

MERO DAIRY (DAIRY MANAGEMENT SYSTEM)
FOR
“Dhulikhel Dairy Farm”



Submitted By:
SANDESH LAMA
NIST College, Banepa
Exam Roll No: 10931/19
TU Registration No: 7-2-1014-12-2019

A Summer Project Submitted to
Faculty of Management, Tribhuvan University
In partial fulfilment of the requirements for the degree of
Bachelor of Information Management

Banepa-09, Kavrepalanchok

October, 2023

STUDENT DECLARATION

This is to certify that I have completed the Summer Project entitled “**Mero Dairy**” under the guidance and supervision of “**Mr. Yubaraj Neupane**” in partial fulfilment of the requirements for the degree of Bachelor of Information Management at Faculty of Management, Tribhuvan University. This is my original work in the year 2023 and I have not submitted it earlier elsewhere.

Date: October 2023

Name: Sandesh Lama

Signature:

TU Symbol No:10931/19

CERTIFICATE FROM THE SUPERVISOR

This is to certify that the summer project entitled “**Mero Dairy**” is an academic work done by **Sandesh Lama** submitted for the partial fulfilment of the requirements for the degree of **Bachelors of Information Management (BIM)** at Faculty of Management, Tribhuvan University under my guidance and supervision. To the best of my knowledge, the information presented by him in the summer project report has not been submitted earlier.

Signature of the Supervisor

Name: Mr. Yubaraj Neupane

Designation: Supervisor

Date: October 2023

LETTER OF APPROVAL

Following a successful presentation and evaluation by the research evaluation committee, this report entitled "**Mero Dairy**" submitted by Mr. Sandesh Lama has been accepted and forwarded to the University for awarding Bachelor of Information Management (BIM) degree.

Signature

Name: Samish Shrestha

Designation: Faculty Coordinator, BIM

Date:-

Signature

External Examiner

ACKNOWLEDGEMENT

It gives me immense pleasure to express my deepest sense of gratitude and sincere thanks to my highly respected and esteemed supervisor **Mr. Yubaraj Neupane**, for his valuable guidance, encouragement and support throughout this project. His useful suggestions for this whole work and co-operative behaviour are sincerely acknowledged.

I am also grateful to the entire BIM Department of NIST College for housing us a seemly environment where I can work with this project. I was pleased to be under the commands of the department to help me from all possible ways. I am also grateful to my teachers, for their constant support and guidance. Likewise, I would like to express my sincere thanks to all my friends and family members who helped me directly or indirectly during this project work. Finally, I am immensely indebted to all my well wishers and people who provided me with insightful feedbacks throughout my work.

EXECUTIVE SUMMARY

Mero Dairy is a web-based application, which is a dairy management system for dairy businesses to manage the records more effectively and efficiently without any hassles like in traditional pen and paper methods. Mero Dairy is a complete platform for the dairy owner/administrator to maintain day-to-day records crucial for any dairy business.

This project was developed after an analysis of the current methods of maintaining records in small-scaled dairy businesses. The system development's core focus is on providing a complete solution platform for dairy owners to manage their records accurately, with minimum efforts in time-efficient manner. For developing the system based on the identified problems and requirements, HTML, CSS, JavaScript, PHP, and MySQL database were used. VsCode was the IDE of choice for coding the system and Safari Browser was used to test and run the system.

This report includes all the descriptions of the Dairy Management System and the activities performed to guide the project to its completion. Here is the complete documentation of the system in an understandable form.

LIST OF FIGURES

Figure 1: Use Case Diagram of Mero Dairy.....	5
Figure 2: Class Diagram of Mero Dairy.....	7
Figure 3: Activity Diagram of Mero Dairy.....	8
Figure 4: Sequence Diagram of Mero Dairy.....	9
Figure 5: Login Page - Sign In Section.....	10
Figure 6: Login Page - Sign Up Section.....	10
Figure 7: Admin Dashboard.....	11

LIST OF TABLES

Table 1: Test Case for Admin Login.....	13
Table 2: Test Case for Adding a New Item.....	14
Table 3: Test Case for Recording a Sale.....	15

LIST OF ABBREVIATIONS

- BIM : Bachelor of Information Management
CSS : Cascading Style Sheets
DBMS: Database Management System
DMS : Dairy Management System
HTML : Hypertext Markup Language
IDE : Integrated Development Environment
JS : JavaScript
MySQL : My Structured Query Language
PHP : Hypertext Preprocessor
RAM : Random Access Memory
RAM : Random Access Memory
SQL : Structured Query Language
TU : Tribhuvan University
UI : User Interface
UML : Unified Modeling Language
URL : Uniform Resource Locator
VS Code : Visual Studio Code
XAMPP : X (cross platform), Apache, MySQL, PHP, Perl

TABLE OF CONTENTS

<i>TITLE PAGE</i>	<i>i</i>
<i>STUDENT DECLARATION</i>	<i>ii</i>
<i>CERTIFICATE FROM THE SUPERVISOR</i>	<i>iii</i>
<i>LETTER OF APPROVAL</i>	<i>iv</i>
<i>ACKNOWLEDGEMENT</i>	<i>v</i>
<i>EXECUTIVE SUMMARY</i>	<i>vi</i>
<i>LIST OF FIGURES</i>	<i>vii</i>
<i>LIST OF TABLES</i>	<i>viii</i>
<i>LIST OF ABBREVIATIONS</i>	<i>ix</i>
Chapter 1: Introduction.....	1
1.1 Background	1
1.2 Introduction of the organization	1
1.3 Current situation of the organization.....	1
1.4 Problem Statement	2
1.5 Objectives.....	2
1.6 Methodology	2
1.6.1 Project Framework.....	2
1.6.2 Data and Information.....	3
1.6.3 Tools Used for Data Collection	3
Chapter 2: Task and Activities Performed	4
2.1 Analysis of tasks, activities, problem, issues	4
2.1.1 Analysis of tasks	4
2.2 Requirement Specification and Analysis.....	4
2.2.1 Functional Requirements	4
2.3 Feasibility Study.....	6
2.3.1 Technical Feasibility	6
2.3.2 Operational Feasibility.....	6
2.3.3 Economical Feasibility	6

2.4 System Design.....	7
2.4.1 Class Diagram.....	7
2.4.2 Activity Diagram	8
2.4.3 Sequence Diagram.....	9
2.5 System Implementation.....	10
2.5.1 Module Description	10
2.5.2 Tools and Technology Used.....	11
2.6 Software Testing.....	12
2.6.1 Unit Testing.....	12
2.6.2 Integration Testing	12
2.6.3 System Testing.....	12
Chapter 3: Discussion and Conclusion.....	16
3.1 Findings.....	16
3.2 Discussion	16
3.3 Conclusion.....	16
3.4 Future Enhancements	17
References.....	18
Appendices.....	a
Appendix A: Screenshots	a
Appendix B: Sample Codes.....	f

Chapter 1: Introduction

1.1 Background

Dhulikhel Dairy Farm is a dairy business which has been successfully running for over the last 16 years. The business has grown steadily over the years along with which the operations, workload, and scale of the business has increased as well. However, the dairy business uses traditional methods of book keeping and records keeping till this day. Therefore, a dairy management system is critical for the dairy business to track its daily operations and maintain accurate records. The dairy management system allows dairy businesses to streamline their basic accounting, billing, inventory management, and supplier management. The proposed dairy management system aims to automate these processes and provide a user-friendly interface for the dairy business owner to manage the operations effectively.

1.2 Introduction of the organization

Dhulikhel Dairy Farm is a small dairy business located at Dhulikhel Buspark, Kavrepalanchok. It was established in 2063 B.S. What started as a tiny dairy shop has now expanded into a sizeable business with improved machinery and equipments, becoming the major provider of dairy products in Dhulikhel. This family run business serves hundreds of customers daily. The proprietor Prithviman Tamang handles the managerial operations, while two staffs are in production department. Likewise, other family members take turns at customer service and cashier.

1.3 Current situation of the organization

Even though the business side is pretty successful and the influx of customers has only seen to increase, the dairy is totally dependent on keeping the records in handwritten form. Registrars are used for manually recording everything including supplier details, daily supply details, sales, billing, and other accounting details. Due to this, the business occasionally has to face issues with these manual records being inconsistent, time-consuming, and inaccurate.

1.4 Problem Statement

Dhulikhel Dairy Farm faces several challenges in managing their operations efficiently. The manual process of maintaining records, inventory management, and supplier management is time-consuming and prone to errors. These challenges can and has led to inaccuracies, inefficiencies, and reduced profitability in the past. Therefore, there is a need for a dairy management system that addresses these challenges and provides a robust platform for the dairy business to manage its operations effectively.

1.5 Objectives

The objectives of the proposed management system are as follows:

- To automate the billing processes
- To create an inventory management module to track dairy product stock, and ensure efficient stock control.
- To maintain accurate records of debtors, creditors and suppliers.
- To provide accurate and timely information for decision-making.
- To provide a user-friendly interface for dairy business owner to manage their operations efficiently.

1.6 Methodology

1.6.1 Project Framework

The proposed management system will be developed using the Waterfall methodology. The waterfall model is a sequential software development process that follows a linear and structured approach. The waterfall model provides a systematic approach to software development and aligns well with the objectives of the Dairy Management System. Here is how the dairy management system was developed using the waterfall model:

1. Requirements Analysis: I gathered and documented the detailed requirements of the DMS, ensuring a clear understanding of the system's functionalities and features.
2. System Design: The system design phase involved creating the architecture, database schema, and user interface design for the DMS based on the SRS.

3. Implementation: This phase focused on the actual coding and development of the DMS modules.

4. Testing: Comprehensive testing was conducted to ensure the DMS functions accurately, handles various scenarios, and produces reliable results.

1.6.2 Data and Information

As a dairy management system is not very common and there were very few instances of such systems, most of the data collection was done through primary sources. Through direct interaction, interview and few questionnaires with the owner of Dhulikhel Dairy Farm I was able to identify the major problems as well as collect requirements for the system. Similarly, I did first hand observation of the clerical processes of the business to gain further insights and specify the requirements.

1.6.3 Tools Used for Data Collection

For the purpose of collecting data about the requirements of the organisation, following tools were used:

i) Direct Interviews

The direct interviews were done with the dairy owner Mr. Prithiviman Tamang as well as his wife, and their son, who are the front desk facilitators for the majority of the time. Higher level management information on requirements and dynamics of the business were collected from the owner . On the other hand, the wife and the son who were mostly involved in the clerical tasks were inquired about the day-to-day workings and functions.

ii) Observation

On-site observation of the workings of the business was done up close. Likewise, the processes of keeping records were closely noted. Nonetheless, the old registrars and records were also observed.

iii) Internet Search

Finally, some information were gathered from thorough research on the internet. Information on the requirements and existing systems of small local businesses were assessed through the internet.

Chapter 2: Task and Activities Performed

2.1 Analysis of tasks, activities, problem, issues

2.1.1 Analysis of tasks

The organization was visited for requirement collection. After the requirement was collected, requirements were broken down into tasks and various modules, and various plans and procedures were created to develop the system. A clear view of organizational needs and various aspects of the business and problems which can be solved by developing the software were found after the visit, and conversation with the stakeholders.

2.2 Requirement Specification and Analysis

The main objective of requirement analysis is to identify and evaluate the needs and expectations of users from the proposed system. It involves gathering and understanding user requirements, system requirements, functional requirements, and non-functional requirements. It helped to have a better understanding of the features and functionalities of the system.

2.2.1 Functional Requirements

Functional requirements define the basic system behavior, it tells us what the system must do, must not do, how the system should behave to inputs, etc. Following are the functional requirements of our DMS:

- The system should automate the accounting process including accounts payable and accounts receivable.
- User authentication and authorization.
- The system should generate bills and invoices for customers.
- The system should manage the dairy business's inventory and track the stock levels.
- The system should manage the dairy business's supplier records including the supply entry, supply level and fat%.



Figure 1: Use Case Diagram of Mero Dairy

2.2.2 Non-functional Requirements

A non-functional requirement is a requirement that is not directly concerned with specific functions but is concerned with properties such as reliability, response, security, safety, etc. Various non-functional requirements of our system are listed below:

- Only authorised user can access the system using the admin's email and password.
- The system should be user-friendly and easy to use.
- The system should be secure and protect sensitive data.
- The system should be scalable to accommodate the dairy's future growth.
- The system should be reliable and available when requested.

2.3 Feasibility Study

2.3.1 Technical Feasibility

The management system will be developed as a web application which requires minimal hardware specifications and simple technologies. Technologies such as HTML, CSS, JavaScript, PHP and MySQL can be used to create the system. VS Code will be used to program and compile the code, and a browser will be used to test the web app. My current device can handle these requirements easily.

2.3.2 Operational Feasibility

The organization currently has a pc, a laptop and a printer for daily uses; the same pc or laptop can be used for using the web application. The system will be easy to use and will integrate with the existing business processes of the dairy business thereby making it operationally feasible as well.

2.3.3 Economical Feasibility

The system will be used and operated in the same computer which the organization uses for daily use. In the distant future, some amount of pc upgrade or maintenance charge might be needed which is economically feasible for the organization. As, the introduction of the new system will decrease the need for notebooks and registrars to maintain records, it will even decrease the cost for the business.

2.4 System Design

System design aims to describe the structure of the software to be implemented, the data models and structures used by the system and the interfaces between system components (Sommerville, 2009). It involves defining how the requirements gathered during earlier stages align with the system components and overall structure. Various UML diagrams are used for showing abstract views and interaction between the components, which is crucial for better implementation of the system.

2.4.1 Class Diagram

A class diagram is a type of static structure diagram that describes the structure of a system by showing the system's classes, attributes, operations, or methods, and the relationships among objects.

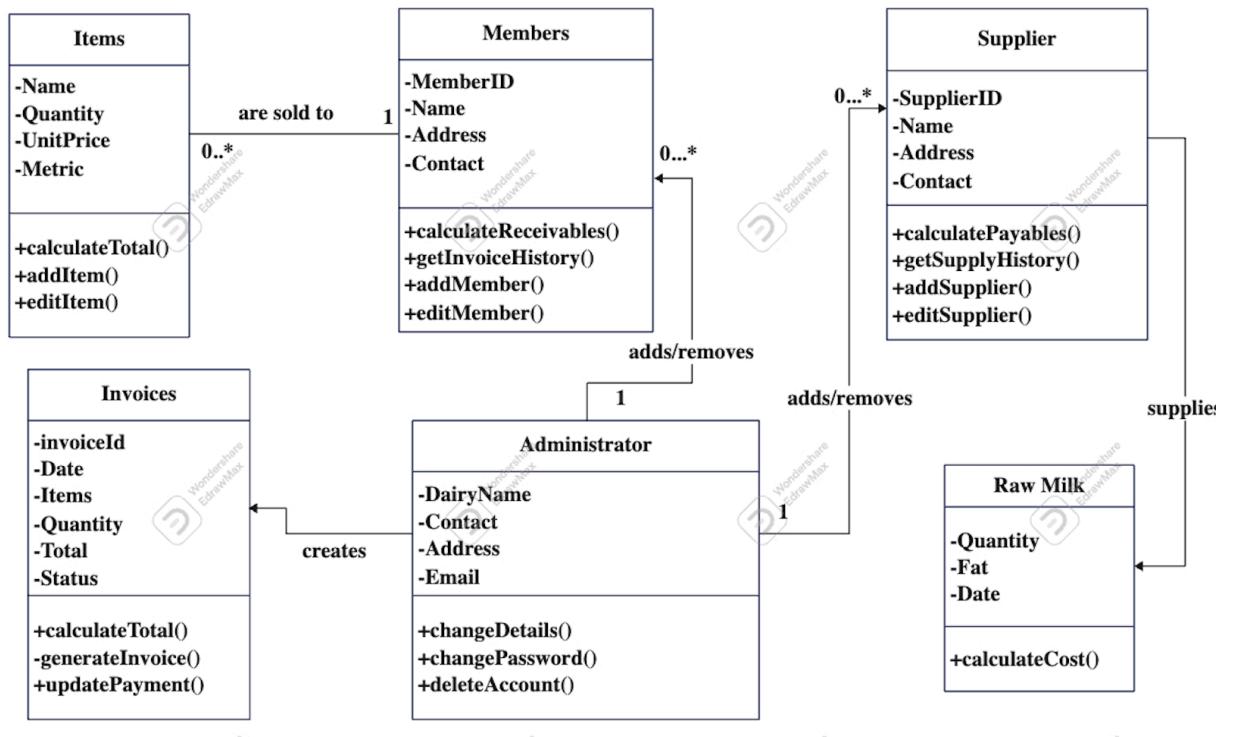


Figure 2: Class Diagram of Mero Dairy

2.4.2 Activity Diagram

An activity diagram represents a series of actions or flow of control in a system like a flowchart or a data flow diagram. They also describe the steps in a use case diagram.

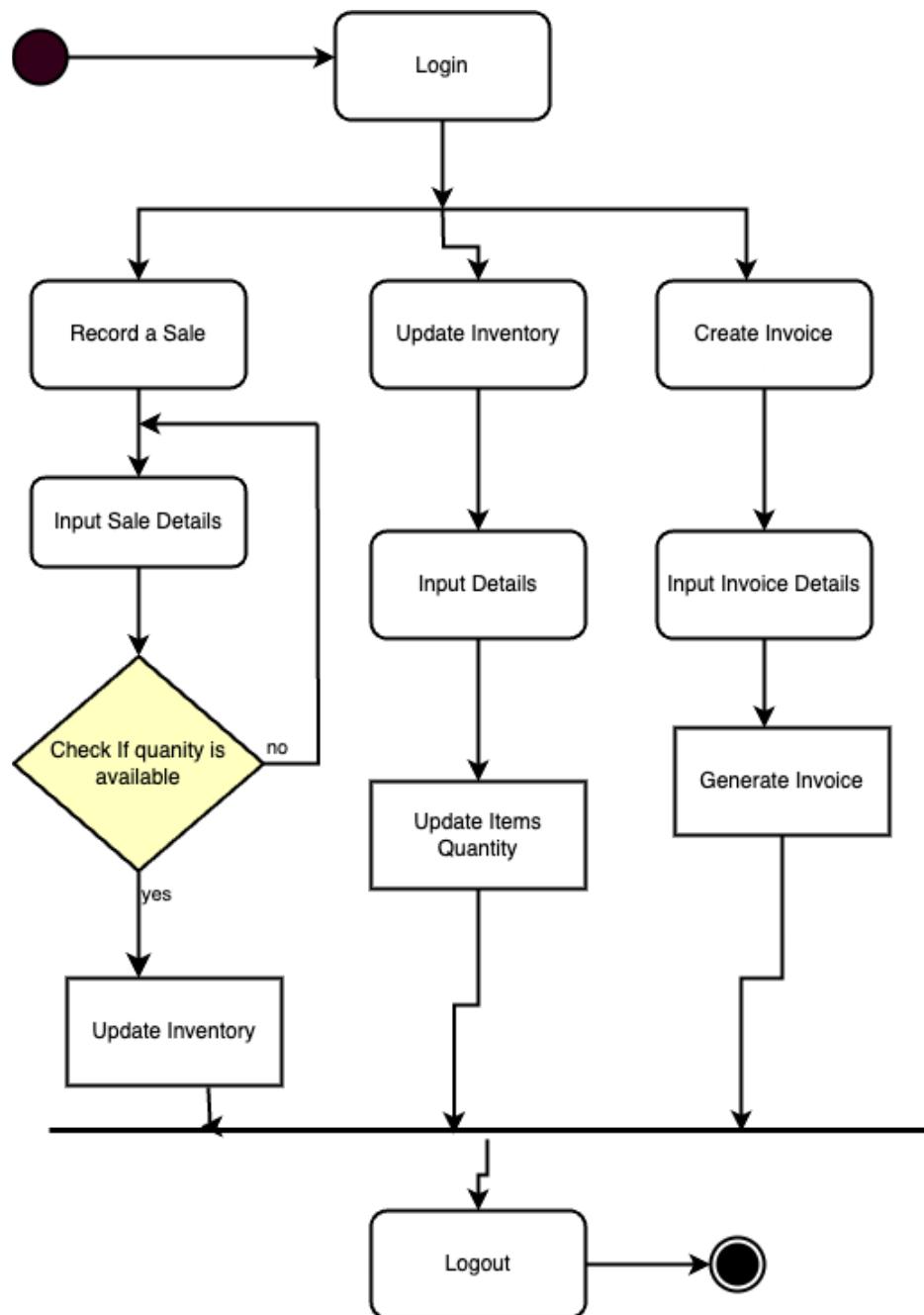


Figure 3: Activity Diagram of Mero Dairy

2.4.3 Sequence Diagram

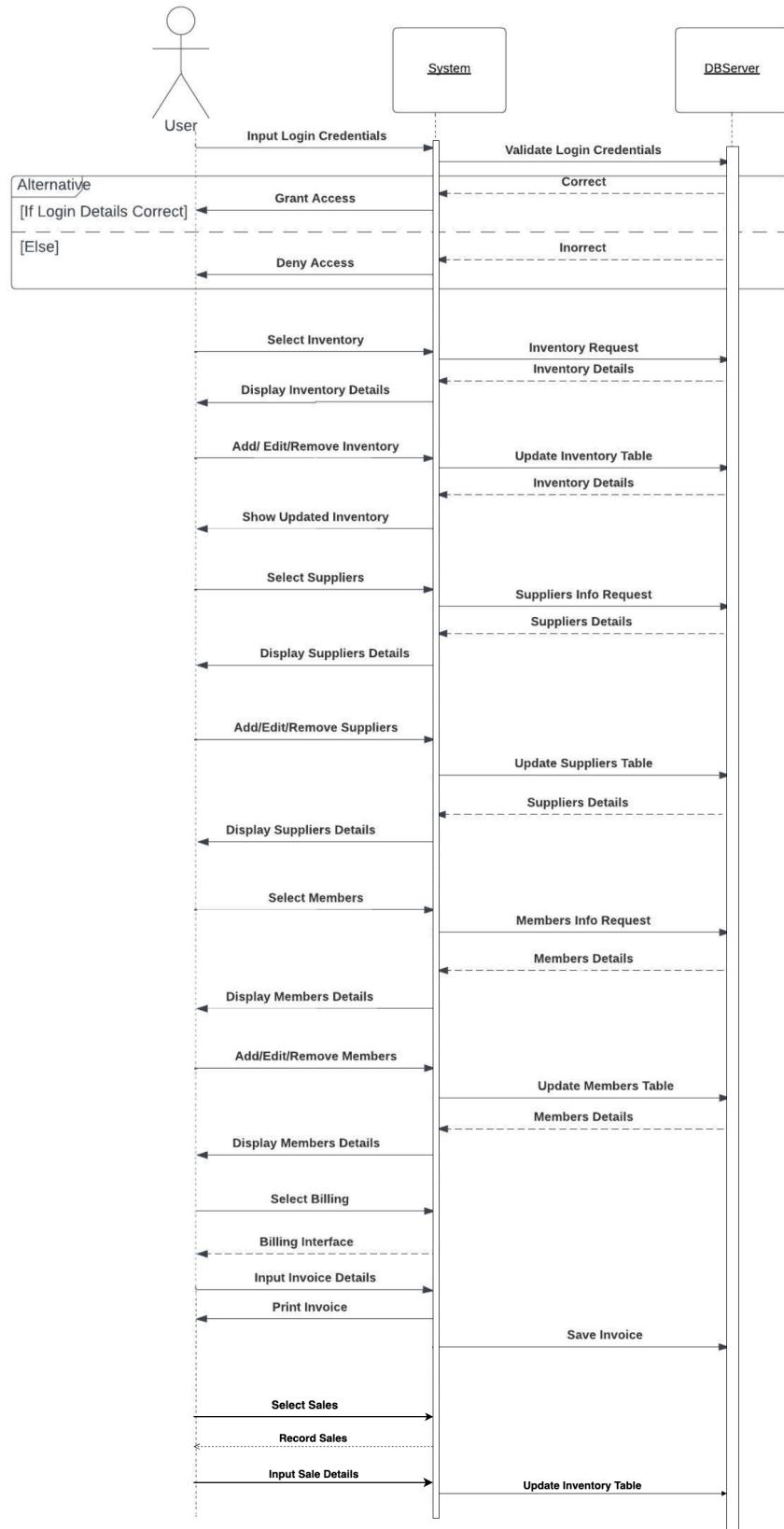


Figure 4: Sequence Diagram of Mero Dairy

2.5 System Implementation

2.5.1 Module Description

The DMS is developed specifically for the dairy owners. For the admin, the system provides login page, a dashboard, and several pages for varied functions such as inventory management, suppliers management, members management, invoice, sales and user settings.

2.5.1.1 Login Panel

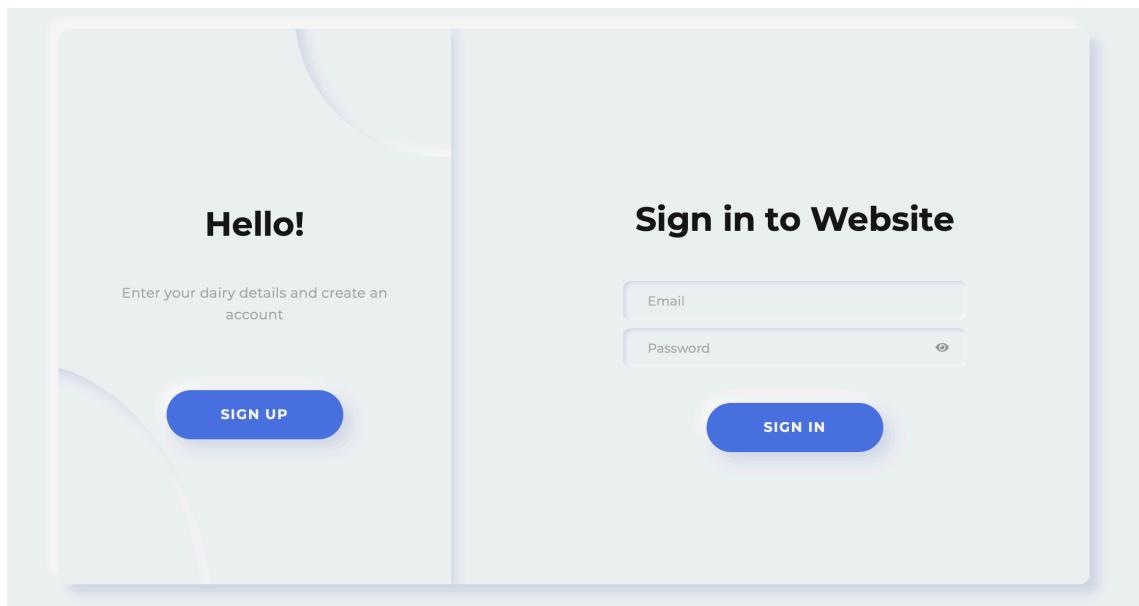


Figure 5: Login Page - Sign In Section

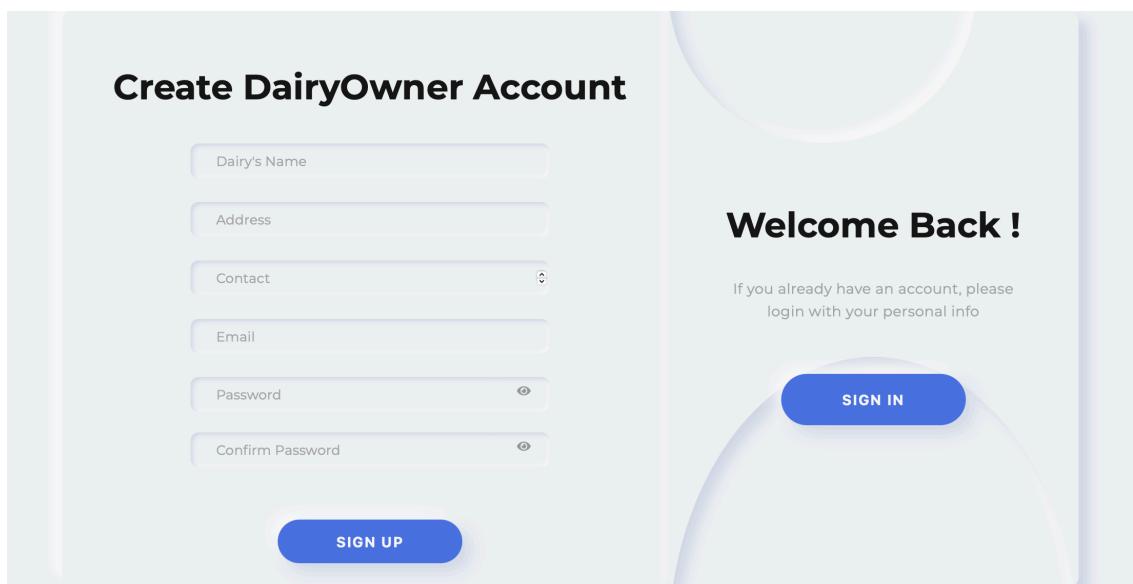


Figure 6: Login Page -Sing Up Section

2.5.1.2 Admin Dashboard

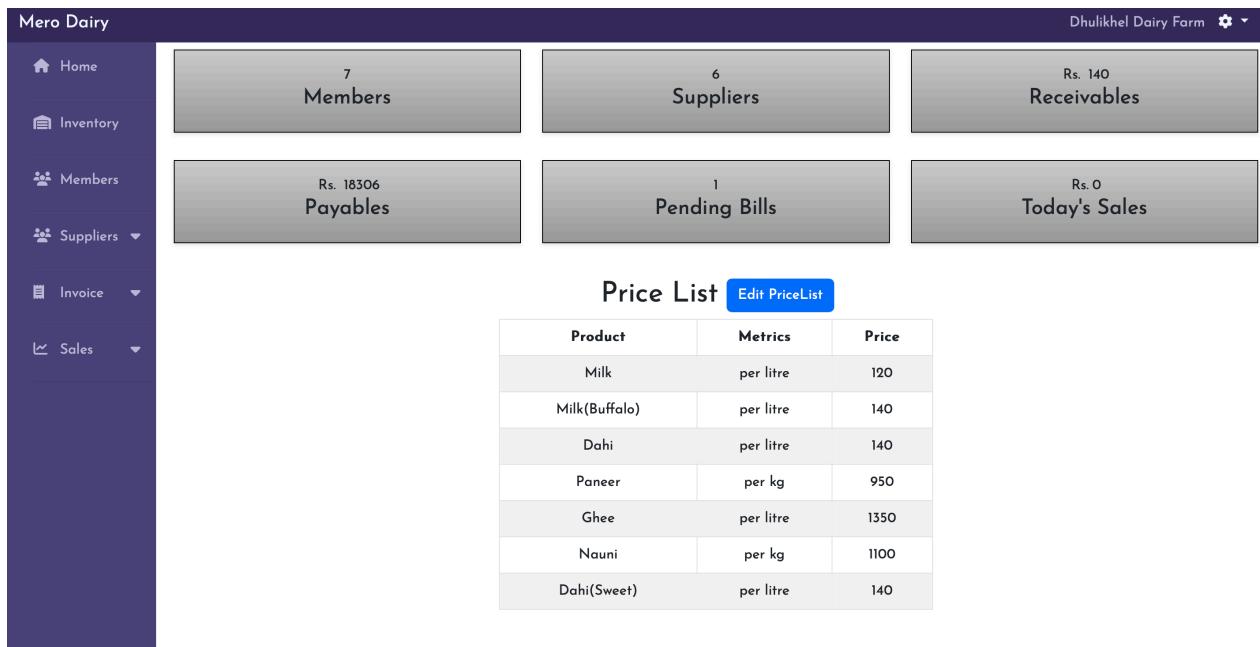


Figure 7: Admin Dashboard

2.5.2 Tools and Technology Used

2.5.2.1 Frontend Tools

Frontend tools are the tools that are used to develop the UI part or the interface for a system. HTML, CSS, Bootstrap CSS, Font Awesome, Google Fonts, JavaScript, and jQuery are used for developing the Frontend of the project.

2.5.2.2 Backend Tools

PHP has been used as the backend language for all the server-side processing and dealing with the dynamic aspects, whereas MySQL is used as the primary DBMS.

2.5.2.3 Other Tools

For system and architecture designs, draw.io and edrawmax.com were used. VsCode was the IDE of choice, and XAMPP was used for running MySQL database server and Apache web server. Mac Pages was used for the documentation of the project.

2.6 Software Testing

Software testing is important for evaluating a software product to find errors, gaps, or missing requirements and to ensure that it meets the specified requirements. It encompasses various steps, techniques, and methodologies to systematically verify and validate software systems.

2.6.1 Unit Testing

Unit testing is the process of testing an individual piece of code like classes, functions, interfaces, and procedures.

The general unit test cases that have been conducted in the system are as follows:

- Ensure that admin can login to the system with correct email and password
- Ensure that admin can add new items, edit the price of items, and remove items
- Add, edit and remove members and suppliers
- Record supply of raw milk and sale of items

2.6.2 Integration Testing

Integration testing is where individual components or modules of a system are combined and tested as a group to ensure their seamless interaction and functionality as a whole. The following test cases have been conducted during integrated testing:

- Ensure admin dashboard is working properly
- Ensure supply of raw milk as well as sale of items updates the inventory
- Ensure unpaid bills amount are added to receivables in members

2.6.3 System Testing

System testing involves evaluation of the entire integrated software application as a comprehensive system. This testing stage aims to verify whether the software meets the specified requirements and functions correctly in its entirety. It checks whether all the components are jointly working or not, whether data is getting transferred over various interfaces or not, etc. System testing encompasses testing of the emergent system properties including performance, security, reliability, compatibility etc. and is important for acceptance by customers.

Table 1: *Test Case for Admin Login*

Project Name:	Dairy Management System			
Module Name:	Login module			
Test Title:	Check admin login functionality			
Description:	The system authenticates the given email and password to decide if the admin can login.			
Precondition:	The UI to enter the credentials should be available and working properly.			
Test Steps:	a) Visit login URL b) Enter email address and password in respective fields c) Press “Enter” or Click on “Sign In” button			
E x p e c t e d Result:	The admin can view the dashboard after successful login but, if login credentials are incorrect then error message should be displayed in the login form.			
Test Case ID	Test Scenario	Test Data	Expected Result	Status
LOGIN01	Check with unregistered email	Email : <u>ddf@gmail.com</u> Password: “Dairy123”	Display “Wrong Email!!”	Pass
LOGIN02	Check with valid email but invalid password	Email : “dhulikhel@ <u>gm</u> ail.com” Password: “df123”	Display “Wrong Password!!”	Pass
LOGIN003	Check with empty input field	Email: Password:	Display “Please fill out this field”	Pass
LOGIN004	Check with valid email address and password	Email : “dhulikhel@ <u>gm</u> ail.com” Password : “Dhulikhel1”	Redirect to admin home page	Pass

Table 2: *Test Case for Adding a New Item*

Project Name:	Dairy Management System			
Module Name:	Adding item module			
Test Title:	Check the operation of adding new dairy item			
Description:	When new item is added, it should be displayed in the home page inside the priceList table.			
Precondition:	Admin account is properly setup.			
Test Steps:	a) Visit login URL b) Admin login with authorised email and password c) IN home page, click on ‘Edit PriceList’ button aside the priceList table d) Click on ‘add item’ button e) Fill the required fields f) Click on ‘Confirm’ button			
E x p e c t e d Result:	The newly added item must be displayed in the PriceTable in the home page.			
Test Case ID	Test Scenario	Test Data	Expected Result	Status
ADD01	Add a new item	Item: Ice Cream Metric: per .5l Price: 250	Display Ice Cream in PriceList table	Pass
ADD02	Try to add an existing item	Item: Milk Metric: per litre Price: 120	Display “Item already exists”	Pass

Table 3: *Test Case for Recording a Sale*

Project Name:	Dairy Management System			
Module Name:	Sale Recording module			
Test Title:	Check that sales decreases quantity in inventory			
Description:	When a sale is recorded or an item, it should reduce the stated quantity from the inventory table for that item.			
Precondition:	There are items in the inventory.			
Test Steps:	a) Visit login URL b) Admin login with authorised email and password c) Navigate to Record a Sale page d) Fill the sale details e) Click on ‘Record Sale’ button			
E x p e c t e d Result:	The inputted quantity of the item should be subtracted from the inventory.			
Test Case ID	Test Scenario	Test Data	Expected Result	Status
SALE01	Item sale with quantity less than in inventory(inventory for dahi: 50l)	Item : Dahi Quantity: 30	Subtract quantity of dahi in inventory by 30l	Pass
SALE02	Item sale with quantity more than in inventory (inventory for milk: 100l)	Item : Milk Quantity: 120	Display “Quantity exceeds inventory level” Make the “Record Sale” button unclickable	Pass

Chapter 3: Discussion and Conclusion

3.1 Findings

After analysing the pain points of the dairy business arising from the use of traditional methods of record keeping, it was found that using those old techniques have proved to increase the cost of the business. Furthermore, records were more difficult to manage and handle. Therefore, if the business was to use this DMS for recording and managing the processes, it would benefit them to maintain records more effectively and efficiently as well as retrieve when required easily without any hassle.

3.2 Discussion

The system **Mero Dairy** is a web application developed for a small-scaled dairy business. The project is focused on creating a digital system for a dairy business to maintain all the important day-to-day records by replacing the original traditional methods. The project solved the problem that was identified during the requirements gathering phase. The system can perform various operations such as updating the prices of the items, updating the inventory as required, updating suppliers and members details, recording raw milk supply, generating invoices, and recording sales.

3.3 Conclusion

The project **Mero Dairy** has been successfully designed, developed and tested to conform with the specifications and the deliverables. The system has made an attempt to effectively introduce concept of using a software to manage the day-to-day operations of a dairy business. After the evaluation of the system with the organization, it was deemed that the system fulfills all the requirements gathered during the initial requirement engineering phase. The evaluation from the user of the system in the organization showed that the system helps to perform the necessary tasks effectively.

3.4 Future Enhancements

The project is currently quite limited in its scope and provides only the most basic of functionalities required to a dairy business. However, in the future, it is aimed at becoming an all in one system for the complete management of dairy businesses. To elaborate, currently the inventories have to be manually added for the items, as it is not possible to know the output and separation of raw milk into different items. Likewise, complete accounting statements and reports cannot be attained as of yet. Therefore, in future iterations accounting functionalities can be integrated. Moreover, invoice system can be enhanced to directly print an invoice with a compatible sized printer for a more professional look. Similarly, Mechanisms for generating sales reports, and analytics can be added. Nonetheless, the system is at the moment only designed to cater to administrator needs. In the future, the system can be improved to incorporate functionalities for staffs and even customers. On the development side, this whole system has been built without using any framework or libraries, so it will be difficult to scale it and make huge changes. Therefore, it is planned to be reengineered by using some front-end frameworks like React. For backend, Spring Boot and Spring Web MVC framework, or ExpressJS, MongoDB, Docker, and NodeJs is planned to be utilised.

References

- Oragui, D. (2023, April 26). Software Documentation Best Practices. Retrieved from <https://helpjuice.com/blog/software-documentation>
- Salfer, J. (2021, December 16). Keys to a successful family dairy business. Retrieved from <https://extension.umn.edu/dairy-news/keys-successful-family-dairy-business>
- Lucid Software. (2018, August 27). How to make a UML Sequence Diagram. Retrieved from https://www.youtube.com/@lucid_software
- Pandey, S. (2022, December 1). How to Write an Effective Project Report. Retrieved from <https://www.brightwork.com/blog>
- Sommerville, I. (2009). Software Engineering.
- T, N. (2021, October 29). Feasibility Study. Retrieved from Binary Terms: <https://binaryterms.com/feasibility-study.html>
- Walter, T. (2019, June 25). Dairy Management System: Features and Benefits. Retrieved from <https://troy-walter.medium.com/>
- w3schools. (n.d.). W3Schools Free Online Web Tutorials. Retrieved August 25, 2023, from <https://www.w3schools.com/>

Appendices

Appendix A: Screenshots

Annex 1: Profile Page

The screenshot shows the 'Profile' section of the Mero Dairy application. On the left is a dark sidebar menu with options: Home, Inventory, Members, Suppliers, Invoice, and Sales. The main content area has two stacked sections. The top section is titled 'Dairy Details' and contains fields for Dairy Name (Dhulikhel Dairy Farm), Contact (9812982090), Address (Dhulikhel-03), and Email (dhulikheldf@gmail.com). A 'Save' button is at the bottom. The bottom section is titled 'Change Password' and contains fields for Current Password, New Password, and Confirm New Password. A 'Change Password' button is at the bottom.

Annex 2: Inventory Page

The screenshot shows the 'Inventory' page of the Mero Dairy application. The sidebar menu is identical to the profile page. The main content area is titled 'Inventory' and displays a table of items. The table has columns for Id, Item, Quantity, Metric, and Options (with edit and delete icons). The data is as follows:

ID	Item	Quantity	Metric	Options
100	Milk	230	Litre	
103	Paneer	198	Kg	
108	Dahi	290	Litre	
109	Ghee	50	Litre	
110	Milk(Raw)	1414.5	Litre	
112	Nauni	26	Litre	
113	Milk(Buffalo)	30	Litre	
117	Dahi(Sweet)	90	Litre	

Showing 1 to 8 of 8 entries

a

Annex 3: Members Page

The screenshot shows the 'Members' section of the Mero Dairy application. The page title is 'Members'. On the left, there is a sidebar with navigation links: Home, Inventory, Members (selected), Suppliers, Invoice, and Sales. A blue button 'Add Member' is located at the top right of the main content area. Below it is a search bar with placeholder 'Search:' and a dropdown 'Show 10 entries'. The main content area displays a table with the following data:

ID	Name	Address	Contact No.	Receivables	Options
1000	Araniko Hotel	Dhulikhel-09	9869708091	0	
1001	Himalayan Horizon	Dhulikhel-02	9813049959	0	
1013	Yubraj Neupane	Dhulikhel-08, Kavre	9813929283	140	
1015	Dhulikhel Hospital	Dhulikhel 09	9810203345	0	
1016	Dhulikhel Lodge Resort	Dhulikhel-03	9810203049	0	
1019	Ram Upreti	Dhulikhel-09, Kavre	9810203346	0	
1024	Mirabel Resort	Dhulikhel-03	9810223412	3220	

At the bottom, it says 'Showing 1 to 7 of 7 entries' and has navigation buttons 'Previous' and 'Next'.

Annex 4: Supply Page

The screenshot shows the 'Raw Milk Supply Details' section of the Mero Dairy application. The page title is 'Raw Milk Supply Details'. On the left, there is a sidebar with navigation links: Home, Inventory, Members, Suppliers (selected), Invoice, and Sales. A blue button 'Record' is located at the top right of the main content area. Below it is a date field 'Date: 04/10/2023'. The main content area displays a table with the following data:

Supplier	Milk Fat	Quantity(litre)	Cost per Fat	Total Cost(Rs.)
Manohar Waiba	7.50	20.00	16.5	2475.00

b

Annex 5: Supply History of Supplier

The screenshot shows the Mero Dairy software interface. On the left, there's a sidebar with navigation options: Home, Inventory, Members, Suppliers (selected), Invoice, and Sales. A modal window titled "Supply History of Manohar Waiba" is open in the center. It contains a table with columns "Date", "Fat", and "Quantity". The data shows three entries: 2023-09-30 (6.5, 20), 2023-10-01 (6.5, 20), and 2023-10-02 (6.5, 10). Below this table, there's a list of suppliers with columns "Id", "Name", "Address", "Phone", "Category", and "Last Supply Date". The list includes Hari Ram Khanal, Manohar Waiba, Menuka Syangbo, Kaila Tamang, Ramesh Bomjan, and Shyam Bahadur Tamang. At the bottom of the modal, there's a "Close" button. The background shows a grid of payment details with columns "Payables" and "Options".

Annex 6: Create Invoice Page

The screenshot shows the Mero Dairy software interface for creating an invoice. The left sidebar has the same navigation options as Annex 5. The main area starts with "Customer Information" fields: Status (Pending), Date (04/10/2023), QR Select Existing Customer, Customer Name (Himalayan Horizon), Address (Dhulikhel-02), and Phone (9813049959). Below this is a table for entering items, with columns "Item", "Quantity", "Unit Price(Rs.)", and "Sub-Total". Three items are listed: Dahi (8 units at 140 Rs., Sub-Total 1120.00), Milk (10 units at 120 Rs., Sub-Total 1200.00), and Paneer (5 units at 950 Rs., Sub-Total 4750.00). At the bottom, there are fields for "Sub Total" (7070.00), "Discount" (70), and "Total" (7000.00). A green "Create Invoice" button is at the bottom right.

Annex 7: Invoice Record Page

ID	Issued Date	Customer	Subtotal	Discount	Total	Status	Options
226	2023-10-02	Dhulikhel Hospital	6550	500	6050	paid	P D
250	2023-10-03	Yubraj Neupane	140	0	140	pending	P U D
251	2023-10-04	Mirabel Resort	3220	0	3220	pending	P U D

Annex 8: Invoice Example

Dhulikhel Dairy Farm
 [Dhulikhel Buspark]
 [Dhulikhel-03, Kavre]
 Phone [+977-9823281271]

INVOICE

Date 2023-10-04
 Invoice # 251

Bill to

Mirabel Resort
 Dhulikhel-03
 9810223412

Items	Quantity	Unit Price	Amount
Milk(Buffalo)	15	140	2,100
Dahi(Sweet)	8	140	1,120
	Subtotal	Rs.	3,220
	Discount	Rs.	0
	Total	Rs.	3,220

Annex 9: Sale Record Page

ID	Sale Date	Items	Quantity	Price	Subtotal
9	2023-10-02	Dahi	10	140	1400
10	2023-10-03	Milk(Buffalo)	10	140	1400
11	2023-10-04	Paneer	2	950	1900

Annex 10: SQL Tables

Table	Action	Rows	Type	Collation	Size	Overhead
Inventories		8	InnoDB	utf8mb4_general_ci	16.0 KiB	-
InvoiceItem		8	InnoDB	utf8mb4_general_ci	16.0 KiB	-
Invoices		4	InnoDB	utf8mb4_general_ci	16.0 KiB	-
Members		7	InnoDB	utf8mb4_general_ci	16.0 KiB	-
PriceList		7	InnoDB	utf8mb4_general_ci	16.0 KiB	-
Sales		5	InnoDB	utf8mb4_general_ci	32.0 KiB	-
Suppliers		6	InnoDB	utf8mb4_general_ci	16.0 KiB	-
SupplyHistory		10	InnoDB	utf8mb4_general_ci	16.0 KiB	-
User		1	InnoDB	utf8mb4_general_ci	16.0 KiB	-
9 tables	Sum				56 InnoDB utf8mb4_general_ci 160.0 KiB	0 B

Appendix B: Sample Codes

Recording invoice details in Database

```
<?php
include('../connect.php');
if (isset($_POST['createInvoice'])) {
    $sql = "SELECT invoiceID from Invoices ORDER BY invoiceID desc LIMIT 1";
    $result = mysqli_query($con,$sql);
    $row = mysqli_fetch_assoc($result);
    $invoiceIdItem = $row['invoiceID'] + 1;
    $stmt = $con->prepare("INSERT INTO InvoiceItem(invoiceID, Item, Quantity,
Price, SubTotal) VALUES (?, ?, ?, ?,?)");
    $rowCount = count($_POST['item']);
    for($i=0;$i < $rowCount; $i++){
        $item = $_POST['item'][$i];
        $quantity = (float)$_POST['quantity'][$i];
        $price = $_POST['Price'][$i];
        $sub_total = (float)$_POST['sub-total'][$i];
        $stmt->bind_param('isid',$invoiceIdItem,$item,$quantity,$price,$sub_total);
        if(!$stmt->execute()){ die('Failed to execute Statement:' . $stmt->error); }}
    $customer = $_POST['customer_name'];
    $date = $_POST['date'];
    $invoice_subtotal = $_POST['invoice_subtotal'];
    $invoice_discount = $_POST['invoice_discount'];
    $invoice_total = $_POST['invoice_total'];
    $status = $_POST['status'];
    $address = $_POST['customer_address'];
    $contact = $_POST['customer_phone'];
    if($status === "pending"){
        $recsql = "UPDATE Members SET Receivables = Receivables + '$invoice_total'
WHERE Name = '$customer'";
    }
}
```

```

mysqli_query($con, $recsql);
$pendingAmt = $invoice_total;
} else{
    $pendingAmt = 0;
}
$sql1 = "INSERT INTO `Invoices` ( `Date`, `Customer`, `SubTotal`, `Discount`,
`Total`, `Status`, `PendingAmount`,`Address`,`Contact`) VALUES ('$date',
'$customer', '$invoice_subtotal', '$invoice_discount', '$invoice_total', '$status',
'$pendingAmt','$address','$contact')";
$res = mysqli_query($con, $sql1);
if($res){
    echo "<script>window.open('generatePdf.php', '_blank');</script>";
    echo '<script>window.location.href = "billing.php";</script>';
} else{ echo "Error"; }
$stmt->close();
mysqli_close($con);
} ?>

```

Delete a Member

```

<?php
    include('../connect.php');
    if(isset($_POST['delete'])){
        $id = $_POST['deleteid'];
        $sql = "DELETE From Members WHERE MemberID='$id' ";
        $res = mysqli_query($con, $sql);
        if($res){   header('location:members.php');
        } else{
            echo 'Error: ' . mysqli_error($con);
        }
    }
?>

```