# A Web-Based Dairy Farm Management System

**Amanda Odawa**

**115187**

**Munge Kirubi**

**122882**

**An Information System project documentation submitted to Strathmore University in partial fulfilment of the requirements for the award of the Bachelor of Business Information Technology at Strathmore University**

**School of Computing and Engineering Sciences**

**Strathmore University**

**Nairobi, Kenya**

**October 2021**

**Declaration and Approval**

We,

Student Name: Amanda Odawa

Student Number: 115187

Sign: _____ Date: _____

Student Name: Munge Kirubi

Student Number: 122882

Sign: _____ Date: _____

declare that this proposal is based on our own work under the supervision of our supervisor:

Supervisor Name: _____

Sign: _____ Date: _____

**Abstract**

Dairy farming is an important contributor toward Kenya's economy. This makes it a rewarding investment. Milk production, cow monitoring, and record keeping are among the activities carried out in a dairy farm. It is therefore important for dairy farms to be managed properly so that these activities are cohesive and run smoothly. A large number of dairy farms in Kenya employ the use of manual hand-written systems to record data. This method of data management is tedious, prone to a lot of errors and does not offer data security. The use of a digital web-based dairy farm management system has significantly decreased the forementioned challenges. This system utilizes the Object Oriented Analysis Design and iterative waterfall methodology. The IDE that was used for implementation of this system is Visual Studio Code because it has plug-ins that made coding run smoothly and efficiency. HTML, CSS and JavaScript were used to make the front-end framework because they are reliable for the creation of web-based systems. JavaScript and PHP were used to create the back-end framework because they are highly productive and robust languages. The system utilizes MySQL as the database due to its secure nature.

**Table of Contents**

Declaration and Approval..... ii

Abstract....iii
List of figures....vii
List of tables....viii
List of abbreviations....ix
CHAPTER 1: Introduction....1
1.1 Background....1
1.2 Problem Statement....1
1.3 Aim....2
1.4 Specific objectives....2
1.5 Justification....2
1.6 Scope and limitations....2
CHAPTER 2: Literature Review....3
2.1 Introduction....3
2.2 Dairy Farming in Kenya....3
2.2.1 Role of Dairy Farming in Kenya....3
2.2.2 History and Evolution of Dairy Farming in Kenya....3
2.3 Improvements to Running a Dairy farm....5
2.3.2 Existing Systems....5
2.3.3 Challenges of the Existing Systems....8
2.4 Gaps in the Existing Systems....9
2.5 Conceptual Framework....9
CHAPTER 3: Methodology....11
3.1 Introduction....11
3.2 Research approach....11
3.3 Methodology....11
3.3 Methodology....11
3.3.1 Requirements Analysis....12

3.3.2 System design....12
3.3.3 Implementation.... 13
3.3.4 Testing.... 13
3.4 Deliverables....13
3.4.1 System Modules....13
CHAPTER 4: System Analysis and Design....15
4.1 Introduction....15
4.2 Requirement specification....15
4.2.1 Functional requirements....15
4.2.2 Non-functional requirements.... 16
4.3 System Architecture....16
4.4 Design Diagrams....17
4.4.1 Use case diagram....17
4.4.2 UML Class Diagram.... 18
4.4.3 Sequence Diagram.... 19
4.4.4 GUI Diagram(mock-ups).... 20
4.4.5 Database Schema.... 23
CHAPTER 5: System Development and Testing....24
5.1 Introduction....24
5.2 Components of the System....24
5.2.1 Input Components....24
5.2.2 Output Components.... 24
5.2.3 Processing Components....24
5.2.4 Storage Components.... 24
5.3 Implementation of the system....25
5.4 Testing of the System....25
5.4.1 Authentication module....25

5.4.2 User Module....27
CHAPTER 6: Conclusion and Recommendation....29
6.1 Introduction....29
6.2 Conclusion....29
6.3 Recommendation....29
6.4 Future Works....29
References....30
Appendix....32
Gantt Chart....32

**List of figures**

Figure 2. 1 Nedap livestock management network system 6

Figure 2. 2 MilkingCloud livestock management system repots summary 7

Figure 2. 3MilkingCloud sample data entry page 7

Figure 2. 4 Afimilk farm management system dashboard 8

Figure 2. 5Conceptual framework 10

Figure 3 1: Iterative Waterfall Model 12

Figure 4 1: System Architecture 17

Figure 4 2: Use case diagram 18

Figure 4 3: UML class diagram 19

Figure 4 4: Sequence diagram 20

Figure 4 5: User create account page 20

Figure 4 6: System log in page 21

Figure 4 7: Report on the staff members 21

Figure 4 8: Feed monitoring page 22

Figure 4 9: Database Schema for Dairy Farm Management System 23

**List of tables**

Table 4 1: Functional requirements 15

Table 5 1: Authentication Module Results 25

Table 5 2: User Module Results 26

## List of abbreviations

| | |
|---|---|
| CSS | Cascading Style Sheets |
| FAO | Food and Agriculture Organization of United Nations |
| GDP | Gross Domestic Product |
| HTML | Hyper Text Markup Language |
| IT | Information Technology |
| KCC | Kenya Co-operative Creameries |
| KDB | Kenya Dairy Board |
| KNAIS | Kenya National Artificial Insemination Service |
| OOAD | Object Oriented Analysis Design |
| p.a | per annum |
| PC | Personal Computer |
| PHP | Personal Home Page (Hypertext Pre-processor) |

# CHAPTER 1: Introduction

## 1.1 Background

Dairy farming is a field of agriculture that deals with the breeding, rearing, and use of dairy animals, especially cows, to produce milk and the many dairy products that come from it. (Webb, 2019)

Kenya's dairy sector is a major contributor towards the nation's Gross Domestic Product (GDP). This makes dairy farms an important part of its economy. However, there are several opportunities to improve individual dairy farms and maximize on revenue. (State Department of Livestock, 2013)

There are a variety of processes that run in a dairy farm. These include milk production, dairy product processing, management and care of livestock, customer services, human resource management, record keeping, finance services and pasture management. Some of the challenges experienced in dairy farms include poor management systems, poor security and mismanagement of funds. (Editor, 2020) This project seeked to improve the management systems used.

A majority of dairy farms in Kenya opt to use a manual paper-based record keeping system. Despite its practicality, implementing a digital system has made data management more effective and efficient. This makes it easier and faster for staff members to keep track of and manage animals. The system has increased correctness of data and make retrieval of data easier, which is beneficial to the management and investors. The system also offers enhanced data security.

## 1.2 Problem Statement

A majority of Kenyan dairy farms implement manual management of data. This method of data recording is functional, but inefficient. Staff members use a lot of their time just to record, file and review data. This results in wastage of time which could otherwise be used to increase the farm's productivity. Accessing this data is a tedious process as one has to manually search through a large number of physical files to find a specific record. Furthermore, data stored in such a manner is prone to damage or misplacement. This makes it

difficult for stakeholders and managers to make well informed and appropriate decisions concerning the farm.

### 1.3 Aim

To create a web-based dairy management system to help in efficient record management.

### 1.4 Specific objectives

i. To examine the evolution of dairy farming practice in Kenya.
ii. To examine existing systems used in management of dairy farms.
iii. To design the dairy farm management system.
iv. To develop the dairy farm management system.
v. To test the dairy farm management system.

### 1.5 Justification

Manual management of data in dairy farms is time consuming, insecure and not easily accessible. This web-based system has not only more data security, but it has also makes it easier for management to access the required information at any time and any place as long as they have internet access. This has also reduced time spent on data recording and resulted in an overall increase in production time. Data security has also been enhanced.

### 1.6 Scope and limitations

The implemented system is web-based. The system was completed in three months' time. It is not used to record any data apart from that of dairy cattle. It records sales of milk from the dairy farm and offers a report on the same. However, lacks a finance module, or a record system for acquired and stored animal feed.

# CHAPTER 2: Literature Review

## 2.1 Introduction

This chapter discusses the existing dairy farm management systems mentioning how each one works and gaps in these systems. It discusses the challenges faced by different stakeholders in dairy farms.

It elaborates on dairy farming as a major contributor to Kenya's economy and its evolution over the years, from the time it was practiced in small-scale until present day. The chapter looks at the period when IT was implemented in dairy farming and how this affected the industry. Three different dairy farm management systems are discussed in the chapter along with their challenges and how these challenges can be minimized.

## 2.2 Dairy Farming in Kenya

### 2.2.1 Role of Dairy Farming in Kenya

Kenya's dairy farming subsector is of high significance to the government as it falls within one of the Big Four Agenda items: food and nutrition security, and manufacturing. It contributes to approximately 8% of the GDP p.a. (Editor, 2020) Apart from revenue, dairy farming is a source of employment. This applies to various fields in the dairy sector including milk production, milk processing and the governing sector (KDB). Dairy farming is also important to the rest of the country's residents. By locally producing milk and high-quality dairy products, Kenyan residents can enjoy the nutritional benefits of said products at a relatively affordable price. (Muriuki, 2011)

### 2.2.2 History and Evolution of Dairy Farming in Kenya

Dairy farming has been carried out in Kenya since the pre-colonial period. However, it was only practiced in small scale as a part of people's diet. Commercial dairy farming was introduced to Kenya in 1900 by colonialists. During late 1960s the use of technology was implemented, which revolutionized dairy farming in Kenya. (State Department of Livestock, 2013)

#### 2.2.2.1 The Colonial Period (1900-1962)

Colonialists imported animals for breeding from South Africa, England and Australia. These were then interbred with local dairy breeds of cattle to boost milk production and reduce chances of diseases. In 1925, KCC was formed to process and market milk from the colonial settlers. Kenyans were not allowed to take part in commercial dairy farming until 1954 when Swynnerton Plan allocated them a quota. In 1951, KDB was established to enforce regulations in milk marketing. (Muriuki, 2011), (State Department of Livestock, 2013)

#### 2.2.2.2 Post-Independence Period (1963-2002)

After independence, policies were put in place with the main focus being inclusivity of indigenous Kenyans in commercial dairy farming. The government took an initiative to develop dairy farming by implementing the policies that were set in place by colonial dairy farmers. These included tick control, veterinary services and artificial insemination. This led to the development of KNAIS in 1965. The use of technology especially in milk processing plants like KCC began during this period. Unfortunately, the policy environment was gradually filled with powerful individuals who were corrupt and exercised poor judgement in decision making. This led to a decline in the industry's economic state, regardless of the new technology they used. (Muriuki, 2011), (State Department of Livestock, 2013)

#### 2.2.2.3 Dairy Market Liberalization Period (2003 onwards)

During this period, the administration focused on correcting the errors of the previous administrations. They succeed in reviving the industry's economic state with the production of milk increasing exponentially. (Muriuki, 2011), (State Department of Livestock, 2013)During this period, several individuals and companies also came up with various tools, technologies and techniques to improve dairy farming. Some of these include digital animal trackers, automated feeding systems and milking machinery.

## 2.3 Improvements to Running a Dairy farm

### 2.3.1 Challenges Experienced by Stakeholders

A majority of small-scale dairy farmers lack management skills resulting in mismanagement of resources and the staff. This may lead to conflicts between the staff and dairy farmer which

will cause a decrease in productivity. Mismanagement of resources results in wastage of fund, leading to a decrease in profits for the dairy farm. A decrease in profits in several dairy farms results in an overall decrease in revenue to the government. This causes a decline in the nation's economy. Staff in dairy farms perform various tasks within a day. Apart from feeding, milking and watering the animals, they are also expected to write down daily reports. Undertaking these activities is tedious and may lead to a decrease in productivity due to demotivation. (Muriuki, 2011)

### 2.3.2 Existing Systems

#### 2.3.2.1 Nedap Livestock Management System

Nepad Livestock Management is a system that is compatible with mobile devices, desktops and can also be used on the web. It offers five solutions to dairy farmers. The first is Nepad CowControl. Its features include heat detection, health monitoring, herd performance trends, cow locating and augmented reality. Nepad CowControl helps to maximize the farm's productivity levels as it keeps track of the animals throughout the day and night. This ensures that the records kept comprehensive and precise which ensures that resources are maximized without waste. The other solutions that Nepad offers are Electronic Concentrate feeding, Sorting and Routing, Milk Yield Recording and Identification (Tracking ISO). These solutions also increase the farm's performance by reducing manual labour. The solutions can be used individually or integrated into a single control system. The entire cycle of the animal can be monitored and managed using this system. (Nedap Livestock Management, n.d.)

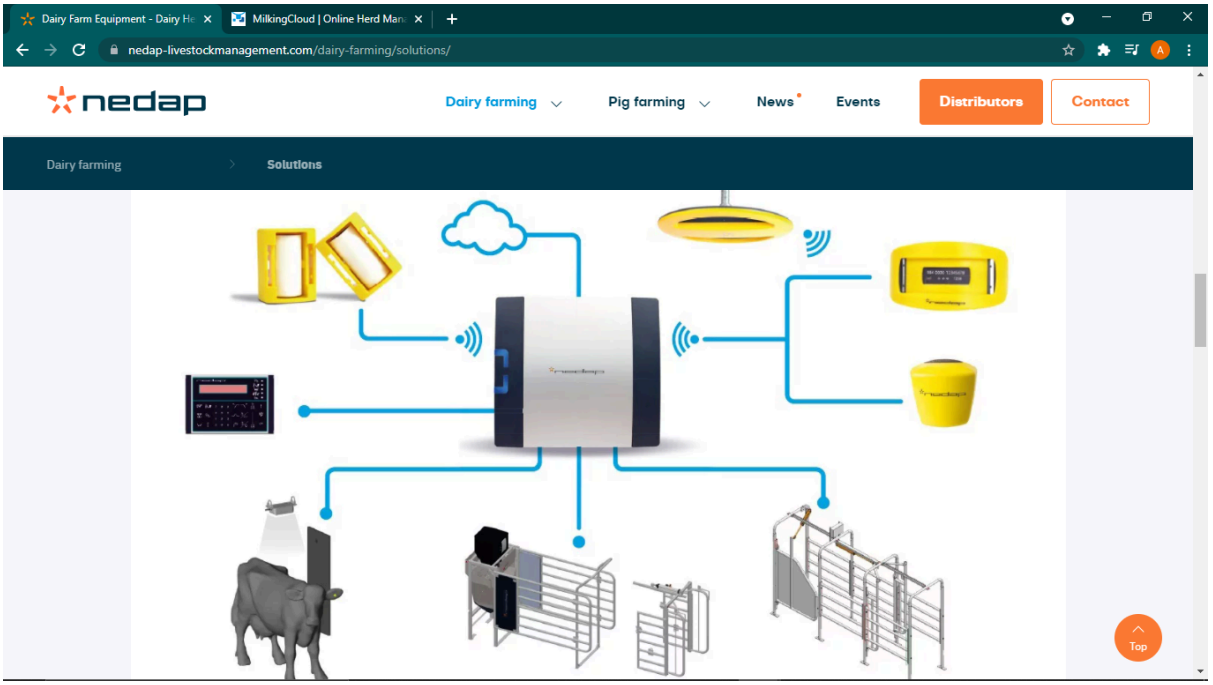



*Figure 2. 1 Nedap livestock management network system*

### 2.3.2.2 MilkingCloud Herd Management Software

This software helps dairy farmers to keep track of their dairy farms via herd management, insemination tracking, cost management, milk yield, feeding management, and examination, vaccination and treatment. The system can be accessed via the web or downloaded into a PC, desktop or mobile phone. First, a user is prompted to create an account by inputting their e-mail, full name, herd count and a new password. A person can then access a demo file or create their own file. There is a free seven-day trial period before a person is asked to purchase the system. This option is attractive to small scale dairy farmers as they are able to try out the system before purchase, therefore they do not incur any losses. (MilkingCloud, 2019)

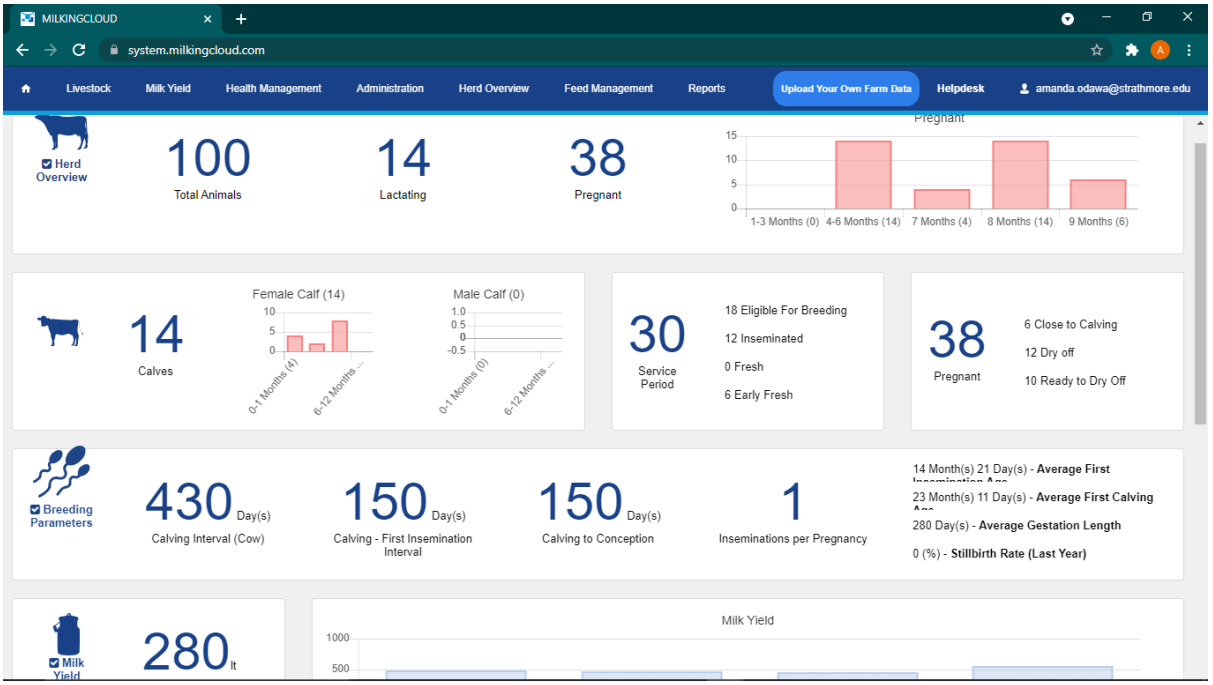



*Figure 2. 2 MilkingCloud livestock management system reports summary*

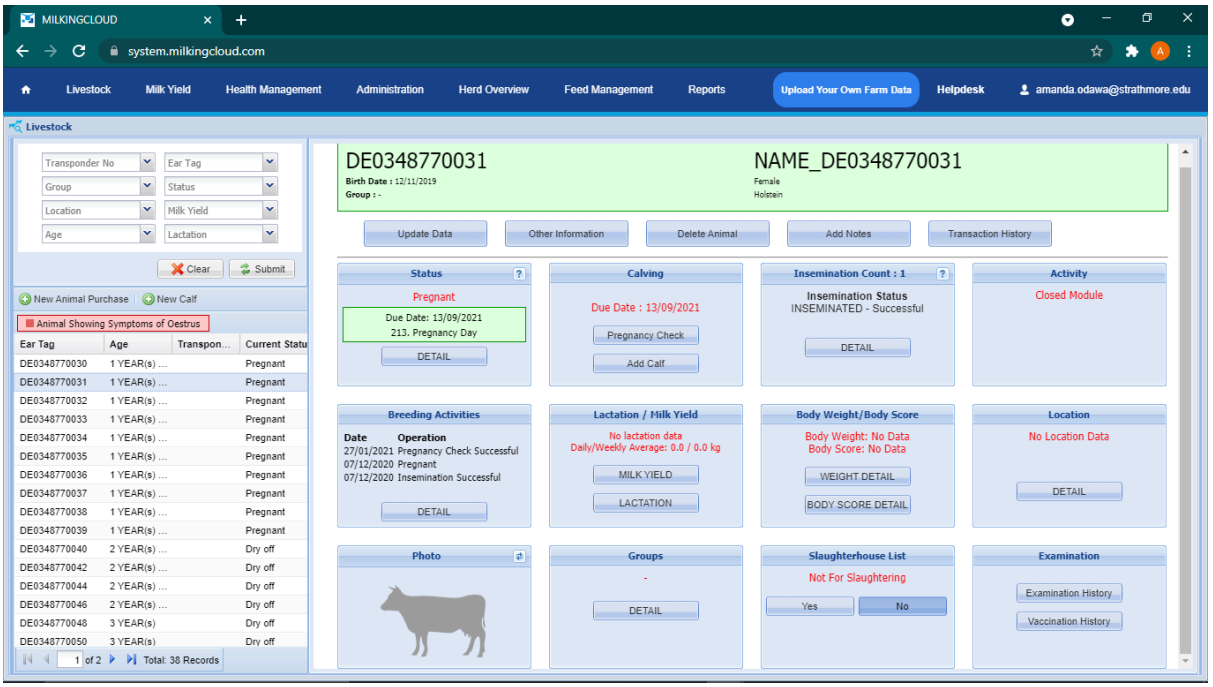
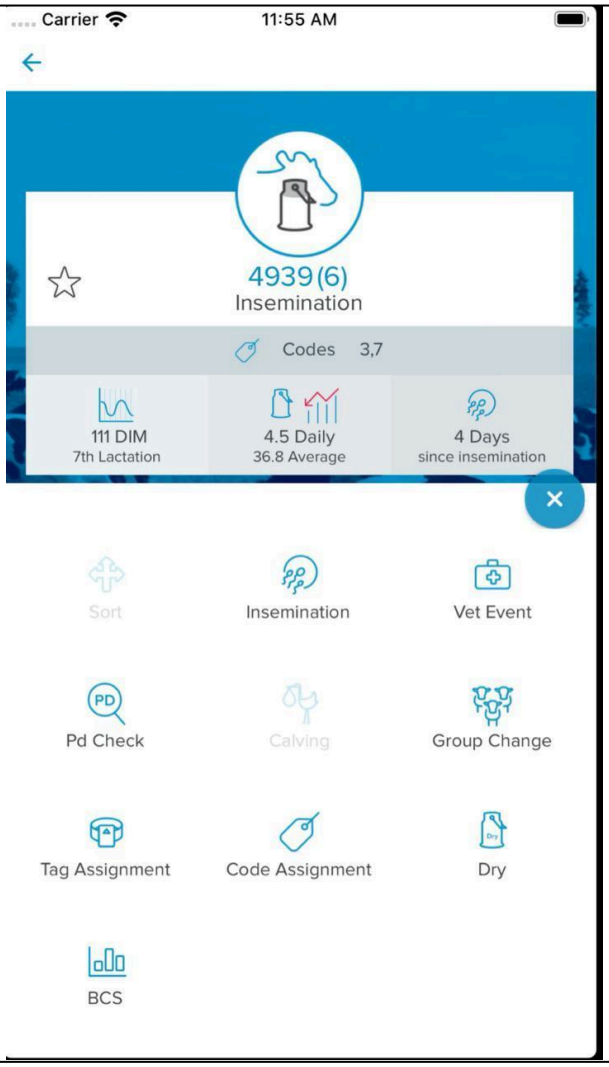



*Figure 2. 3MilkingCloud sample data entry page*

### 2.3.2.3 Afimilk Farm Management System

The Afimilk Farm Management System makes use of a sensor that is put on the animal's leg. This sensor relays data about the animal via the cloud and the farmer receives the data from the web in a PC, desktop or mobile phone. The information that the sensor sends to the farmer includes animal identification, heat detection, timely calving alerts, movement and rest monitoring, and sensitive lameness detection. All these are vital to ensure maximum animal health with minimal manual effort, as the data is automatically collected. This is a




time-saving system which guarantees the farmer that they can monitor their livestock at any time. (Afimilk Ltd., 2020)

### 2.3.3 Challenges of the Existing Systems

Nedap is a complex system which requires a user to have a user manual. Inexperienced users will still have a hard time using the system without prior training. Although it offers a wide range of solutions, they are not integrated within a single system. The system requires a user to combine all components that they will need for management of their dairy farm. This process is tedious and can be stressful for a person with limited IT knowledge. Furthermore, Nedap does not have an option of viewing and testing the system before purchase. This may discourage dairy farmers from using the livestock management system.

The layout of MilkingCloud is not user friendly for people who are using a livestock management system for the first time. Its home page is overcrowded with numerous graphs, figures and reports which may overwhelm its user.

Afimilk offers a sufficient number of services, while maintaining simplicity. However, it is inconvenient for dairy farmers with a small budget for integrating a management system with their farms. This is because the sensors and digital animal trackers are costly.

## 2.4 Gaps in the Existing Systems

Nedap does not provide an integrated interface making the installation process long and tedious. Maintenance of such a system will require a lot of time and money. This can be improved by having a single system which can be downloaded and installed once, making maintenance easier.

MilkingCloud is a bulky system making it unattractive to users. Nedap is complex, which can be off putting to users. Having a simplified system which addresses the most immediate needs of a dairy farm will make it more appealing to potential users.

Afimilk uses individual trackers for livestock which is costly. Instead, a system that does not rely on digital trackers and sensors will be more efficient.

## 2.5 Conceptual Framework

The implemented system prompts the user to input details of the cows, staff, milk production and sales. The system then receives data on daily feed intake, vitamins and medication, veterinary check-ups and records, which are updated regularly.

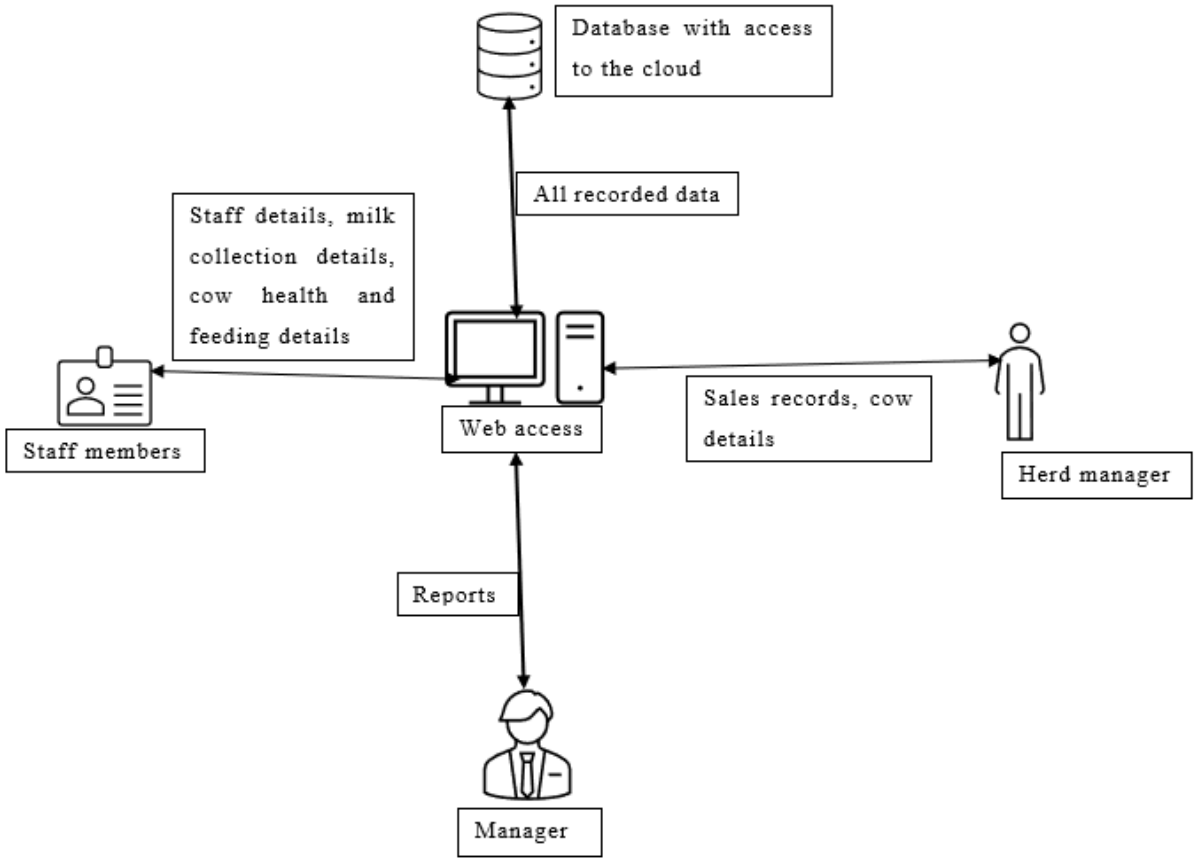



*Figure 2. 5Conceptual framework*

# CHAPTER 3: Methodology

## 3.1 Introduction

The research approach that was used is OOAD as it focuses on implementation of real-world entities to solve problems. Predictive process modelling was applied, specifically the iterative waterfall methodology. This research approach was divided into in four stages: requirements analysis, system design, implementation and testing. The chapter will also discuss the system modules.

## 3.2 Research approach

The research approach used in developing the system is Object Oriented Analysis Design (OOAD). The OOAD solves problems by implementing real-world entities as objects. The research approach focused on breaking down the problem into smaller modules and solving these individual modules, then combining the results to achieve the final system. OOAD made it possible to describe the system as objects and their interactions before coding. This made it easier for its development. Modification of the system also became simplified due to the different modules, as the developers were able to focus only on the areas that needs to be altered without disrupting the entire system. (Pasupathy, Dr. Bhavani, & Nadu, 2015)

## 3.3 Methodology

## 3.3 Methodology

This project employed the waterfall methodology, which made use of predefined stages that had to be completed sequentially. Specific goals and milestones were used to control movement from one phase to the next. The methodology made troubleshooting simple due to the breakdown of the project into smaller modules. (The Economic Times, 2021) Iterative waterfall model provided relevant feedback from each phase to the previous phase which was useful when making changes. (Sayan, 2018)

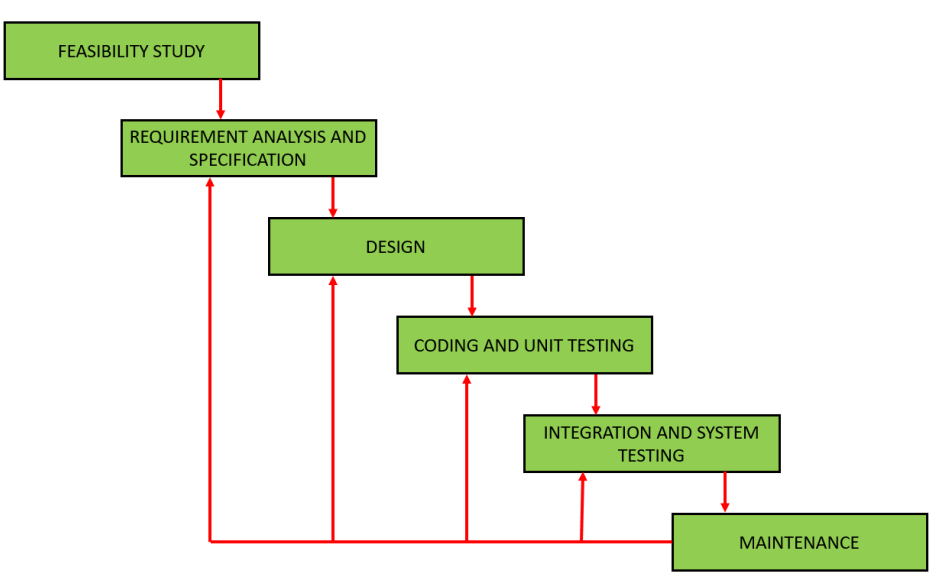



*Figure 3 1: Iterative Waterfall Model*

### 3.3.1 Requirements Analysis

This stage focused on how the functional and non-functional requirements would be gathered. They were collected via research papers, analysis of similar existing models, user experience and carrying out informal interviews with workers at dairy farms.

### 3.3.2 System design

The system design was implemented using analysis and design diagrams. These diagrams promoted a better understanding of the project. They include:

1. Use Case Diagram - Shows the different interactions between actors and the information they access.
2. Class diagram - Describes a system's structure by displaying the classes, attributes of objects and the relationships between them.
3. Sequence diagram – A diagram that describes a specific order on how objects work hand in hand.
4. Database schema - This is a visual representation of the structure of the database in highlighting the primary keys, foreign keys, rows etc.
5. Graphical User Interface design – This is an interface that enables the user to interact with the system by use of icons and pointers.

### 3.3.3 Implementation

The implementation of the web-based system employed the use of HTML, CSS, PHP, and JavaScript. MySQL was used as the database due to its robust and secure nature. The IDE that was used is Visual Studio Code. This source-code editor has features that make writing code easy, and additional plug-ins that made work run smoother and faster.

### 3.3.4 Testing

This was the final phase of the project. System testing was carried out to find errors and failures and improve system reliability. Regression testing was done to check that any changes made did disrupt previously working code. (JavaTpoint, n.d.)

## 3.4 Deliverables

### 3.4.1 System Modules

The system consists of five modules. They include information module, cow monitoring module, milk collection module, sales module and reports module.

#### 3.4.1.1 Information Module

This module is accessed by the managers and system admin. Information concerning cows, staff member, milk collected and milk sales is displayed here for viewing. This module keeps track of all employees and cattle.

#### 3.4.1.2 Cow Monitoring Module

The module contains data that is used to keep track of the cows' health status. It is used to record information about veterinary services and feeding the animals. This data is recorded by the staff and accessed by them when needed for reference.

#### 3.4.1.3 Milk collection module

The users of this module record information concerning daily milk yield by individual cows. This module is also be used for pricing of milk.

#### 3.4.1.4 Sales Module

The herd managers record details about customers and their purchases. The selling price of the milk is also be included here. This module is important for the generation of the reports module.

#### 3.4.1.5 Reports Module

This module is available for access by managers and the system admin. It contains a summary of the farm's sales. This is an important module because it makes it easier for the management to make financial decisions and other adjustments to the daily running of the farm.

# CHAPTER 4: System Analysis and Design

## 4.1 Introduction

The chapter focuses on describing the system's functions and behaviour. It highlights the functional and non-functional requirements of the system. The system's architecture and design diagrams are also discussed in the chapter.

## 4.2 Requirement specification

Software requirement analysis is the process consisting of collection and interpretation of facts and identification of problems. It is a problem-solving technique that specifies the functionalities of a system. This section discusses the different services that the system performs.

### 4.2.1 Functional requirements

Functional requirements are the features of the system that enable users to accomplish tasks. This section mentions the various functions that the system is expected to perform, with regard to different users.

*Table 4.1: Functional requirements*

| USER | FUNCTIONAL REQUIREMENT |
|---|---|
| All users | Create account<br>Log-in<br>Log-out |
| Manager | View sales milk sales report<br>View dashboard<br>View staff info |
| Milker | Input, update and delete milk collection data |
| Feeder | Input, update and delete cow feeding data |
| Herd manager | Add new cows to the system<br>Update and delete cow data<br>Input, update and delete milk sales data<br>Input, update and delete vaccination data |

Apart from these users, the System Admin is able to log-in; log-out; view sales milk sales report; view dashboard; view staff info; input, update and delete milk collection data; input, update and delete cow feeding data; add new cows to the system; update and delete cow data; input, update and delete milk sales data; and input, update and delete vaccination data.

### 4.2.2 Non-functional requirements

Non-Functional requirements refer to specification of how the system should behave. This section focuses on the non-functional requirements of the system.

They include:

1. Scalability - This is the ability of the application to handle a growing number of users and the ease of managing of the application. This system can handle an unlimited number of users without crashing.
2. Data Integrity - This is the accuracy, completeness, and consistency of a system. The database maintains accuracy and consistency of data.
3. Security - This is the ability of an application to secure business data and client user data. The system hashes all user passwords.
4. Performance - Describes how fast a software system responds to user activities under specified workload. The system has the capability to be deployed and be used as a web platform. Use of PHP has ensured a fast system response time.

## 4.3 System Architecture

The system consists of a front-end framework and back-end framework. The user interacts with an electronic device which is connected to the web server. Upon keying in an input, it is sent as a request to the web server. The server retrieves data from the database and file system. This data is sent as a response to the device and viewed as output by the user.

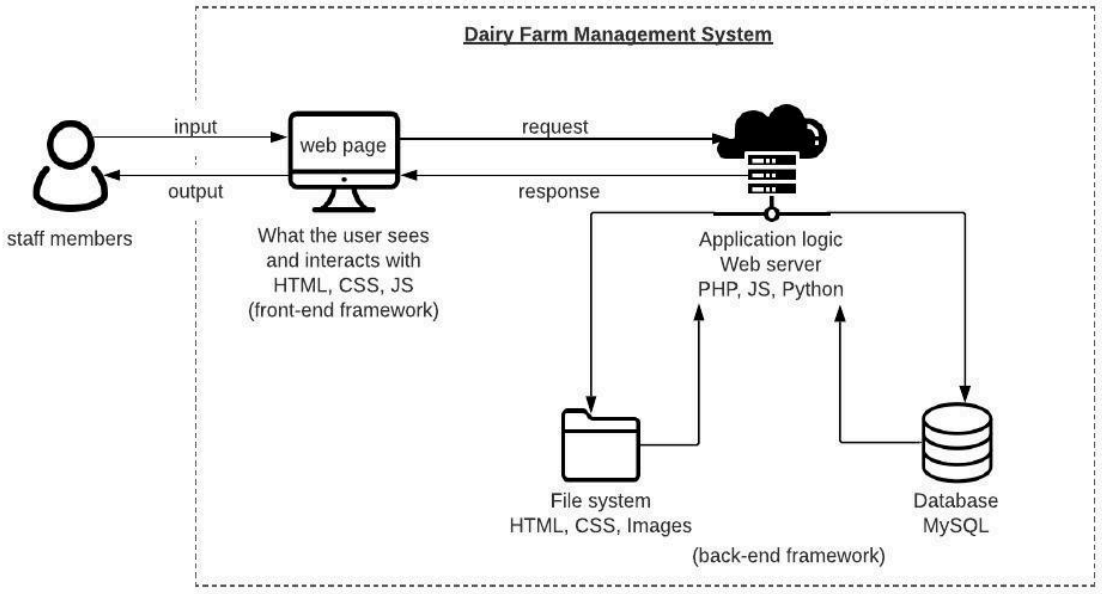



*Figure 4 1: System Architecture*

### 4.4 Design Diagrams

This section includes the design diagrams used in projects that implement the OOAD approach. This is because the project used the forementioned methodology.

#### 4.4.1 Use case diagram

This design diagram illustrates the various functions of the system and the actors who interact with them. All users, apart from the system admin, are eligible to create an account. All users can log in and log out. Feeders manage feeding records and milkers manage milk collection data. The herd manager can register new cows, manage veterinary records and record sales. The general manager and system admin can access milk sales reports. The system admin can also perform functions of the other users.

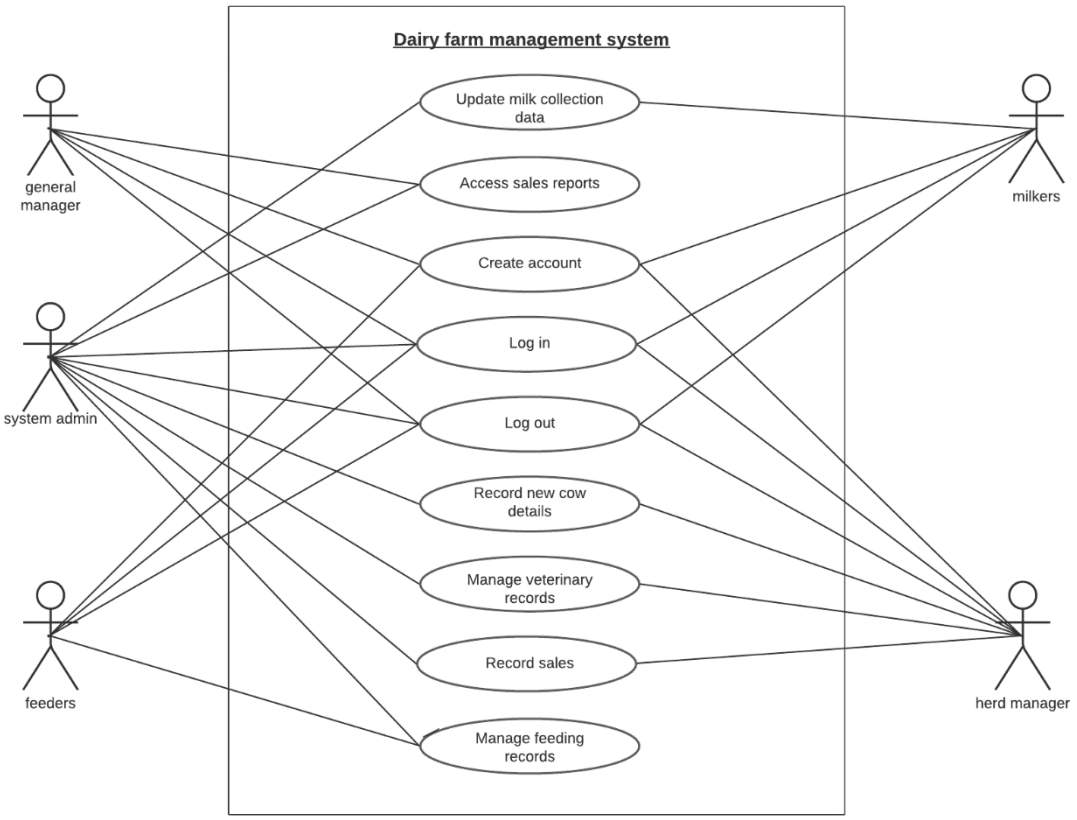



*Figure 4 2: Use case diagram*

### 4.4.2 UML Class Diagram

This diagram shows the various objects that will form the system and interact with it. It also includes attributes of the various classes and operations carried out by some of them. The diagram also includes the multiplicities between related classes.

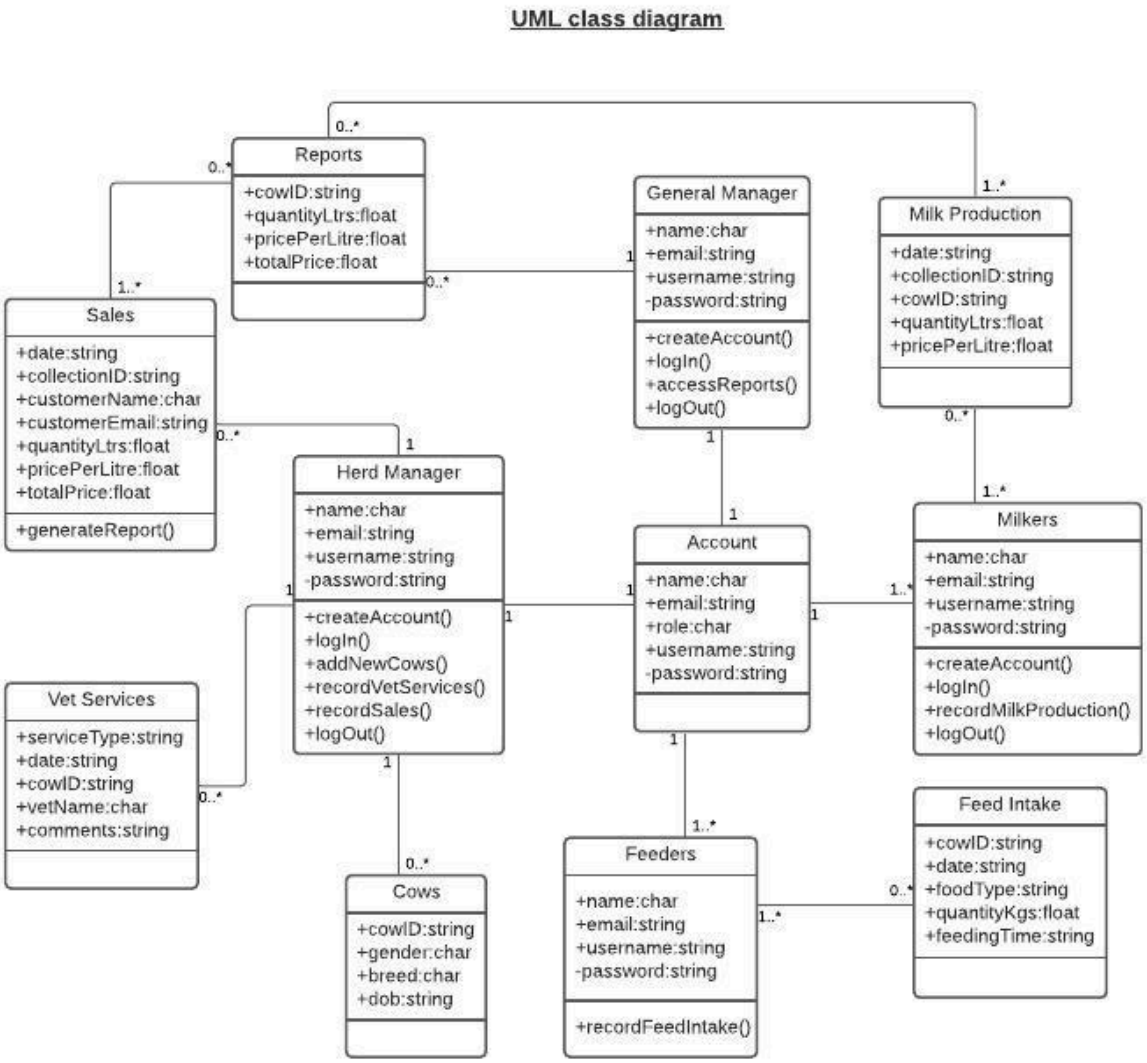



*Figure 4 3: UML class diagram*

### 4.4.3 Sequence Diagram

A sequence diagram shows different processes or objects that occur simultaneously and include the messages exchanged between. The diagram shows the log in process whereby the different roles of the staff are displayed and distributed. The different staff members perform their different roles to achieve the farm's goals that include managing veterinary records, adding feeds given to the cows and adding amount of milk collected.

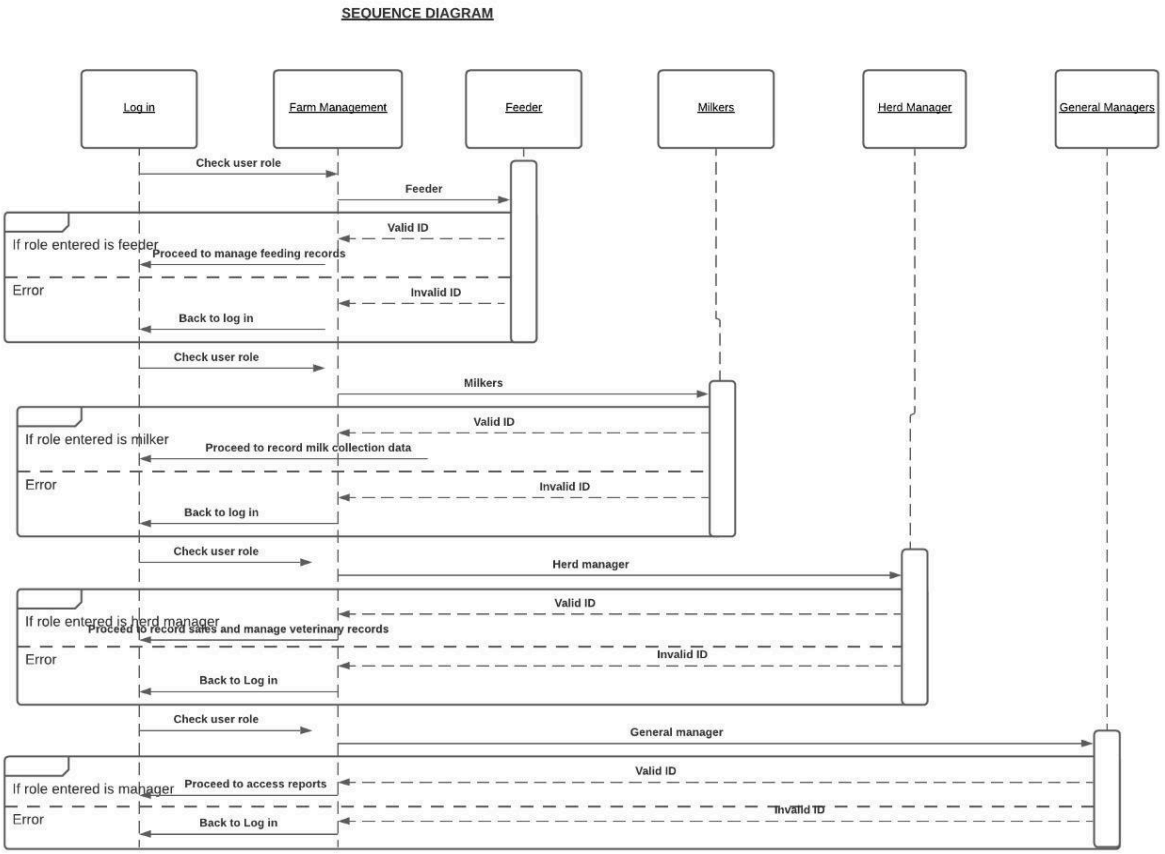



*Figure 4 4: Sequence diagram*

### 4.4.4 GUI Diagrams

The GUI diagrams used were mock-ups. Mock-ups show what the system displays as the user interacts with it. Displayed below are four mock-ups of the system.

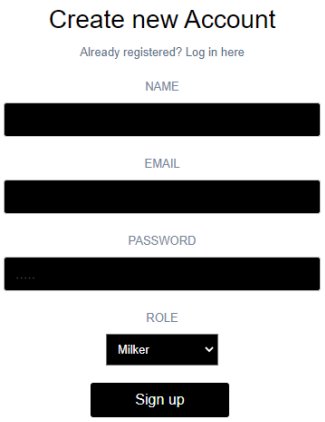



*Figure 4 5: User create account page*

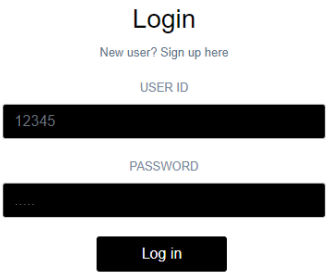



*Figure 4 6: System log in page*

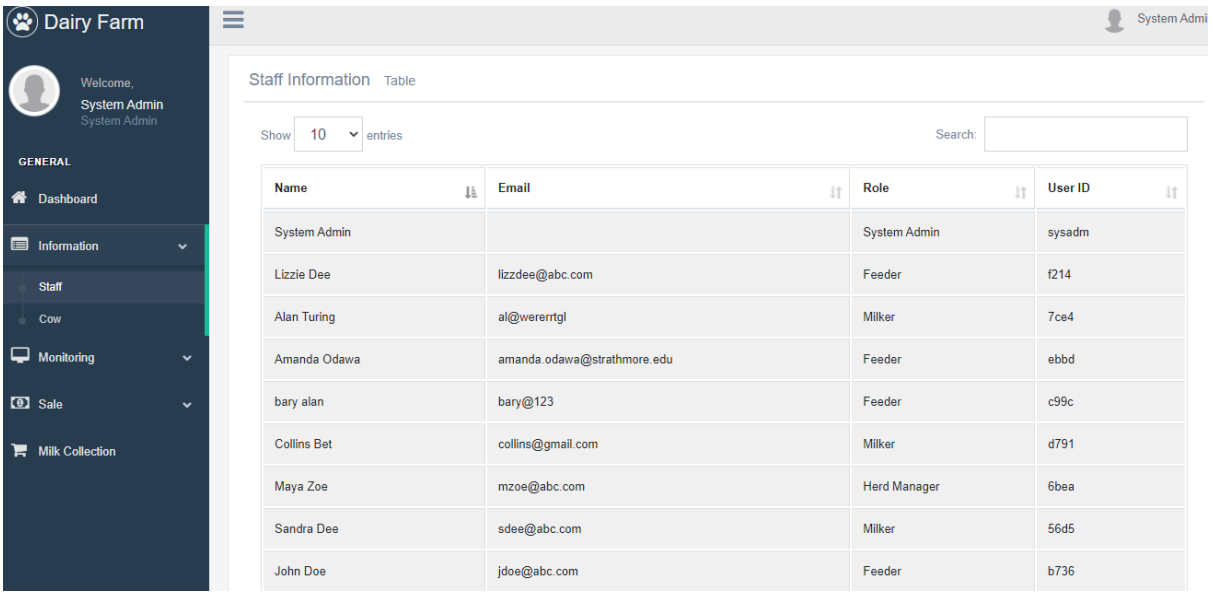



*Figure 4 7: Report on the staff members*

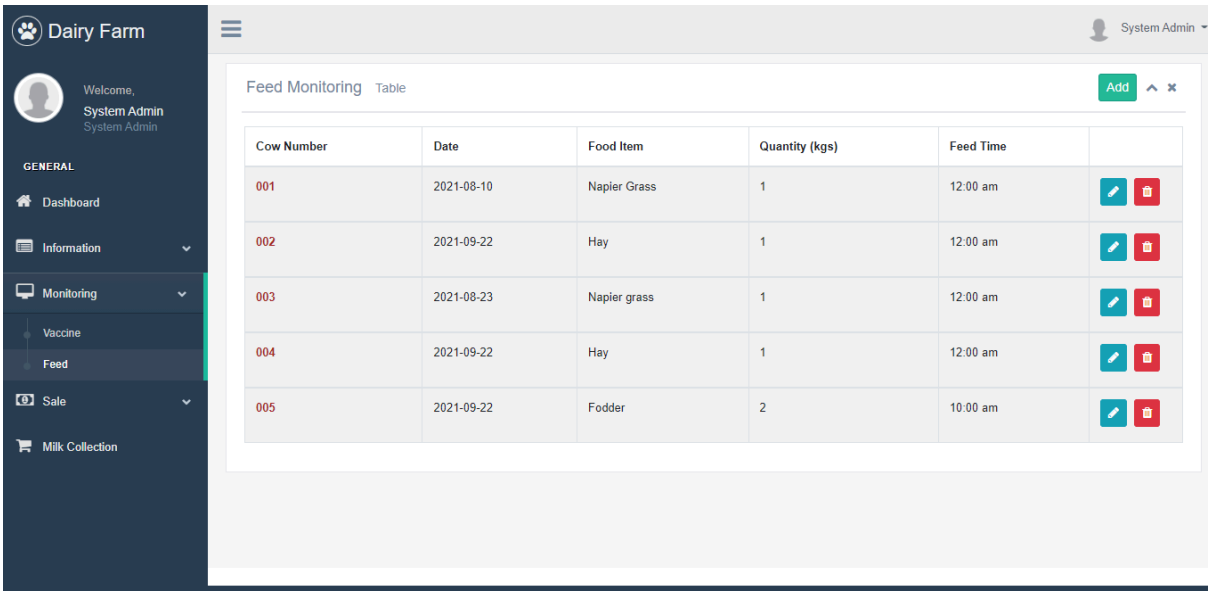



*Figure 4 8: Feed monitoring page*

### 4.4.5 Database Schema

A database schema is a visual represntation of a design that describes the structure of a database. As seen in the diagram, the staff table is linked to the feeders table, milk production table and the cow information table as well. Different staff with different roles perform

different activities. The amount of milk produced and the sales generated are all recorded in the reports table.

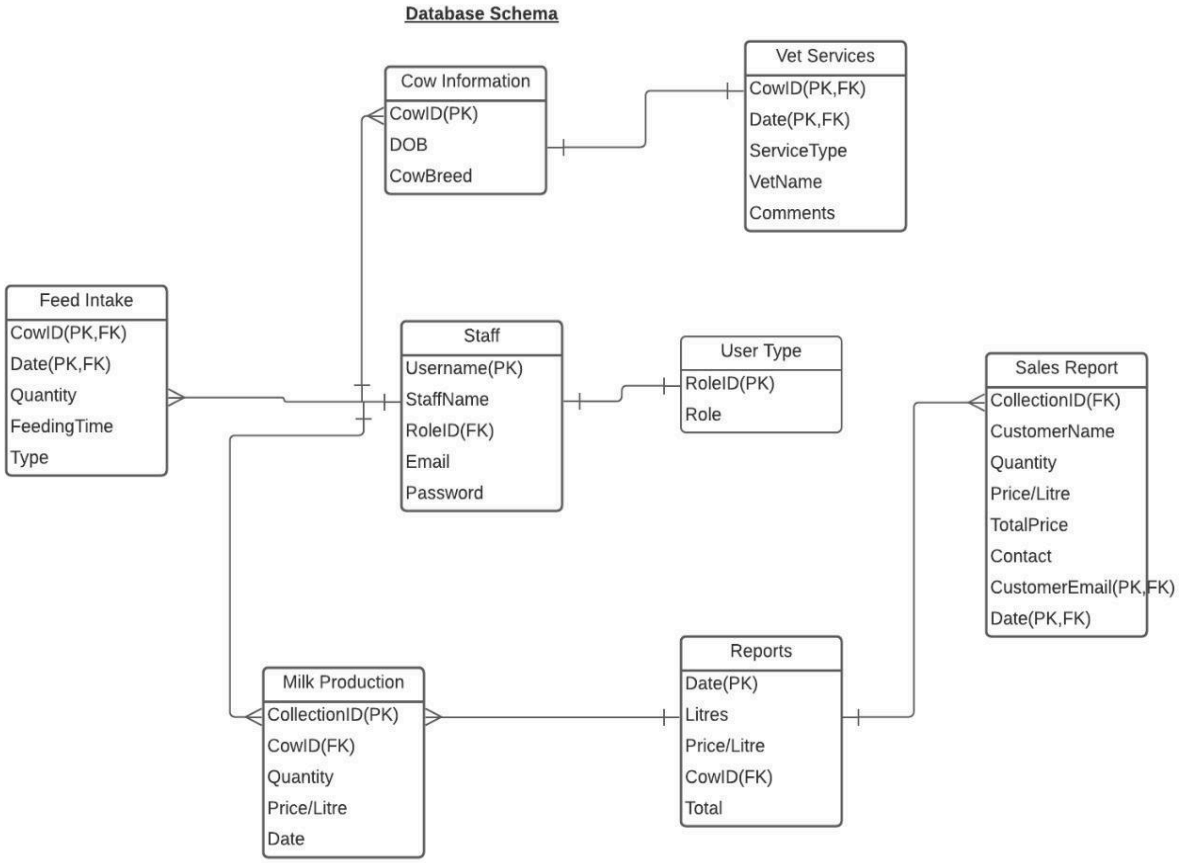



*Figure 4 9: Database Schema for Dairy Farm Management System*

## CHAPTER 5: System Development and Testing

## 5.1 Introduction

This section discusses how the system was tested after its implementation. System implementation is the process of defining how the system should be built, while system testing is validating the functionality of the different modules in the system and ensuring they perform the specified functional and non-functional requirements.

## 5.2 Components of the System

The components of the system include the input, output, processing and storage.

### 5.2.1 Input Components

The input components of this system include the keyboard and the mouse which were used to key in details into the system.

### 5.2.2 Output Components

The output component of this system is the monitor which will be used to display the inputs of the system.

### 5.2.3 Processing Components

The processing components of this system were the operating system, processor and RAM.

### 5.2.4 Storage Components

The storage component of the system is the Hard Disk Drive (HDD) which is the primary storage engine for the system.

## 5.3 Implementation of the system

The system was developed on a Lenovo Laptop with an Intel core i5-8250U processor with a CPU speed of 1.60GHz, RAM of 8GB and a windows 10, 64-bit operating system. The system was developed using Visual Studio Code. PHP and JavaScript were used for the backend development of the system. HTML, CSS, JavaScript and bootstrap were used in the

front-end development of the system. The inputs that were provided by the user were stored on the Xampp server.

### 5.4 Testing of the System

This section displays some of the tests done on the system and their outcomes:

#### 5.4.1 Authentication module

*Table 5 1: Authentication Module Results*

| Test case | Description | Test Data | Expected outcome | Actual Results | Status (Pass/Fail) |
|---|---|---|---|---|---|
| 1 | Verify that a milker will be able to log in with a valid user ID and password | User ID is '97d6' and password is 'kings' | Milker will be directed to the milk collection page and successfully logged in | As expected | Pass |
| 2 | Verify that a feeder will be able to log in with a valid user ID and password | User ID is '6d00' and password is 'caleb' | Feeder will be directed to the feed monitoring page and successfully logged in | As expected | Pass |
| 3 | Verify that a herd manager will be able to log in with a valid user ID and password | User ID is 'dafd' and password is 'prince1' | Herd Manager will be directed to the cow information page and successfully logged in | As expected | Pass |
| 4 | Verify that a manager will be | User ID is 'f9e3' and | Manager will be directed to the | As expected | Pass |

| | | | | | |
|---|---|---|---|---|---|
| | able to log in with a valid user ID and password | password is 'jim' | dashboard page and successfully logged in | | |
| 5 | Verify that a system administrator will be able to log in with a valid user ID and password | User ID is 'sysadm' and password is '#0#1' | System administrator will be directed to the dashboard page and successfully logged in | As expected | Pass |
| 6 | Verify that user cannot log in with an invalid username and valid password | User ID is 'FFFF' and password is 'caleb' | User alerted of wrong log in credentials | As expected | Pass |
| 7 | Verify that user cannot log in with a valid username and invalid password | User ID is '6d00' and password is 'CALEB' | User alerted of wrong log in credentials | As expected | Pass |

### 5.4.2 User Module

*Table 5 2: User Module Results*

| **Test Case** | **Description** | **Test Data** | **Expected Outcome** | **Actual Results** | **Status (Pass/fail)** |
|---|---|---|---|---|---|
| 8 | Check if the milker can add new milking | Milk Data | Cow number, date, liter and price per liter | As expected | Pass |

| | | | | | |
|---|---|---|---|---|---|
| | details, edit and delete | | either added, edited or deleted | | |
| 9 | Check if the feeder can add new feeding details, edit and delete | Feeding Data | Cow number, food item, quantity, date and feeding time either added, edited or deleted | As expected | Pass |
| 10 | Check if the herd manager can add new vaccination, sales and cow details, edit and delete | Vaccination Data, Sales data, Cow Data | Vaccination – Cow number, date, vaccine type either added, edited or deleted<br>Sales data – Collection number, customer name, amount, price per liter, date either added, edited or deleted<br>Cow data – Cow ID, cow breed, date either added, edited or deleted | As expected | Pass |
| 11 | Check if the manager can view milk sales report | Milk Sales report | View data collected on sales of milk | As expected | Pass |
| 12 | Check if the system | | Carry out all user | As expected | Pass |

| | | | | | |
|---|---|---|---|---|---|
| | administrator can carry out all user requirements | | requirements and add new staff to the system | | |

## CHAPTER 6: Conclusion and Recommendation

### 6.1 Introduction

This chapter describes the conclusions reached after the system was developed, as well as the recommendations that can be made regarding its application. It also emphasizes the system's potential for future improvements.

### 6.2 Conclusion

In conclusion the system developed was able to solve the problem identified and met the aim of the project. The problem identified was the manual record keeping of livestock records. The aim was to create a web-based management system to reduce manual labour in farms. The researcher therefore developed a system to solve this problem. The system allows specific users to log in to specific pages and make changes that their credentials allow them. A manager can view the final reports that are inputted. This reduces manual manipulation of data. This project will enable farmers to know what they are doing right and wrong, this will make them be able to make necessary changes that will improve the farm.

### 6.3 Recommendation

The dairy farm management system was tested on an Apache HTTP web server which is reliable, flexible and efficient: It is recommended that the dairy farm management system be deployed on an Internet Information Service for the best performance.

The system was developed in a PHP 7 environment with some backward compatibility to PHP 5.6. The production server should have at least PHP 7.0.1 installed as a minimum to improve the user's experience.

### 6.4 Future Works

The current system is web-based to which other updates can be added later. Some of the updates include payments methods such as M-PESA, PayPal, VISA cards to enable both Kenyans and foreigners to purchase milk

### References


Afimilk Ltd. (2020). *Afimilk*. Retrieved from Afifarm: https://www.afimilk.com/afifarm/

Editor. (2020, March 9). *Food Business Africa*. Retrieved from The dairy industry in Kenya: https://www.foodbusinessafrica.com/the-dairy-industry-in-kenya-production-capabilities-investments-innovations-and-trends/

JavaTpoint. (n.d.). *Software Testing*. Retrieved from Java T Point: https://www.javatpoint.com/system-testing

MilkingCloud. (2019). *MilkingCloud*. Retrieved from Milking Cloud: https://www.milkingcloud.com/?gclid=CjwKCAjw_o-HBhAsEiwANqYhpwIvQZjdqpdSWfKzoJ-pw9beejI9PzcTGF-nrnqzgO4mVFncD31sNRoCz-gQAvD_BwE

Muriuki, H. (2011). *Dairy Development in Kenya.* Rome: FAO.

Nedap Livestock Management. (n.d.). *Nedap Livestock Management*. Retrieved from Nedap CowControl:

https://www.nedap-livestockmanagement.com/dairy-farming/solutions/nedap-cowcontrol/

Pasupathy, S., Dr. Bhavani, R., & Nadu, T. (2015). An Efficient Methodology for Developing and Maintaining Consistent Software Using OOAD Tools. *International Journal of Innovative Research in Science, Engineering and Technology*, 539-544.

Sayan, P. (2018, May 28). *Software Engineering: Iterative Waterfall Model*. Retrieved from Geeks for Geeks: https://www.geeksforgeeks.org/software-engineering-iterative-waterfall-model/?ref=rp

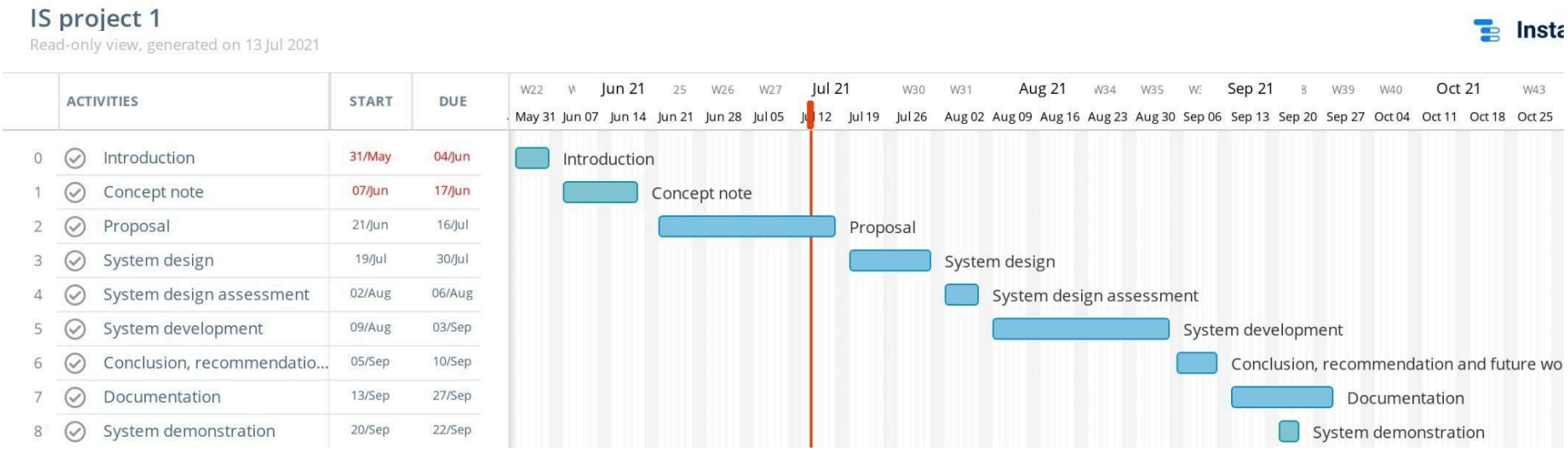
State Department of Livestock. (2013). *Sessional Paper No. 5 of 2013 on the National Dairy Development Policy.* Ministry of Agriculture, Livestock and Fisheries.

The Economic Times. (2021, July 13). Waterfall Model. New Delhi, India.

Webb, B. (2019, December 23). *Dairying*. Retrieved from Britannica: https://www.britannica.com/topic/dairying

# Appendix

## Gantt Chart




*Appendix 1: Gantt chart for project development*