Tribhuvan University

Faculties of Humanities and Social Sciences

Online Sweet shop

A E-commerce project report

I*n partial fulfillment of the requirements for the Bachelor Degree in Computer Application*

Submitted to Department of Computer Application

Divya Gyan College

Putalisadak, Kathmandu

Submitted by
Sujan Baral()
Susanta Timsina()

Date of Submission: 2080

Under the Supervision of

Mr. Srijan Shah

Tribhuvan University Faculty of Humanities and Social Sciences

SUPERVISOR'S RECOMMENDATION

I hereby recommend that this project prepared by **Sujan Baral and Susanta Timsina** entitled "**ONLINE SWEET SHOP**" in partial fulfillment of the requirements for the degree of Bachelor of Computer Application is recommended for the final evaluation.

SIGNATURE
Srijan Shah
SUPERVISOR
Project Coordinator
Department of IT
Divya Gyan College
Putalisadak, Kathmandu

LETTER OF APPROVAL

This is to certify that this project prepared by **SUJAN BARAL & SUSANTA TIMSINA** entitled "**ONLINE SWEET SHOP**" in partial fulfillment of the requirements for the degree of Bachelor in Computer Application has been evaluated. In our opinion it is satisfactory in the scope and quality as a project for the required degree.

SIGNATURE of Supervisor	SIGNATURE of HOD/ Coordinator
Mr.Srijan Shah Supervisor Divya Gyan college Putalisadak,Kathmandu	**Mrs. Annu Khanna** Co-ordinator Divya Gyan college Putalisadak,Kathmandu
SIGNATURE of External Examiner	SIGNATURE of External Examiner

ABSTRACT

Online sweet shop is a web-based application that can be used to ease the customer life. In online shopping, customers aren't required to stand in queues to pay the price of products that they have purchased. **Online sweet shops** provide the facility to shop from their office or home, so it saves traveling time. It is the major benefit of online shopping. Customers can purchase from their homes, workplaces as per their comfort. It is easy to cancel the transactions in online shopping. The user can then view the complete specification of each product. They can also view the product reviews and also write their own reviews. The application also provides a drag and drop feature so that a user can add a product to the shopping cart by dragging the item into the shopping cart. The main emphasis lies in providing a user friendly search engine for effectively showing the desired results and its drag and drop behavior.

Keyword-_Online, Shopping, Sweet , User_

ACKNOWLEDGEMENT

TABLE OF CONTENT

List of Tables

List of Figures

List of Abbreviations

COD – Cash On Delivery

ER – Entity Relationship

GC- Google Chrome

OSS – Online Sweet Shop

UC – Use Case

UI – User Interface

Chapter 1: Introduction

1.1 Introduction

It is an Online sweet shop where customers can order their sweets from this e-commerce shopping site. It links customers and shopkeepers directly as a medium to connect them. Some sweets Shop cannot afford to pay extra for the entire ambience. It provides them with the platform to expand their sweets business.

Online sweet shop is a web-based application that can be used to ease the customer life. In online shopping, customers aren't required to stand in queues to pay the price of products that they have purchased. Online sweet shops provide the facility to shop from their office or home, so it saves traveling time. It is the major benefit of online shopping. Customers can purchase from their homes, workplaces as per their comfort. It is easy to cancel the transactions in online shopping. The user can then view the complete specification of each product. They can also view the product reviews and also write their own reviews. The application also provides a drag and drop feature so that a user can add a product to the shopping cart by dragging the item into the shopping cart. The main emphasis lies in providing a user friendly search engine for effectively showing the desired results and its drag and drop behavior.

Keywords: *Sweetopia, online, product*

.

1.2 Problem Statement

The challenges encountered by the existing system serve as a major drawback to the realization of efficiency and customer satisfaction. The experience of ordering in this shopping shop is not pleasant for the customers. This project aims to develop online shopping for customers with the goal that it is very easy to shop your loved things from an extensive number of online shopping sites available on the web. With the help of this you can carry out online shopping from your home. There is no compelling reason to go to the crowded shops or shopping centers during festival seasons. You simply require a PC or a laptop and one important payment sending option to shop online. To get to this online shopping system all the customers will need to have an email and password to login and proceed your shopping . The login credentials for an online shopping system are under high security and nobody will have the capacity to crack it easily. Upon successful login the customers can purchase a wide range of things. can be dispatched using an online shopping system. Not just these, you can also purchase from outside nations by a few clicks on your mouse. And of course you will get your requested ordered items at your doorstep. will have to make long queues before placing their orders especially during peak hours and then the ordering staff will record customer orders. Having placed their order, the customer must then wait near the counter until their order is ready for collection. The other problem with the current system is that the customers are not able to see the packed sweets before they place their order and also, they only have to pay for an order online.

1.3 Objectives

The main objective of this project is to develop an application which gives provision to the shop owners to flourish their business by uploading sweets and details and will invariably lead to higher customer retention and acquisition rates.

The objectives of Online sweet shop are as follows:
• To develop a web-based sweets ordering system.

1.4 Scope and Limitation

The existing system refers to the system that is currently being followed by the sweets deliveries. Presently all the functionalities are done manually. If a customer wants to order an item, he should visit the shop and order the goods, pay for the items and get the goods slips. Then he should go to the sweets counter to get the sweets and take the bill.. This makes the person quite difficult and tedious since they must stay in the queue. The main disadvantage is time consuming and it makes it difficult for the manager to know the customer's past history so to avoid these limitations and make the system working more accurately it needs to be computerized.

1.4.2 Scope of System

To overcome the restrictions of the above system, an Online sweet shop system is proposed which has the following advantages, people can successfully order the sweet using the system. There will be a lesser requirement of staff at the back counter. The system will help in reduction of labor cost involved and also reduces the space required to set up restaurants in the restricted area. As it is an automated system it is less probable to make any mistakes. The customers can avoid the long queues at the counter, with a reasonable speed of execution and maximum throughput.

1.4.1 Limitations of Existing System

The limitations of the existing systems are listed as follows:
• There is a lack of understanding between the customers and the person from the admin side.
• It is time consuming for the customers to wait for their order.

- The process of taking orders from the customer is very tedious and makes it impossible to deliver goods on time.
- The record keeping system is poor which might cause loss of vital records of the past and cannot protect files from unauthorized access.
- It is difficult for Managers to supervise all the sides of restaurants like kitchen, floor and customer simultaneously.

1.5 Report Organization

This is the report organization for the Online sweet shop system which also includes charts/diagrams to illustrate the system architecture and design. Furthermore, it contains information regarding the tools and technologies used to build the system.

Table 1.1- Outline of the Report

Introduction	Introduction Problem Statement Objective Scope and Limitations Report Organization
Requirement and Feasibility Analysis	Background Study Literature Review
Methodology	Requirement Analysis Feasibility Analysis System Design

Implementation and Testing	Tools and Technology Test Cases
Conclusion and Recommendations	Lessons Learnt Conclusion Future Recommendations

Chapter 2: Background Study and Literature Review

2.1 Background Study :

Nowadays, every technical field is trying to put human life at ease. The past couple years there has been enormous growth of internet restaurants. It's just a single window for ordering from a varied series of restaurants. Basically, "Online sweet shop" can be defined as a simple and convenient way for the customers to purchase sweets online, without having to go to the Shop. This system is very useful to those who are very busy at work or at home and do not have the time to go outside. Customers don't really need to have technical knowledge to operate it. Because it is designed in a very modest way. It provides a complete dashboard with information.

In short, it's easy, convenient, completely transparent sweets shopping software and also customer-friendly online ordering system.

2.2 Literature Review :

In order to develop the proposed system first requirement analysis is done considering past researchers and existing systems. Required data is gathered by study of manuals. Then the system is implemented. A theme for the app will be selected ideas of the group members. Then each member is requested to complete their allocated task. Finally, the system is integrated. Once the goal was set, the requirement collection process began to get information about the online food ordering system history and past works on this domain.

Following methods have been used for requirement collection:

Literary Analysis

Literature has been reviewed and critically analyzed to find what kind of works have been suggested in the past.Requirements gathering from the various references of the existing ecommerce site such as *'Amazon'* and *'Daraaz'*, Etc. are the reference site for gathering the visual information and the imaginary concepts for the project. How those systems implement the user-friendly environment. As a developer I've used both platforms for the 'Atomic Design' approach in which user requirements are fulfilled.

Observation

Number of shopping websites have been reviewed to find the current trend in developing web applications for shops. Along with this, a typical shop ordering system has been reviewed to get an insight of the online order processing system.

Study of manuals

Manuals and reports based on online shopping site systems were obtained and studied and a lot of information concerning the system to be produced was obtained.

Brainstorming

Brainstorming has been used for the creative and idea generation by the team members. The requirements for the system somewhat collected from the brainstorming method as well.

Chapter 3: System Analysis and Design

3.1 System Analysis

The system analysis of the system is done by conducting requirement analysis, feasibility analysis, data modeling and process modeling as follows:

3.1.1 Requirement Analysis

The requirement analysis of online sweet shops is done through finding the functional requirements and non-functional requirements for the system.

3.1.1.1 Functional Requirements

This subsection contains the functional requirements for the online sweet shop. Features from the proposal are refined into use case diagrams and to best capture the functional requirements of the system.
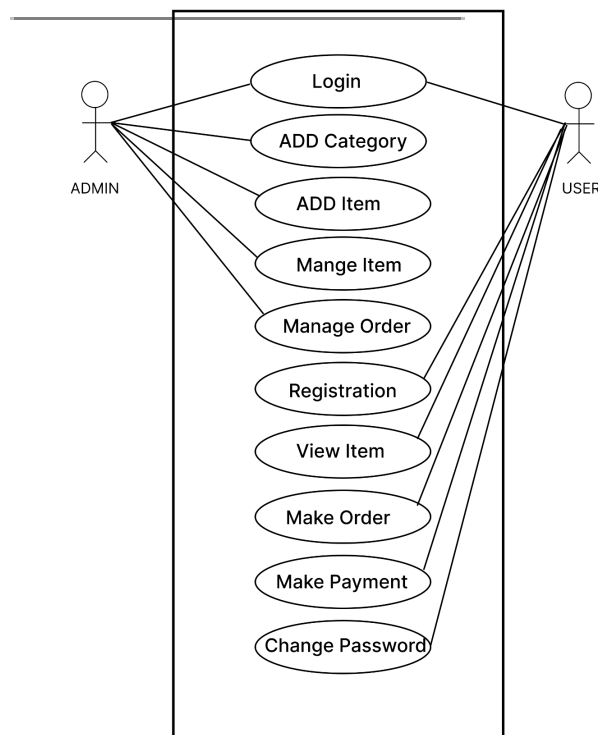
Figure 3.1: Use Case Diagram of Online Sweet Shop .

Table 3.1: UC-001 Login

Use Case ID	UC-001
Use Case Name	Login
Actors	Customer, Admin,
Brief Description	This use case describes how a user logs into the Online sweet shop. The actors starting this use case are Customers, Admin, and. The system asks users to input username and password. If the system approves the login, the users are allowed to perform required actions.
Triggers	The user wants to login in the system from one of his/her accounts.
Preconditions	The system is operational. The user has an account in the system. The user is trying to log in with their valid username and password. The user is not already logged in into the system.
Normal Flow	1. The use case begins when the user accesses the system. 2. The system prompts the user for their account credentials. 3. The user enters their username and password. 4. The system verifies the user credentials and grants the entry to the system. 5. The user gains access to the system's functionality. 6. Use case ends successfully.
Alternate Flows	3A1. User is not registered in the system. 3A2. If not registered, the system allows the user to sign up in the system. 4A1. Invalid Username or Password. 4A2. Users are already logged in the system. 4A3. User is requested to re-enter the username and password.

Key Scenarios	A. Success Scenarios
	
	    1. Valid username and password
	
	    2. Access to the system
	
	B. Failure Scenarios
	
	    1. Fail due to authentication failure
	
	    2. Fail due to invalid username and password
	
	    3. Fail due to inadequate critical resources
	
	    4. Fail due to non-responding system

Post conditions	A. Successful Completions
	
	    1. The user has logged into the system successfully.
	
	    2. The user accesses the system functionality.
	
	B. Failure Conditions
	
	    1. The user has not been logged into the system.

Special Requirement	1. The user shall be a registered user to log into the system. 2. The system shall keep the track of the users who accessed the system.

Table 3.2: UC-002 Place Order

Use Case ID	UC-002
Use Case Name	Place Order
Actors	Customer, Sweet ordering site
Brief Description	This Use Case describes the process by which orders are entered into the system by the customers. A customer accesses the site from the workplace or from home, views the menu of Shop, selects Sweet items, and places an order for the items to be delivered.
Triggers	The customer wants to place an order in the system from one of his/her accounts.
Preconditions	The system is operational. The customer has registered into the system.

	The customer is logged into the system.
Normal Flow	1. The Use Case starts when the customer places an order. 2. The customer selects the desired sweet items. 3. The customer presses the Checkout button. 4. The system displays the Checkout Screen if a customer is logged in. 5. The customer logs in if not logged in. 6. The customer will enter payment information. 7. The customer will select submit. 8. The system will verify the information and save the order as pending 9. The system forwards the order to the website employees. 10. The order is marked confirmed, an order id is returned to the customer. 11. The use case ends successfully.
Alternate Flows	3A1. If a customer is not registered in the system, prompt the customer to register. 5A1. Invalid payment information. 5A2. The customer wants to choose invalid mode of payment. 6A1. If a customer asks to order another food item, return to step 4. 8A1. The order is not registered in the system. 9A1. The customer doesn't get the order id.

Key Scenarios	A. Success Scenarios
	    1. Valid registered customer
	    2. Access to system 3. Select and order sweet items.
	    4. Make payment.
	B. Failure Scenarios
	    1. Fail due to authentication failure
	    2. Fail due to invalid username and password
	    3. Fail due to inadequate critical resources
	    4. Fail due to non-responding system
	    5. Fail due to insufficient balance.

Post conditions	A. Successful Completions
	1. The customer has logged into the system successfully.
	2. sweet order is shopped in the system with a status of "pending".
	3. The system has received the customer's payment details.
	4. The food is delivered successfully.
	B. Failure Conditions
	1. The customer cancels the order.
	2. The system terminates the order.
Special Requirement	1. The customer shall receive order id through email.
	2. The system shall keep the track of the customer's order.
	3. The system shall track the order status.

Table 3.3: UC-003 Make Payment

Use Case ID	UC-003
Use Case Name	Make Payment
Actors	Customer, Online sweet shop
Brief Description	Allows any registered customer to make payment after ordering sweets of their choice. If the customer is not registered then at first customer should register themselves with then can order easily. The customer should provide all the payment related information. Customers can make payments only after providing valid payment details for ordering purposes.
Triggers	The customer wants to make the payment for his/her order from one of his/her accounts.
Preconditions	The system is operational. The customer has registered and logged into the system. The customer has valid payment details.
Normal Flow	1.The use case begins when the customer enters into the system to make payments. 2.The system provides a login option to customers. 3.A customer enters his/her username and password. 4.The system verifies the customer's entered username and password against the registered information.

	5.The customer chooses the add to cart option.
	6.Customer selects the required sweet item.
	7.Customer presses the checkout button.
	8.Customer selects the mode of payment.
	9.Customer enters the payment details.
	10.Customer pays for the order.
	11.The use case ends successfully.
Alternate Flows	4A1. Invalid username or password. 4A2. Customer is requested to re-enter the username and password. 7A1. Customer selects the unavailable mode of payment. 8A1. Customer enters invalid payment details.

Key Scenarios	A. Success Scenarios
	    1. The order payment is made.
	    2. Customer views payment success information.
	B. Failure Scenarios
	    1. Fail due to authentication failure
	    2. Fail due to invalid username and password
	    3. Fail due to inadequate critical resources
	    4. Fail due to non-responding system
	    5. Fail due to insufficient balance.

Post conditions	A. Successful Completions
	    1. Enters into the system.
	    2. Customer selects the required items.
	    3. The customer makes the payment successfully.
	    4. The customer receives the payment confirmation.
	B. Failure Conditions
	    1. The payment detail entered by the customer is invalid.
	    2. The payment system might not be working at that moment.

Special Requirement	1.The system shall deduct the discounted amount from the total amount paid by the customer.

3.1.1.2 Non-Functional Requirements

The non-functional requirements of Online sweet shop are as follows:

Performance Requirement:

i.   The system shall be able to take orders from the customer.

ii. The user shall be able to login into the system.

iii. The user shall be able to update the profile.

Usability Requirement:

i. The user shall be able to use the system in an easy manner.

Availability Requirement:

i.   The system is available 100% for the user and is used 24hrs a day and 365 days a year.

ii. The system shall be operational 24hrs a day and 7 days a week.

Environmental Requirement:

i. The system shall require a localhost server, database server and a web browser to run successfully.

Compatibility Requirement:

i. The system shall be compatible across all platforms under the required environment.

Security Requirement:

i.   The user password shall be in encrypted format in the database.

ii. Every user shall have a unique Session while logging into the system.

3.1.2 Feasibility Analysis

The feasibility analysis of online sweet shop is done through measuring the following feasibilities which are explained as follows: [1]

3.1.2.1 Technical Feasibility

The system can be implemented in various technologies which are presently available as well as in all technologies which will be implemented in future.

Software Requirement

Operating System: Windows (7 or windows 10 or more)

Web Browser: Mozilla FF and Google Chrome

XAMPP

3.1.2.2 Operational Feasibility

This project is feasible to operate. The current mode of operation provides adequate throughput and response. So this project is completely operational and can be operated on any platform.

3.1.2.3 Economic Feasibility

There hasn't been any economic feasibility analysis but it can be done on the basis of Function Point Analysis or Kilo Line of Code method.

3.1.2.4 Schedule Feasibility

Here is the Gantt chart showing the probability of the project to be completed within its scheduled time limits, by a planned due date.

Project Timeline

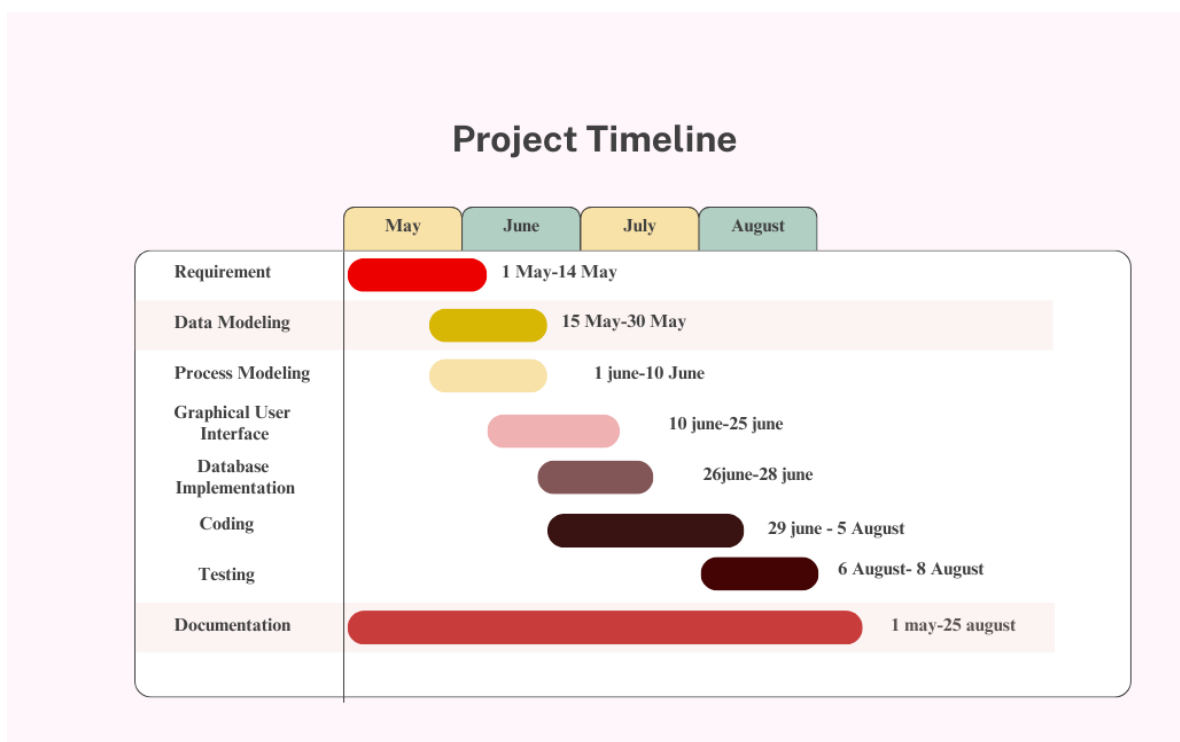|	May	June	July	August
Requirement	1 May-14 May			
Data Modeling	15 May-30 May			
Process Modeling	1 june-10 June			
Graphical User Interface	10 june-25 june			
Database Implementation	26june-28 june			
Coding	29 june - 5 August			
Testing	6 August- 8 August			
Documentation	1 may-25 august			

Figure 3.2: Gantt Chart of Online sweet shop [4]

3.1.3 Object Modeling (Class Diagram)

For data modeling, the class diagram of online sweet shop is shown below as:
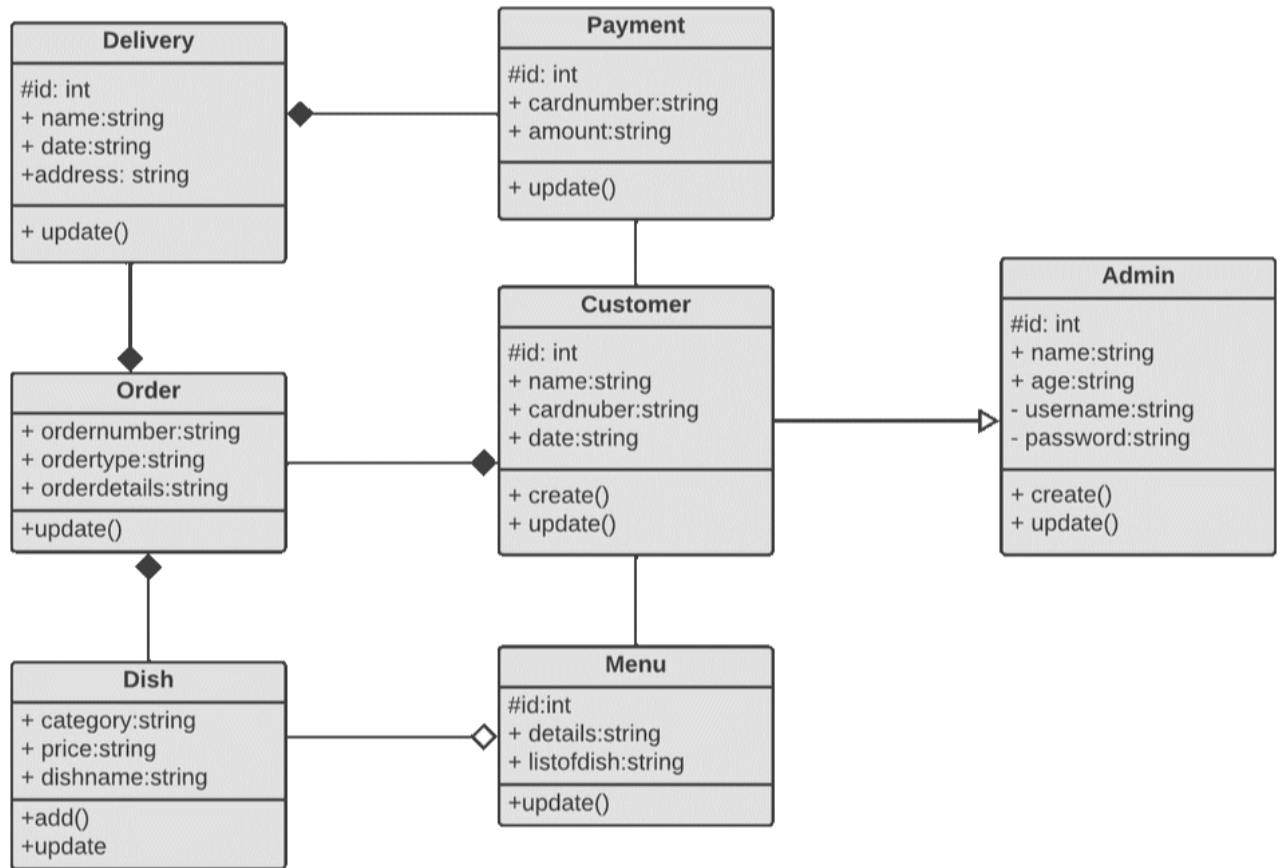
Figure 3.3: Class Diagram of Online sweet shop

Here, one admin manages this shop. Also, one sweet shop has many sweet items. One customer makes one or many orders. And many orders can have one or many items. Each entity has its own primary key attribute. For example, Customer has customer ID as a key attribute and admin has admin ID as a key attribute. [2]

3.1.4 Dynamic Modeling (DFD)

Sequence Diagram

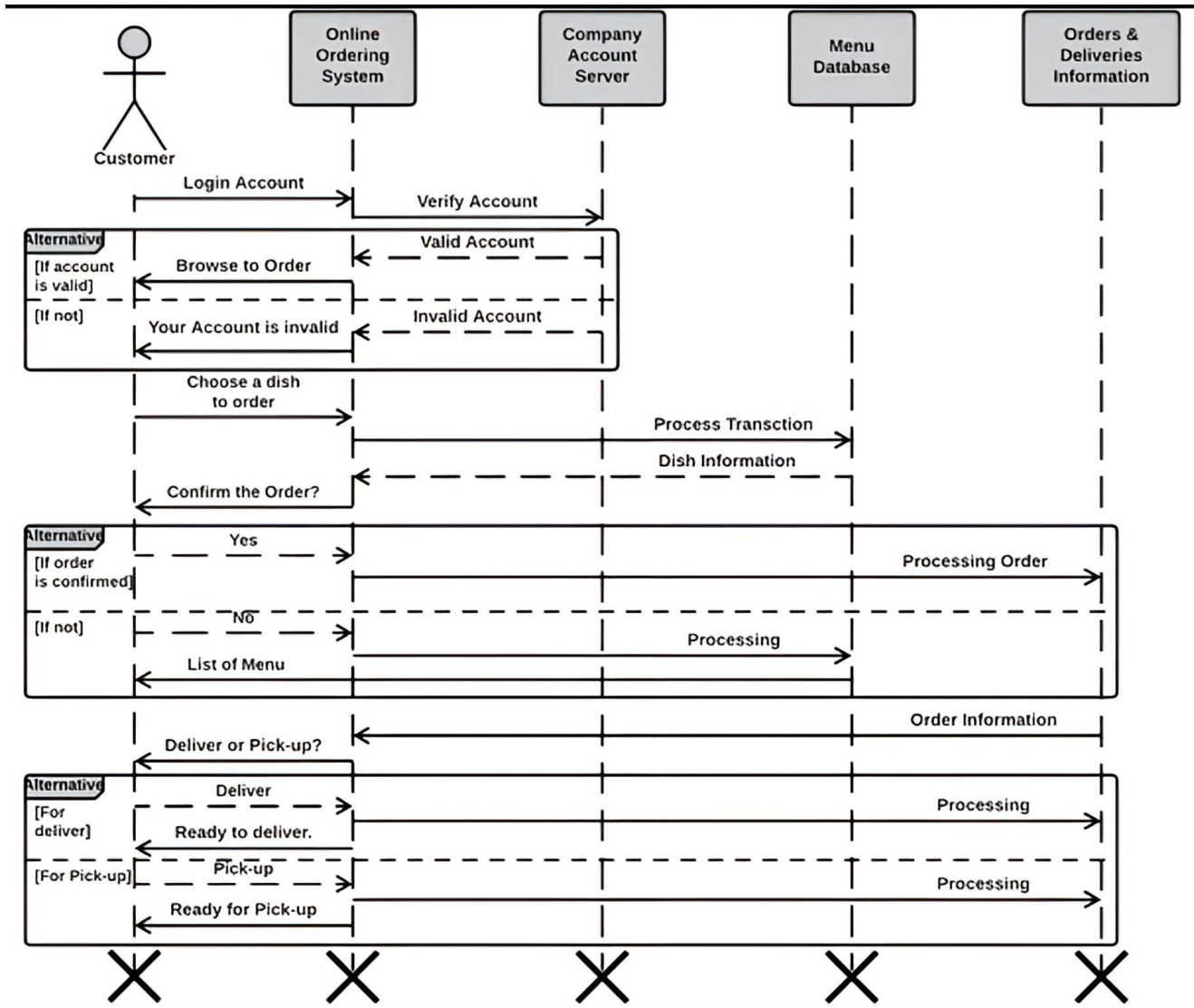
Figure 3.4: Sequence Diagram of Online sweet shop

3.1.4.2 Process Modeling

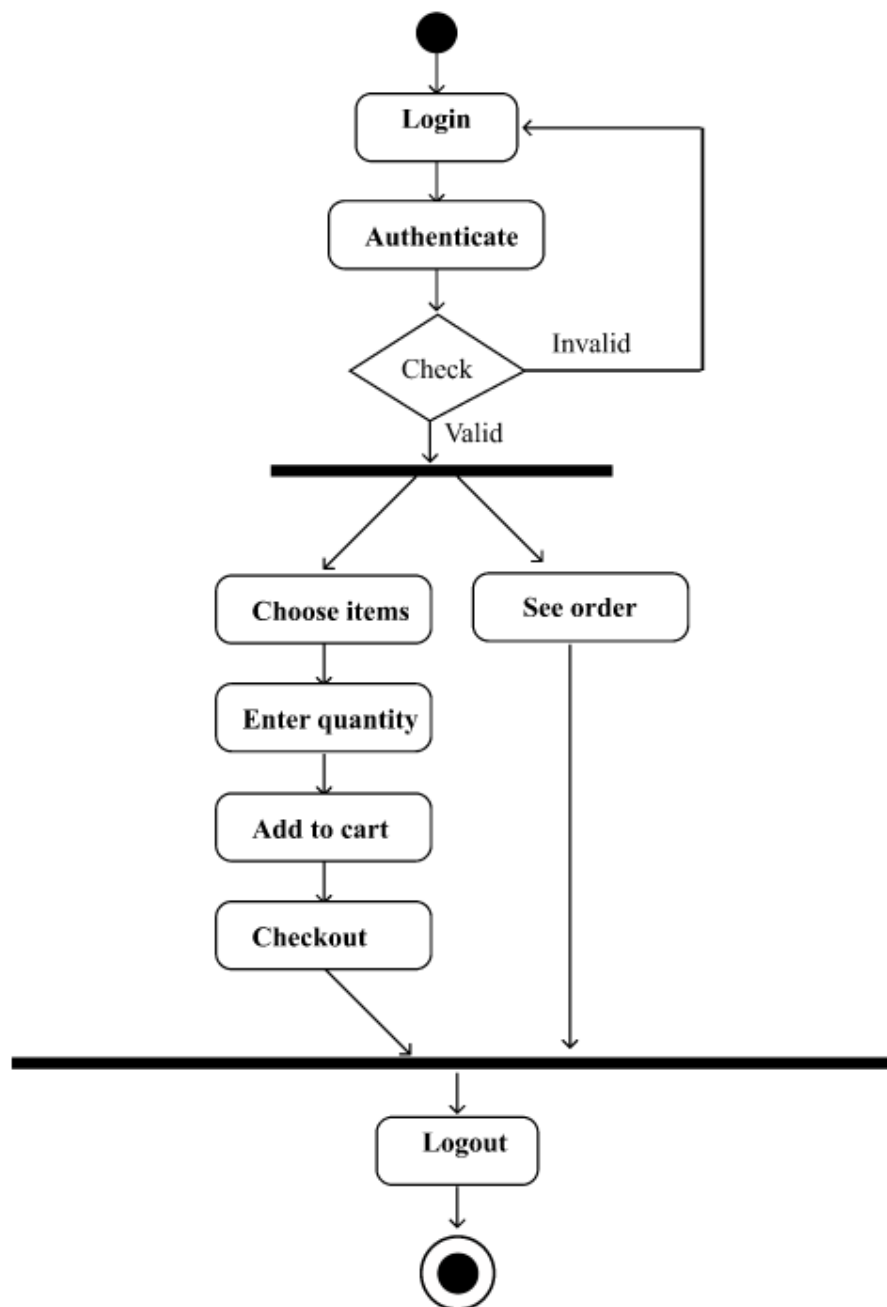   **Activity Diagram**

Activity Diagram of Users

Figure 3.5: Activity Diagram of User
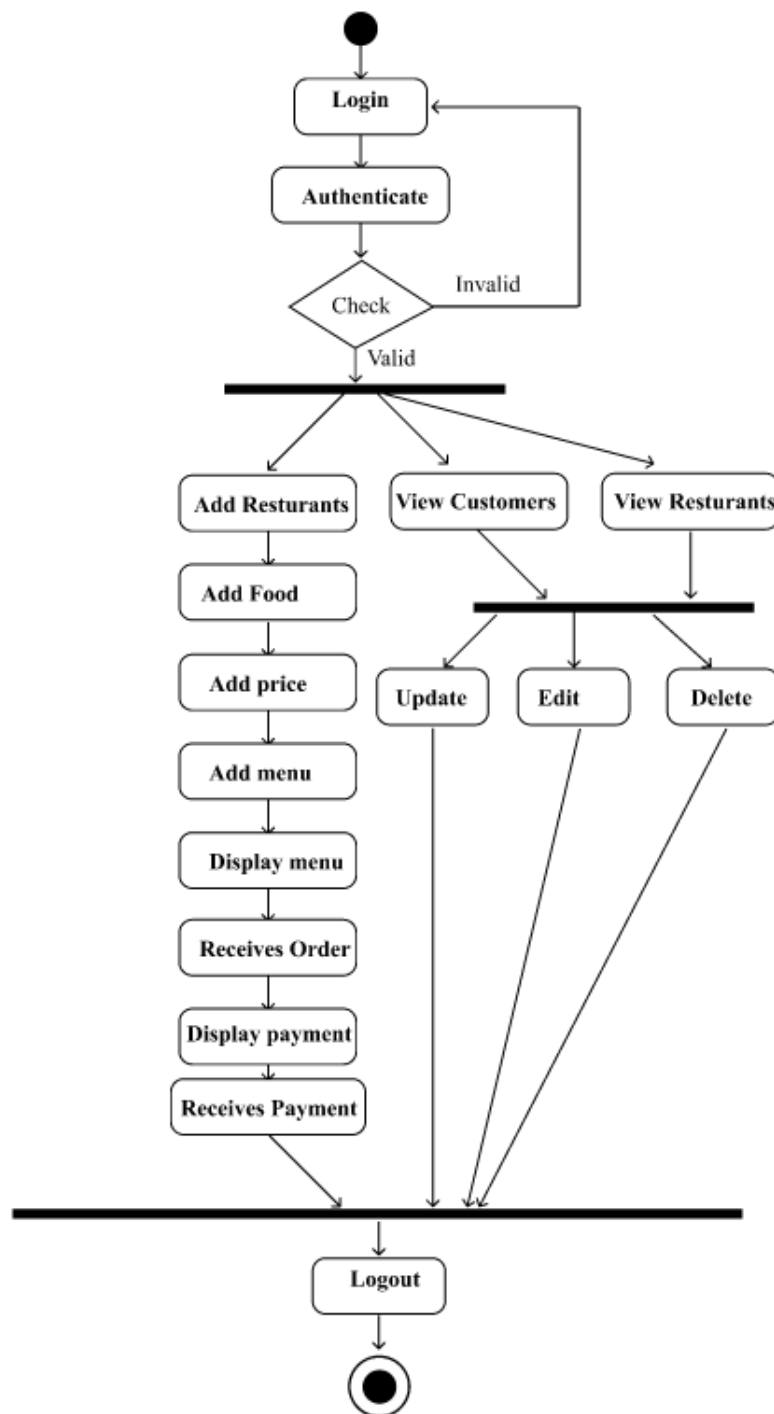
Activity Diagram of Admin

Fig 3.6.: Activity Diagram of Admin

3.2 System Design
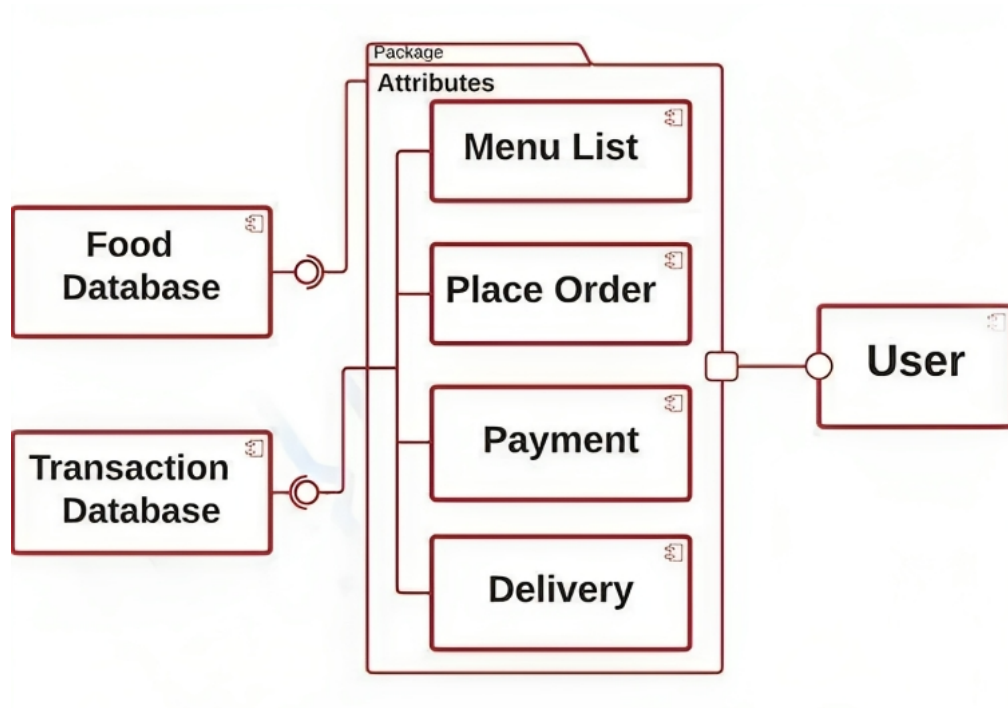
3.2.1. Component Diagram

Fig 3.7.: Component Diagram

Deployment Diagram
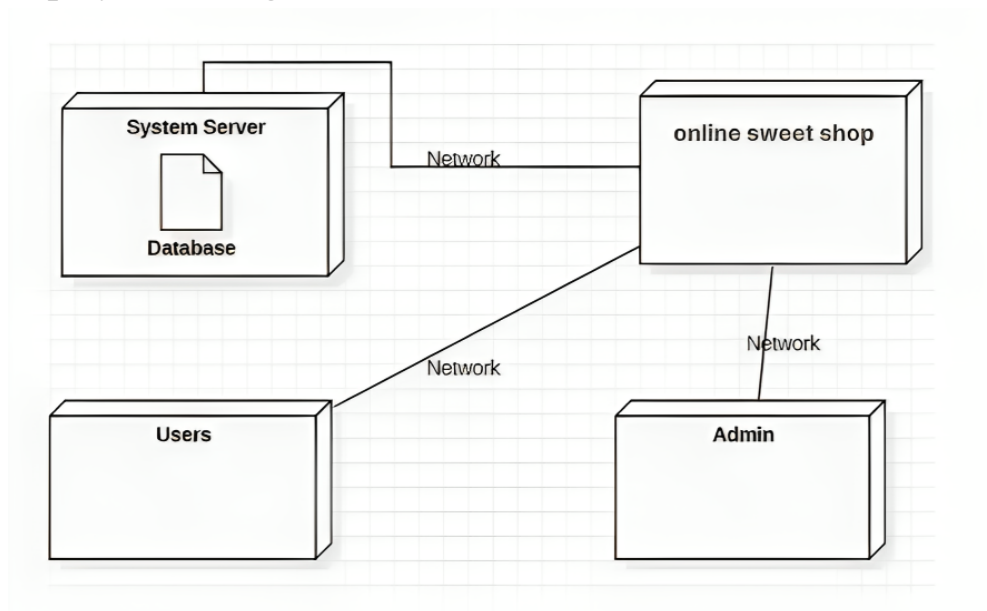
Fig 3.8: Deployment Diagram

3.2.5 Algorithm Details

Algorithm used for Performance analysis Using Standard Deviation

Step-1:-Start

Step-2:-Count the number of orders received by the system for each and every Branches and shop the data in an Array.

Step-3:-Calculate the standard deviation using following method

```
function Stand_Deviation($arr)
{
    $num_of_elements = count($arr);
    $variance = 0.0;
    $average = array_sum($arr)/$num_of_elements;
    foreach($arr as $i)
    {
        $variance += pow(($i - $average), 2);
    }
    return (float)sqrt($variance/$num_of_elements);
}
```

Step-4:-Compare every order count of each restaurant with the standard deviation to analysis the performance

```
if($ocount>$sd)
    {
        $p="High";
    }
    elseif ($ocount==$sd)
{
    $p="Average";
    }
    else
    {
    $p="Low"
    }
```

Step-5:-Display $p for each Restaurant.

Step-6:-Stop.

Algorithm for Rating System For Branches

Step-1:-Start
Step-2:-Select all the ratings given to the Branches from the database.
Step-3:-Calculate Average rating from all those rating

```
foreach($itemRating as $rate){
        $ratingNumber+= $rate['ratingNumber'];
        $count += 1;
        if($rate['ratingNumber'] == 5) {
                $fiveStarRating +=1;
        } else if($rate['ratingNumber'] == 4) {
                $fourStarRating +=1;
        } else if($rate['ratingNumber'] == 3) {
                $threeStarRating +=1;
        } else if($rate['ratingNumber'] == 2) {
                $twoStarRating +=1;
        } else if($rate['ratingNumber'] == 1) {
                $oneStarRating +=1;
        }
    }
    $average = 0;
    if($ratingNumber && $count) {
            $average = $ratingNumber/$count;
    }
    $averageRating = round($average, 0);
```

Step-4:-Calculate % of each rating number

```
$fiveStarRatingPercent = round(($fiveStarRating/5)*100);
$fiveStarRatingPercent=!empty($fiveStarRatingPercent)?$fiveStarRatingPercent.'%':'0%';
$fourStarRatingPercent = round(($fourStarRating/5)*100);
$fourStarRatingPercent=!empty($fourStarRatingPercent)?$fourStarRatingPercent.'%':'0%';
$threeStarRatingPercent = round(($threeStarRating/5)*100);
$threeStarRatingPercent=!empty($threeStarRatingPercent)?$threeStarRatingPercent.'%':'0%';
$twoStarRatingPercent = round(($twoStarRating/5)*100);
$twoStarRatingPercent=!empty($twoStarRatingPercent)?$twoStarRatingPercent.'%':'0%';
$oneStarRatingPercent = round(($oneStarRating/5)*100);
$oneStarRatingPercent=!empty($oneStarRatingPercent)?$oneStarRatingPercent.'%':'0%';
```

Step-5:-Display the average rating and highest percentage.
Step-6:-Stop.

Chapter 4: Implementation and Testing

4.1 Implementation

The tools and techniques used to implement the system and the implementation details of various modules of online sweet ordering system are as follows:

4.1.1 Tools Used

The tools used for the implementation of online sweet shop are listed below:

Draw.io

It was used for designing the system designs such as system flowchart, ER diagram, relational model, architectural design, use case diagram etc.

Balsamiq

It was used for designing the user interface of the system.

HTML CSS JavaScript

HTML, CSS and JavaScript were used for the front-end development. HTML was used for the web page elements. CSS was used for providing its own styling to the components. JavaScript was used for client-side validations and adding dynamic components to the website. [3]

BOOTSTRAP

Bootstrap was used for adding responsive components in our system as well as other bootstrap components such as buttons, cards, navbar etc.

PHP

PHP was used as the back-end programming language. The project is completely based on PHP.

MySQL

MySQL was used for the implementation of the database system. We have used it for creating databases and tables, performing updates on the data and deleting the data from the database. [3]

XAMPP

As mentioned above HTML CSS and PHP are the languages used in this project. As PHP is a server-side language it requires a server to be interpreted. Therefore, we have used the XAMPP server. This software was used to connect to Apache and MySQL.

Sublime Text Editor

The code of the system was written using Sublime Text Editor.

4.1.2 Implementation Details of Modules

The major functional modules of online sweet shop and their implementation is shown in the figure below:
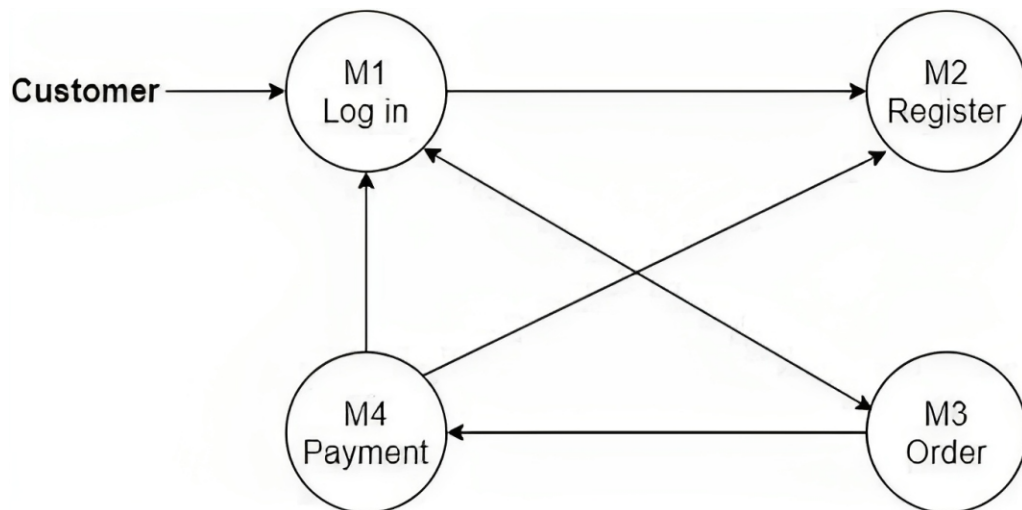
Figure 4.1: Implementation of Modules

In the above figure, there are four modules M1, M2, M3 and M4. Each module is related to the other module as shown in the figure. If the customer is not registered he/she must register into the system then only he/she can login into the system. Likewise, for the payment module the customer must be logged in the system. When the customer wants to make an order he/she must be registered in the system.

The descriptions of modules is shown in the table below:

Table 4.1: Descriptions of Modules

Code	Module Name	Description	Arguments	Return Type
M1	Login Module	This module has been used for the logging into the system.	Username Password	Boolean
M2	Sign Up Module	This module has been used for registering a new user into the system.	Username Password Email Address Contact	Boolean
M3	Place sweets Order Module	This module has been used for placement of order when the customer confirms the order.	Order ID Customer ID Item name Quantity Price Total	Void
M4	Payment Module	This module accepts the payment from the user and shops the payment details in the database.	Email Card Number Date Amount Customer Name	Boolean

4.2 Testing

The testing for online sweet shop is done by testing the unit and system modules like login module, sign up module, place order module and payment module. [1]

4.2.1 Test Cases for Unit Testing

The test cases for the login module and sign up module are as follows:

25

Table 4.2: Test Case 001-Login

Project Name: ONLINE SWEET SHOP						
Test Case						
Test Case ID: TC_001				**Test Designed by: Sujan Baral** **Susanta Timsina**		
Test Priority (Low/Medium/High): Medium				**Test Designed date: 2023-07-6**		
Module Name: Login Screen				**Test Executed by: Sujan Baral**		
Test Title: Verify login with valid username and password				**Test Execution date: 2023-07-6**		
Description: Test the OFOS login page						
Pre-conditions: User has valid username and password						
Dependencies:						
Step	**Test Steps**	**Test Data**	**Expected Result**	**Actual Result**	**Status (Pass/Fail)**	**Notes**
1	Navigate to login page		Login page should open	As Expected User is navigated to Login page of system	Pass	**Then the page moves to another page.**
2	Provide valid username	User Name = Sujan76	Credential can be entered	As Expected	Pass	
3	Provide valid password	Password = Sujan123	Credential can be entered	As Expected	Pass	
4	Click on LOGIN button		User should be able to login to an application	As Expected i.e. User is able to log in to the system	Pass	
Post-conditions: **User is validated with the database and successfully logs into the OFOS system.** **The account session details are logged in to the database.**						

Table 4.3: Test Case 002-Sign Up

Project Name: ONLINE SWEET SHOP						
Test Case						
Test Case ID: TC_002				**Test Designed by: Sujan Baral** **Susanta Timsina**		
Test Priority (Low/Medium/High): Medium				**Test Designed date: 2023-07-6**		
Module Name: Sign Up Screen				**Test Executed by: Sujan Baral**		
Test Title: Register new customer into the system				**Test Execution date: 2023-07-6**		
Description: Test the OFOS sign up page.						
Pre-conditions: User has all the necessary details.						
Dependencies:						
Step	**Test Steps**	**Test Data**	**Expected Result**	**Actual Result**	**Status (Pass/Fail)**	**Notes**
1	Navigate to sign up page		Sign up page should open	As Expected i.e. User is navigated to Sign Up page of system	Pass	
2	Provide all required information	User Name = Suman Password = Suman123 Email= suman76@ gmail.com Address =Dharan	Credential can be entered	As Expected	Pass	
3	Click on SIGNUP button		User should be able to sign up to the system	As Expected i.e. User is able to registered to the system	Pass	
Post-conditions: **User credentials are validated and successfully registered to the OFOS system.**						

4.2.2 Test Cases for System Testing

The test cases for place order and payment module are as follows:

Table 4.4: Test Case 003-Place Order

Project Name: ONLINE SWEET SHOP						
Test Case						
Test Case ID: TC_003				Test Designed by: Sujan Baral Susanta Timsina		
Test Priority (Low/Medium/High): High				Test Designed date: 2023-07-6		
Module Name: sweets placed order				Test Executed by: Sujan Baral		
Test Title: Verify place order with valid customer				Test Execution date: 2023-07-6		
Description: Test if the order module is working properly or not.						
Pre-conditions: Customers has a user account in the system.						
Dependencies:						
Step	**Test Steps**	**Test Data**	**Expected Result**	**Actual Result**	**Status (Pass/ Fail)**	**Notes**
1	Navigate to menu page		Menu page should open	As Expected i.e. Customer is navigated to Menu page of online sweet shop	Pass	
2	Select the desired sweet to cart	sweet = Rasbari, Barfi	sweet items should be selected	As Expected	Pass	
3	Validate Customer credentials	Username = sujan Password = Sujan123	Customer Credential is validated	As Expected	Pass	
4	Click on Add to cart button		Customer should be able to have their order in a cart	As Expected i.e. User is able to add their order on cart	Pass	
Post-conditions: Customer has placed the order successfully. The order details are loaded into the database.						

	Project Name: ONLINE SWEET SHOP					
Test Case						
Test Case ID: TC_004				**Test Designed by: Sujan Baral** **Susanta Timsina**		
Test Priority (Low/Medium/High): High				**Test Designed date: 2023-07-6**		
Module Name: Payment				**Test Executed by: Sujan Baral**		
Test Title: Payment Verification				**Test Execution date: 2023-07-6**		
Description: Test if payment module is working accurately or not.						
Pre-conditions: Customer has a user account in the system.						
Dependencies:						
Step	**Test Steps**	**Test Data**	**Expected Result**	**Actual Result**	**Status (Pass/ Fail)**	**Notes**
1	Navigate to payment page		Payment page should open	As Expected i.e. Customer is moved to payment page of Online sweet shop	Pass	
2	Enter Payment Details	Email=suman76@gmail.com VISACard no = 86372538137	Payment Details should be entered	As Expected	Pass	
3	Payment details verification		Payment details are verified	Customer will get The product	Pass	
4	Click on Pay Now button		Customer should be able to pay respective amount	Customer is able to make payment	Pass	
Post-conditions: **The customer makes the payment successfully.**						

Table 4.5: Test Case 004-Payment

Chapter 5: Conclusion and Future Recommendations

5.1 Lesson Learnt

This project has helped to learn how to develop web-based online sweet shops and implement it across various platforms. It helped to get familiar with both client-side and server-side programming languages. It helped to learn how to host a web application locally on a host computer. It helped to learn how a customer could make orders and add items to cart. It also helped to know how to implement payment gateway using Stripe. It also helped to get familiar with Bootstrap and its components. It helped to learn how to do pair programming and finish the project within the schedule. It helped to know how to implement theoretical knowledge gained from various subjects in practical life. This project has helped to gain great skills for project management and software development.

5.2 Conclusion

It is presented as to why and how the online sweet shop can be used and built. With a private login system customers can place a secure online order and also can view or receive the updates in real-time. It allows the customers to navigate through the menus and customize their orders. The experience in developing this software was to show the abilities of wireless communication and in refining the business management and decent service delivery. etc. By this application the customer can access their adored sweets items in their place itself.

5.3 Future Recommendations

The following section describes the work that will be implemented with future releases of the software.

- Customize orders: Allow customers to customize .
- Enhance User Interface by adding more user interactive features. Provide Deals and promotional Offer details to the home page. Provide Recipes of the Week/Day to Home Page
- Payment Options: Add different payment options such as Esewa,Khalti, PayPal, Cash and Gift Cards etc. Allow to save payment details for future use.
- Allow to process an order as a Guest
- Order Process Estimate: Provide customer a visual graphical order status bar
- Order Status: Show only Active orders to Restaurant Employees
- Delivery Options: Add delivery option
- Order Ready notification: Send an Order Ready notification to the customer .

REFERENCES

[1]   I. Sommerville, Software Engineering, 10th ed. London: Pearson Education Limited, 2016.

[2]   A. Silberschatz, H. Korth and S. Sudarshan, DATABASE SYSTEM CONCEPTS, 6th ed. New York: McGraw-Hill, 2011, pp. 39-55, 259-321.

[3]   P. Deitel and H. Deitel, Internet & WorldWideWeb HOW TO PROGRAM, 4th ed. New Jersey: Pearson Education, Inc., 2008.

[4]   "Build native PowerPoint timelines online | Office Timeline Online", https://www.officetimeline.com/, 2021. [Online]. Available:
  https://online.officetimeline.com/app/#/new-from-template?templateId=db9d8356-68f1 4b2b-b9a5-f8efe7fd88f7&fromLandingPage=true.

[5]   "(DOC) An Online Sweet shop Documentation | Monica Diaz - Academia.edu", https://www.academia.edu
2021[Online].Available:https://www.academia.edu/33253493/An_Online   sweet   shop System_Documentation

[6]   "Daraz : Sweets Delivery Service in Kathmandu and Pokhara", https://daraz.com/, 2020. [Online]. Available: https://daraz.com/

APPENDICES

i.   Home Page
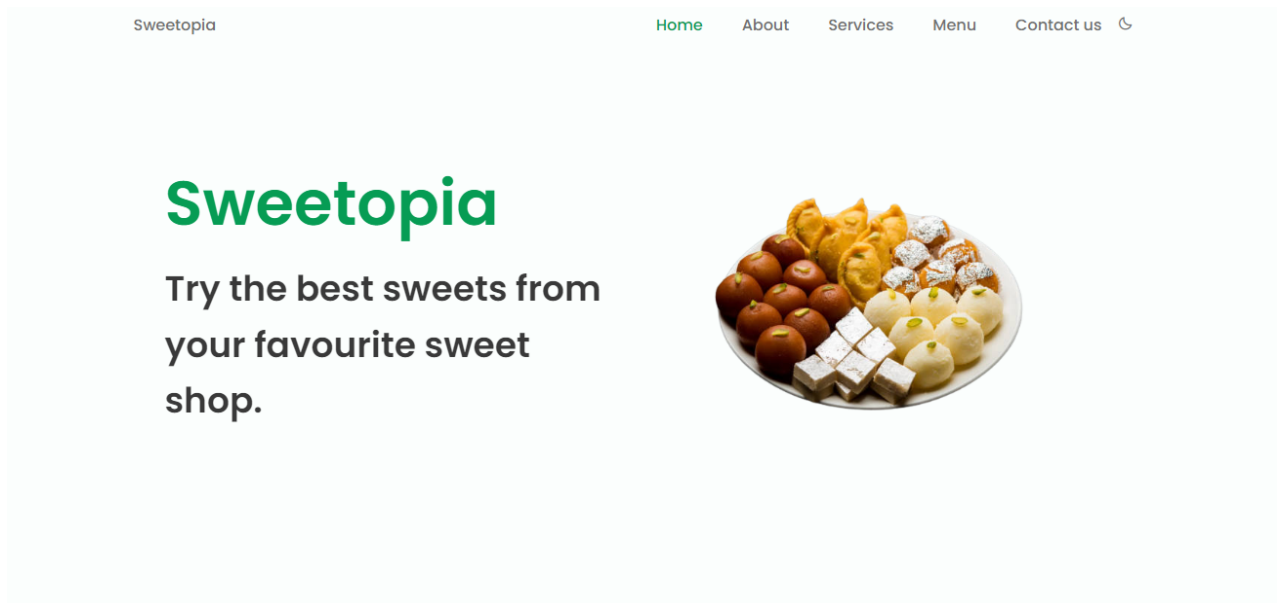
ii.   Login Page

iii.   Menu Page

iv.   Admin Dashboard