Dairy Farm Record Keeping System

**Jason Muthuo Gikonyo**

**Student ID: 100949**

**An Information System Project 1 documentation submitted to the Faculty of Information Technology in partial fulfilment of the requirements for the award of the Bachelor’s Degree in Business Information Technology of   
Strathmore University**

**Faculty of Information Technology  
Strathmore University  
Nairobi, Kenya**

**September 2018**

# Declaration

I declare that this work has not been previously submitted and approved for the award of a Bachelor’s degree by this or any other University. To the best of my knowledge and belief, the documentation contains no material previously published or written by another person except where due reference is made in the documentation itself.

Student’s signature:

………………………………….………….. [*Signature*]

………………………………….………….. [*Date*]

**Approval**

The Information System Project 1 documentation of Jason Muthuo Gikonyo was reviewed and approved *(for examination)* by:

Supervisor’s signature:

………………………………….………….. [*Signature*]

………………………………….………….. [*Date*]

# Abstract

This project aims to solve the issue of the lack of good record keeping among farmers. It is using a computer-based solution that automates the record keeping process and provides brief analysis from the data obtained. The lack of record-keeping among farmers is a serious economical concern which impacts farmers by being unable to track the farms status and expenses properly. A computer-based solution can be used to streamline this process hence ensuring that they can keep large quantity of records safely and providing the analysis of the data to facilitate decisions. Though the problem has already been addressed for large scale farms, the small-scale farms do not have an economical solution to the problem. This means that this project will be able to address the problem but at a lower cost. The project will utilize the Extreme Programming methodology which will have an emphasis on time management and fast development with constant testing. This means that the final product will be of high quality and delivered on time.

**Table of Contents**

[Declaration ii](#_Toc527213802)

[Abstract iii](#_Toc527213803)

[List of Figures vi](#_Toc527213804)

[List of Tables vii](#_Toc527213805)

[Acknowledgements viii](#_Toc527213806)

[Chapter 1: Introduction 1](#_Toc527213807)

[1.1 Background 1](#_Toc527213808)

[1.2 Problem Statement 1](#_Toc527213809)

[1.3 Aim 1](#_Toc527213810)

[1.4 Specific Objectives 2](#_Toc527213811)

[1.5 Justification 2](#_Toc527213812)

[1.6 Scope and Limitations 2](#_Toc527213813)

[Chapter 2: Literature Review 3](#_Toc527213814)

[2.1 Introduction 3](#_Toc527213815)

[2.2 Farming Record Keeping 3](#_Toc527213816)

[2.3 Technologies used for farm records and analysis 3](#_Toc527213817)

[2.4 Desktop-based record-keeping systems 4](#_Toc527213818)

[2.5 Gaps(summary) 4](#_Toc527213819)

[Chapter 3: System Development Methodology 5](#_Toc527213820)

[3.1 Introduction 5](#_Toc527213821)

[3.2 Extreme Programming Methodology 5](#_Toc527213822)

[3.2.1 Planning 6](#_Toc527213823)

[3.2.2 Designing 6](#_Toc527213824)

[3.2.3 Coding 6](#_Toc527213825)

[3.2.4 Testing 6](#_Toc527213826)

[3.3 Method to be used to Gather the Functional and Non-Functional Requirements 7](#_Toc527213827)

[3.4 List of Design Diagrams that will be drawn in Chapter 4 7](#_Toc527213828)

[3.5 List of Development Tools that will be used 7](#_Toc527213829)

[3.6 Method to be used to test the developed system 8](#_Toc527213830)

[3.7 Domain of Execution 8](#_Toc527213831)

[3.8 Proposed Modules and System Architecture 8](#_Toc527213832)

[Chapter 4: System Analysis and Design 9](#_Toc527213833)

[4.1 Introduction 9](#_Toc527213834)

[4.2 System Analysis 9](#_Toc527213835)

[4.3 System Design 9](#_Toc527213836)

[Chapter 5: System Testing 13](#_Toc527213837)

[5.1 Introduction 13](#_Toc527213838)

[5.2 Test Environment 13](#_Toc527213839)

[5.3 Test Cases 13](#_Toc527213840)

[5.4 Test Results 13](#_Toc527213841)

[Chapter 6: Conclusion and Recommendations for Future Work 14](#_Toc527213842)

[6.1 Conclusion 14](#_Toc527213843)

[6.2 Recommendations for Future Work 14](#_Toc527213844)

[References 15](#_Toc527213845)

[Appendix A: Timeline of Activities 16](#_Toc527213846)

[Appendix B: TurnItIn Similarity Index 17](#_Toc527213847)

[Appendix C: User Manual 18](#_Toc527213848)

[Appendix D: Marking Guide 19](#_Toc527213849)

# List of Figures

[Figure 3.1: Extreme Programming Methodology. 5](#_Toc527452932)

[Figure 3.2: System architecture 8](#_Toc527452933)

[Figure 4.1: Dairy Farm record-keeping system use case 10](#_Toc527452934)

[Figure 4.2: Dairy record-keeping system sequence diagram 10](#_Toc527452935)

[Figure 4.3: Class Diagram 10](#_Toc527452936)

[Figure 4.4: Database Schema 10](#_Toc527452937)

[Figure 4.5: System Wireframe 10](#_Toc527452938)

# List of Tables

[Table 1: Requirement Specification 9](#_Toc527452987)

# Acknowledgements

# Introduction

## Background

This project aims to solve the issue of the lack of good record keeping among farmers using a computer-based solution that automates the record keeping process and provides brief analysis from the data obtained. The issue of inability of farmers to keep good records is one which is caused by the lack of the fundamental knowledge of its importance.

This issue is also caused by lack of good record keeping systems whether digital or manual. This prevents them from having the ability to analyse the productivity of their cattle in relation to the produce and various inputs used (example animal feeds). It also affects the farmer economically since without good records, they are not able to keep track of the money inflows and outflows leading to risky business decisions which brings in a loss.

This project is important since it will provide a farmer a simple system to follow when inputting new records. It will provide a brief analysis on the data which can be used in decision-making.

## Problem Statement

Farmers, especially in small and medium have a problem in the administrative component of farming. This is brought on by how farmers perceive tasks like record keeping. They believe that they are unnecessary tasks, yet they have a large impact in farming. This brings about losses since they rely on their memory for all the farming details. In the dairy farming, records are seriously important since details like vaccinations and diseases records, are important since it will allow a farmer to know the status of the cows.

## Aim

This project aims to create a platform for farmers to perform record keeping for their farms and provide them with some analysis from the data provided, that will be by providing farmers with a computer-based system will a facility to input the various data like seeds bought and the produce gained. The project has an analysis module which will give simplified analysis on data provided.

## Specific Objectives

* To identify a simple and comprehensive way of making and keeping records.
* To review techniques for creation of computer-based solutions like information systems and programs.
* To develop a computer-based solution that will help in record keeping and their analysis to make decisions.
* To test the created record keeping system using sample data from a small-scale farm.

## Justification

Farmers stand to benefit from the implementation of the system since it will give them a quality method of making and keeping farm records in a way that will allow them to revisit them and use them as a guide. The project will ensure that the records will be stored and are easily retrieved. The project will have an impact especially in the modern period since farms are becoming more automated. Unlike other proposed and implemented solutions, this project will seek to be cost effective means to give farmers a means to keep good farms because of the analysis gained.

## Scope and Limitations

This project covers the following areas like A recording section where the user will be able to input records. A review section where previously inputted data can be viewed. An analysis section that will present the data to the user. The project does not contain an accounting section though will have slight analysis based on expenditure and income.

Limitations that may occur are a lack of proper knowledge of farming operations which will hinder the effectiveness of the project. This can be overcome by researching and talking to a shareholder beforehand.

Secondly, lack of enough time to complete a comprehensive project. This can be slightly fixed using good planning and reduction of the project scope.

Furthermore, an issue with multiple bugs in the project due to inexperience which can be overcome by careful planning and learning more about the language to be used.

# Literature Review

## Introduction

One of the most underrated parts of farming is the administrative section that is the record keeping and analysis. This is due to the misconception that it is not important part and has no impact on anything (Harlow, 2018). This is far from the truth since records are the backbone of modern farms. They are the physical representation of the farms productivity. A record can be used as means of keeping track of the farms financial state and can be used to document the medical background of each cow hence can be used to give credibility to the quality of the products you sell.

This chapter will seek to explain the problem of record keeping among farmers in detail. It will bring out the importance of record keeping and the different types of records that can be kept by a farmer. It will, as well, address the proposed solution in depth by explain the choice of the technology to be used. It will further explain the reasons the solution is important, with respect to multiple factors such as cost.

## Farming Record Keeping

In farming, the most underrated part is the administration section especially record keeping. Record keeping is used to bring accountability in the farm. For cattle, it is essential since it allows for proper management of the cattle ("Record keeping and management planning", 2018). These records can detail many different things including breeding records, financial records and others ("Record keeping | Infonet Biovision Home.", 2018). This covers all aspects of the farm hence allowing the farmer to keep track of all their cattle and finances as well as their stock easily. Records allow for different functions including legal functions where a farmer can use the feeding and medical records to show the quality of the products especially if it is meat ("Record keeping and management planning", 2018).

## Technologies used for farm records and analysis

Most basic of technology used for record keeping is using ledges or papers. This is using hand written notes which are dependent on the writer’s perspective. This means that it is difficult since it means they are not reliable. It is also time consuming since the layout of the records needs to be recreated every time. It has restricted analysis capability ("Establishing and Using a Farm Financial Record-Keeping System - eXtension", 2018). Mobile can also be used to keep records using either USSD technology which relies on an intermediary company who will store this content for the farmer, but it is extremely hard process since it is not a very user-friendly system and lacks many components that might make it easy to use. For mobile, we could also use android-based applications. This are used as a substitute for desktop and web-based systems. They have some capacity to perform analysis of the data.

Web based applications can be used as well. They will rely on a hosting company for the website. Will require the farmers have access to the internet to access the application. It has very good storage system and can perform high analysis. Lastly desktop-based applications are also used to make, store and analysis records and it does not rely on access to the internet to function. It will need a computer to run and will need a computer literate individual to use it.

## Desktop-based record-keeping systems

This are the most extensive type of record keeping and analysis system. These systems are most common since they are the easiest to manage and can be used by individual farmers without requiring any connection to the internet or another company. It has faster data processing than most other types of technology. They provided the most intensive analysis of data that the farmer provides ("Establishing and Using a Farm Financial Record-Keeping System - eXtension", 2018). They are easiest to learn how to uses and can be easily upgraded and new modules added but takes more time to learn how to uses. The initial cost of using the desktop-based system is high and may not be used by new farmers ("Establishing and Using a Farm Financial Record-Keeping System - eXtension", 2018).

## Gaps(summary)

In summary, a desktop-based solution to the record keeping issue is required and should be pursued to reduce the losses the farmers incur due to the lack of comprehensive planning using data they have. Though for small scale farmers, a computerised solution may cost them initially, they will be able to recover the costs and will increase the productivity of the farm since to gain the most from it, it will need to be properly managed and future actions should be planned ("Establishing and Using a Farm Financial Record-Keeping System - eXtension", 2018). It will also require the farmer to learn how to use a computer to properly use the solution properly. They may hire someone else to use the system.

# System Development Methodology

## Introduction

This project will utilize the Object-Oriented Analysis and Design as the main approach. This is because this project works as a collection of objects that work together to get a result. This approach also a developer to create the system using the divide and conquer method which will allow for an easier analysis and design of the project. This allows for code-reuse therefore enhancing the systems efficiency and its quality. OOAD utilizes a combined design and analysis to allow for improved communication between the users and the developer.

The project is developed using the Extreme programming methodology (XP). XP is an example of the agile software development methodologies that relies on adaptability and fast development of projects. It is perfect for this project since it utilises time management and fast development of good systems. XP ensures that the project is done within a strict time frame.

## Extreme Programming Methodology

A close up of text on a white background

Description generated with high confidence

Figure 3.1: Extreme Programming Methodology.

Extreme programming is an agile methodology. It was created to create systems in the shortest time possible. It was created to create adaptable and agile systems with strict time frames. This methodology has a high degree of customer involvement (Programming, 2018). The customer is treated like the part of the development team. This means that the customers views are treated highly. XP has the least iteration cycles therefore it is required that the system is created as fast as possible to allow for the product to be delivered on time. XP has five stages in its software development life cycle which make sure that the product is created is of high quality and is created in time.

### Planning

This is the first stage of the methodology in which the goals of the entire project are set, and the cycles are determined. This will mean that the client will be met, and the requirements will be found, and the timeline will be set. Afterwards, work will begin. This stage is critical and will be revisited often when needed to get all the project’s requirements.

### Designing

At this stage, the developer would create a simple design after defining the main parts of the future code since simplicity is the main principle of the XP methodology. This stage is where all code is designed, and the GUI screens are designed. This is the backbone of the project since it forms the foundation of project which can accept any change since the beginning system is simple and can easily modified.

### Coding

In XP, refactoring is a major part since code is made simple and allows for code reuse depending on the area without losing functionality. The project is constantly being integrated in a repository since this methodology gives priority to coding than any other task to ensure that the final product is of high quality (Nayab, 2018).

### Testing

The testing phase is not an individual phase but occurs during the development phase. This is done to get all bug and error in code early to ensure that the system will be perfect before release. It is done early to get the customer’s review of the system and get recommendations on what is needed. This makes sure that the system is created to the client’s specifications.

## Method to be used to Gather the Functional and Non-Functional Requirements

Performing site visits to observe how a dairy farm is handled and which details need to be recorded.

Reviewing documentations and other details from sources to get an understanding about other similar systems.

Conducting interview with different dairy farmer to get an understanding of the details they need to record and get their thoughts on how a system should work to help them.

## List of Design Diagrams that will be drawn in Chapter 4

Use case diagram. This is a diagram that is used to identify, clarify and categorise system requirements. It is necessary since it holds functional requirements un an easy to read and tracking format. They also represent the goal between an actor and the system.

Class diagram. This is a diagram that are used to show the classes of the system, their interrelationships and the operations and attributes of the classes. It is essential since it is used to inform about the system’s working in the form of the classes.

Sequence diagram. This is a diagram that is used to show the interactions between objects in the sequential order those interactions occur. This diagram will be used to represent how the system flows between processes handled by classes.

Database schema. This is a diagram that is used to show how the different tables and how they relate to one another. This is important since it show how the tables which are used to store the data are related to each other and how data from each can be used with relation to each other.

GUI diagram. This is a diagram that is used to represent how the screens will look. This includes wireframes which are a design mock-up of the proposed graphical windows. They are important since they can be used as a blueprint of the windows and can be customised earlier to improve its user-friendliness.

## List of Development Tools that will be used

The project is using Java 8 which provides a simple but comprehensive library to create robust systems that work very well. The IDE to be used will be NetBeans which is the most user-friendly IDE with the best tools to develop java-based projects. The project will utilise a MySQL database management system. This is because it is a good DBMS with good control measures which are perfect for this kind of projects.

## Method to be used to test the developed system

Stress testing which is a method of testing that can be used to check if the performance of the software is satisfactory under unfavourable conditions. This is necessary for this project since it has an analysis module which could fail if conditions are extreme.

Functional testing which is a testing method which ensures that the functional requirements are met and work as needed. It is important since it ensures that the system works as intended.

## Domain of Execution

This project is a desktop-based system since the clients for this project will easily be able to use a computer easily and most of the clients do not have good internet connection or none. This will mean that they will not be able to use a web-based system.

## Proposed Modules and System Architecture

The project has the following modules: a module for inputting records from the farm, a module to display the data that is stored and finally a module to display the analysis of the data

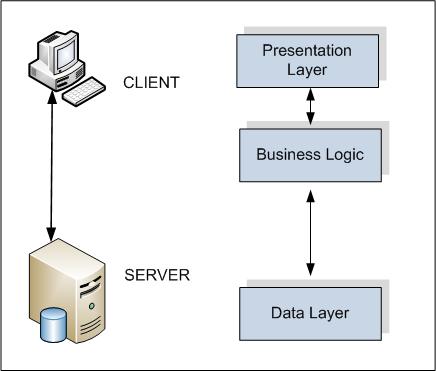
.

Figure 3.2: System architecture

# System Analysis and Design

## Introduction

In this chapter, the project’s functional and non-functional requirements will be stated alongside a review from the shareholder collaboration that lead to the chosen requirements. This chapter will also contain the system design diagrams. This diagram will be the proposed design and are subject to changes if required. They are the foundation and can be used as a blueprint of the system.

## System Analysis

|  |  |  |
| --- | --- | --- |
| Requirement ID | Requirement Category | Requirement Description |
| FR1 | Functional | The system allows a user to input new records into the database. |
| FR2 | Functional | the system allows a user to update older records. |
| FR3 | Functional | The system allows a user to modify records. |
| FR4 | Functional | The system allows an administrator to add a new user. |
| FR5 | Functional | The system displays the current records in tables for the user to view. |
| FR6 | Functional | The system presents the data in the saved records in an analysis for the user. |
| NFR1 | Security | The system encrypts all passwords to ensure security. This prevents unauthorised access to the system which can easily be gained if the passwords in the database are not encrypted |
| NFR2 | Performance | The system should have a high performance even in extreme conditions. This is to ensure that the system will be able to handle data that it receives and keeps records. |
| NFR3 | Usability | The system should be easy to use and learn. This is to make it easier for farmers to get used to the system faster and utilize it as it is intended. |
| NFR4 | Usability | The system should come with documentation to facilitate future updates and maintenance. |
| NFR5 | Reliability | The system should work efficiently and be reliable. This to ensure the protection of the data saved by the farmers which are critical for the best possible running of the system. |

Table 1: Requirement Specification

The shareholder has been presented the above requirements and have reviewed it. They believe that the above stated requirements are those necessary for the system. The rating given for the above system was 8 out of 10 and was advised that the requirement about an accounting module was not required and needs to be removed.

## System Design

Figure 4.1: Dairy Farm record-keeping system use case

Figure 4.2: Dairy record-keeping system sequence diagram

Figure 4.3: Class Diagram

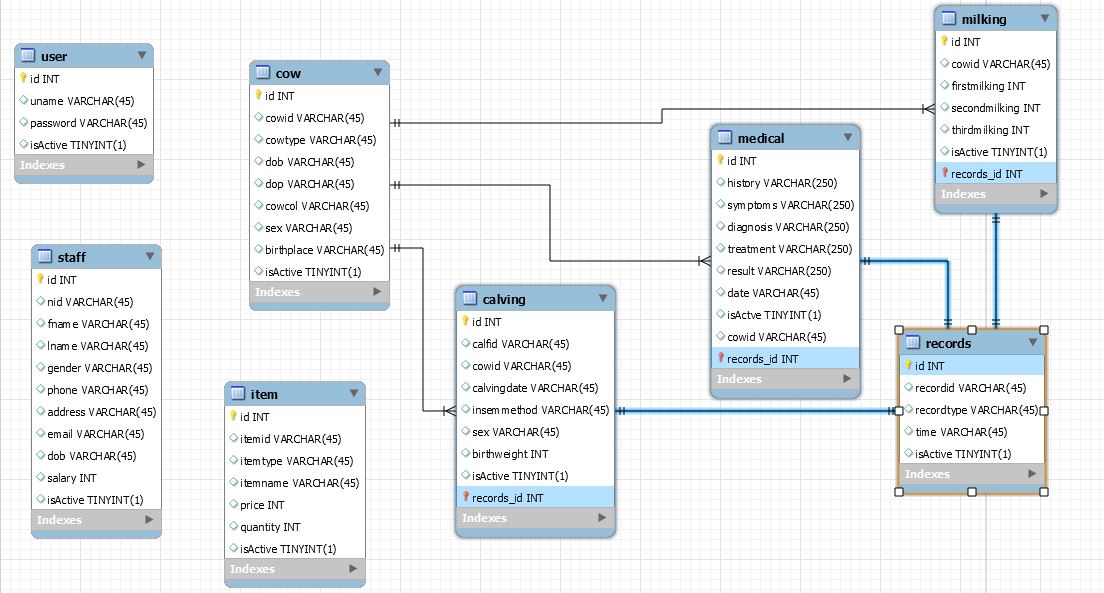


Figure 4.4: Database Schema

A screenshot of a cell phone

Description generated with very high confidence

Figure 4.5: System Wireframe

# System Testing

## Introduction

This chapter will show the several tests and the result of those test cases. This chapter will help ensure that the requirements specified will be properly executed. Test are used to make sure that the system runs as needed. The test environment will be specified to ensure that the future tests done get the same results. This makes it certain that if the system is run and retested in an another computer with the same specification as the original test environment

## Test Environment

The test environment is a Lenovo Yoga 510-14ISK with an Intel core i3-6006u with a CPU clock speed of 2.00 GHz. The test computer has a RAM of 4 GB. The system utilizes a Windows 10 Home operating system and has a system type of 64 bit operating system with a x64 bit processor. The screen size of the system is 14 inches with a screen resolution of 1366 x 768. The system utilises a MySQL server with a Java 8 environment.

## Test Cases

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Test ID | Related requirement | Inspection check | Pre-condition | Test Data | Priority Level |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

Table 2: Test Cases

## Test Results

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Test ID | Expected result | Actual result | Status | Recomendation |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

Table 3: Test Results

# Conclusion and Recommendations for Future Work

## Conclusion

***Notice the singular form (not “Conclusions”)***

## Recommendations for Future Work

# References

1. Record keeping and management planning. (2018). Retrieved from <https://www.dpi.nsw.gov.au/animals-and-livestock/beef-cattle/management/business-management/record-keeping-mgt>
2. Record keeping. (n.d.). Retrieved from <https://www.infonet-biovision.org/AnimalHealth/Record-keeping#simple-table-of-contents-7>
3. Harlow, I. (2018). Good recordkeeping benefits small farms - Farm and Dairy. Retrieved from <https://www.farmanddairy.com/top-stories/good-recordkeeping-benefits-small-farms/236647.html>
4. Establishing and Using a Farm Financial Record-Keeping System - eXtension. (2018).Retrieved from <https://articles.extension.org/pages/11140/establishing-and-using-a-farm-financial-record-keeping-system>
5. Programming, L. (2018). Life Cycle of Extreme Programming. Retrieved from <https://hygger.io/blog/life-cycle-of-extreme-programming/>
6. Nayab, N. (2018). Understanding the Extreme Programming Life Cycle Phases. Retrieved from <https://www.brighthubpm.com/methods-strategies/88996-the-extreme-programming-life-cycle/>

# Appendix A: Timeline of Activities

***Add a visible diagram that shows the work to be done during specific periods in the course of the project. This should be in the form of a Gantt chart.***

****

# Appendix B: TurnItIn Similarity Index

A screenshot of a cell phone

Description generated with very high confidence

# Appendix C: User Manual

# Appendix D: Marking Guide