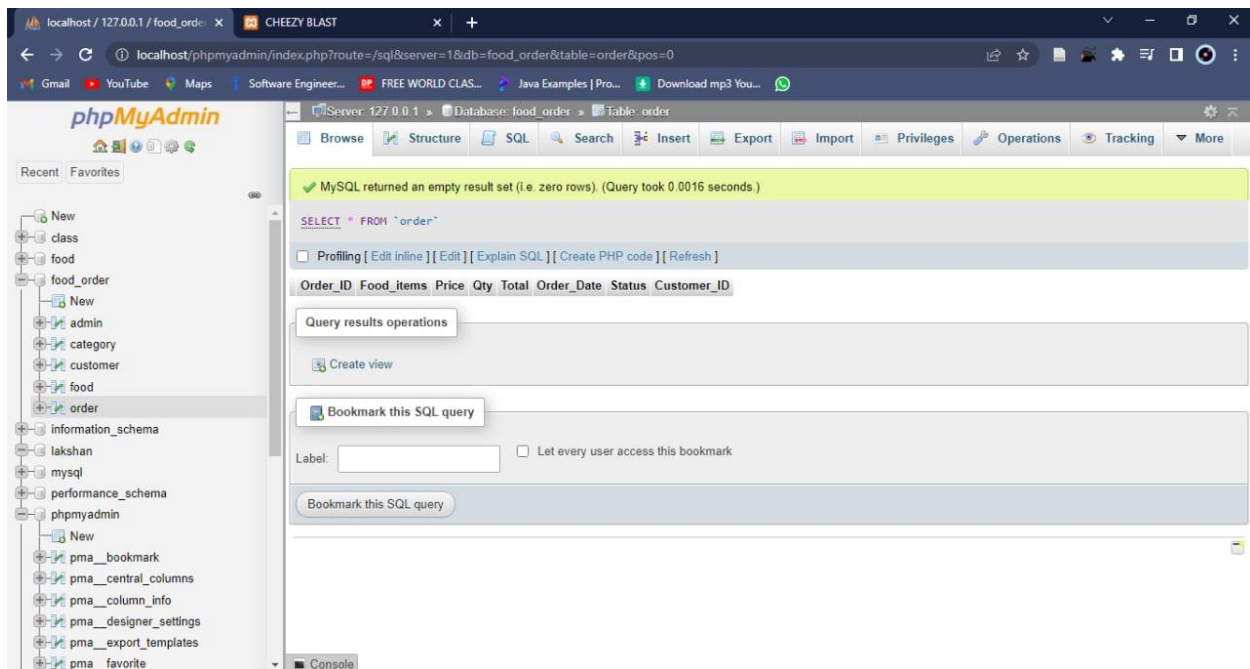
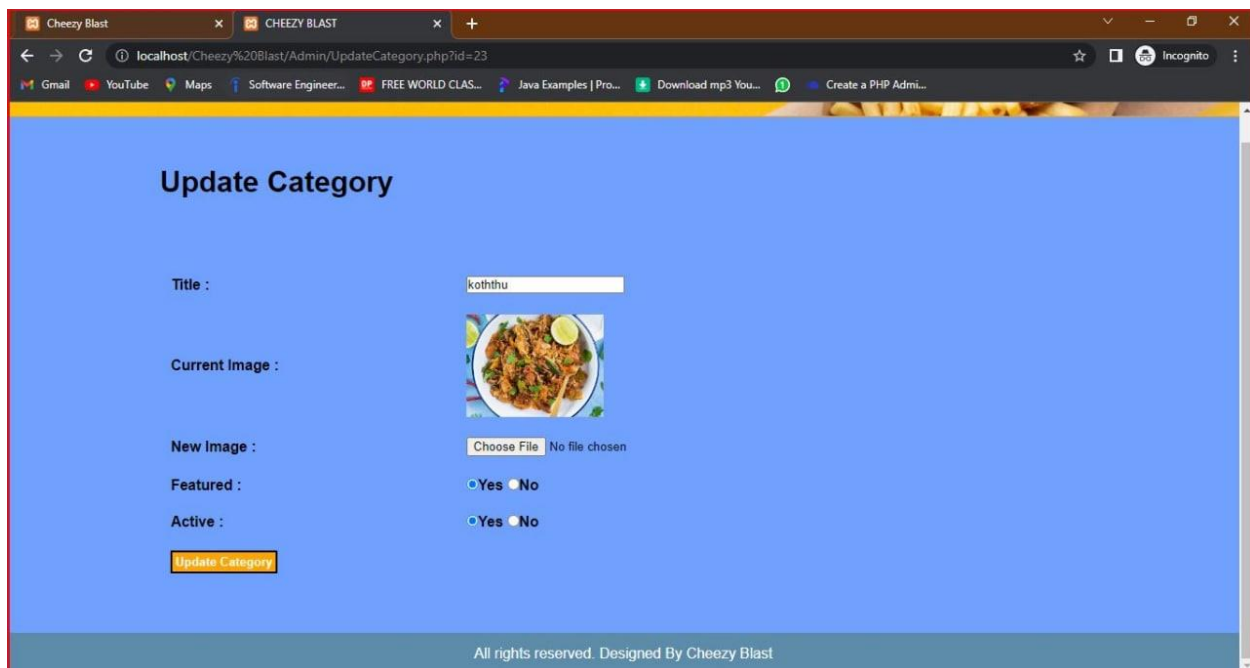


Figure 30-Database of Order Table

This appendix mainly related chapter 5– Implementation under User Interfaces

Backend user interface

Update Category



The screenshot shows a web browser window with the title 'Cheezy Blast' and the URL 'localhost/Cheezy%20Blast/Admin/UpdateCategory.php?id=23'. The page has a blue background and a yellow header. The main content area is titled 'Update Category' and contains the following form elements:

- Title :** A text input field containing the value 'koththu'.
- Current Image :** A small image of a dish, likely koththu, with a lemon wedge on top.
- New Image :** A file upload area with a 'Choose File' button and the text 'No file chosen'.
- Featured :** Radio buttons for 'Yes' (selected) and 'No'.
- Active :** Radio buttons for 'Yes' (selected) and 'No'.
- Update Category** : A yellow button with black text.

At the bottom of the page, there is a footer that reads 'All rights reserved. Designed By Cheezy Blast'.

Figure 31-Interface of Update Category

Category

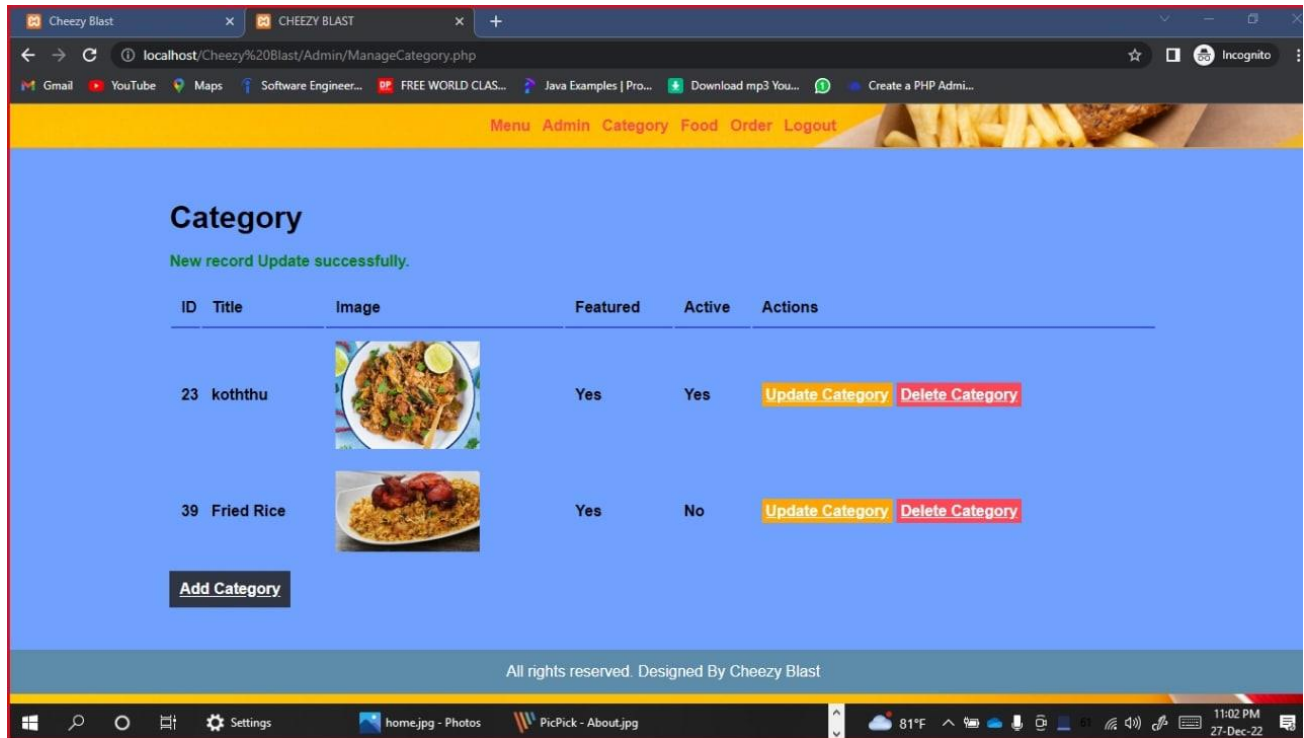


Figure 32-Interface of Category

Admin

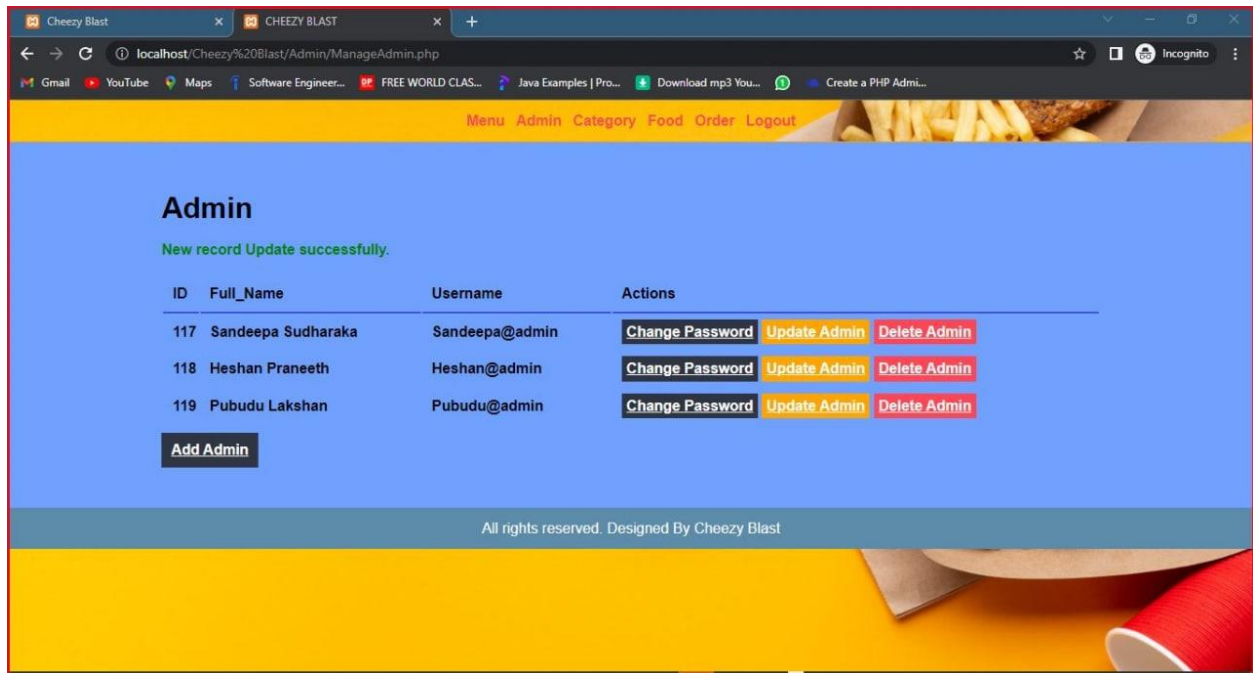


Figure 33-Interface of Admin

This appendix mainly related chapter 6 – Implementation under Code Segments

Backend Code Segment

Update Category

```
<?php include('C:\PHP\htdocs\Cheezy Blast\Partials\Menu.php');?>

<link rel="stylesheet"href="../../css/admin.css">

<?php $SITEURL = 'http://localhost/Cheezy%20Blast/'; ?>

<!-- Main section -->

<div class="Main">

    <div class="wrapper">

        <h1>Update Category</h1>

        <?php

            $id=$_GET['id'];

            $sql="SELECT * FROM category WHERE Category_ID=$id;";

            if(mysqli_query($conn,$sql)){

                $row=mysqli_fetch_assoc(mysqli_query($conn,$sql));

                $id=$row['Category_ID'];

                $title=$row['Title'];

                $current_image=$row['Category_Image_Name'];

                $featured=$row['Featured'];

                $active=$row['Active'];
```



```

    }

    else

    {

header("location:http://localhost/Cheezy%20Blast/Admin/ManageAdmin.php");

    }

?>

<form action="" method="Post" enctype="multipart/form-data">

<table class="tbl-full">

    <tr>

        <td>Title :</td>

        <td><input type="text" name="Title "
placeholder="Category Title" value="<?php echo $title;?>" ></td>

    </tr>

    </br>

<tr>

        <td>Current Image :</td>

        <td>

<?php

    if($current_Image != "")

    {

        ?>

```

```

        <?php
        }
        else
        {
            echo "<b><p style='color:red'>Image Not Added.</p></b>";
        }
    ?>
</td>

</tr>

</br>

<tr>

        <td>New Image :</td>

        <td><input type="File" name="image" ></td>

</tr>

</br>

<tr>

        <td>Featured :</td>

        <td><input
        <?php
        if($featured=="Yes"){echo"checked";} ?> type="radio" name="Featured" value="Yes">Yes

        <input
        <?php
        if($featured=="No"){echo"checked";} ?> type="radio "
        name="Featured" value="No">No</td>

</tr>

</br>

```

```

        <tr>

            <td>Active :</td>

            <td><input
                if($featured=="Yes"){echo "checked";} ?> type="radio" name="Active" value="Yes">Yes

                <input <?php if($featured=="No"){echo "checked";} ?> type="radio "
name="Active" value="No">No</td>

        </tr>

        <tr>

            <td colspan="2">

                <input type="submit" name="submit"
value="Update Category" class="btn-2">

                <input type="hidden" name="current_image" value="<?php echo
$current_Image;?>">

                <input type="hidden" name="id" value="<?php echo $id;?>">

            </td>

        </tr>

    </table>

</form>

<?php
if(isset($_POST['submit']))
{
    $id=$_POST['id'];
    $title=$_POST['Title'];

```

```

$current_Image=$_POST['current_image'];

$featured=$_POST['Featured'];

    $active=$_POST['Active'];

//get image

if(isset($_FILES['image']['name']))

{

    $image_name=$_FILES['image']['name'];

    if($image_name!="")

    {

        // new image

        //rename

        $ext = end(explode('.', $image_name));

        $image_name="Food_Category_".rand(000,999).'.'.$ext;

        $source_path=$_FILES['image']['tmp_name'];

        $destination_path="../images/Category/".$image_name;

        $upload = move_uploaded_file($source_path,$destination_path);

        if($upload==false)

        {

            $_SESSION['upload'] ="<b><p style='color:green'>Failed to upload image.</p></b>";

```

```

        header('location:'.$SITEURL.'Admin/ManageCategory.php');

        die();

    }

//remove

    if($current_Image !='')

    {

        $remove_path='../images/Category/'.$current_Image;

        $remove=unlink($remove_path);

        if($remove==false)

        {

            $_SESSION['Faiildremove'] ="<b><p style='color:green'>Failed to upload
image.</p></b>";

            header('location:'.$SITEURL.'Admin/ManageCategory.php');

            die();

        }

    }

}

else

{

    $image_name=$current_Image;

}

```

```

}

else

{

$image_name=$current_Image;

}


$sql3="UPDATE category SET

Title='$title',

Category_Image_Name='$image_name',

Featured='$featured',

Active='$active'

WHERE Category_ID='$id'

";


if ($conn->query($sql3) === TRUE) {


$_SESSION['update']      ="<b><p      style='color:green'>New      record      Update
successfully.</p></b>";

header('location:'.$SITEURL.'Admin/ManageCategory.php');

}

else {

```

```

$_SESSION['update'] ="<b><p style='color:red'>New record Update Faile.</p></b>";

header('location:'.$SITEURL.'Admin/ManageCategory.php');

}

}

?>

</div>

</div>

<?php include('C:\PHP\htdocs\Cheezy Blast\Partials\Footer.php');?>

```

Category

```

<?php

include('C:\PHP\htdocs\Cheezy Blast\Partials\Menu.php');

$SITEURL = 'http://localhost/Cheezy%20Blast/';

?>

```

```

<!-- Main section -->

```

```

<div class="Main">

```

```

    <div class="wrapper">

```

```

        <h1>Category</h1></br>

```

```
<?php

if(isset($_SESSION['add']))

{

    echo $_SESSION['add'];

    unset($_SESSION['add']);

}

if(isset($_SESSION['upload']))

{

    echo $_SESSION['upload'];

    unset($_SESSION['upload']);

}

if(isset($_SESSION['delete']))

{

    echo $_SESSION['delete'];

    unset($_SESSION['delete']);

}

if(isset($_SESSION['remove']))

{

    echo $_SESSION['remove'];

    unset($_SESSION['remove']);

}
```



```

}

if(isset($_SESSION['NoCategoryFound']))
{
    echo $_SESSION['NoCategoryFound'];
    unset($_SESSION['NoCategoryFound']);

}

if(isset($_SESSION['update']))
{
    echo $_SESSION['update'];
    unset($_SESSION['update']);

}

if(isset($_SESSION['Faidremove']))
{
    echo $_SESSION['Faidremove'];
    unset($_SESSION['Faidremove']);

}

?>

<br>

```

```

<table class="tbl-full">

    <tr>

        <th>ID</th>

        <th>Title</th>

        <th>Image</th>

        <th>Featured</th>

        <th>Active</th>

        <th>Actions</th>

    </tr>

    <tr>

        <?php

            $sql = "SELECT * FROM category";

            $res=mysqli_query($conn,$sql);

            $count=mysqli_num_rows($res);

            // $count=mysqli_num_rows($conn->query($sql));

            // if($count>0)

            if($count>0)

            {

                while($row=mysqli_fetch_assoc($res))

                {

                    $id=$row['Category_ID'];

                    $title=$row['Title'];

```

```

$image_name=$row['Category_Image_Name'];

                                $featured=$row['Featured'];

                                $active=$row['Active'];

                                ?>

<tr>

<td><?php echo $id;?></td>

<td><?php echo $title;?></td>

<td>

                                <?php

                                if($image_name!="")

                                {

                                    ?>

                                <?php

                                }

                                else

                                {

```

```

                                echo"<b><p
style='color:green'>Image  Not Added.</p></b>";

                                }

                                ?>

                                </td>

                                <td><?php echo $featured ?></td>

                                <td><?php echo $active;?></td>

                                <td>

                                <!-- <a href="" class="btn-2">Update
Category</a> -->

                                <a href="<?php echo
$SITEURL?>Admin/UpdateCategory.php?id=<?php echo $row['Category_ID']; ?> " class="btn-
2">Update Category</a>

                                <a href="<?php echo
$SITEURL?>Admin/DeleteCategory.php?id=<?php echo $row['Category_ID'];
?>&image_Name=<?php echo $row['Category_Image_Name']; ?>"class="btn-dlt">Delete
Category</a>

                                </td>

                                </tr>

                                <?php

                                }

```

```

    }

    else

    {

        ?>

<tr>

        <td colspan="6"><b><p style='color:green'>New
record created successfully.</p></b></td>

        </tr>

<?php

    }

    ?>

</table>

<br>

<a href="AddCategory.php" class="btn-1">Add Category</a>

```

</div>

</div>

<?php include('C:\PHP\htdocs\Cheezy Blast\Partials\Footer.php');?>

Admin

<?php include('C:\PHP\htdocs\Cheezy Blast\Partials\Menu.php');?>

<div class="Main">

<div class="wrapper">

<h1>Admin</h1></br>

<?php

if(isset(\$_SESSION['add']))

{

echo \$_SESSION['add'];

unset(\$_SESSION['add']);

}

if(isset(\$_SESSION['delete']))

```
{  
  
    echo $_SESSION['delete'];  
  
    unset($_SESSION['delete']);  
  
}
```

```
if(isset($_SESSION['update']))  
{  
  
    echo $_SESSION['update'];  
  
    unset($_SESSION['update']);  
  
}
```

```
if(isset($_SESSION['UserNotFound']))  
{  
  
    echo $_SESSION['UserNotFound'];  
  
    unset($_SESSION['UserNotFound']);  
  
}
```

```
if(isset($_SESSION['PasswordNotMatch']))  
{  
  
    echo $_SESSION['PasswordNotMatch'];  
  
}
```

```

        unset($_SESSION['PasswordNotMatch']);

    }

    if(isset($_SESSION['ChangePwd']))
    {

        echo $_SESSION['ChangePwd'];

        unset($_SESSION['ChangePwd']);

    }

?>

</br>

<table class="tbl-full">

    <tr>

        <th>ID</th>

        <th>Full_Name</th>

        <th>Username</th>

        <th>Actions</th>

    </tr>

<?php

```



```

admin";

$sql = "SELECT ID, Full_Name, Username FROM

$result = $conn->query($sql);

if ($result->num_rows > 0) {

    // output data of each row

    while($row = $result->fetch_assoc()) {

        //$_GET['id'] = $row['ID'];

    }

}

<tr>

<td><?php echo $row['ID'] ?></td>

<td><?php echo $row['Full_Name'] ?></td>

<td><?php echo $row['Username'] ?></td>

<td>

<a href="<?php echo
$SITEURL?>Admin/UpdatePassword.php?id=<?php echo $row['ID']; ?>"class="btn-1">Change
Password</a>

<a href="<?php echo
$SITEURL?>Admin/UpdateAdmin.php?id=<?php echo $row['ID']; ?>" class="btn-2">Update
Admin</a>

<a href="<?php echo
$SITEURL?>Admin/Delete.php?id=<?php echo $row['ID']; ?>" class="btn-dlt">Delete
Admin</a>

```

```

Admin</a> --

<!-- <a href="<?php  ?>" class="btn-dlt">Delete

</td>

</tr>

<?php

}

} else {

    echo "0 results";

}

//$conn->close();

?>

</table>

</br>

<a href="Add-Admin.php" class="btn-l">Add Admin</a>

</div>

</div>

<?php include('C:\PHP\htdocs\Cheezy Blast\Partials\Footer.php')?>

```

Appendix E

This appendix mainly related chapter 6– Testing and Evaluation

| Project Details | Project Name | Cheezy Blast | | | | | | | | | |
|-----------------|---|-------------------------------------|---|------------------------------------|---|---------------|--------|----------|----------|------------------|----------|
| | Module Name | Cagegery Selection | | | | | | | | | |
| | Cretd By | Heshan | | | | | | | | | |
| | Reviewd By | Sandeepa | | | | | | | | | |
| | Created Date: | 26-Dec-22 | Reviewed Date | 27-Dec-22 | | | | | | | |
| | | | | | | | | | | | |
| | | | | | | | | | | | |
| Test Case ID | Test Description/Test case name | Prerequisite | Test Steps | Input Data | Expected Result | Actual Result | Status | Severity | Priority | Executed by | Comments |
| TC_Category_001 | select category and check availability of items | Login to thsystem as registred user | 1. browse cheezyblast.lk
2. login by giving user name and password
3. select category
4. select sub category
5. check available items | catgary name and sub category name | Showing all available items relvant to selected category and sub category | As expected | pass | | | Sandeepa, Tester | |
| TC_Category_002 | select category and check sold out items | Login to thsystem as registred user | 1.browse cheezyblast.lk
2. login by giving user name and password
3. select category
4. select sub category
5. check sold out items | catgary name and sub category name | Showing all sold out items relvant to selected category and sub category | As expected | pass | | | Sandeepa, Tester | |

Figure 34-Test case of category

| Project Details | | | Project Name | Cheezy Blast | | | | | | | |
|-----------------|--|-------------------------|--|-----------------|---|---------------|--------|----------|----------|------------------------|----------|
| | | | Module Name | Inquery | | | | | | | |
| | | | Created by | Heshan | | | | | | | |
| | | | Reviewed by | Sandeepa | | | | | | | |
| | | | Created Date | 25/12/2022 | Review Date | 26/12/2022 | | | | | |
| Test Case ID | Test Description/Test case name | Prerequisite | Test Steps | Input Data | Expected Result | Actual Result | Status | Severity | Priority | Executed by | Comments |
| TC_Inquery_001 | Enter all the fields correctly and send requests | Vist the cheezyblast.lk | 1. Browse the web site "cheezyblast.lk"
2. Sing in or Sign Up to website
3. Order the food
4. click to my order
5. Click inquer
6. Given valid details for text area
7. Click on send button | Message | Alert message as successfully send the data | As expected | Pass | | | Sandeepa - Test leader | |
| TC_Inquery_002 | Keep text area empty in given form | Vist the cheezyblast.lk | 1. Browse the web site "cheezyblast.lk"
2. Sing in or Sign Up to website
3. Order the food
4. click to my order
5. Click inquer
6. Given text area empty
7. Click on send button | Message (Empty) | Alert message as fill all the fields | As expected | Pass | | | Sandeepa - Test leader | |

Figure 35-Test case of inquiry

Reference

- [1 "Domino's Pizza," [Online]. Available: <https://m.dominos.lk.com/>. [Accessed 20 12 2022].
]
- [2 "Sponge," [Online]. Available: <https://www.sponge.lk/>. [Accessed 20 12 2022].
]
- [3 "KFC," [Online]. Available: <https://www.kfc.lk/>. [Accessed 20 12 2022].
]
- [4 "Pizza Hut," [Online]. Available: <https://www.pizzahut.lk/>. [Accessed 20 12 2022].
]
- [5 "Taco Bell," [Online]. Available: <https://tacobell.lk/>. [Accessed 20 12 2022].
]
- [6 "Fact Finding," [Online]. Available: <https://www.w3schools.in/dbms/fact-finding> . [Accessed 23 12
] 2022].
- [7 "Use Case Diagram," [Online]. Available: https://en.wikipedia.org/wiki/Use_case_diagram.
] [Accessed 21 12 2022].
- [8 "Class Diagram," [Online]. Available: https://en.wikipedia.org/wiki/Class_diagram. [Accessed 21 12
] 2022].
- [9 "Activity Digrams," [Online]. Available: [https://www.visual-paradigm.com/guide/uml-unified-
\] modeling-language/what-is-activity-diagram/](https://www.visual-paradigm.com/guide/uml-unified-modeling-language/what-is-activity-diagram/). [Accessed 21 12 2022].
- [1 "Sequence Diagram," [Online]. Available: https://en.wikipedia.org/wiki/Sequence_diagram.
0] [Accessed 21 12 2022].
- [1 "EER Diagram," [Online]. Available: <https://www.edrawsoft.com/article/what-is-eer-diagram.html>.
1] [Accessed 21 12 2022].
- [1 "Star UML," [Online]. Available:
2] [https://en.wikipedia.org/wiki/StarUML#:~:text=StarUML%20is%20a%20software%20engineering,o
n%20Windows%2C%20Linux%20and%20MacOS..](https://en.wikipedia.org/wiki/StarUML#:~:text=StarUML%20is%20a%20software%20engineering,on%20Windows%2C%20Linux%20and%20MacOS..) [Accessed 22 12 2022].
- [1 "phpMyAdmin," [Online]. Available: <https://www.phpmyadmin.net/>. [Accessed 22 12 2022].
3]
- [1 "Windows 10," [Online]. Available: https://en.wikipedia.org/wiki/Windows_10. [Accessed 22 12
4] 2022].

- [1 "Visual Studio Code," [Online]. Available: https://en.wikipedia.org/wiki/Visual_Studio_Code.
5] [Accessed 22 12 2022].
- [1 "XAMPP," [Online]. Available: <https://www.apachefriends.org/>. [Accessed 22 12 2022].
6]
- [1 "RAM," [Online]. Available: [https://www.techtarget.com/searchstorage/definition/RAM-random-](https://www.techtarget.com/searchstorage/definition/RAM-random-access-memory)
7] [access-memory](https://www.techtarget.com/searchstorage/definition/RAM-random-access-memory). [Accessed 22 12 2022].
- [1 "Processor," [Online]. Available: [https://www.intel.com/content/www/us/en/business/small-](https://www.intel.com/content/www/us/en/business/small-business/resources/ram-vs-processor.html#:~:text=of%20the%20Processor-,The%20processor%2C%20also%20known%20as%20the%20CPU%2C%20provides%20the%20instructions,computer%20think%20and%20work%20faster..)
8] [business/resources/ram-vs-processor.html#:~:text=of%20the%20Processor-,The%20processor%2C%20also%20known%20as%20the%20CPU%2C%20provides%20the%20instructions,computer%20think%20and%20work%20faster..](https://www.intel.com/content/www/us/en/business/small-business/resources/ram-vs-processor.html#:~:text=of%20the%20Processor-,The%20processor%2C%20also%20known%20as%20the%20CPU%2C%20provides%20the%20instructions,computer%20think%20and%20work%20faster..) [Accessed 22 12 2022].
- [1 "CSS," [Online]. Available:
9] [https://en.wikipedia.org/wiki/CSS#:~:text=Cascading%20Style%20Sheets%20\(CSS\)%20is,Web%2C%20alongside%20HTML%20and%20JavaScript..](https://en.wikipedia.org/wiki/CSS#:~:text=Cascading%20Style%20Sheets%20(CSS)%20is,Web%2C%20alongside%20HTML%20and%20JavaScript..) [Accessed 22 12 2022].
- [2 "HTML," [Online]. Available: [https://www.hostinger.com/tutorials/what-is-](https://www.hostinger.com/tutorials/what-is-html#:~:text=HTML%20stands%20for%20HyperText%20Markup,such%20as%20tags%20and%20attributes..)
0] [html#:~:text=HTML%20stands%20for%20HyperText%20Markup,such%20as%20tags%20and%20attributes..](https://www.hostinger.com/tutorials/what-is-html#:~:text=HTML%20stands%20for%20HyperText%20Markup,such%20as%20tags%20and%20attributes..) [Accessed 22 12 2022].
- [2 "JavaScript," [Online]. Available: [https://developer.mozilla.org/en-](https://developer.mozilla.org/en-US/docs/Web/JavaScript#:~:text=JavaScript%20(JS)%20is%20a%20lightweight,Apache%20CouchDB%20and%20Adobe%20Acrobat..)
1] [US/docs/Web/JavaScript#:~:text=JavaScript%20\(JS\)%20is%20a%20lightweight,Apache%20CouchDB%20and%20Adobe%20Acrobat..](https://developer.mozilla.org/en-US/docs/Web/JavaScript#:~:text=JavaScript%20(JS)%20is%20a%20lightweight,Apache%20CouchDB%20and%20Adobe%20Acrobat..) [Accessed 22 12 2022].
- [2 "MySQL," [Online]. Available: [https://dev.mysql.com/doc/refman/8.0/en/what-is-](https://dev.mysql.com/doc/refman/8.0/en/what-is-mysql.html#:~:text=MySQL%20is%20a%20database%20management%20system.&text=It%20may%20be%20anything%20from,system%20such%20as%20MySQL%20Server..)
2] [mysql.html#:~:text=MySQL%20is%20a%20database%20management%20system.&text=It%20may%20be%20anything%20from,system%20such%20as%20MySQL%20Server..](https://dev.mysql.com/doc/refman/8.0/en/what-is-mysql.html#:~:text=MySQL%20is%20a%20database%20management%20system.&text=It%20may%20be%20anything%20from,system%20such%20as%20MySQL%20Server..) [Accessed 22 12 2022].
- [2 "Unit Testing," [Online]. Available: [https://smartbear.com/learn/automated-testing/what-is-unit-](https://smartbear.com/learn/automated-testing/what-is-unit-testing/#:~:text=A%20unit%20test%20is%20a,subroutine%2C%20a%20method%20or%20property..)
3] [testing/#:~:text=A%20unit%20test%20is%20a,subroutine%2C%20a%20method%20or%20property..](https://smartbear.com/learn/automated-testing/what-is-unit-testing/#:~:text=A%20unit%20test%20is%20a,subroutine%2C%20a%20method%20or%20property..) [Accessed 23 12 2022].
- [2 "Black BoxTesting," [Online]. Available: <https://www.guru99.com/black-box-testing.html>. [Accessed
4] 23 12 2022].
- [2 "Security Testing," [Online]. Available: [https://www.geeksforgeeks.org/software-testing-security-](https://www.geeksforgeeks.org/software-testing-security-testing/)
5] [testing/](https://www.geeksforgeeks.org/software-testing-security-testing/). [Accessed 23 12 2022].
- [2 "Integration Testing," [Online]. Available:
6] [https://www.techtarget.com/searchsoftwarequality/definition/integration-](https://www.techtarget.com/searchsoftwarequality/definition/integration-testing#:~:text=Integration%20testing%20%D%2D%20also%20known,be%20coded%20by%20different%20programmers..)
[testing#:~:text=Integration%20testing%20%D%2D%20also%20known,be%20coded%20by%20different%20programmers..](https://www.techtarget.com/searchsoftwarequality/definition/integration-testing#:~:text=Integration%20testing%20%D%2D%20also%20known,be%20coded%20by%20different%20programmers..) [Accessed 23 12 2022].
- [2 "System Testing," [Online]. Available: <https://www.guru99.com/system-testing.html>. [Accessed 23
7] 12 2022].

Responsibility matrix

| Name | Index | Task |
|---------------|----------|---|
| HMP Laksahan | E2041348 | Chapter 1,2,7 and Diagram |
| JRS Sudharaka | E2041378 | Chapter 5, Database Design and Test Cases |
| THP Amarasena | E2041305 | Chapter 3,4,6 and Diagrams, Test cases |