



COLLEGE OF COMPUTING AND INFORMATION SCIENCES
SCHOOL OF COMPUTING AND INFORMATICS SCIENCES

THE PROJECT MANAGEMENT PLAN REPORT

FOR THE LIVESTOCK RECORDS KEEPING SYSTEM

BY

GROUP 9

84%.

A PROJECT REPORT SUBMITTED TO THE SCHOOL OF COMPUTING AND INFORMATICS
TECHNOLOGY
FOR THE PROGRESS OF THE LIVESTOCK RECORDS KEEPING SYSTEM.
ACCOMPLISHED IN PARTIAL FULFILMENT OF FINAL EXAMINATION FOR FIRST YEAR
SEMESTER TWO OF BACHELOR OF
SCIENCE IN COMPUTER SCIENCE OF MAKERERE UNIVERSITY

LECTURER

MADAM FIONA SSOZI

DEPARTMENT OF COMPUTER SCIENCE

SCHOOL OF COMPUTING AND INFORMATICS TECHNOLOGY, MAKERERE UNIVERSITY

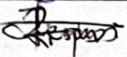
APRIL, 2010

MASIIKA RAYMOND GODWIN.

DECLARATION

We Group 9 hereby declare that this project report is original and has not been published and/ or submitted by any other group to the instructor of this unit for the project award of Makerere University year one semester two in the department of computer science or any other course before.

NAME	STUDENT NUMBER	REGISTRATION NUMBER	SIGNATURE
MUGUME IAN	1800725478	18/U/25478/PS	
OKELLO ANDREW PETERS	1800725473	18/U/25473/PS	
OKORI BARBARA	1800700547	18/U/547	
AKANKUNDA SHIVAN	1800725536	18/U/25536/PS	

MASIIKA RAYMOND 1800725504 18/U/25504/PS 

1 - 05

2 - 10

3 - 09

4 - 18

5 - 07

6 - 20

7 - 10

8 - 05

84%

Well Done!

Table of contents

1. Declaration.....	2
2. Acknowledgement.....	4
3. Introduction.....	5
3.1 Features of the proposed system.....	5
3.2 Functionality.....	5
4. System Analysis.....	6
4.1 Analysis of the existing system.....	6
4.2 Problem in the current existing system.....	6
4.3 Recommendations.....	6
5. Feasibility report.....	7
5.1 Background.....	7
5.2 The Current Situation.....	8
5.3 The Proposed Solution.....	8
5.4 Costs and Benefits.....	8
5.5 Recommendations	9
6. Requirements elicitation technique.....	10
6.1 Interviews.....	10
7. Requirements Analysis.....	11
7.1 User requirements.....	11
7.2 System requirements.....	11
7.3 Functional requirements.....	11
7.4 Non-functional requirements.....	12
8. Data Flow Diagram.....	13
8.1 Context diagram.....	13
8.2 Level 0 DFD.....	14
8.3 Use Case Diagram.....	15
8.4 Entity Relationship Diagram.....	17
8.5 Activity Diagram.....	18
9. Architectural Design.....	19
10. User Interfaces and their usefulness.....	24
11. Recommendation for a Conversion Strategy.....	29
12. User Documentation.....	30
13. Evaluation Forms.....	36
13.1 A member's evaluation form of other members' effort in the group.....	36
13.2 Self Evaluation form.....	37

ACKNOWLEDGEMENT

It took nearly one month to finally come up with a group project on **LIVESTOCK RECORDS KEEEEPING SYSTEM**. We would like to thank people who have helped us throughout the duration of project development. All the wonderful ideas, everything. It was a great experience working on the project of *System Analysis and Design*.

It actually still remains that no project ever can be executed proficiently and efficiently without sharing the particular ideas, technical expertise and innovative thoughts put forwarded by the technical and non-technical veterans.

We are also grateful to our lecturer madam **Fiona Ssozi** for being constantly there for us whenever we needed help. We are grateful too because everyone at the livestock farm were friendly to us. Great thanks goes to Mr. Denis Kamugine the sole owner of the Livestock Farm for all the information he provided to us about records keeping.

Last but not least, we would like to thank all those who have helped us in various ways. I mean we had a few challenges here and there but they were always there to help.

Problem Statement

Kamugine Dairy farm is located in Wakiso district, Nsangi along Masaka road. Mr. Denis Kamugine the owner of the dairy farm keeps Exotic Friesian cows because of the high quantities of milk they produce. His employees do the milking of the cows which they later sell to the customers near the farm.

Daily records of the farm activities are manually kept in books being entered in by his employees and later Mr. Denis balances the books himself. Mr. Denis finds difficulties in manual calculations, retrieving past records and also report generation.

3. Introduction

Livestock Records keeping System is a software application to maintain the records related to sales, incomes, expenses, profits and losses, records search, stock and report generation and all necessary requirements for the farm to manage day to day activities. The main objective of the application is to automate the existing system of manually maintaining the records of sales, incomes, expenses, profits and losses, records search, stock and report generation.

3.1 Features of the proposed system.

Our proposed system will be able to have the following features once implemented to use;

1. The system will be able to automatically calculate the total income, profits and losses, expenses, stock.
2. Records are to be maintained in a database and any record related to the farm is generated and edited easily.
3. Keeps record of consumables and delivery dates of cows.
4. Customised report designing and generating.
5. Configurable as per user's requirements.
6. A search button to look for past details of records on the farm.
7. Users will be able to create new accounts and login easily because its customisable and configurable.
8. It will be easy to use because of its friendly interface and since most people today are using Android applications.

3.2 Functionality

- 1) More economical and safer means of storing and keeping track of information.
 - 2) Easier access to information like management reports and stock as well as more accurate and faster results from statistical analyses.
 - 3) Reduces errors and eliminating of long and repetitive manual processing.
 - 4) Greater accountability and transparency in operations.
 - 5) More reliable security for sensitive and confidential information.
 - 6) User friendly interface.
 - 7) Fast access to the database.
 - 8) Search facility for past records.
 - 9) Less error.
- 10) All book keeping difficulties in managing the farm activities have been rectified by implementing the computerisation

4. System Analysis.

4.1 Analysis of the existing system.

The existing system deals mainly with Exotic Friesian cows because of the plenty of liters of milk produced per day. Mr.Denis Kamugine the owner of the dairy farm has one kraal for all his cows each with their respective names like Susan, Matilda, Bihogo and many others which help him distinguish them from one another. His employees collect liters of milk both in the morning and evening time in twenty literjerry cans and later sell them to nearby customers around the farm, a cup of milk being sold at one thousand Uganda shillings only.

He keeps track of farm activities through daily record keeping where all the farm records are kept in books. Record keeping is mainly done by him when he comes back from his official government work in the evening time at around five o'clock. Information recorded includes vaccines and feeds like dregs from supplier, sales, liters of milk collected per day, profits and losses, worker details and expenses. Losses are made when milk gets spoilt, cows get sick and die, cows are stolen and when the workers steal money from the farm earnings. Profits are made when milk and cows are sold and other products. Profits and losses are calculated manually in record books of the farm once in a while because of it being tiresome to calculate the past records.

4.2 Problems in the Current system.

- 1) Difficulty in report generating: More calculations are required to generate reports of the previous records and financial statements.
- 2) A lot of paper work: Since almost every record is kept in books, it is liable to be damaged by water, fire or easily get lost.
- 3) Lack of proper storage for the records books.
- 4) Slow retrieval of data especially from records books of different years to obtain information like sales ,profits and losses.
- 5) Time consuming: A lot of time is taken to calculate monthly profits and losses, reports, salaries for workers etc.
- 6) Manual control: All calculations to generate the report is done manually so there is greater chance to committing errors in the calculations.

4.3 Recommendations.

The only solution for the problem is to adopt the computerised support system to support a records keeping system which will help in the accurate and faster managements.

- 1) The system should be able to generate monthly and yearly reports about income, expenditure, profits and losses.
- 2) The system should be able to update the livestock information like stock of feeds left, profits and losses of recent days.
- 3) The system should allow a user to carry out a simple search about past records to quickly retrieve data easily.
- 4) The system should be able to keep records of consumables and delivery dates of cows.

- 5) The system should be able to keep records in a database and any record related to the farm is generated and edited easily.

5. Feasibility study report.

5.1 Background

Using the Mnemonic "SCOPE", the terms of reference for the Livestock Records Keeping System are the following;

S – System Boundary: The system will serve the farm owner, employees and generally the Farm management.

C - Constraints: Non-functional requirements like retrieving password, user registration and user login

O – Objectives: System that supports the keeping of Livestock records of the farmer at his farm.

P – Permission: Farm Management (Farm Owner)

E – End products: Livestock Records Keeping System(LRKS)

Reasons for the Feasibility study.

- ❑ Define the problem.
- ❑ Set project bounds.
- ❑ Identify users of the system.
- ❑ Specify resources to be made available for the project.
- ❑ Propose general hardware & software system options.
- ❑ Perform a value assessment.
- ❑ Assess project risks.
- ❑ Recommend whether to proceed with the project or abandon it.

5.2 The current situation

The existing system deals mainly with Exotic Friesian cows because of the plenty of litres of milk produced per day. Mr. Denis Kamugine the owner of the dairy farm has one kraal for all his cows each with their respective names like Susan, Matilda, Bihogo and many others which helps him distinguish them from one another. His employees collect litres of milk both in the morning and evening time in twenty litre jerrycans and later sell them to near by customers around the farm, a cup of milk being sold at one thousand Uganda shillings only.

He keeps track of farm activities through daily record keeping where all the farm records are kept in books. Record keeping is mainly done by himself when he comes back from his official government work in the evening time at around five o'clock. Information recorded includes vaccines and feeds

like dregs from supplier, sales, litres of milk collected per day, profits and losses, worker details and expenses. Losses are made when milk gets spoilt, cows get sick and die, cows are stolen and when the workers steal money from the farm earnings. Profits are made when milk and cows are sold and other products. Profits and losses are calculated manually in record books of the farm once in a while because of it being tiresome to calculate the past records.

Problem in the Current system.

- 1) Difficulty in report generating : More calculations are required to generate reports of the previous records and financial statements.
- 2) A lot of paper work : Since almost every record is kept in books, it is liable to be damaged by water, fire or easily get lost.
- 3) Lack of proper storage for the records books.
- 4) Slow retrieval of data especially from records books of different years to obtain information like sales ,profits and losses.
- 5) Time consuming : A lot of time is taken to calculate monthly profits and losses, reports, salaries for workers etc.
- 6) Manual control : All calculations to generate the report is done manually so there is greater chance to committing errors in the calculations.

5.3 The proposed solution

System requirements.

These are requirements that are needed to incorporate the desired functionality in the system. This therefore called for the description of the properties of the system and this had to address both the software and hardware requirements.

System requirements are:

- | | |
|------------------------|---|
| Tools for development: | Android studio. |
| Platform: | Android operating system 4.2 and above. |
| Hardware: | Android smart phones. |

5.4 Costs and benefits

This refers to computing anticipated costs and benefits.

The purpose is to answer the questions like;

- Is the project justified (because benefits outweigh costs)?
- Can the project be done with in the given cost constraints?
- What is the minimal cost to attain such a certain system?

Costs

Development costs

Hardware and software
Data conversion costs
Vendor installation

Operational costs

Software upgrades
Licensing for software
Hardware upgrades
User training
Uploading the App to Google play Store
Hardware repairs

Benefits

Tangible benefits

The system is contributed to achieving the following benefits;

- The new system reduces the costs of buying books for records keeping.
- The new system keeps records safely without being easily destroyed.

Intangible benefits

- The new system allows the user to access records about the user to access records about the farm from any where.
- The new allows the user to advertise widely and better for his farm products.
- Error reductions in mathematical calculations.
- Better and more timely information

5.5 Recommendations

The study concluded that the implementation of the Project is feasible, whereas the State and external financial support is essential.

Based on the information presented in this feasibility study, it is recommended that the Farm Management approves the Proposed system and begins project initiation. The findings of this feasibility study show that the Proposed system will be highly beneficial to the Farm and has a high probability of success.

6. Requirements elicitation technique.

6.1 Interviews.

An interview is a planned meeting during which one obtains information from another person.

The reasons as to why we used Interviews to gather information were;

- ❖ To investigate issues in a depth way.
- ❖ To discover how individuals think and feel about a topic and why they hold certain opinions.
- ❖ To inform decision making, strategic planning and resource allocation.
- ❖ To deepen understanding and explain statistical data.

Our team of four were able to interview the owner of the farm through the following questions where we gathered information from;

1. What is the name of your farm?
2. What kind of livestock (cows) do you keep and why?
3. How do you distinguish your livestock (cows)?
4. Do you keep all the cows in one kraal?
5. How many litres of milk do you collect a day?
6. How do you keep track of the activities of the farm?
7. Who does the record keeping?
8. How often does he/she put down records?
9. Where do you keep your records?
10. What exactly do you record down?
11. How do you generate income and what other sources of income?
12. When do you make losses?
13. How do you calculate your losses and profit?
14. What method do you use to calculate your losses and profit?

Farm owner: DENNIS KAMUGYENE
Contact: 0772446011 / 0702446011

7. Requirements Analysis.

Requirements Analysis also called requirements engineering refers to the process of determining user expectations for a new or modified product. These features called requirements must be quantifiable, relevant and detailed. In software engineering, such requirements are often called functional specifications.

7.1 User requirements.

These specify the requirements a user expects from software to be developed in a software project. The above line is comprised of the preferred features by the user in the system on which the interviews and comments from the owner of the farm, who is going to use the system. The system should be able to;

- Restrict unauthorized login attempt to the system.
- Allow the user enter and read data on a daily basis.
- Provide user friendly interfaces for easy interactions.

7.2 System requirements.

These are requirements that are needed to incorporate the desired functionality in the system. This therefore called for the description of the properties of the system and this had to address both the software and hardware requirements.

System requirements are:

Tools for development: Android studio.

Platform: Android operating system 4.2 and above.

Hardware: Android smart phones.

7.3 Functional requirements.

Functional requirement 1.

ID: FR-000-001

Title: Stock.

Description: The system shall be able to accept and record daily entries of the available stock and update available information about the farmer's stock in terms of number of feeds for cows, vaccines for cows.

Functional requirement 2.

ID: FR-000-002

Title: Record Income.

Description: The system shall be able to accept and record daily entries of income attained by the farm from the user, find their total and display it accordingly. Income refers to activities that generate money for the farmer for example selling litres of milk, ghee, selling meat and cows and others.

Functional requirement 3.

ID: FR-000-003

Title: Record Summary.

Description: The system shall be able to display records of previous entries for different days and a

monthly summary. Under monthly summary, the system shall be able to calculate and display profit made per month.

Functional requirement 4.

ID: FR-000-004

Title: Record expenses.

Description: The system shall be able to accept and record daily entries of expenses from the user, find their total and display it accordingly. Expenses may include amount spent on vaccines for the cows, feeds for cows, salaries for workers etc.

Functional requirement 5.

ID: FR-000-005

Title: Others.

Description: The system shall be able to accept and record daily entries of Consumables, Losses, and Delivery dates of cows.

7.4 Non-functional requirements.

Non-Functional requirement 1.

ID: NFR-100

Title: User Login.

Description: Given that a user has registered, then the user should be able to login to the mobile application. The login information will be stored on the phone and in the future the user should be logged in automatically.

Non-Functional requirement 2.

ID: NFR-200

Title: Retrieve password.

Description: Given that a user has registered and forgotten the password, then the user should be able to retrieve the password by email.

Non-Functional requirement 3.

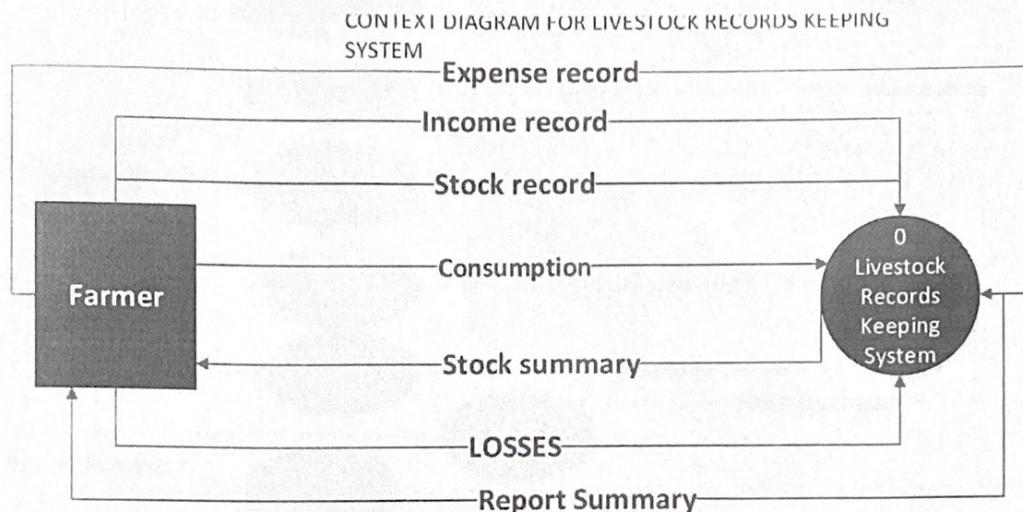
ID: NFR-300

Title: User registration.

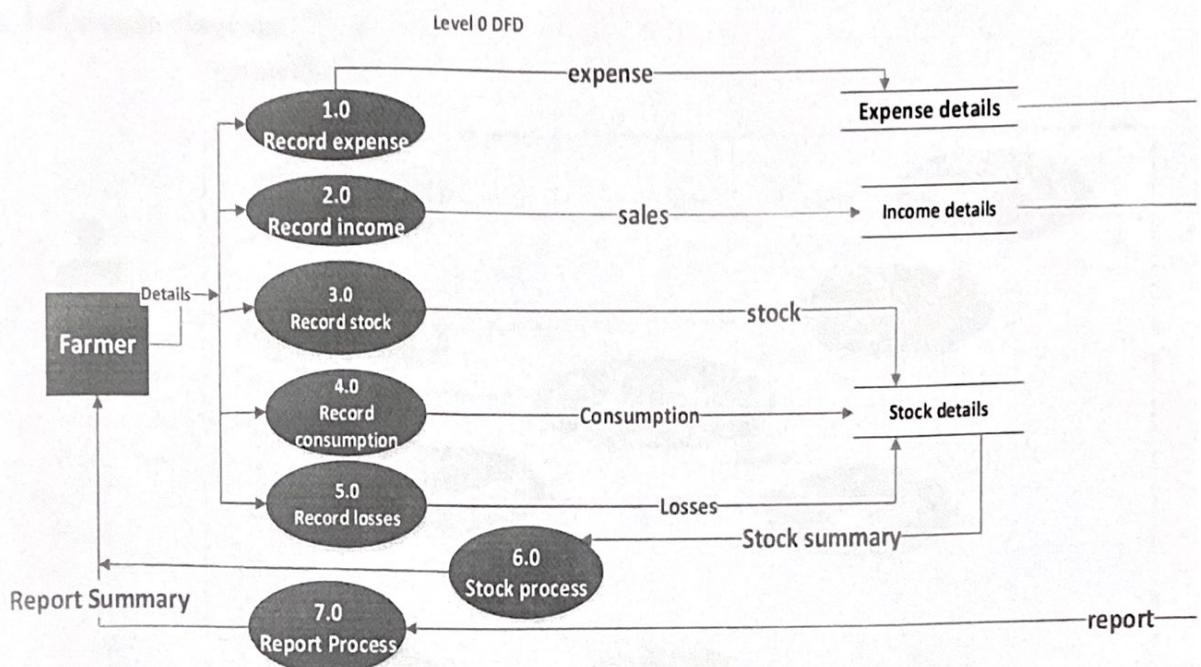
Description: Given that the user has downloaded the mobile application, then the user should be able to register through the mobile application. The user must provide their bio-data and provide a regularly used phone number.

8. Data Flow Diagram

8.1 Context Diagram



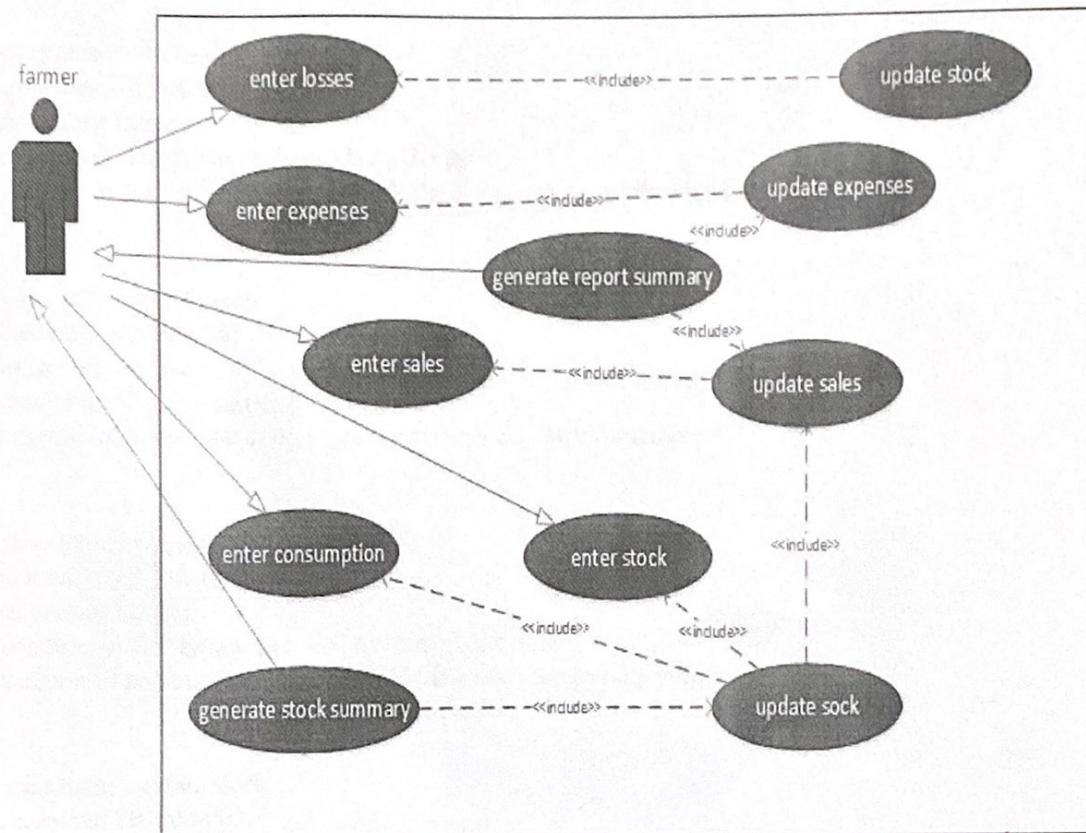
Farm owner?



You can have only 1 entity

8.3 Use case diagram.

Use case diagram



Use case name: enter expenses

requirements: FR-000-004

player (actor): farmer

pre-conditions: The farmer has spent on the farm.

Main course of action: The farmer records the items and prices they have spent on the farm

Use case name: update expenses

requirements: FR-000-003

player (actor): system

pre-conditions: The user submitted expenses
main course of action: The system records and stores new information.

Use case name: enter sales
requirements: FR-000-002
player (actor): farmer
pre-conditions: The farmer has sold from the farm
main course of action: The farmer records the items and prices they have sold.

Use case name: update sales
requirements: FR-000-003
player (actor): system
pre-conditions: the user entered sales
main course of action: the system records and stores new information

Use case name: enter consumption
requirements: FR-000-005
player (actor): farmer
pre-conditions: The farmer has used an item on the farm
main course of action: The farmer records the items and quantity consumed

Use case name: update stock
requirements: FR-000-001
player (actor): system
pre-conditions: The farmer enters stock of feeds and vaccines
main course of action: system records and stores new information of the stock

Use case name: generate stock summary
requirements: FR-000-001
player (actor): System
pre-conditions: stock has been updated
main course of action: system displays the stock summary

Use case name: generate record summary
requirements: FR-000-003

player (actor): system

pre-conditions: either the system updates sales or expenses

main course of action: systems displays the summary of the previous records.

8.4 Entity Relationship Diagram

Tables and their attributes

FarmerOwner (farmOwnerNo, fname, phoneNumber, expenses, stock, agenda)

Farm (farmName, farmNo, location)

Supplier (sname, phoneNumber, sNo)

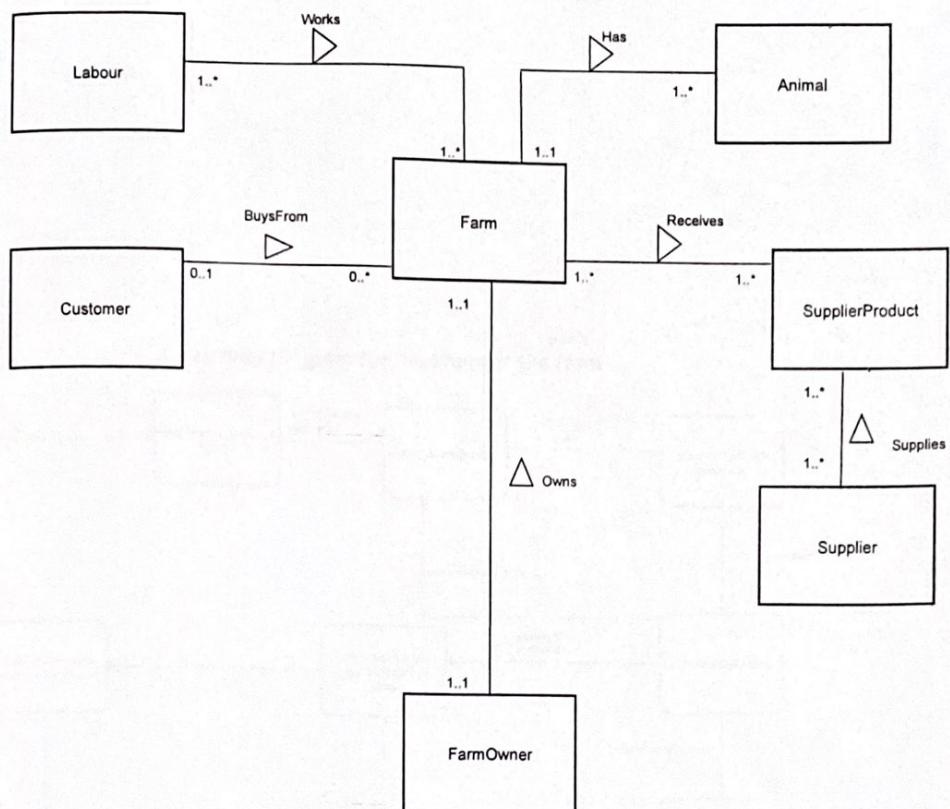
Customer (cname, phoneNumber, cNo, productBought)

Animal (tagNo, name, DOB)

Labour (lname, labourID, phoneNumber, salary)

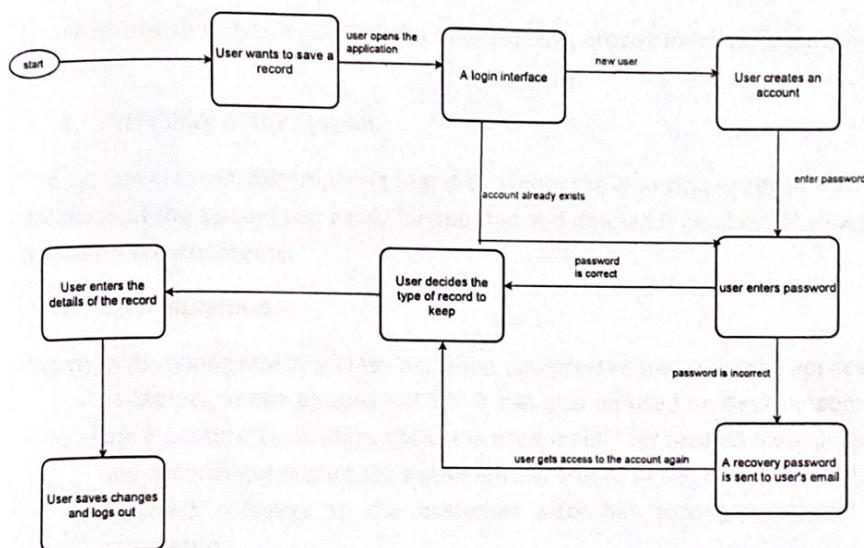
FarmProduct (farmProductNo, price, dateSold, profit, loss)

SupplierProduct (productNo, productName, productCost)



9

An activity diagram for the users at the farm



<https://peppomjocoigkepgbmcfnnindia/index.html>

9. Architectural Design

Introduction

The goal of architecture design is to provide our team with an up to date and relevant assessment of best practices related to the system, design, development, deployment and operation of the livestock farm records management system. System architecture mainly deals with how the system should perform and what functions will be required.

Application Architecture

Below is how the system handles the data storage, processing data, interface and how it ensures security measures.

1. Scalability of the system.

The system is extensible implying that it can meet the changing needs at the farm since the information in the database of the system can easily be updated and deleted if need be. The system is also able to run on multiple hardware environments.

2. Web integration.

The records management system has been compressed into a mobile application that can run on gadgets such as tablets, smart phones but still it can also be used on desktop computers. The system will be part of the E commerce strategy though it is used mainly for records management but still during the transaction any records generated are automatically stored in the database of the system. The system also sends feedback message to the customer after him buying a product and the process is successfully completed.

3. Total cost of ownership and initial costs.

We were able to access the total cost of ownership which includes purchases, fees altogether called **hard costs** that the farm management will have to inject into the system development project.

Soft costs such as training people on how to use the system were considered though they are not easy to determine

4. Legacy system interface requirements.

The farm is using traditional means to handle its data which implies that it had no existing system. Because of such a scenario our system will not need to interface with any legacy system since in reality there is no existing system at the farm apart from keeping their records in books and files.

5. Processing options.

The System processes data both in batches and online. If a customer wants to order for any item from the farm the system uses online processing option to carry out the assignment but if a farmer or any worker at the farm wants to keep a record, there is no need for the system to be online instead batch processing is convenient for such a task.

6. Security issues.

The system allows users to create passwords for their applications which is very crucial in avoiding unauthorized access to somebody's account. This implies that the information can only be accessed by certain personnel with appropriate permission.

Data Store Architecture.

Below is the description of the different strong entities in our logical model design of our system.

FarmProduct.

This entity allows the user to enter the prices of the different products, date when the product was sold, sold/loss made, profit/loss made, the type of the product and farm product number. This mainly processes and stores any profit or loss made on the sold product.

Animal.

This entity allows the farm owner to enter details about the animals reared on the farm the it processes and stores the results in the database.

Labour.

This is the entity that allows the owner of the farm to enter the details of people working on the farm and mainly processes and stores each salary of the different workers on the farm.

Farm.

This an entity that allows the farm owner to keep records and details of the farm.

Customer.

The customer entity enables the farm owner to keep details about its customers and the different products they burn.

Supplier.

This entity allows the farm owner to keep records and details of its suppliers.

SupplierProduct.

This entity enables the farm owner to keep records of the different products supplied to the farm by the suppliers.

FarmOwner.

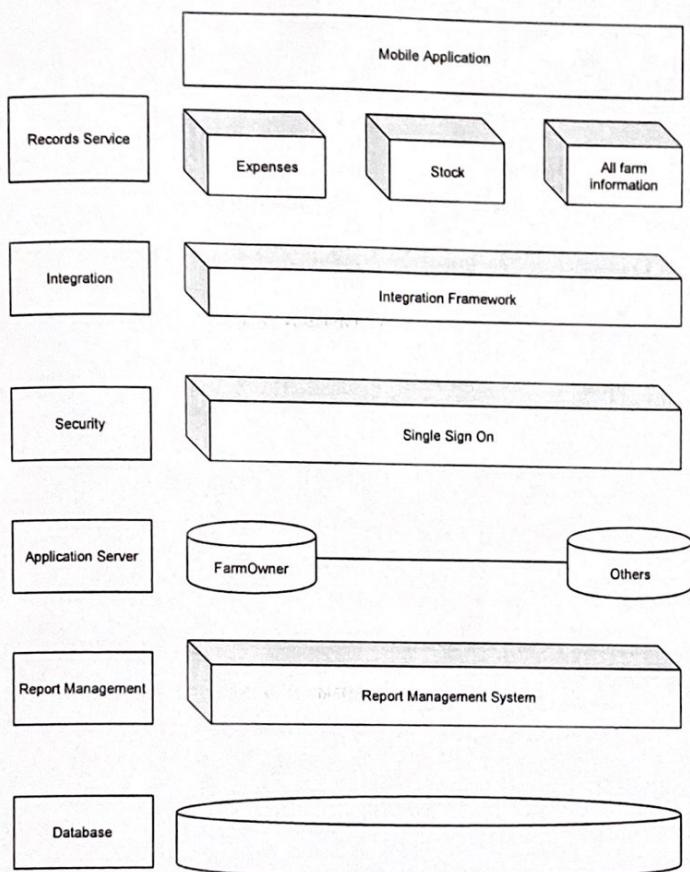
This entity allows the farm owner to enter his biodata information, expenses, stock related to the farm.

Entity	Entity Description
FarmProduct	Information about the product.
Animal	All information about animals reared.

Labour	Personal information and salary of the farm workers.
Farm	Information about the farm.
Customer	Information about customers.
Supplier	Details about farm suppliers.
SupplierProduct	Information about the products supplied by suppliers.
FarmOwner	Personal information about the farm owner.

Data Conversion Architecture.

Data	Source	Target	Anticipated Volume
FarmProduct information	Worker in charger	FarmProduct relation	10GB
Animal information	Farm Owner	Animal relation	
Labour information	Farm Owner	Labour relation	

RECORDS MANAGEMENT SYSTEM ARCHITECTURE BLOCK DIAGRAM

10. User interfaces.

1

Livestock Records Keeping System

Username

Password

Login

[Forgotten password?](#)

[Sign up](#)

2

Livestock Records Keeping System
Create new account

First name

Last name

User name

Email

Phone number

Password

Confirm password

Upload photo

Gender

I'm not a robot

Submit

Interface 1's importance: Enables a User to Login to the system.

Interface 2's importance: Enables a new User to register/sign-up for the system

3

Livestock Records Keeping System

✓ Account created successfully

Username

Password

Login

[Forgotten password?](#)

[Sign up](#)

Interface 3's importance: Used to provide a notification message to the new User upon successful registration.

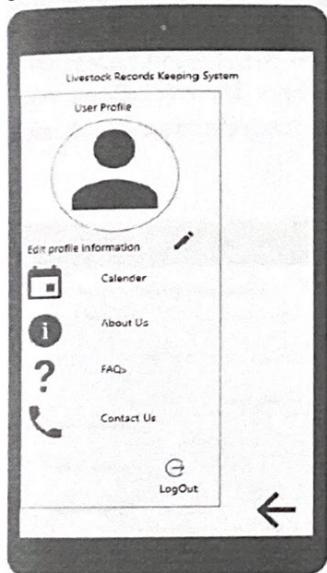
24



4

interface 4's importance: It shows the Home screen with all the tabs in it.

5



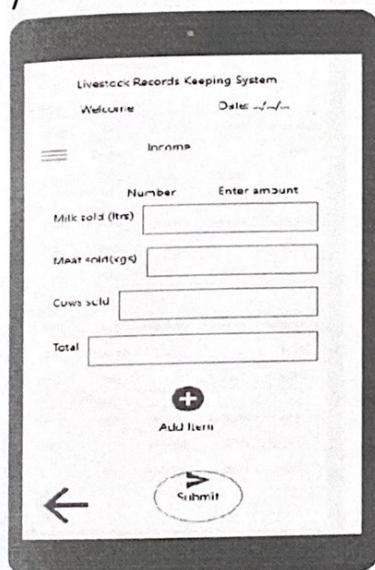
Interface 5's importance: Once the side bar has been clicked on, it will display a menu including User profile information and the ability to edit it, calendar to keep up to date, Frequently asked questions and also the contact details of the farm.

6



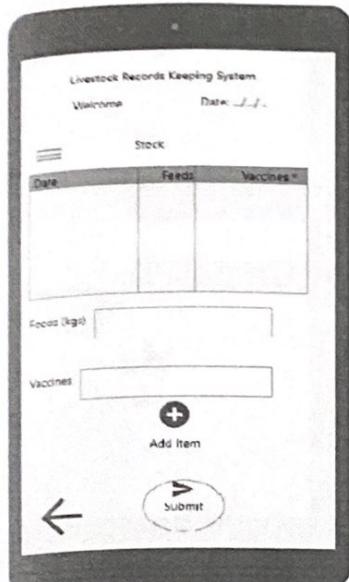
interface 6's importance: An 'expenses' tab to enable the user enter in details for expenditure like amount spent on Vaccinations, feeds and salary for labour and later calculate the total expenditure and submit the Total to the system's database.

7



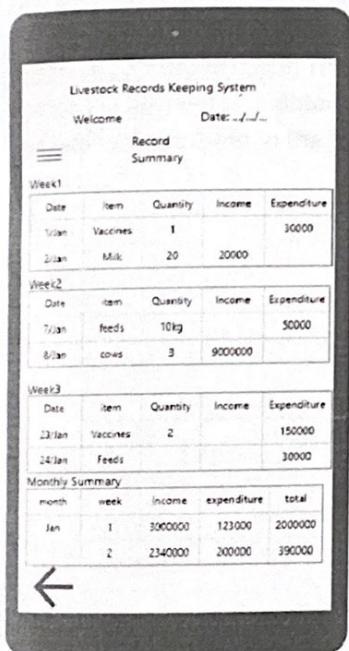
interface 7's importance: An 'Income' tab to enable the user enter in details for incomes earned at the farm like amount of milk and meat sold and later calculates the Total and submits it to the system's database.

8

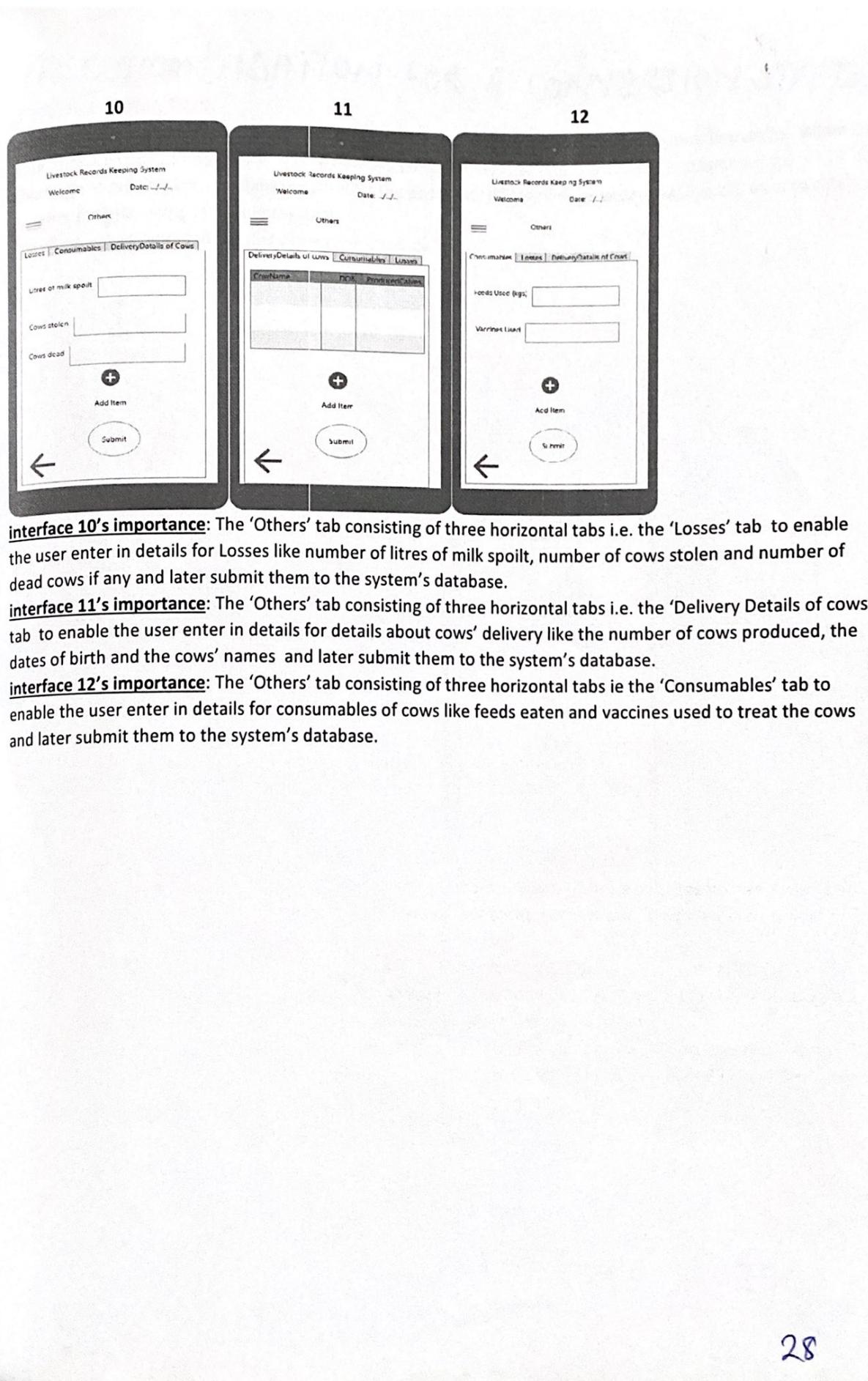


interface 8's importance: A 'Stock' tab to enable the user enter in details for Stock like feeds and vaccines bought for the cows and later submit them to the system's database.

9



interface 9's importance: A 'Record Summary' tab to enable the user input in details for the week in terms of incomes and expenditures spent on each respective item and the system will later generate a month record details.



RECOMMENDATION FOR A CONVERSION STRATEGY

11. PARALLEL CONVERSION.

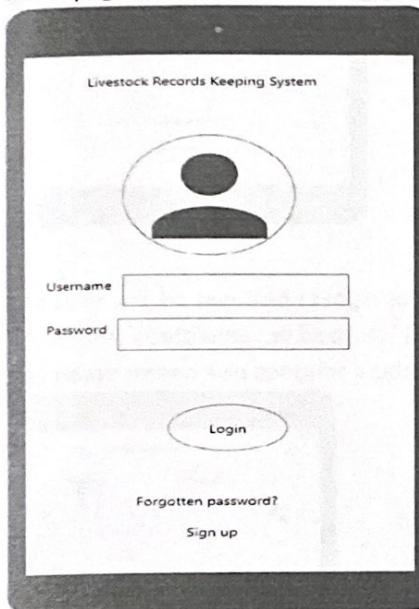
Parallel conversion refers to running the old system and the new system at the same time, in parallel. When the same results can be gained over time, the new system is put into use and the old one is stopped. One advantage of running both systems in parallel is the possibility of checking new data against old data to catch any errors in processing in the new system.

Therefore, we recommend Parallel conversion strategy.

12. User Documentation

A First-time user of the mobile application should be able to see the login page when he/she opens the application, if the user is not does have an account or is not registered, he/she should be able to click the "create new account" button located at the bottom of the log in page.

If the user chooses to stay logged in, then the next time they open the app they should be able to go to the home page first.

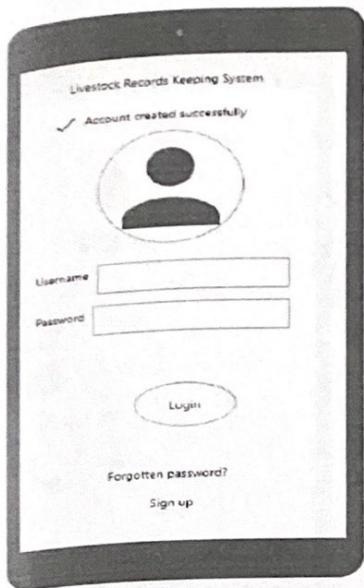


User Login: reference NFR-100

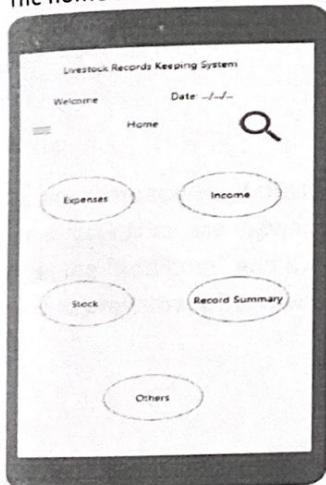
The image shows the 'User registration' screen of the Livestock Records Keeping System. It has a title 'Create new account' at the top. Below it are several input fields: 'First name', 'Last name', 'User name', 'Email', 'Phone number', 'Password', 'Confirm password', 'Upload photo', and 'Gender'. There is also a checkbox labeled 'I'm not a robot' with a checked box. At the bottom is a blue 'Submit' button with a white arrow icon.

User registration: reference NFR-300

For a new user creating a new account/registering for the first time, he/she will receive a notification on the login page upon a successful account creation.

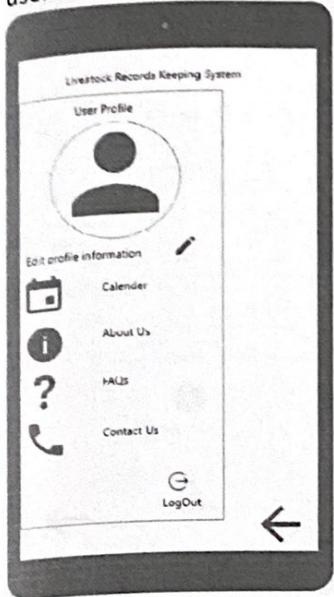


The User will be required to login to the system and once login is successful, a home screen of the application will appear containing tab buttons like Expenses, Others, Stock, Record Summary and Income. The home screen also contains a side tab, a search button and current date.

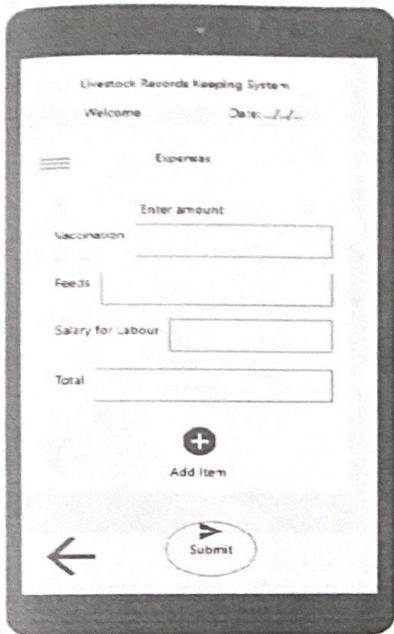


Once the clicks/taps on the side bar tab, a side menu will appear has the profile information of the user, here user can view and edit their profile information. There is also information about FAQs and also information about the system and a calendar to keep the user up to date with farm activities that may be helpful to the

user. A log-out button to log-out of the system.

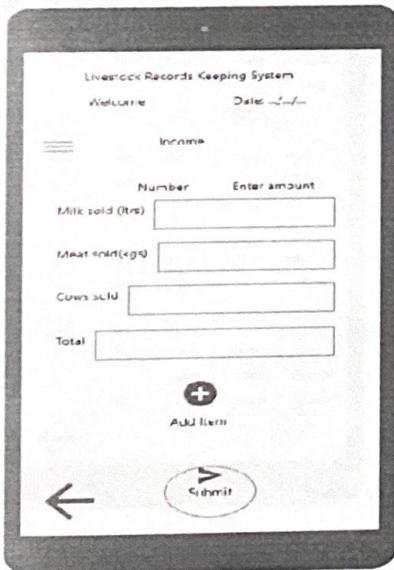


On the home screen, if the user clicks on the “Expenses” button, an interface will appear prompting him/her to input records for the expenses and the system will be able to generate the total. The User will then have to first press the “Add item” and then the “submit” button to store records in the system’s database. The User will have to press the left arrow button to return to the “Home” screen. Reference: FR-000-004



On the home screen, if the user clicks on the “Income” button, an Interface will appear prompting him/her to Input records for the Income and the system will be able to generate the total. The User will then have to first press the “Add item” and then the “submit” button to store records in the system’s database. The User will have to press the left arrow button to return to the “Home” screen.

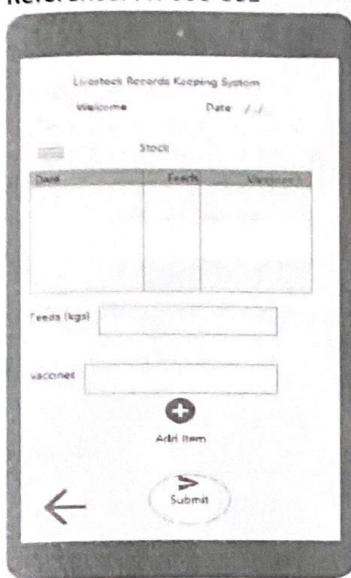
Reference: FR-000-002



On the home screen, if the user clicks on the “Stock” button, an Interface will appear prompting him/her to Input records for the Stock . The User will then have to first press the “Add item” and then the “submit” button to store records in the system’s database. The User will have to press the left arrow button to return to the

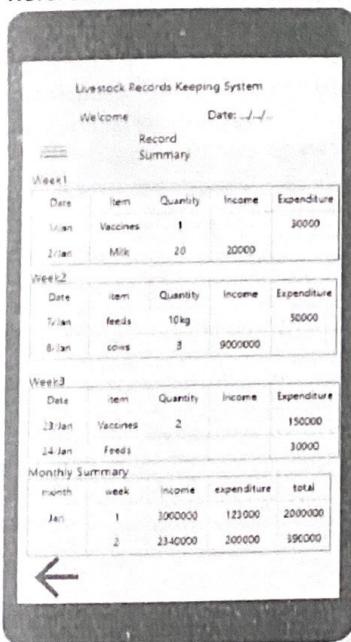
"Home" screen.

Reference: FR-000-001



On the home screen, if the user clicks on the "Record summary" button, an Interface will appear containing previous records of entries for the week(s) and their monthly summary as shown below. The User will have to press the left arrow button to return to the "Home" screen.

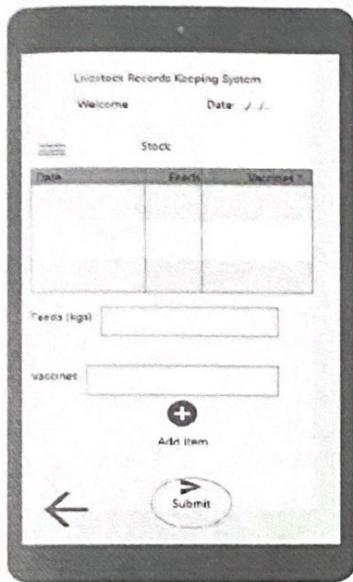
Reference: FR-000-003



On the home screen, if the user clicks on the "Others" button, an Interface will appear containing three horizontal tabs prompting him/her to Input respective records for each tab clicked on. The User will then have

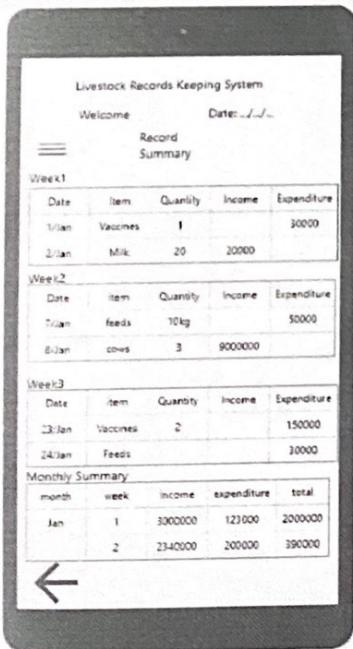
"Home" screen.

Reference: FR-000-001



On the home screen, if the user clicks on the "Record summary" button, an Interface will appear containing previous records of entries for the week(s) and their monthly summary as shown below. The User will have to press the left arrow button to return to the "Home" screen.

Reference: FR-000-003



On the home screen, if the user clicks on the "Others" button, an Interface will appear containing three horizontal tabs prompting him/her to Input respective records for each tab clicked on. The User will then have

to first press the “Add item” and then the “submit” button to store records in the system’s database. The User will have to press the left arrow button to return to the “Home” screen. reference: FR-000-005

The image displays three sequential screenshots of a mobile application interface for a "Livestock Records Keeping System".

Screenshot 1: Shows the "Others" tab selected. It contains fields for "Litres of milk spoilt", "Cows stolen", and "Cows dead". Below these are "Add Item" and "Submit" buttons, and a back arrow icon.

Category	Item 1	Item 2	Item 3
DeliveryDetails			
Consumables			
Losses			

Screenshot 2: Shows the "DeliveryDetails" tab selected. It contains a table with three columns labeled "CowName", "DDP", and "Quantity". Below the table are "Add Item" and "Submit" buttons, and a back arrow icon.

CowName	DDP	Quantity

Screenshot 3: Shows the "Consumables" tab selected. It contains fields for "Foods Used (kg)" and "Vaccines Used". Below these are "Add Item" and "Submit" buttons, and a back arrow icon.