
# School Of Technology
# Course: BSD 2206
# Project title: Dairy Farming Management System

## Final Document

**Reg_No: 20/04289**
**Name: Wainaina David Kinyanjui**
**Supervisor: Mr. Fred Omondi**
**Date Submitted: 14th December 2022**

## i. Declaration

I Wainaina David  Kinyanjui of Registration Number 20/04289 hereby declare that the work I have done was not copied, and was done solely by me. The project has never been presented anywhere else .

## ii. Acknowledgements

I would like to acknowledge God first for the mercies He offered me when doing this project. Special thanks to my project supervisor Mr Fred Omondi for his guidance and support throughout the project.

## iii. The Abstract /Executive summary

This document provides an overview of the Dairy Farming Management System, a comprehensive software solution designed to streamline everyday dairy farming operations. The research problem was identified at Mbogoli farm which is a medium-sized dairy farm in Nandi County that uses analogue methods to record daily activities. The farm includes a herd of fifty dairy cows, and a simple notebook is used to record the daily milk production, feed intake, and health of the cows. This information is then utilized to plan the daily operations on the farm and make decisions about how to increase the cows' output. The already developed Dairy management system will assist farmers at Mbogoli Farm in managing their dairy cows more effectively. Daily milk production, feed consumption, and cow health will be monitored by the system. The system will also record the number of animals, daily milk production, and daily milk consumption. The system will also involve the milk purchasing corporation, which will determine the price at which it acquires the milk.

The problem statement was to develop a competitive dairy farm management system for the farm. The developed system is geared for use by small-scale dairy farmers. The data collection methods that were used in development of this system were: Conducting research on the area of research (Mbogoli Farm), Observation of dairy farms and issuing Questionnaires to small-scale dairy producers.

The design Methodology that was used in development of the system was the Agile methodology which is an approach to design that stresses collaboration, consumer feedback, and iterative development. Complex projects that demand a high degree of adaptability are ideally suited to the agile methodology. For the construction of the Dairy Farming Management System, the agile methodology is optimal. The agile methodology provided a high degree of adaptability and receptivity to change, which was crucial for a project of this type. In addition, the agile methodology promoted collaboration among project stakeholders, which was essential for the effective development of the system The first phase of the agile methodology was to break down the project's needs into tiny, achievable user stories. Each user narrative was independently produced and tested, and then slowly incorporated into the overall project.
In conclusion, the Dairy Farming Management System is a comprehensive software solution designed to streamline everyday dairy farming operations. The use of the Agile methodology enabled the development of the system with a high degree of adaptability and receptivity to change. The system is designed to assist small-scale dairy farmers in managing their dairy cows more effectively by recording the number of animals, daily milk production, and daily milk consumption. In addition, the system is integrated with the milk purchasing corporation, which will determine the price at which it acquires the milk. The system is expected to increase the efficiency of dairy farms and help small-scale farmers make better decisions on how to increase the output of their cows.
The number of words used in this document are (12770)

# Table of Contents

# CHAPTER ONE: INTRODUCTION

## 1.1Background

Mbogoli Farm is a medium-sized dairy farm in Nandi County that uses analog methods to record daily activities. The farm includes a herd of fifty dairy cows, and a simple notebook is used to record the daily milk production, feed intake, and health of the cows. This information is then utilized to plan the daily operations on the farm and make decisions about how to increase the cows' output.

The developed Dairy management system will assist farmers at Mbogoli Farm in managing their dairy cows more effectively. Daily milk production, feed consumption will be monitored by the system. This data will be utilized to plan the daily operations on the farm and make decisions regarding how to increase the cows' output.

The system will also record the number of animals, daily milk production, and daily milk consumption. The system will also involve the milk purchasing corporation, which will determine the price at which it acquires the milk.
Farmers on Mbogoli Farm will be able to manage their dairy cows more effectively and increase their productivity with the help of the suggested technology.

## 1.2 problem statement

I Developed a competitive  Dairy Farming Management System for Mbogoli Farm.

Existing at Mbogoli Farm is an analogue system for monitoring daily milk production, feed consumption, and cow health. This information is then utilized to organize daily farm activities and make decisions regarding how to increase cow output.
The current approach is inefficient and does not assist the farmers of Mbogoli Farm in managing their dairy cows more effectively.

## 1.3. Scope of the study

The scope of the developed Dairy Farming Management System is as follows: -

a) To monitor the productivity of the cows
b) To record the number of animals, milk output per day, and consumption per day
c) Involve the milk purchasing company in the process of determining where and at what price the company purchases milk.

## 1.4 proposed solution.

I intend to develop a Dairy Farming Management System for the Mbogoli Farm.

The Dairy management system will aim to help the farmers in Mbogoli Farm to manage their dairy cows more efficiently. The system will track the daily milk production and feed intake. This information will be used to plan the daily activities on the farm and make decisions on how to improve the productivity of the cows. The system will also register the number of animals, daily milk production, and daily consumption. The system will also entail the milk buying company where the company purchases the milk by setting the price.

## 1.5 Significance of the project.

The significance of the Dairy Farming Management System lies in its ability to help small-scale dairy farmers increase their output and improve their efficiency. By recording the number of animals, daily milk production, and daily milk consumption, the system will help farmers make better decisions on how to increase the output of their cows. In addition, the system is integrated with the milk purchasing corporation, which will determine the price at which it acquires the milk. This will enable farmers to maximize their profits and ensure that they receive a fair price for their milk. Ultimately, the system will help small-scale dairy farmers improve their quality of life.

## 1.3 research objectives

To develop a competitive Dairy Farming Management System that will achieve the following: -

a) To assist the farmers of Mbogoli Farm in more effectively managing their dairy cows.
b) To assist in the registration of the number of animals, daily milk production, and daily consumption.
c) To involve the milk buying company in the process of determining where the company purchases the milk and at what price.
d) To assist in the monitoring of the cows' productivity.

## 1.4 system objectives

The new Dairy Management System is intended to achieve the following objectives:

a) To analyze the dairy production
b) To design the automatic process of tracking dairy production and consumption
c) To implement resource management on dairy farms
d) To test and evaluate the accuracy of dairy production data

# CHAPTER TWO: LITERATURE REVIEW

The use of information technology in dairy farming is not a new concept. In the past decade, various studies have been conducted to explore the potential of technology in dairy farming. For example, Gopalakrishnan (2015) explored the potential of using mobile applications to monitor and manage dairy farms. The study found that mobile applications can be used to monitor the cows' health, track their movements, and record data on feed consumption. In addition, the study found that mobile applications can be used to manage the finances of a dairy farm.

## 2.1. Systems developed there before

The first system that was developed is the Dairy Herd Management System (DHMS). The DHMS was created to manage large dairy herds of over 500 cows. The system included features such as milk production tracking, feed management, health monitoring, and breeding management. The DHMS was designed to be used by dairy farmers and veterinarians.

The second system that was developed is the Dairy Management System (DMS). The DMS was created to manage small to medium dairy herds of up to 500 cows. The system included features such as milk production tracking, feed management, health monitoring, and breeding management. The DMS was designed to be used by dairy farmers, veterinarians, and extension workers.

The Dairy Herd Management System (DHMS) is a comprehensive system that was created to handle big dairy herds of over 500 cows. The system contains capabilities such as tracking milk production, feed management, health monitoring, and breeding management. The DHMS is a useful instrument that may be utilized by dairy producers and veterinarians to properly manage dairy herds. One of the shortcomings of the DHMS is that it is a sophisticated system that requires a high level of training and experience to operate efficiently. In addition, implementing and maintaining the DHMS is a costly endeavour.

The Dairy Management System (DMS) is a simplified system designed to handle smaller to medium-sized dairy herds (up to 500 cows). The system contains capabilities such as tracking milk production, feed management, health monitoring, and breeding management. The DMS is a user-friendly system that allows dairy producers, veterinarians, and extension agents to manage dairy herds efficiently. The DMS lacks all of the capabilities of the DHMS, which is one of its drawbacks. In addition, implementing and maintaining the DMS is less expensive.

In a study conducted by Sharma et al. (2016), the authors explored the potential of using radio-frequency identification (RFID) technology for tracking and managing dairy farms. The authors found that RFID tags attached to the cows can be used to track their movements, feed

consumption, and health. In addition, the authors found that RFID tags can be used to monitor the herd size, milk production, and the amount of feed consumed by the cows.

The studies by Gopalakrishnan (2015) and Sharma et al. (2016) demonstrate the potential of using information technology to manage dairy farms.

### 2.2. *Reasons for the new system*

The proposed new Dairy management system differs in key respects from the DHMS and the DMS. The new system is geared for use by small-scale dairy farmers, whereas the DHMS and DMS are developed for use by large-scale dairy farmers. Secondly, the new system is intended for usage in combination with a milk-buying company, whereas the DHMS and DMS lack this capability. Thirdly, implementing and maintaining the new system is less expensive than the DHMS.

The development of the Dairy Farming Management System builds on these studies by providing a comprehensive software solution to streamline everyday dairy farming operations. The system is designed to assist small-scale dairy farmers in managing their dairy cows more effectively by recording the number of animals, daily milk production, and daily milk consumption. In addition, the system is integrated with the milk purchasing corporation, which will determine the price at which it acquires the milk. Thus, the system promises to increase the efficiency of dairy farms and help small-scale farmers make better decisions on how to increase the output of their cows.

*"Gopalakrishnan, S. (2015). Mobile applications for dairy farm management. International Journal of Dairy Science, 10(2), 107-114."*

*"Sharma, M., Pathak, R., Dixit, S., & Pandey, A. (2016). Milk production management using RFID technology: A review. International Journal of Advanced Research in Computer Science, 7(2), 9-17."*

# Chapter 3 Methodology

## 3.1 Target population

The target population for the Dairy Farming Management System is small-scale dairy farmers. These are farmers who own and operate dairy farms with fewer than 500 cows. The system is designed to help these farmers manage their dairy cows more effectively by recording the number of animals, daily milk production, and daily milk consumption. In addition, the system is integrated with the milk purchasing corporation, which will determine the price at which it acquires the milk. This will enable farmers to maximize their profits and ensure that they receive a fair price for their milk.

## 3.2 Sample selection

The sample selection for the Dairy Farming Management System will be based on convenience sampling. This means that the sample will be selected based on ease of access and availability. The sample will include small-scale dairy farmers from different parts of the country who are willing to participate in the study. The sample size should be large enough to provide reliable results, but small enough to be manageable.

## 3.3 Data collection method(s)

*Interviews with small-scale dairy farmers:* In Kenya, data will be gathered through interviews with small-scale dairy producers. The goal of the interviews is to gather information regarding the farmers' everyday activities, the difficulties they experience, and the solutions they have applied.
*Focus groups with small-scale dairy* farmers Data will be gathered through focus groups with small-scale dairy producers in Kenya. The goal of the focus groups is to collect information regarding the farmers' everyday activities, the obstacles they face, and the solutions they have applied.
*Questionnaires for small-scale dairy producers:* Small-scale dairy farmers in Kenya will be surveyed using questionnaires to obtain data. The goal of the surveys is to collect data regarding the farmers' everyday activities, the obstacles they face, and the solutions they have applied.
*Observation of dairy farms*: Information will be gathered through observation of dairy farms in Kenya. The goal of the observations is to collect data on the farmers' everyday activities, the obstacles they face, and the solutions they have applied.
*Review of current literature on dairy farming Data* will be gathered by reviewing existing literature on dairy farming. The goal of this study is to gather information regarding the farmers' everyday operations, the obstacles they face, and the solutions they have applied.
*Secondary data analysis:* Data will be gathered by secondary data analysis. The objective of the secondary data analysis is to gather information about the everyday activities of farmers, the obstacles they face, and the solutions they have adopted.

## 3.4. Development methodology

The development methodology for the Dairy Farming Management System will be Agile. Agile is a user-centric approach to software development that involves iterative and incremental development. It is based on the idea that the requirements of the system are constantly changing and evolving, and the system must be able to adapt to these changes. Agile methodology also involves collaboration between developers, users, and stakeholders. The development process will involve the continuous delivery of small pieces of

functionality, which can then be tested, refined, and improved. The system will be designed to be responsive to change and will be tested regularly to ensure that it meets the requirements of the users.

# Chapter 4: System Requirement Specifications

## 4.1 purpose

The purpose of this SRS is to define the requirements for the Dairy Farming Management System. This system is intended to provide farmers with an easy way to keep track of their dairy cows and produce. The system will allow farmers to track the productivity of their cows, as well as the quality of their milk. The system will also incorporate the milk purchasing company where they buy the milk by setting the price. In addition, the system will provide farmers with the ability to track the financial performance of their business.

The intended audience for this SRS is farmers, milk buying company, application designers, and developers.

## 4.2System perspective

The system will be a web-based application system that is not integrated with any other system.
The following are the main components of the system
*Mbogoli Farm Database* - This will store information about the dairy cows, milk production and feed consumption.
*Dairy Farming Management System* - This will be the software that farmers will use to input data and generate reports. It will also be used by the milk company to generate payments and monthly statements.
*Milk Purchasing Corporation* - This will be the organization that will purchase the milk from the farmers.

## 4.3System Functionality

The system will have the following functions:

*Input data* - Farmers will input data into the system about the dairy cows, milk production, feed consumption, and cow health. This data will be used to generate reports.

*Generate reports* - The system will generate reports based on the data that is inputted. These reports will be used to plan the daily activities on the farm and make decisions about how to increase the cows' output.

*Determine price* - The system will determine the price at which the milk will be purchased by the milk purchasing corporation. The price will be based on the quantity of milk produced and the quality of the milk.

The system will not be responsible for milking the cows or feeding the cows. This will be the responsibility of the farmers. The system will help the farmers by giving them information about the cows so that they can make decisions about how to increase milk production.

## 4.4 User Characteristics

***Farmers*** **-** The farmers will have a high school education. They will have experience in dairy farming. They will have basic computational skills. The farmers will use the system to input data about the dairy cows, milk production, feed consumption, and cow health. This data will be used to generate reports.
***Milk Buying Company*** **-** The Milk Buying Company will have a college education. They will have experience in the milk industry. They will have Expertise technical skills in Vb.Net. The Milk Buying Company will use the system to determine the price at which it will purchase the milk from the farmers.
***Admin*** - The Admin will have a college education. They will have experience in managing systems. They will have Expertise technical skills in Vb.Net. and MYSQL databases. The admin will use the system to manage the system.

## 4.5 General Constraints

### Regulatory policies

The system should allow farmers to input data about the dairy cows, milk production, feed consumption, and cow health.
The system should allow the Milk Purchasing Corporation to determine the price at which it will purchase the milk from the farmers.
The system should allow the admin to manage the system.

### Hardware limitations specific to the system

The system should be able to store data about the dairy cows, milk production, feed consumption, and cow health.
The system should be able to generate reports based on the data that is inputted.
The system should be easy to use.
The system should be able to run on a PC with Windows operating system.
The system should have a printer for printing reports.
.

### Safety and Security

To ensure that only authorized users may access the system, it should be password-protected. To avoid data loss, the system should regularly be backed up. To avoid unwanted access, the machine should be housed in a secure area, like a locked room or cabinet. It is important to keep an eye out for any unusual or suspicious activity on the system, such as unwanted access attempts. To make sure the system is operating correctly, it should undergo frequent testing.

# 4.6 Assumptions and dependencies

The following are the assumptions made about the System

A. It is assumed that the farmers will have a high school education.
B. It is assumed that the farmers will have basic technical skills.
C. It is assumed that the Milk Purchasing Corporation workers will have a college education.
D. It is assumed that the system will be easy to use.

All of these assumptions could affect the system if they are incorrect. For example, if the farmers do not have the necessary skills to use the system, the system will not be effective.

***The system depends on the following:***

Mbogoli Farm Database - This will store information about the dairy cows, milk production, feed consumption, and cow health.
Farmers - these are the primary users of the system
Milk Purchasing Corporation - This will be the organization that will purchase the milk from the farmers.
Without the above dependencies the system will not achieve the main Objectives

## 4.7 Specific Requirements

### Functional requirements

***Inputs and Outputs***

*The system will have the following inputs:*
Data about the dairy cows, milk production, feed consumption.

*The system will have the following outputs:*
Reports about the dairy cows, milk production, feed consumption, and milk sales statements.

The data about the dairy cows, milk production, and feed consumption will be inputted into the system by the farmers. The reports generated by the system will be used by the farmers to plan the daily activities on the farm and make decisions about how to increase the cows' output. The milk sales statements will be generated by the Milk Purchasing Corporation after

they determine the price at which it will purchase the milk from the farmers. All of this data will be stored in a single Firebase Database.

***Processing***

The system will validate the input data to ensure that it is complete and accurate. The system will generate reports based on the data that is inputted. These reports will be used to plan the daily activities on the farm and make decisions about how to increase the cows' output. The system will determine the price at which the milk will be purchased by the milk purchasing corporation. The price will be based on the quantity of milk produced.

*The system will execute the processes in the following sequence:*

The farmers will input data about the dairy cows, milk production, and feed consumption. The system will generate reports about the dairy cows, milk production, feed consumption, and cow health. The system will determine the price at which the milk will be purchased by the milk purchasing corporation and show the monthly pay of the farmer.

The system will display an error message if the input data is incomplete or inaccurate. The system will display an alert message if the milk production is below the average milk production. The system will display an alert message if the feed consumption is above the average feed consumption. The system will use simple addition of daily production of the animals and accumulate it monthly where it will be used to generate total monthly payments for the farmer by the company.

### Non-Functional requirements

**Performance Requirement**

The system shall be able to record and store information on the dairy cows with a response time of less than 2 seconds.

**Capacity Requirements:** The system shall be able to store information on up to 1000 dairy cows.

**Availability Requirements**: The system shall be available 24 hours a day, 7 days a week.

**Recovery Requirements:** In the event of a power failure, the system shall be able to recover all information that has been entered into the system since the last backup.

## 4.8 User Interface Requirements

The Dairy Farming Management System will have a user-friendly interface. The interface will be a graphical user interface that is easy to use and navigate. The interface will conform to the GUI standards. The screen layout will be constrained to the standard buttons and functions. The error message display will conform to the standards.

***The system will have the following interfaces***

1. **Login page (Admin, Farmers, Company):** This page will have a GUI designed using Vb.Net which will allow the users to select their functionalities in the system. Once the button is selected the user will be taken into their respective pages.
2. **Signup page:** This is the page whereby the company and the farmers sign up into the system.
3. **Farmers Homepage:** This page will welcome the farmers into the system. The farmer will be able to access the Animal page from this page. This page will display daily dairy production in form of text and feed daily feed consumption in form if grid view. It will also allow the user to add(buy) or remove/(Sell) animals in the farm.
4. **Animal page:** This simple page will be accessible to farmers and admin. The farmers have access to its database and can modify the data inside.
5. **Admin page:** This is the page for the super user who will have the oversight of the entire system. This user can add/remove any users of the system. The admin logins will be in the database and he doesn't need to sign up.
6. **Company page:** This is the page which is strictly designed for the milk buying company that the farmer has agreed to sell the milk to. This page will allow the company to set the price for milk, buy the milk and generate milk statements.

## Hardware Interfaces

**Keyboard** - Used to input data into the system
**Printers** - Used to print reports generated by the system
**200Gb disk storage:** For storing the system and all its components
**2.4ghz processing power:** The machine to run this system must have a processor with the stated speed.

### Software Interface

**Firebase Database** - This will be used to store information collected and generated by the system.

**windows 10pro - This will be the operating system on which the system will run on.**
**Vb.net -** This is the programming language in which the system will be designed with.

**Visual Studio 2019 -** This is the system in which the system will run on i.e., failure of this software the system won't be available to execute the tasks.

### Other requirements

**Security Requirements:** The system shall have security features to protect the information stored in the system from unauthorized access.
**Safety Requirements**: The system shall be safe to use.

# Chapter5: Design Specifications

## 5.1 System Overview

The purpose of the system is to help farmers on Mbogoli Farm to manage their dairy cows more effectively and increase their productivity. The system will track the number of animals, daily milk production, and daily animal consumption. The system will also include the milk purchasing corporation, which will determine the price at which the milk is acquired. Farmers on Mbogoli Farm will be able to manage their dairy cows more effectively and boost their productivity with the help of the suggested technology. The system will assist farmers in keeping track of their cows and milk output so that they may make decisions about how to boost milk production.

### 5.1.1 Data flow diagrams level 0

The system will track the number of animals, daily milk production, and daily animal consumption. The system will also include the milk purchasing corporation, which will determine the price at which the milk is acquired. Farmers on Mbogoli Farm will be able to manage their dairy cows more effectively and boost their productivity with the help of the suggested technology. The system will assist farmers in keeping track of their cows and milk output so that they may make decisions about how to boost milk production. The system would also assist farmers in selling their milk at a reasonable price.

The following is the dairy farming management system level 0 DFD.

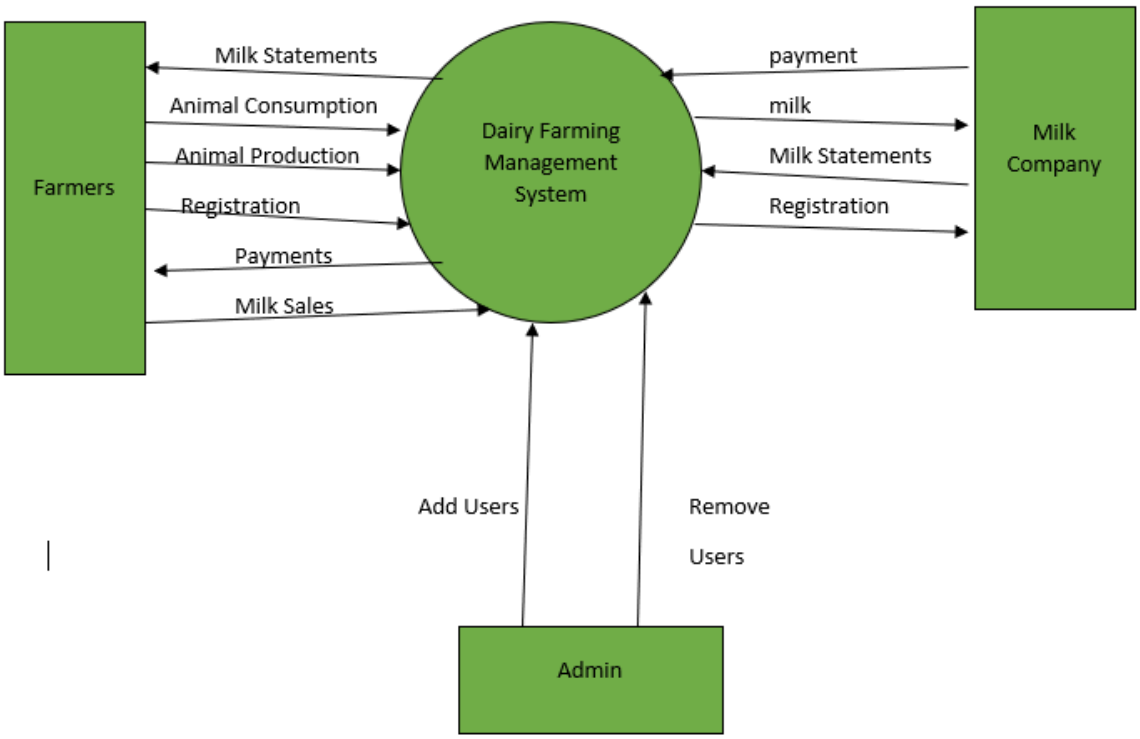


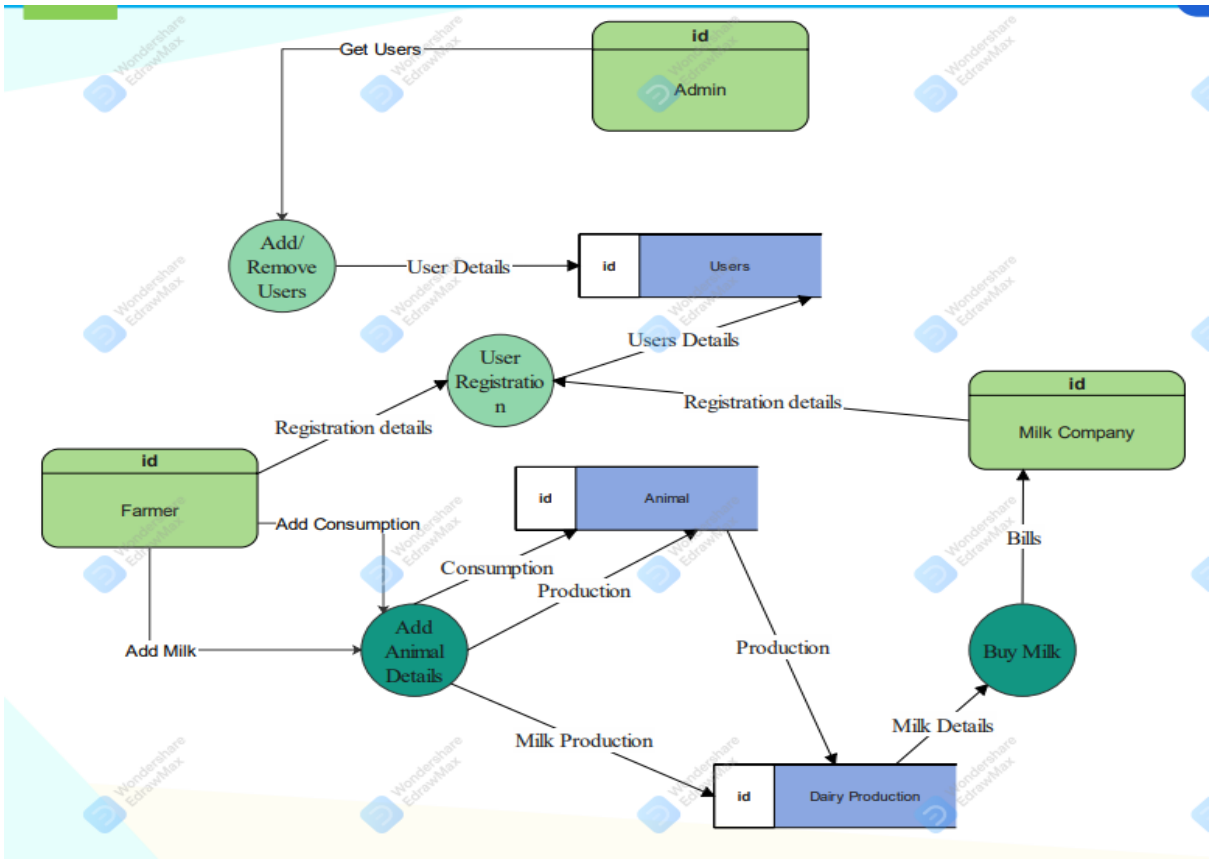

Fig 1.

## 5.1.2 Data flow level 1

A level one dfd diagram is a high-level view of a system that shows the main components of the system and how they interact. It is useful for understanding the big picture of a system and for identifying the major areas of focus for further investigation.

The diagram typically shows four main types of components: inputs, outputs, processes, and data stores. Inputs are the external factors that impact the system, such as user input or environmental conditions. Outputs are the results that the system produces, such as a report or a file. Processes are the actions that the system takes in order to achieve its outputs, such as data manipulation or communication with other systems. Data stores are the repositories for the data that the system uses, such as a database or a file system.

In a level one dfd diagram, the inputs, outputs, processes, and data stores are represented as rectangles, and the interactions between them are represented as arrows. The diagram is read from left to right and top to bottom. The inputs are at the left, the outputs are at the right, the processes are in the middle, and the data stores are at the bottom.




## 5.1.2 Software Architecture

The dairy farming management system software design is the following:

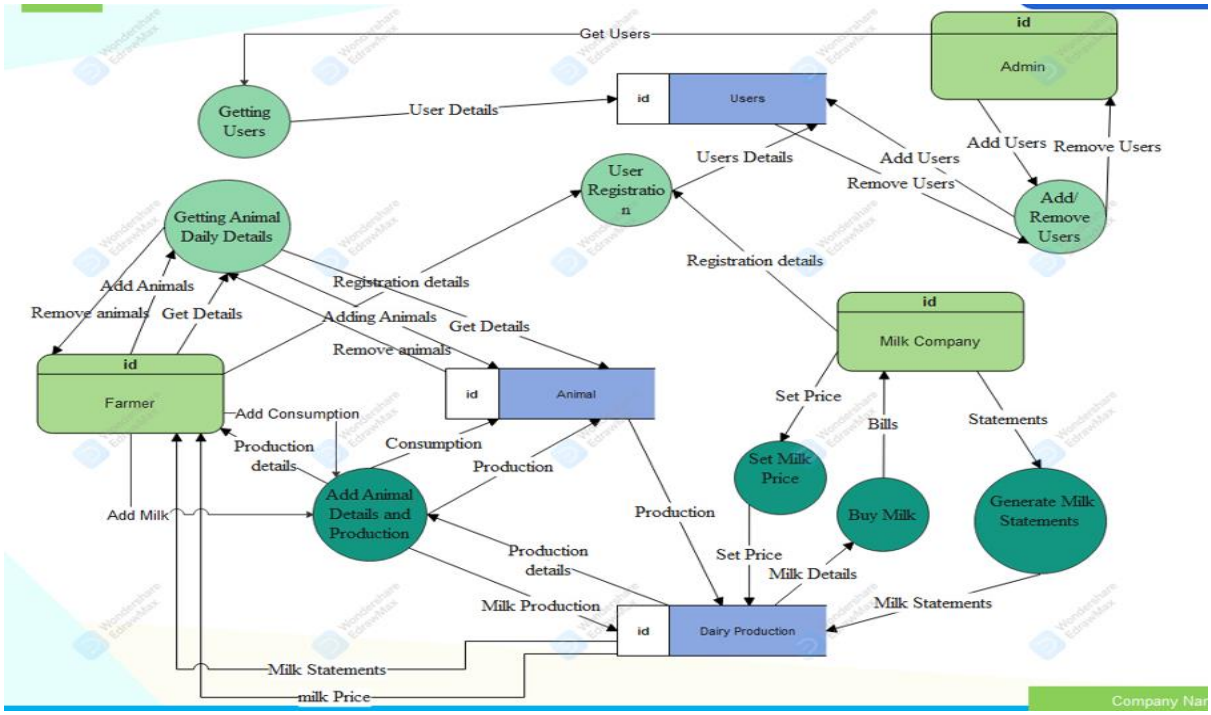



Fig 2.

# 5.2. Data base design

## 5.1.3 Logical Design

Entities And Attributes

An entity is an object, person, place, or concept about which information is stored. An attribute is a characteristic or quality of an entity.

In a database, entities are represented by tables, and attributes are represented by columns in those tables.

In this Project the database has the following Entities:

**Admin** This Entity Stores Admin Login Credentials
**Farmer**: This Entity Stores Farmers Details And Dairy Production Of his/her farm
**Animal** This entity has the animal details
**Company** Carries the company credentials and also its relation to the farmer

The a bove Entities have the following attributes

1. **Admin:** admin_id, admin_password, farmer_id, company_id
2. **Farmer**: farmer_id, farmer_password, farmer_production, animal_production,animal_sn, company_id
3. **Animal:** animal_sn, animal_production, animal_consumption, farmer_id
4. **Company:** company_id, company_name, farmer_id

Nomalization

1. **Admin**
   Admin( admin_id, admin_password, farmer_id)
   Admin( admin_id, admin_password, company_id)
   Admin( admin_id, admin_password)
   Farmer( farmer_id, farmer_password)
   Company( company_id, company_password)

2. **Farmer:**
   Farmer( farmer_id, farmer_password, farmer_production)
   Company(company_id, company_name)
   Animal(animal_sn, animal_production)

3. **Animal:**

   Animal( animal_sn, animal_production, animal_consumption )
   Farmer( farmer_id )

4. **Company:**
   Company:(company_id, company_name)
   Farmer: (farmer_id)

Data Dictionary

admin_id: The unique identifier for the administrator.
admin_password: The password associated with the administrator account.
farmer_id: The unique identifier for the farmer.
company_id: The unique identifier for the company.
farmer_password: The password associated with the farmer account.
farmer_production: The amount of produce the farmer has produced.
animal_production: The amount of produce the animal has produced.
animal_consumption: The amount of produce the animal has consumed.
company_name: The name of the company.

Entity Relationship Diagram (ERD)

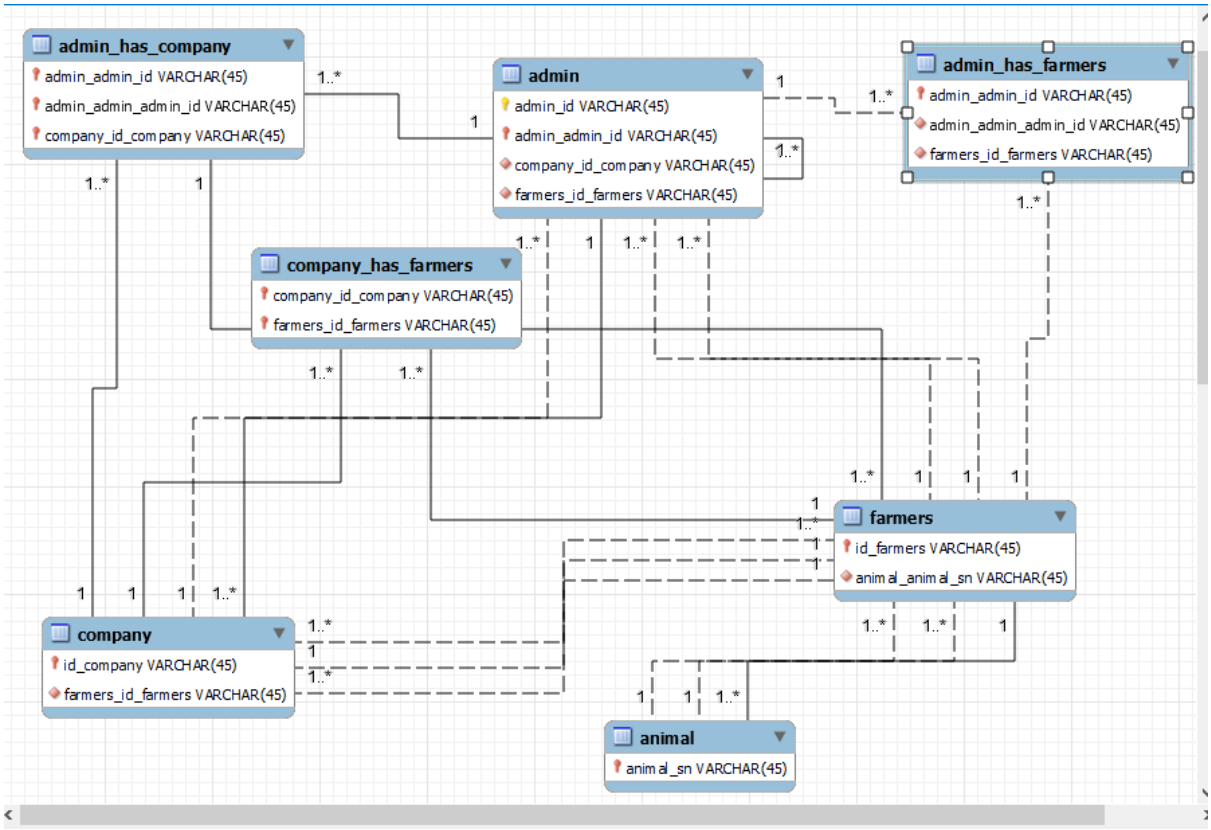
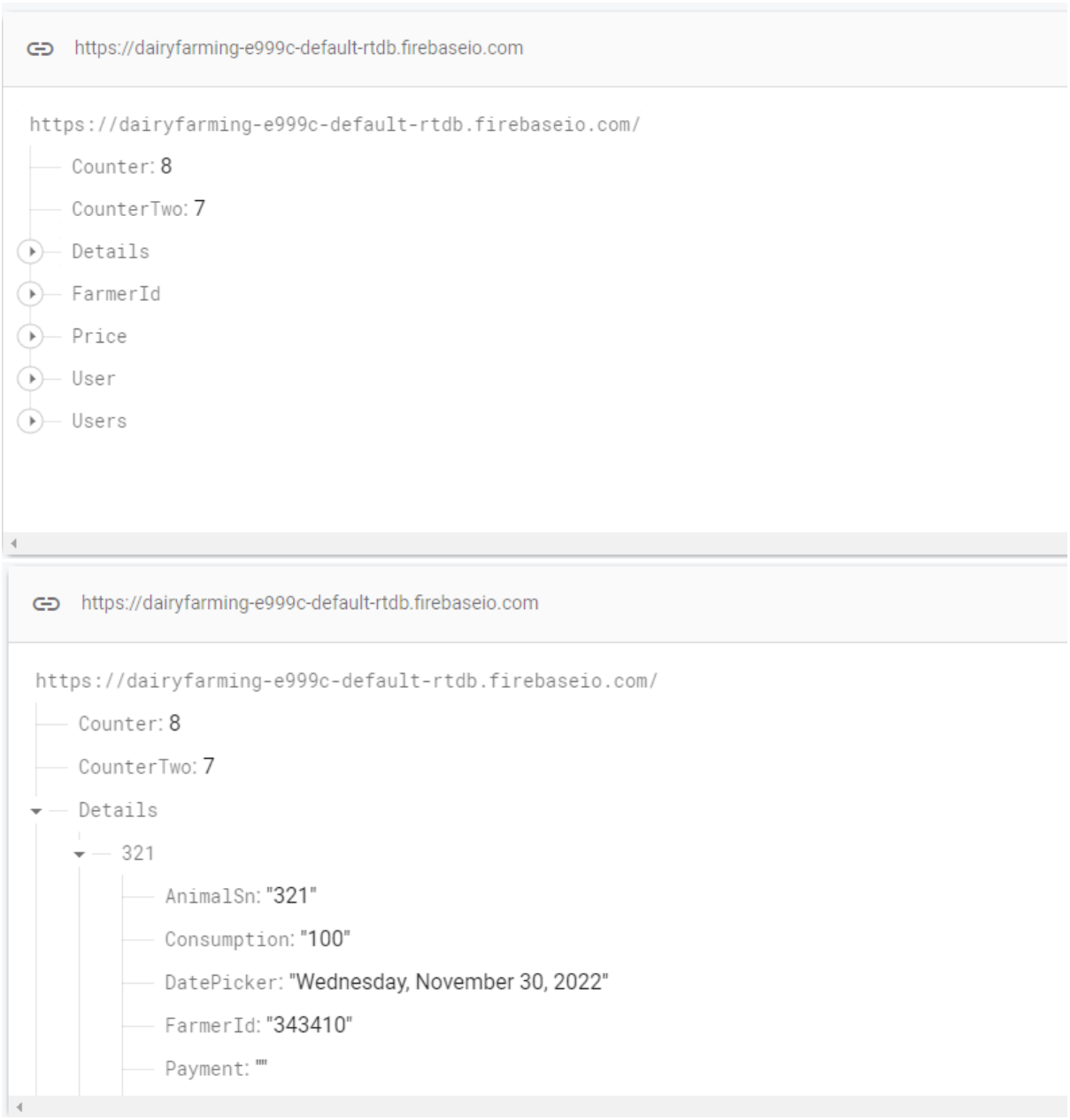



Fig 3

Physical Design

The databased used for this system is NoSql which is filebased. The database is called firebase.

**Diagrams**



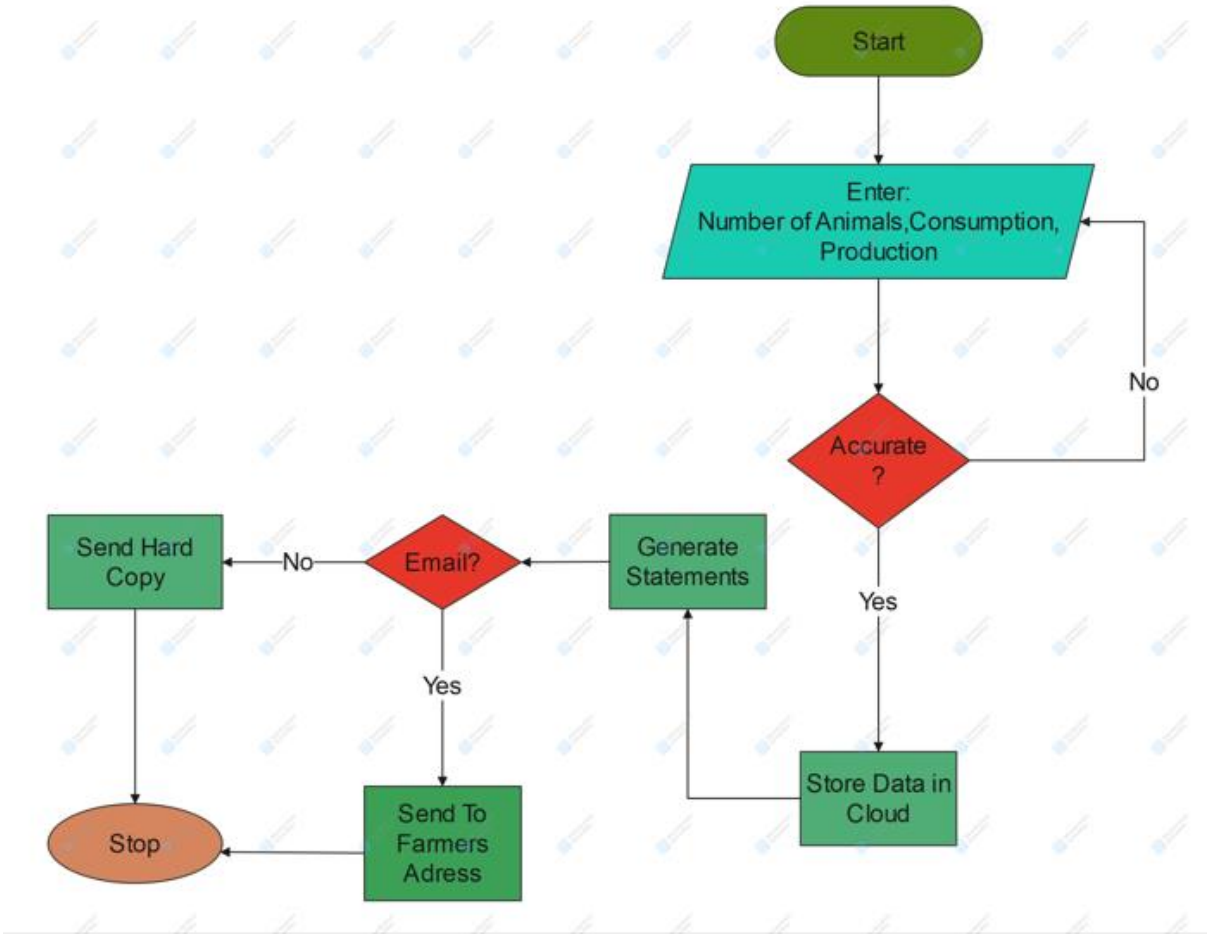

This database does not require any storage since it is Realtime and runs in the cloud

## 5.3 Software design

**The system has two processes as follows**

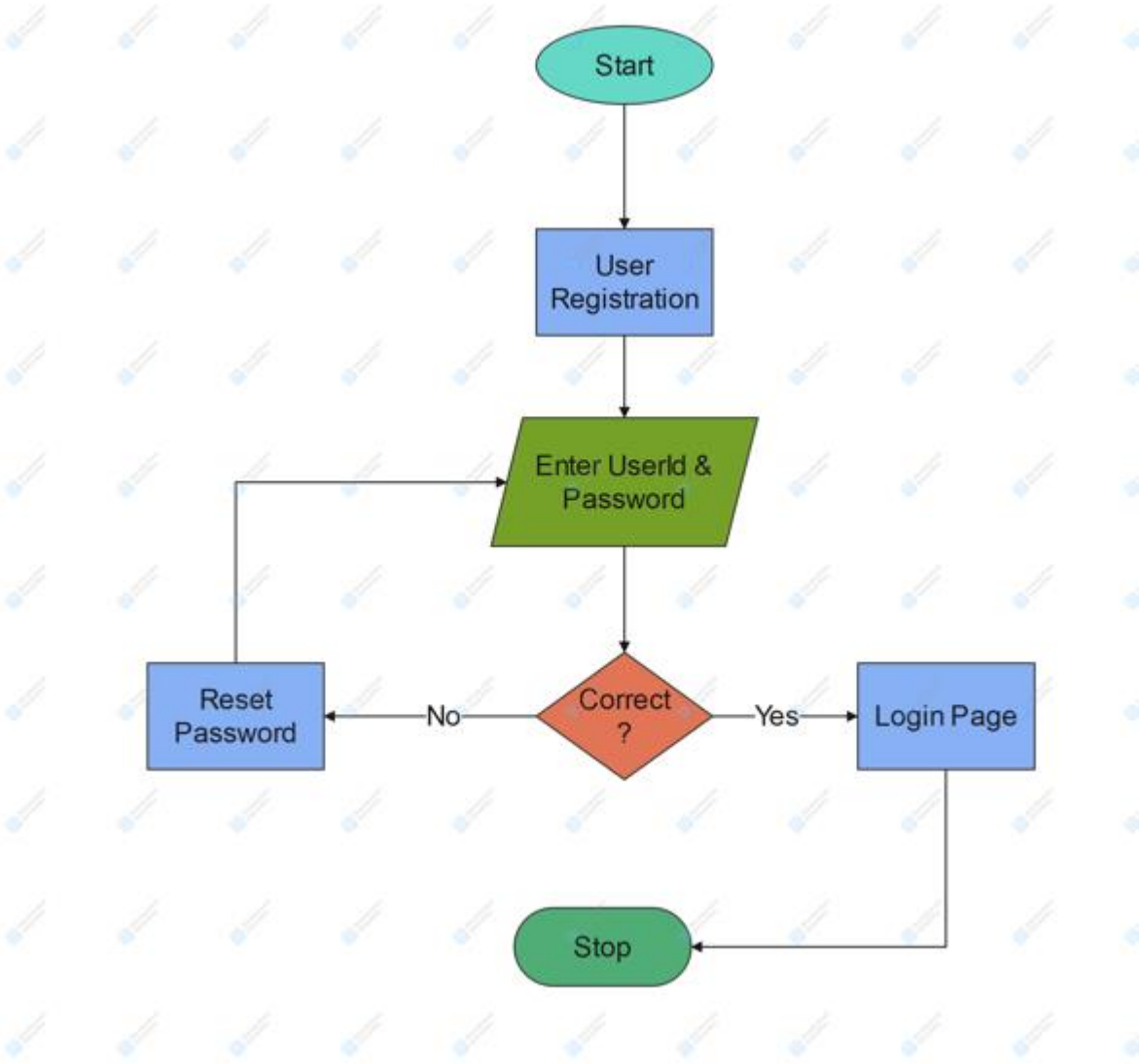
**a)**

1. Start
2. Enter: Number of animals, Consumption and production
3. Check accuracy of the data
4. Store in cloud
5. Generate milk statements
6. Send statements via email or hard copy
7. stop




**B)**

1. Start
2. Register user
3. Enter User Id and password
4. If correct login, else Reset Password
5. stop




## 5.4. User Interface Design

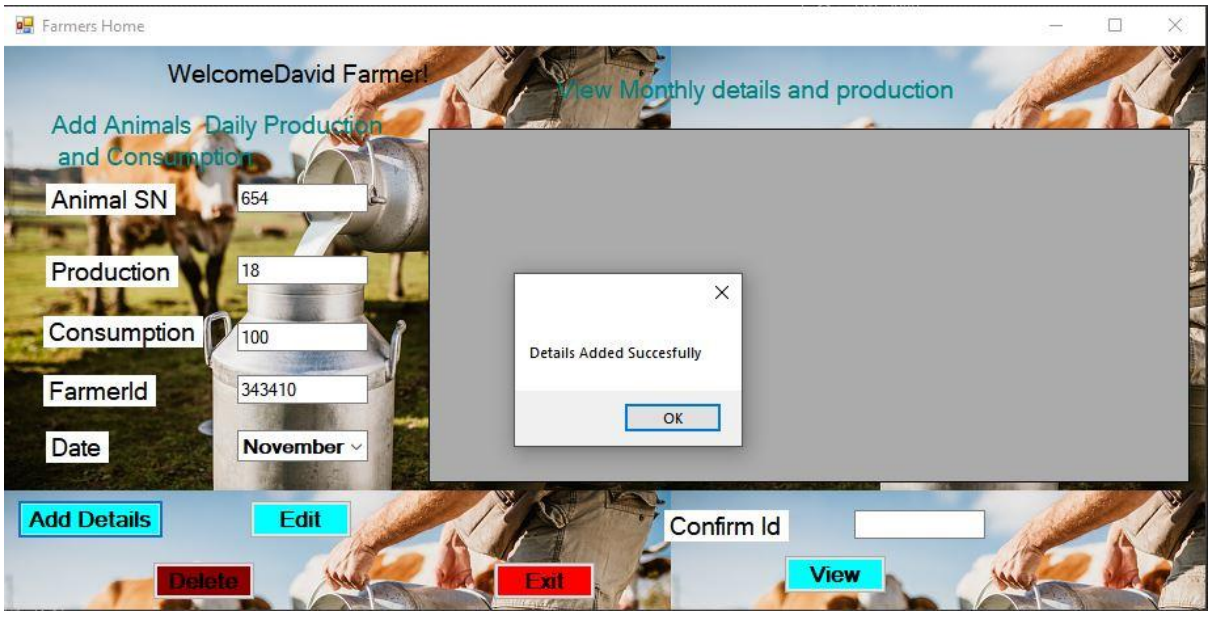
### 5.4.1 Input user interface Design


User Registration

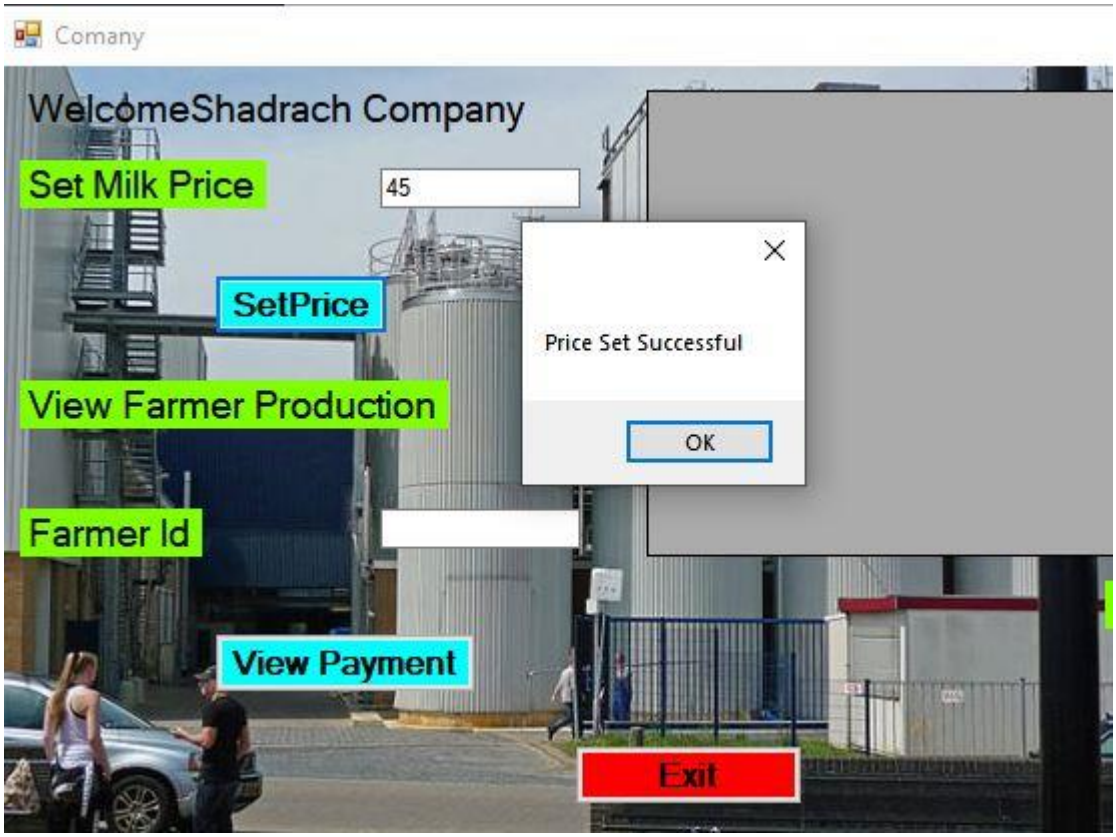

Registration
Please Continue To register
User Id
343410
Password
Registration Successful
OK
Farmer
Register

User Login

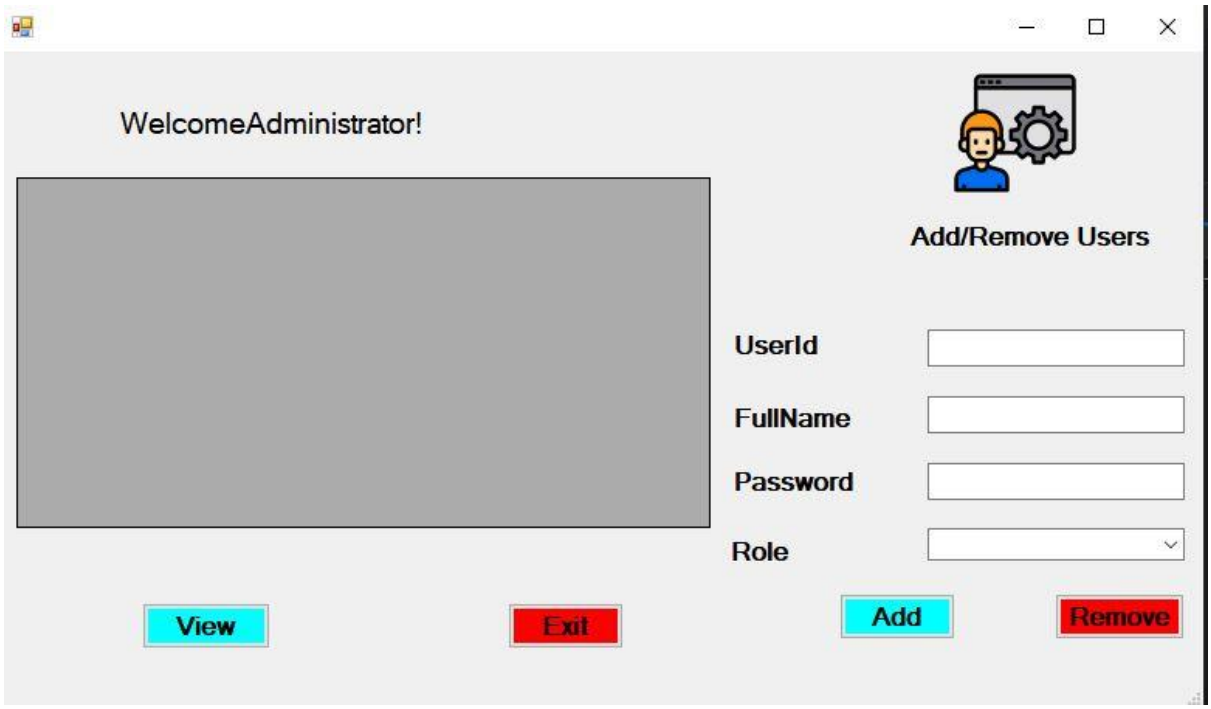

Farmers Home input

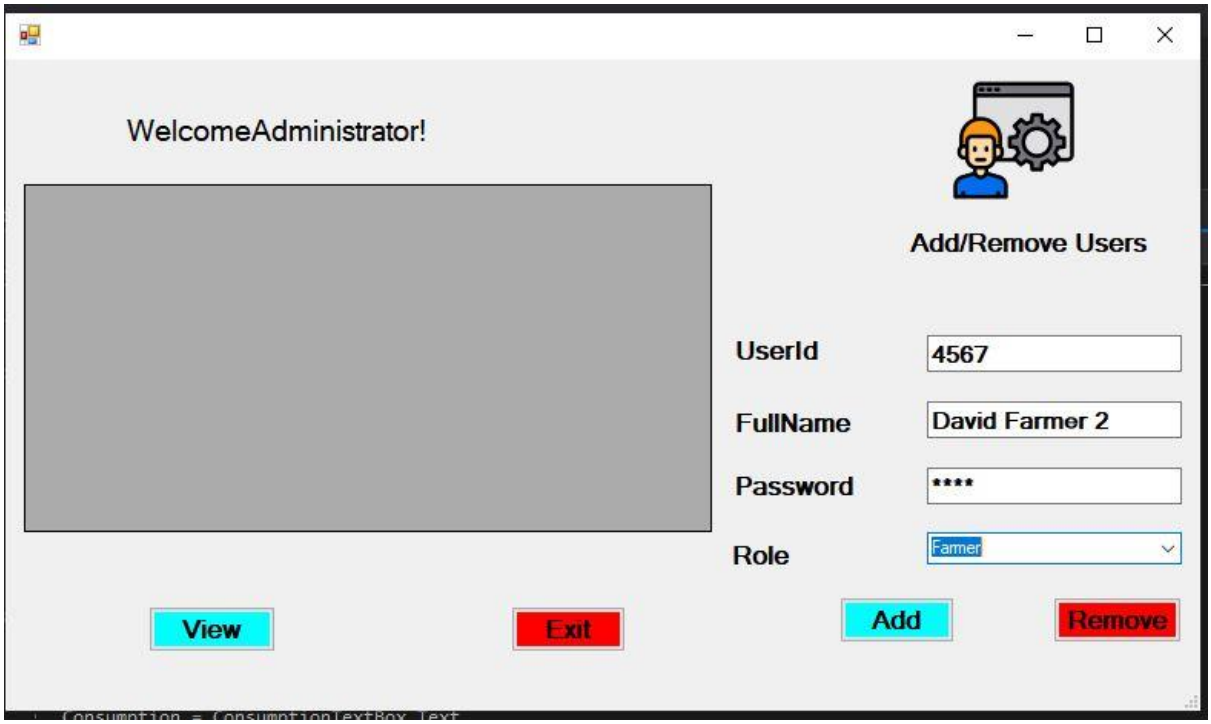



Company Home and Input


Admin Home and input

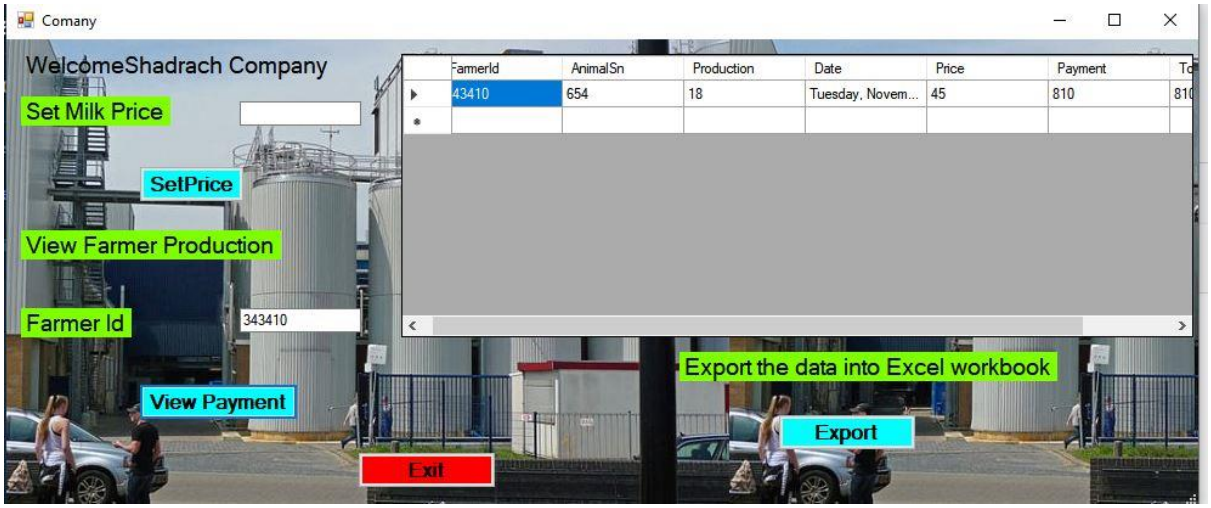


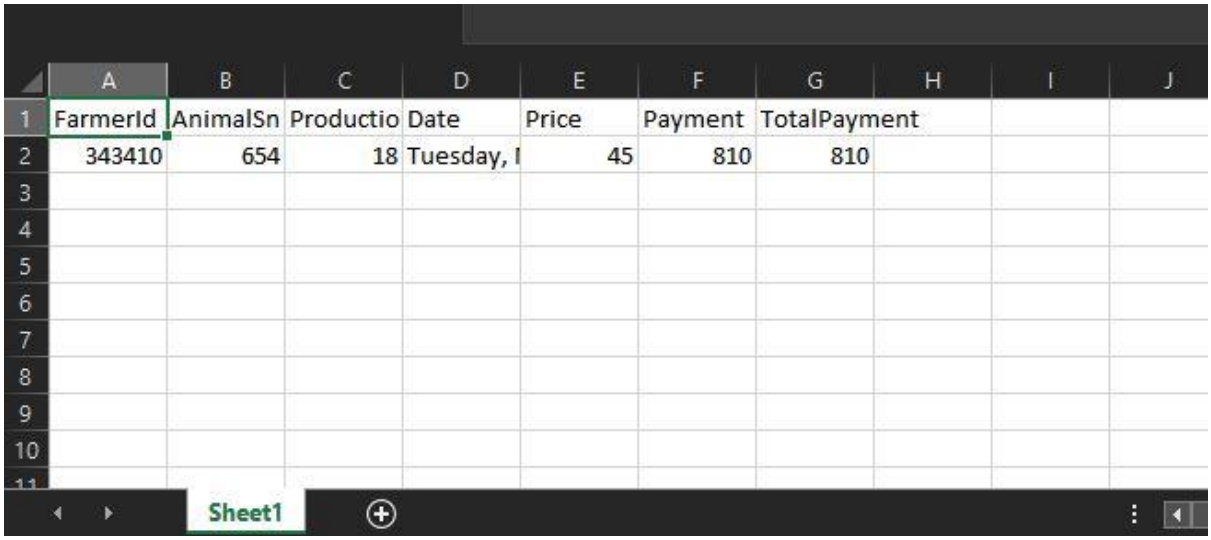




## 5.3.2 Output user interface Design

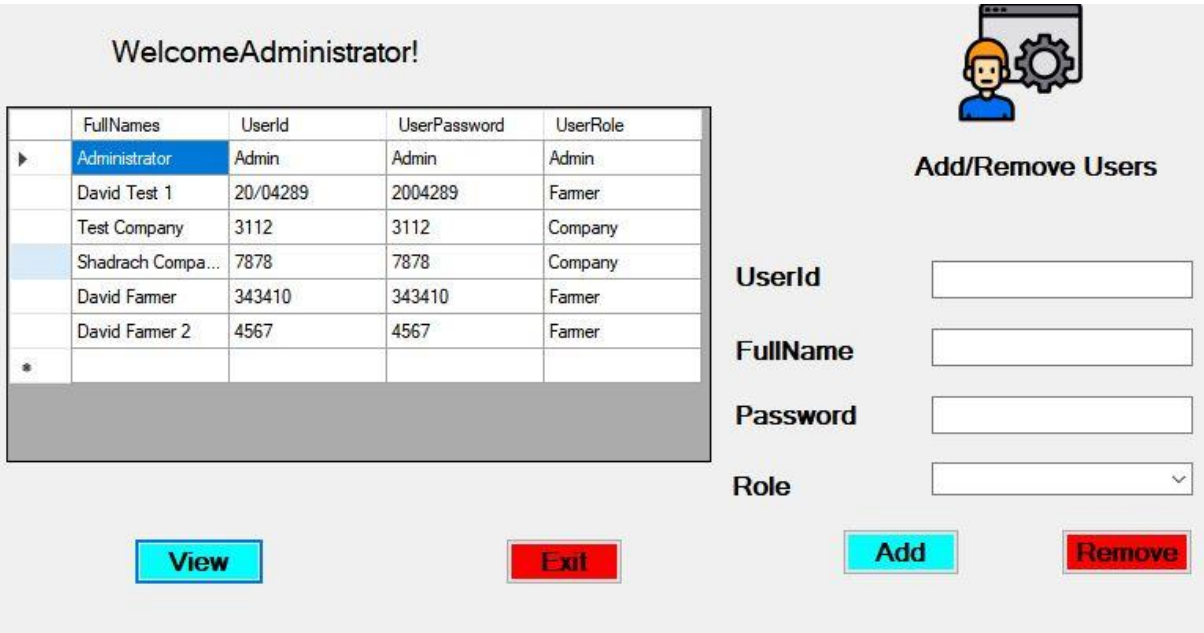
Farmers Output


Company outputs




Admin Output




## 5.5. Test Plan

### Introduction

The Dairy Farming Management System will be tested to ensure that it meets the requirements specified by the client. Testing will be done to ensure that the system is able to track the number of animals, daily milk production, and daily animal consumption. Testing will also be done to ensure that the system is able to manage the milk purchasing corporation.

### 5.5.1 Test Plan

The system will be tested to ensure that it meets the requirements specified by the client. The system will be tested for functionality as well as for non-functional requirements such as scalability and performance. A test plan will be generated which will outline the various tests to be carried out. The test plan will be executed in two phases, with a test report being generated at the end of each phase.

Functional testing will ensure that the system is able to track the number of animals, daily milk production, and daily animal consumption. Non-functional testing will ensure that the

system is able to manage the milk purchasing corporation. Functional testing will be done by manually inputting data into the system and checking that the output is as expected. Non-functional testing will be done by running the system through a series of stress tests to ensure that it can handle large amounts of data.

The project management issues that are required to properly execute effective tests are as follows:

- The test plan must be approved by the client.
- The test plan must be reviewed and updated as needed.
- The test plan must be followed.
- The test team must be properly trained.
- The test team must have the necessary resources.
- The test team must be able to communicate effectively.
- The test team must be able to work together.

### 5.5.2 Software (SCIís) to be tested

The Software to be tested is Dairy Farming Management System

## 5.5.3 Testing strategy

Testing will be done using the following strategies
**Unit testing** is a process where individual software components are tested to ensure that they work as expected.
**Integration testing** is a process where components are tested to ensure that they work together as expected.
**Validation testing** is a process where the system is tested to ensure that it meets the requirements specified by the client.
**High order testing** is a process where the system is put through a series of stress tests to ensure that it can handle large amounts of data.

### 5.5.3.1 Unit testing

Unit testing is a process where individual software components are tested to ensure that they work as expected. In unit testing, individual units of code are tested to ensure that they are functioning correctly. Unit tests are typically written by the developers who create the code. Unit tests are used to verify that the code meets the requirements specified in the design document. Unit tests are also used to find defects in the code.
There are several components that will undergo unit testing:
**The database:** The database will be tested to ensure that it can store and retrieve data as expected.

**The milk tracking system:** The milk tracking system will be tested to ensure that it can track the number of animals, daily milk production, and daily animal consumption.
**The milk purchasing system:** The milk purchasing system will be tested to ensure that it can manage the milk purchasing corporation.

### 5.5.3.2 Integration testing

Integrated testing is a process where components are tested to ensure that they work together as expected. In integrated testing, individual units of code are tested to ensure that they are functioning correctly when they are integrated with other units of code. Integrated tests are typically written by the developers who create the code. Integrated tests are used to verify that the code meets the requirements specified in the design document. Integrated tests are also used to find defects in the code.

The integration testing will begin with testing the User Interfaces. The admin must be able to access all the users of the system so it will be tested first. The database integration with the user interfaces will follow. And then we will integrate the farmers and the milk purchasing system and finally test if the overall objectives are met before realising to the final user.

### 5.5.3.3 Validation testing

Validation testing is a process where the system is tested to ensure that it meets the requirements specified by the client. In validation testing, the system is tested to ensure that it meets the functional and non-functional requirements specified in the design document. Validation tests are typically written by the developers who create the code.

The validation of the system will begin with the login and registration forms which will validate the user inputs and authenticate them. The other validation will be conducted on the farmers side where they input their dairy records. The database should be able to validate the details of each and every user without failure.

#### 5.5.3.4 High-order testing

High order testing is a process where the system is put through a series of stress tests to ensure that it can handle large amounts of data. In high order testing, the system is tested to ensure that it can handle a large number of users, a large amount of data, and a large number of transactions.

There are several types of high order tests that will be conducted:
**Load testing:** Load testing is a type of test that is used to determine how the system responds to a large number of users.
**Stress testing:** Stress testing is a type of test that is used to determine how the system responds to a large amount of data.

**Performance testing**: Performance testing is a type of test that is used to determine how the system responds to a large number of transactions.
**Capacity testing:** Capacity testing is a type of test that is used to determine how the system responds to a large number of users, a large amount of data, and a large number of transactions.

## 5.5.4 Testing resources and staffing

The following are the testing resources to be used in testing the Dairy Farming Management system.
**Test management tool:** A test management tool is a software application that is used to plan, execute, and track the testing of a software project.
**Test case management tool:** A test case management tool is a software application that is used to manage the test cases for a software project.
**Bug tracking tool:** A bug tracking tool is a software application that is used to track the bugs that are found in a software project.
**Test automation tool:** A test automation tool is a software application that is used to automate the testing of a software project.
**Performance testing tool:** A performance testing tool is a software application that is used to test the performance of a software project.

For the staffing the following staffs shall be deployed
**Project manager:** The project manager is responsible for overall project management.
**Lead test engineer:** The lead test engineer is responsible for test planning and execution.
**Test engineer:** The test engineer is responsible for writing and executing test cases.
**Software developer:** The software developer is responsible for developing the code.
**Database administrator:** The database administrator is responsible for managing the database.

### 5.5.4.1 Test work products

**Test plan:** The test plan is a document that outlines the approach that will be taken to testing the system.
**Test cases:** The test cases are documents that describe the steps that need to be taken to test the system.
**Test reports:** The test reports are documents that contain the results of the testing, as well as any bugs that were found.
**TestLogs:** These logs are produced as the system is being debugged in order to record all the transactions happening in the system.

### 5.5.4.2 Test record keeping

The following are ways to keep track of test results:
**Test logs:** Test logs are documents that contain a record of all the tests that were conducted, as well as the results of those tests.
**Test databases:** Test databases are software applications that store the results of tests in a database.
**Test management tools:** Test management tools are software applications that can be used to track the results of tests.

### 5.5.4.3 Test metrics

The following are the test metrics that will be used during the testing activity:
**Pass/fail rate:** The pass/fail rate is the percentage of tests that pass.
**Code coverage:** Code coverage is the percentage of code that is covered by tests.
**Test case execution time:** Test case execution time is the amount of time it takes to execute a test case.
**Bug density:** Bug density is the number of bugs per thousand lines of code.

### 5.5.4.4 Testing tools and environment

The test environment for the Dairy Farming Management System should be similar to the production environment. The test environment should have the same hardware, software, database, and network configuration as the production environment. The test environment should also have a copy of the production data.

The following are the testing tools
**Test management tool:** A test management tool is a software application that is used to plan, execute, and track the testing of a software project.
**Test case management tool:** A test case management tool is a software application that is used to manage the test cases for a software project.
**Bug tracking tool:** A bug tracking tool is a software application that is used to track the bugs that are found in a software project.

**Test automation tool:** A test automation tool is a software application that is used to automate the testing of a software project.
**Performance testing tool:** A performance testing tool is a software application that is used to test the performance of a software project.

### 5.5.4.5 Test schedule

The test Schedule will have the following criteria which can change if the client decides a shorter one

1) Test planning: 2 weeks

2) Test case development: 4 weeks
3) Test execution: 6 weeks
4) Test reporting: 2 weeks

## 5.5.5 Test Procedure

The following are the procedures and tactics to be put in place when testing the software

**Test planning:** The test team will develop a test plan that outlines the approach that will be taken to testing the system.
**Test case development:** The test team will develop test cases that describe the steps that need to be taken to test the system.
**Test execution**: The test team will execute the test cases and track the results.
**Test reporting:** The test team will generate test reports that contain the results of the testing, as well as any bugs that were found.

### Testing tactics:

**Black box testing:** Black box testing is a type of testing where the test team does not have access to the source code. Black box testing is typically used to test the functionality of the system.
**White box testing:** White box testing is a type of testing where the test team has access to the source code. White box testing is typically used to test the code for defects.
**Gray box testing:** gray box testing is a type of testing where the test team has access to some of the source code. Gray box testing is typically used to test the code for defects.

### Test cases:

**functional test cases:** Functional test cases are used to test the functionality of the system.
**non-functional test cases:** Non-functional test cases are used to test the performance of the system.

## 5.5.6 Software (SCIís) to be tested

The software to be tested is Dairy Farming Management System

## 5.5.7 Testing procedure

The overall procedure for software testing can be divided into four main phases: test planning, test case development, test execution, and test reporting.

**Test planning:** In the test planning phase, the test team develops a test plan that outlines the approach that will be taken to testing the system.
**Test case development:** In the test case development phase, the test team develops test cases that describe the steps that need to be taken to test the system.
**Test execution:** In the test execution phase, the test team executes the test cases and tracks the results.
**Test reporting:** In the test reporting phase, the test team generates test reports that contain the results of the testing, as well as any bugs that were found.

## 5.5.7.0 Unit test cases

Unit testing is a process where individual software components are tested to ensure that they work as expected. In unit testing, individual units of code are tested to ensure that they are functioning correctly. Unit tests are typically written by the developers who create the code. Unit tests are used to verify that the code meets the requirements specified in the design document. Unit tests are also used to find defects in the code.

### 5.5.7.1 Stubs and/or drivers for the database

A stub is a piece of code that is used to stand in for another piece of code. A driver is a piece of code that is used to drive another piece of code. In the case of the database, a stub or driver may be used to stand in for the database code.

### 5.5.7.2 Test cases for the database

The test cases for the database are as follows:
**Verify that the database can be accessed**. The test team will need to ensure that they can connect to the database and run queries against it.
**Verify that the database can store data**. The test team will need to insert data into the database and verify that it is stored correctly.
**Verify that the database can retrieve data.** The test team will need to query the database and verify that the data is returned correctly.
**Verify that the database can update data**. The test team will need to update data in the database and verify that the changes are saved correctly.
**Verify that the database can delete data.** The test team will need to delete data from the database and verify that it is removed correctly.

#### 5.5.7.3 Purpose of tests for the database

The purpose of the tests for the database is to verify that the database is functioning correctly. The tests will need to ensure that the database can be accessed, that data can be stored and retrieved, and that the database can be updated and deleted as needed.

#### 5.5.7.4 Expected results for the database

The following results are typically expected:

1. The database can be accessed.
2. The database can store data.
3. The database can retrieve data.
4. The database can update data.
5. The database can delete data.

#### 5.5.7.5 Stubs and/or drivers for the milk tracking system

A stub is a piece of code that is used to stand in for another piece of code. A driver is a piece of code that is used to drive another piece of code. In the case of the milk tracking system, a stub or driver may be used to stand in for the milk tracking system code.

#### 5.5.7.6 Test cases for the milk tracking system

The test cases for the milk tracking system are as follows
Verify that the milk tracking system can be accessed.
Verify that the milk tracking system can track milk production.
Verify that the milk tracking system can track milk sales.
Verify that the milk tracking system can track milk inventory.
Verify that the milk tracking system can track milk delivery.

#### 5.5.7.7 Purpose of tests for the milk tracking system

The purpose of the tests for the milk tracking system is to verify that the system is functioning correctly. The tests will need to ensure that the system can be accessed, that milk production can be tracked, that milk sales can be tracked, that milk inventory can be tracked, and that milk delivery can be tracked.

#### 5.5.7.8 Expected results for the milk tracking system

The expected results for the milk tracking system are as follows
The milk tracking system can be accessed.
The milk tracking system can track milk production.
The milk tracking system can track milk sales.
The milk tracking system can track milk inventory.
The milk tracking system can track milk delivery.

## Integration testing

Integrated testing is a process where components are tested to ensure that they work together as expected. In integrated testing, individual units of code are tested to ensure that they are functioning correctly when they are integrated with other units of code. Integrated tests are typically written by the developers who create the code. Integrated tests are used to verify that the code meets the requirements specified in the design document. Integrated tests are also used to find defects in the code.

The integration testing will begin with testing the User Interfaces. The admin must be able to access all the users of the system so it will be tested first. The database integration with the user interfaces will follow. And then we will integrate the farmers and the milk purchasing system and finally test if the overall objectives are met before realising to the final user.

## Testing procedure for integration

The test team will develop a test plan that outlines the approach that will be taken to testing the system.
The test team will develop test cases that describe the steps that need to be taken to test the system.
The test team will execute the test cases and track the results.
The test team will generate test reports that contain the results of the testing, as well as any bugs that were found.

## Validation testing

Validation testing is a process where the system is tested to ensure that it meets the requirements specified by the client. In validation testing, the system is tested to ensure that it meets the functional and non-functional requirements specified in the design document. Validation tests are typically written by the developers who create the code.

The validation of the system will begin with the login and registration forms which will validate the user inputs and authenticate them. The other validation will be conducted on the farmers side where they input their dairy records. The database should be able to validate the details of each and every user without failure.

## Testing procedure for validation

The following are the validation testing procedures
The test team will develop a test plan that outlines the approach that will be taken to testing the system.
The test team will develop test cases that describe the steps that need to be taken to test the system.
The test team will execute the test cases and track the results.

The test team will generate test reports that contain the results of the testing, as well as any bugs that were found.

### Expected results

The following are the expected results of validation testing
The system can validate users
Can track and keep all the records
The system can add or remove the users
The system can handle large amount of data

### Pass/fail criterion for all validation tests

The pass/fail criterion for all validation tests will be as follows:
**All test cases must pass**. This means that the system must meet all of the requirements that are outlined in the test cases.
**All test cases must be completed.** This means that the test team must have executed all of the test cases.
**All bugs must be fixed**. This means that the system must be free of any bugs that were found during testing.

## High-order testing

High order testing is a process where the system is put through a series of stress tests to ensure that it can handle large amounts of data. In high order testing, the system is tested to ensure that it can handle a large number of users, a large amount of data, and a large number of transactions.

There are several types of high order tests that will be conducted:
**Load testing:** Load testing is a type of test that is used to determine how the system responds to a large number of users.
**Stress testing:** Stress testing is a type of test that is used to determine how the system responds to a large amount of data.
**Performance testing**: Performance testing is a type of test that is used to determine how the system responds to a large number of transactions.
**Capacity testing:** Capacity testing is a type of test that is used to determine how the system responds to a large number of users, a large amount of data, and a large number of transactions.

### Recovery testing for Database

The purpose of the recovery testing is to verify that the database is functioning correctly. The tests will need to ensure that the database can be accessed, that data can be stored and retrieved, and that the database can be updated and deleted as needed.

### Security testing for database

The purpose of the security testing is to verify that the database is secure. The tests will need to ensure that the database can only be accessed by authorized users, that data is only accessible to authorized users, and that the database is protected from unauthorized access.

### Stress testing for database

The purpose of the stress testing is to verify that the database can handle a high volume of traffic. The tests will need to ensure that the database can be accessed by a large number of users, that data can be stored and retrieved quickly, and that the database can be updated and deleted as needed.

### Performance testing for database

The purpose of the performance testing is to verify that the database is performing as expected. The tests will need to ensure that the database can be accessed quickly, that data can be stored and retrieved quickly, and that the database can be updated and deleted as needed.

### Alpha/beta testing for database

The purpose of the alpha/beta testing is to verify that the database is functioning correctly. The tests will need to ensure that the database can be accessed, that data can be stored and retrieved, and that the database can be updated and deleted as needed.

### Pass/fail criterion for all validation tests for database

**All test cases must pass.** This means that the database must meet all of the requirements that are outlined in the test cases.
**All test cases must be completed**. This means that the test team must have executed all of the test cases.
**All bugs must be fixed**. This means that the database must be free of any bugs that were found during testing.

### Recovery testing for the milk tracking system

The purpose of the recovery testing is to verify that the system is functioning correctly. The tests will need to ensure that the system can be accessed, that milk production can be tracked, that milk sales can be tracked, that milk inventory can be tracked, and that milk delivery can be tracked.

### Security testing for the milk tracking system

The purpose of the security testing is to verify that the system is secure. The tests will need to ensure that the system can only be accessed by authorized users, that milk production can only be tracked by authorized users, that milk sales can only be tracked by authorized users, that milk inventory can only be tracked by authorized users, and that milk delivery can only be tracked by authorized users.

### Stress testing for the milk tracking system

The purpose of the stress testing is to verify that the system can handle a high volume of traffic. The tests will need to ensure that the system can be accessed by a large number of users, that milk production can be tracked quickly, that milk sales can be tracked quickly, that milk inventory can be tracked quickly, and that milk delivery can be tracked quickly.

### Performance testing for the milk tracking system

The purpose of the performance testing is to verify that the system is performing as expected. The tests will need to ensure that the system can be accessed quickly, that milk production can be tracked quickly, that milk sales can be tracked quickly, that milk inventory can be tracked quickly, and that milk delivery can be tracked quickly.

### Alpha/beta testing for the milk tracking system

The purpose of the alpha/beta testing is to verify that the system is functioning correctly. The tests will need to ensure that the system can be accessed, that milk production can be tracked, that milk sales can be tracked, that milk inventory can be tracked, and that milk delivery can be tracked.

### Pass/fail criterion for all validation tests for the milk tracking system

**All test cases must pass.** This means that the database must meet all of the requirements that are outlined in the test cases.
**All test cases must be completed**. This means that the test team must have executed all of the test cases.
**All bugs must be fixed**. This means that the database must be free of any bugs that were found during testing.

## 5.5.8 Testing resources and staffing

**Test management tool:** A test management tool is a software application that is used to plan, execute, and track the testing of a software project.

**Test case management tool:** A test case management tool is a software application that is used to manage the test cases for a software project.
**Bug tracking tool:** A bug tracking tool is a software application that is used to track the bugs that are found in a software project
**Test automation tool:** A test automation tool is a software application that is used to automate the testing of a software project.
**Performance testing tool:** A performance testing tool is a software application that is used to test the performance of a software project.

For the staffing the following staffs shall be deployed
**Project manager:** The project manager is responsible for overall project management.
**Lead test engineer:** The lead test engineer is responsible for test planning and execution.
**Test engineer:** The test engineer is responsible for writing and executing test cases.
**Software developer:** The software developer is responsible for developing the code.
**Database administrator:** The database administrator is responsible for managing the database.

### 5.5.9 Test work products

**Test plan:** The test plan is a document that outlines the approach that will be taken to testing the system.
**Test cases:** The test cases are documents that describe the steps that need to be taken to test the system.
**Test reports:** The test reports are documents that contain the results of the testing, as well as any bugs that were found.
**TestLogs:** These logs are produced as the system is being debugged in order to record all the transactions happening in the system.

### 5.5.10 Test record keeping and test log

The following are ways to keep track of test results:
**Test logs:** Test logs are documents that contain a record of all the tests that were conducted, as well as the results of those tests.
**Test databases:** Test databases are software applications that store the results of tests in a database.
**Test management tools:** Test management tools are software applications that can be used to track the results of tests.

## 5.6 Implementation Strategy

## 5.6.1.0 Introduction

Implementation is the process of converting a new or a revised system design into an operational
one.

There are several factors that need to be considered when choosing an implementation strategy for a Dairy Farming Management System. The first factor is the size of the dairy farm. If the dairy farm is large, it will require a different implementation strategy than if the farm is small. The second factor is the number of cows on the farm. If there are a large number of cows, the system will need to be able to handle a large amount of data. The third factor is the level of automation on the farm. If the farm is highly automated, the system will need to be able to interface with the farm's equipment. The fourth factor is the budget for the project. The final factor is the timeline for the project.

After considering all of these factors, the best implementation strategy for the Dairy Farming Management System is to use a phased approach. In the first phase, the system will be implemented on a small dairy farm. This will allow the team to test the system and make sure that it works as intended. In the second phase, the system will be implemented on a larger dairy farm. This will allow the team to test the system's ability to handle a large amount of data. In the third phase, the system will be implemented on a highly automated dairy farm. This will allow the team to test the system's ability to interface with the farm's equipment.

The phased approach is the best implementation strategy because it will allow the team to gradually increase the complexity of the project. This will minimize the risk of the project and increase the chances of success.

## 5.6.1.1 Implementation Phase Deliverables

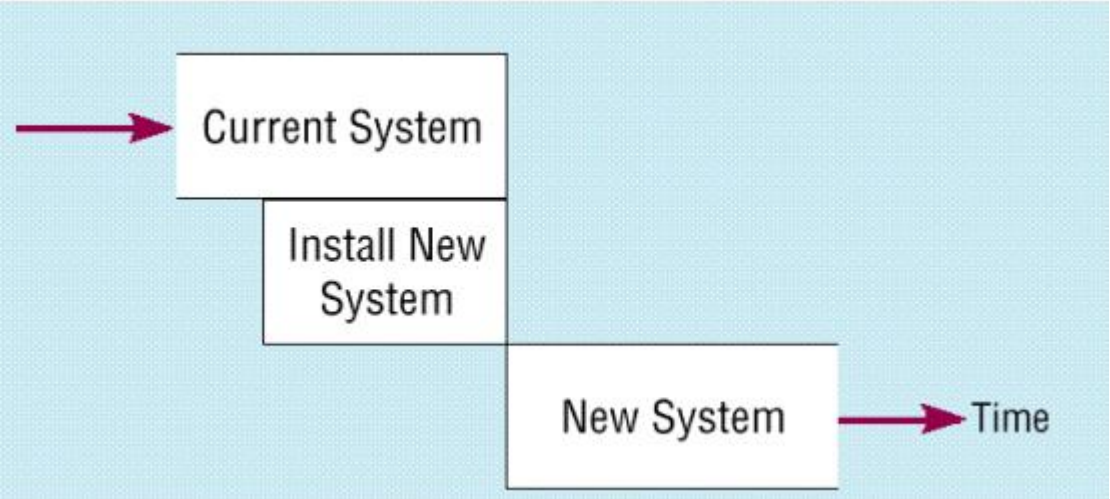
### 5.6.1.1.1 Installation & Conversion Plans

The aim of conversion is to put the tested system into operation. Conversion is the process of moving from old system to new system.
The installation and conversion plans will be used to install the system and convert the data from the old system to the new system. The software and hardware installation plan will be used to install the system on the dairy farm. The data conversion plan will be used to convert the data from the old system to the new system. The site and facility remodelling plan will be used to remodel the dairy farm to accommodate the new system. The training plan will be used to train the farmers on how to use the new system. The software maintenance plan will be used to maintain the new system.

#### 5.6.1.1.2 Conversion Plan

For this System we will have a detailed meeting with the manager of mbogoli farm and i will inform him about the conversion plan. For this System we will use Direct Installation.
The direct installation method will be used to install the new system on Mbogoli dairy farm. This method will involve installing the new system on the farm and then Feeding the data from the old system (Note Books) to the new system. This method is the best choice for the Mbogoli farm because it will allow the team to quickly install the new system and convert the data from the old system.

The diagram below illustrates direct installation




#### 5.6.1.1.3 File Conversion

Since the farm has been using manual systems where they record data in A4 sized papers the main task in this section will be inputting the already data in the system commonly known as data entry. I will then train them how to do data entry in such a way the data will be captured in the database as the data stored in the database will be easily retrieved.

#### 5.6.1.1.4 Creating Test Files

Once they get the flow of how to go about data entry, we will then create test files where I will then supervise them as they test. We will then delete the data from the database once the test is done.

#### 5.6.1.1.5 Conversion Of physical facilities

A communication network is set of methods that users work to transfer a worth message.
The institution must ensure to have the following in place:
(i)Computer with windows operating system
(ii)Stable Network
(iii)Electricity

### 5.6.2.0 Training plan

For our system we will use the One-to-one training method which will consist the following:

The training plan will involve training the farmers on how to use the new system. The farmers will be trained on how to input data into the system, how to retrieve data from the system, and how to use the system to make decisions about the dairy farm. The farmers will be given a manual that will explain how to use the system. The farmers will be given a test at the end of the training to make sure that they understand how to use the system.

The training plan will be implemented over a period of two weeks. The first week will be spent training the farmers on how to use the system. The second week will be spent testing the farmers on their knowledge of the system.

The training plan is necessary because it will ensure that the farmers are able to use the new system. The training plan will also ensure that the farmers understand how to use the system to make decisions about the dairy farm.

### 5.6.3.0 Resistance to change plan

There are several ways to reduce resistance to the system change. these include Identifying the defects in the present system, convincing them that the changeover will improve quality of work, Opening communication between the users and project teams

The first way to reduce resistance to the system change is to identify the defects in the present system. The farmers should be made aware of the defects in the system so that they can see that the new system will be an improvement.
The second way to reduce resistance to the system change is to convince the farmers that the changeover will improve the quality of work. The farmers should be given a demonstration of the new system so that they can see the benefits of the new system.
The third way to reduce resistance to the system change is to open communication between the users and project teams. The farmers should be given a chance to voice their concerns about the new system.

The resistance to change plan is necessary because it will ensure that the farmers are willing to accept the new system. The resistance to change plan will also ensure that the farmers understand the benefits of the new system.

#### 5.6.4.0 Software maintenance plan

The Software Maintenance will be as follows:
**Identification-**Whenever there is an issue the users will report the issue.

**Analysis-**The issue will then be analyzed and solutions will be proposed.
**Design**-Depending on the solutions suggested, the best solution will be designed to meet the users' specification.
**Implementation**-Once the design is completed, the I will implement the changes.
**System Testing**-After the implementation of the changes required the whole system will be tested together with the changes made.
**Use Testing**-The user who raised the concern of the issue will be given the first priority for testing the system first after the implementation of the issues.
**Deployment**-Once the user test confirms everything is fine and working as expected the system will be released to all users.

For this System in order to enhance availability of maintenance the following shall be set up in the local area of the farm
**Information Centre / Help desk:** The information centre will be responsible for answering questions about the new system. The help desk will be responsible for providing support for the new system.

**Resident expert:** The resident expert will be responsible for providing support for the new system.

**miscellaneous considerations:** The other things to consider include providing recovery and backup, disaster recovery, and PC maintenance.

The software maintenance plan is necessary because it will ensure that the new system is supported. The software maintenance plan will also ensure that the new system is able to provide support for itself.

#### 5.6.5.0 Conclusion

The above information outlines the best implementation strategy for a Dairy Farming Management System. The phased approach is the best implementation strategy because it will allow the team to gradually increase the complexity of the project. This will minimize the risk of the project and increase the chances of success.

# CHAPTER 6: RESULTS AND CONCLUSIONS

## 6.1 Results

The test results showed that the system was able to handle the data from the old system and that the farmers were able to use the system to make decisions about the dairy farm. The farmers were also able to understand how to use the system to make decisions about the dairy farm. The system also was able to facilitate buying and selling of milk to the company.

## 6.2 Conclusion

In conclusion I was able to deduce that the above tests having passed the system can well function in a dairy farm that has internet and access to a computer.

The overall project will greatly improve dairy farming on Mbogoli farm since there will be an automated system to manage the farm hence productivity will rise.

## 6.3 Recommendations

Livestock farming is a demanding, 24-7 operation. It's not for the faint of heart or those looking for an easy life. But for those willing to put in the long hours and hard work, it can be a very rewarding way of life.

Dairy farming is a particularly demanding type of livestock farming. Dairy cows need to be milked twice a day, every day, 365 days a year. They need to be fed a nutritious diet and given plenty of clean water to drink. And they need to be kept healthy and free from disease.

The developed dairy farming management system will help dairy farmers to manage their cows effectively and efficiently. Such a system can provide farmers with information on the health and productivity of their cows, and help them to make decisions on feed and herd management.

The developed Dairy farming Management System is one of a comprehensive dairy management system. It includes a cow management software application, which provides farmers with information on the productivity and reproductive status of their cows. The system also includes a feed management module, which helps farmers to formulate and manage their cows' diets it also works in tandem with the milk buying company.

The developed system is not the only dairy management system available on the market. There are a number of other systems available, from different manufacturers. Some are more comprehensive than others, and some are more expensive than others. But all of them have the potential to help dairy farmers to improve the management of their cows.

The key to choosing the right dairy management system for a particular farm is to consider the specific needs of the farm. Different farms will have different requirements, and there is no one-size-fits-all solution. But all dairy management systems have the potential to help farmers to improve the management of their cows.

# GLOSSARY

## References

Deng, X. (2017). Dairy Farming: Latest research advances. Elsevier.

Galloway, J., & Cole, D. (2017). Dairy Herd Management. John Wiley & Sons.

Kosikowski, F. V., & Cole, W. J. (2017). Cheese and Fermented Milk Foods. Elsevier.

McDonald, P., & Edwards, R. A. (2017). Animal nutrition. Pearson Education.

Nebel, B. L., & Wright, A. D. (2017). Dairy Science and Technology. CRC Press.

Schaefer, D. M. (2017). Large Dairy Herd Management. John Wiley & Sons.

Weiss, W. P. (2017). Dairy Cattle Science. John Wiley & Sons.
What is a Dairy Cow? (n.d.). Retrieved from https://www.thespruce.com/what-is-a-dairy-cow-2805181

Daly, K. J., & Magnier, M. (2017). Dairy farming: The story of milk production. New York, NY: Scholastic Inc.

Dairy Farmers of America. (n.d.). Dairy Farming. Retrieved from https://www.dfamilk.com/our-story/where- milk-comes-from/dairy-farming/

Davies, D. L. (2016). An introduction to dairy farming. Oxford, UK: Wiley-Blackwell.

Earl, G., & Beever, D. E. (2017). Dairy cow welfare. Wallingford, UK: CABI.

Lewis, R. J. (2016). The modern milkman: A sustainable delivery model for raw milk. Philadelphia, PA: New Society Publishers.

National Dairy Council. (n.d.). Dairy Farming. Retrieved from https://www.nationaldairycouncil.org/Consumers/Dairy-Farming

Wilkinson, J. (2019). The ethical dairy: A farm animals' perspective. London, UK: Bloomsbury Publishing.

Wynn, P. G. (2018). Dairy farming: Principles, practices, and profits. London, UK: CRC Press.
"Animal husbandry." Wikipedia, The Free Encyclopedia. Wikimedia Foundation, Inc., 11 Apr. 2018. Web. 17 Apr. 2018.

"Dairy Herd Management System." Wikipedia, The Free Encyclopedia. Wikimedia Foundation, Inc., 1 Feb. 2018. Web. 17 Apr. 2018.

"Dairy Management System." Wikipedia, The Free Encyclopedia. Wikimedia Foundation, Inc., 1 Feb. 2018. Web. 17 Apr. 2018.

"Dairy Farming." Wikipedia, The Free Encyclopedia. Wikimedia Foundation, Inc., 10 Apr. 2018. Web. 17 Apr. 2018.

"Agriculture." Wikipedia, The Free Encyclopedia. Wikimedia Foundation, Inc., 11 Apr. 2018. Web. 17 Apr. 2018.

"Economy." Wikipedia, The Free Encyclopedia. Wikimedia Foundation, Inc., 13 Apr. 2018. Web. 17 Apr. 2018.

Azzam, A. M., & Tamime, A. Y. (2007). Dairy Processing: Principles and Practice (3rd ed.). Oxford, UK: Blackwell Publishing.

Butterfield, W. E., & Martin, S. W. (2000). Dairy herd management (5th ed.). Upper Saddle River, NJ: Prentice-Hall.

Capper, J. L. (2008). Dairy science and technology (2nd ed.). Boca Raton, FL: CRC Press.

Fox, P. F., & McSweeney, P. L. H. (Eds.). (1998). Advanced dairy chemistry: Proteins (Vol. 1, 3rd ed.). New York, NY: Springer.

Fox, P. F., & McSweeney, P. L. H. (Eds.). (2004). Dairy chemistry and physics (2nd ed.). New York, NY: Springer.

Graham, J. M., Corbett, J. L., & dam, H. (2007). Dairy herd management: Principles and practices (6th ed.). Clifton Park, NY: Thomson Delmar Learning.

Harlander, S. K., & Qureshi, M. A. (Eds.). (2005). Dairy science and technology handbook (2nd ed.). Boca Raton, FL: CRC Press.

Jay, M. J. (2007). Modern food microbiology (7th ed.). New York, NY: Springer.

Jordan, F., & Rowe, R. (2007). Dairy processing improvement opportunities for small-scale enterprises. Food Control, 18(5), 421-429.

Kent, N. L. (2007). Dairy Goat Handbook (2nd ed.). Clifton Park, NY: Thomson Delmar Learning.

Kroger, M. (2007). Food processing plants: Design and layout (2nd ed.). Boca Raton, FL: CRC Press.

Moine, J. P. (2007). Dairy cattle nutrition (2nd ed.). Clifton Park, NY: Thomson Delmar Learning.

Nollet, L. M. L., & Toldrá, F. (Eds.). (2007). Handbook of dairy foods analysis (2nd ed.).

https://www.researchgate.net/publication/228127527_Factors_affecting_the_choice_of_an_implementation_strategy_for_a_dairy_farm_management_information_system

https://www.semanticscholar.org/paper/Phased-Implementation-of-an-Expert-System-for-a-in-a-Eckert/7181c1b8126b7f0b1a1b7d436f8feb2f428a39cf

https://www.google.com/search?q=phased+implementation&rlz=1C1CHBF_enUS747US747&oq=phased+implementation&aqs=chrome..69i57j69i60l3j0l2.3796j0j7&sourceid=chrome&ie=UTF-8

https://hbr.org/1993/07/phased-implementation

https://www.google.com/search?q=phased+implementation+advantages&rlz=1C1CHBF_enUS747US747&oq=phased+implementation+advantages&aqs=chrome..69i57j0l5.2155j0j7&sourceid=chrome&ie=UTF-8

R. J. Miller, "Integration testing," in Encyclopedia of Software Engineering, vol. 2, J. J. Marciniak, Ed. New York: Wiley, 1994, pp. 527-532.

M. J. Harrold and J. C. Jones, "Stubs and drivers," in Encyclopedia of Software Engineering, vol. 2, J. J. Marciniak, Ed. New York: Wiley, 1994, pp. 537-541.

J. C. Jones, "Validation testing," in Encyclopedia of Software Engineering, vol. 2, J. J. Marciniak, Ed. New York: Wiley, 1994, pp. 545-549.

J. C. Jones, "High-order testing," in Encyclopedia of Software Engineering, vol. 2, J. J. Marciniak, Ed. New York: Wiley, 1994, pp. 553-557.

J. C. Jones, "Recovery testing," in Encyclopedia of Software Engineering, vol. 2, J. J. Marciniak, Ed. New York: Wiley, 1994, pp. 561-565.

# Appendices Appendix1:

## Sample Code

```
Imports FireSharp.Config
Imports FireSharp.Response
Imports FireSharp.Interfaces
Public Class Admin
    Private fcon As New FirebaseConfig With
        {
        .AuthSecret = "m5KIfvB60pJPjJx8Ti15R08LqREXCaZhlRcfddL8",
        .BasePath = "https://dairyfarming-e999c-default-rtdb.firebaseio.com/"
    }
    Private client As IFirebaseClient
    Public Shared Fname
    Private Sub Admin_Load(sender As Object, e As EventArgs) Handles MyBase.Load
        Label1.Text += Fname + "!"
        Try
            client = New FireSharp.FirebaseClient(fcon)
        Catch ex As Exception
            MessageBox.Show(ex.ToString())
        End Try
    End Sub

    Private Sub Button1_Click(sender As Object, e As EventArgs) Handles AddButton.Click
Condition
        Dim NewUser As New MyUser() With
         {
        .UserId = UserIdTextBox.Text,
        .FullNames = NameTextBox.Text,
        .Password = PasswordTextBox.Text,
        .UserRole = UserComboBox.Text
         }



        Dim setter = client.Set("Users/" + UserComboBox.Text + UserIdTextBox.Text, NewUser)
        Dim res = client.Get("CounterTwo")
        Dim Counter = Integer.Parse(res.ResultAs(Of String)) + 1

        Dim roll As New RoleUserId()
        With roll
```

No issues found  Ln: 13  Ch: 9  SPC  CR

## List of figures

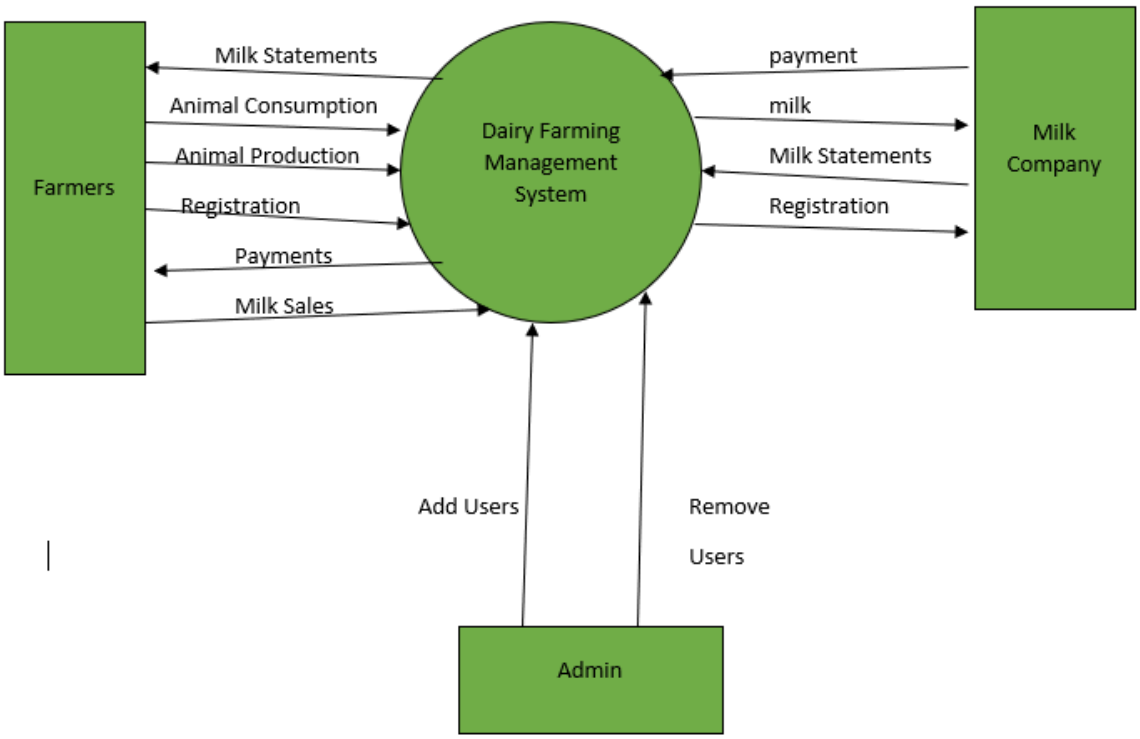

Farmers
Milk Statements
Animal Consumption
Animal Production
Registration
Payments
Milk Sales
Dairy Farming Management System
payment
milk
Milk Statements
Registration
Milk Company
Add Users
Remove Users
Admin

Fig 2

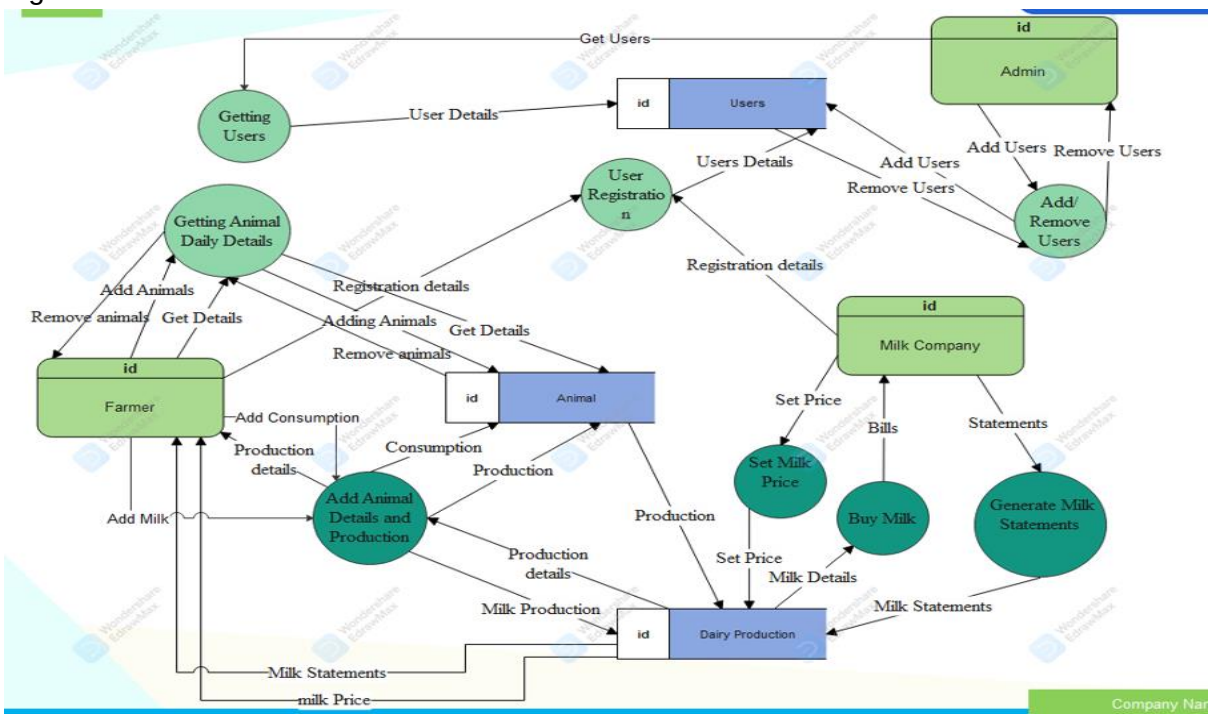
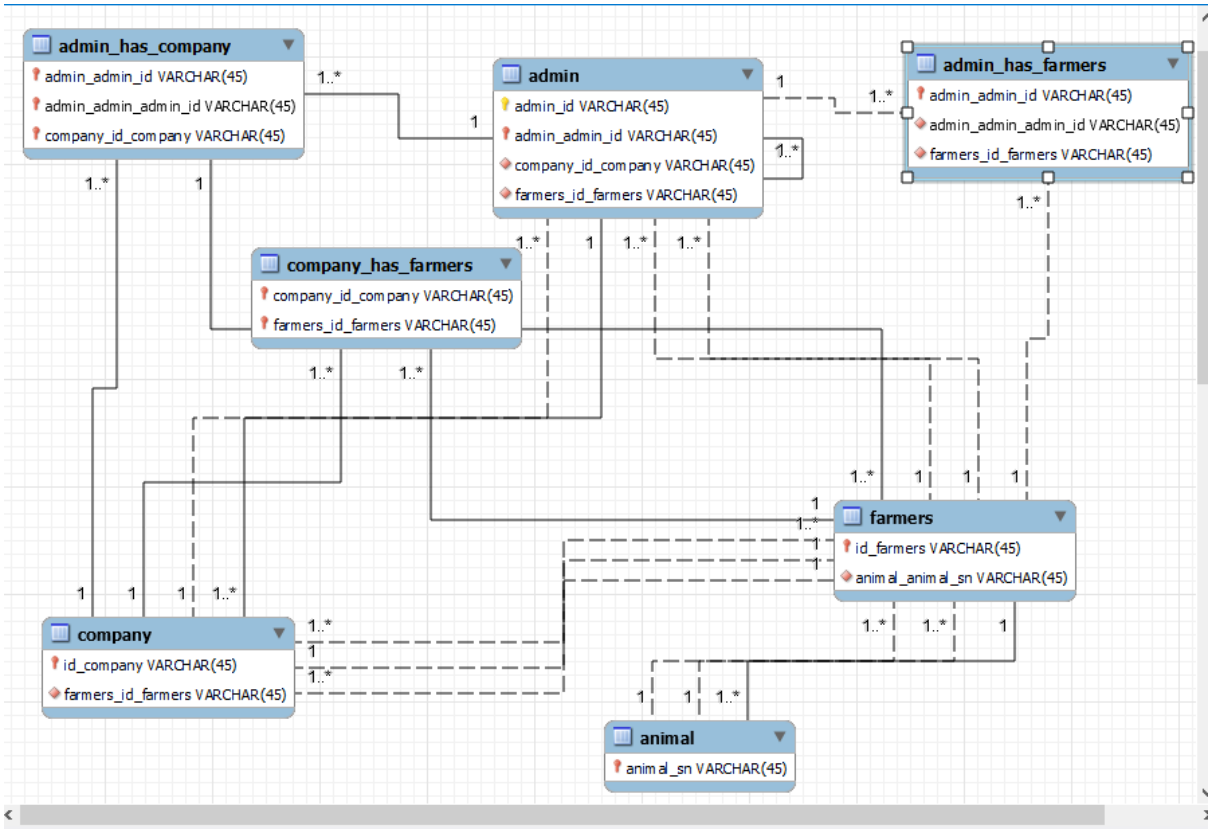

Get Users
id
Admin
Getting Users
User Details
id
Users
Users Details
Add Users
Remove Users
Add Users
Remove Users
User Registration
Add/ Remove Users
Getting Animal Daily Details
Registration details
Registration details
Add Animals
Remove animals
Get Details
Adding Animals
Get Details
Remove animals
id
Milk Company
id
Farmer
id
Animal
Set Price
Bills
Statements
Add Consumption
Production details
Consumption
Production
Set Milk Price
Buy Milk
Generate Milk Statements
Add Milk
Add Animal Details and Production
Production
Production details
Set Price
Milk Details
Milk Production
Milk Statements
id
Dairy Production
Milk Statements
milk Price

Fig 3


admin_has_company
admin_admin_id VARCHAR(45)
admin_admin_admin_id VARCHAR(45)
company_id_company VARCHAR(45)
admin
admin_id VARCHAR(45)
admin_admin_id VARCHAR(45)
company_id_company VARCHAR(45)
farmers_id_farmers VARCHAR(45)
admin_has_farmers
admin_admin_id VARCHAR(45)
admin_admin_admin_id VARCHAR(45)
farmers_id_farmers VARCHAR(45)
company_has_farmers
company_id_company VARCHAR(45)
farmers_id_farmers VARCHAR(45)
farmers
id_farmers VARCHAR(45)
animal_animal_sn VARCHAR(45)
company
id_company VARCHAR(45)
farmers_id_farmers VARCHAR(45)
animal
animal_sn VARCHAR(45)