MAKERERE UNIVERSITY

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

THE PROJECT MANAGEMENT PLAN REPORT

FOR THE POULTRY RECORD KEEPING SYSTEM

BY

GROUP 4

A PROJECT REPORT SUBMITTED TO THE SCHOOL OF COMPUTING AND INFORMATICS SCIENCES
FOR THE PROGRESS OF THE POULTRY RECORD KEEPING SYSTEM.
ACCOMPLISHED IN PARTIAL FULFILMENT OF FINAL EXAMINATION FOR SECOND YEAR SEMESTER ONE OF BACHELOR OF
SCIENCE IN COMPUTER SCIENCE OF MAKERERE UNIVERSITY

LECTURER
DR. PETER WAKHOLI
DEPARTMENT OF COMPUTER SCIENCE
SCHOOL OF COMPUTING AND INFORMATICS TECHNOLOGY, MAKERERE UNIVERSITY

OCTOBER, 2017

DECLARATION

We Group 4 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 two semester one in the department of computer science or any other course before.

NAME	STUDENT NUMBER	REGISTRATION NUMBER	SIGNATURE
SENYANGE RICHARD	216001192	16/U/1102	
NINSIIMA SHEILLA	216001139	16/U/964	
ARIHO IGNATIUS	216001851	16/U/1811	
MUBAGIZI ALLAN	216005099	16/U/7136	

Table of Contents

1.0 FACT FINDING

Our team of four interviewed miss carol who owns Candi poultry farm located in Lubowa Wakiso district. We used face to face interview and asked her the following questions during the interview.

1. What is the name of your organization?

2. What kind of birds do you keep and why?
3. How do you distinguish your birds?
4. Do you keep all birds in one poultry house or?
5. How many eggs do you collect and how do you count them?
6. How do you keep track of the activities on the farm?
7. Who does the record keeping and why is it that its him/her who does the record keeping?
8. How often does he put down the records?
9. Where do you keep your records and why?
10. What exactly do you record down?
11. How do you generate income?
12. When do you make losses?
13. How do you calculate your losses and profits?
14. When do you calculate your losses and profits and why?
15. What method do you use to calculate your losses and profits?

1.1 ANALYSIS OF THE EXISTING SYSTEM

The existing system deals mainly with layers because of their high market and Miss carol the owner of the farm has three poultry houses where she keeps the hens according to their ages. She collects her eggs in a bucket which she later places on trays and counts the number of trays collected.

She keeps track of her activities on the farm through daily record keeping where she keeps all her records in books. Each poultry house has its own record book and in most cases, she records different information in different books. These records are mainly made by her two employees since they are ever at the farm who later present the information to her for balancing. Information recorded includes; salaries of her employees, feeds recorded in kg, water recorded in jerrycans, medication, sales, number of birds on the farm, motarity rate, house renovation, number of eggs collected and losses.

She makes losses when eggs break, birds get sick, birds are stolen, birds die and she benefits from the farm through selling eggs and off layers. She uses approximately on a monthly bases Microsoft excel to calculate her profits and losses which she does once in a while.

1.2 DRAW BACK OF THE SYSTEM

- Difficulty in report generating: they require more calculations to generate report of previous records and financial statements

- Manual control: all calculations to generate the report is done manually so there's a greater chance to get errors
- time consuming: every work is done manually so they can't generate the report in the session or as per requirement and it is time consuming

  A lot of paper work: existing system required a lot of paper work. Loss of even single record led to a difficult situation because all papers are needed to generate the report.

1.3 PROPOSED SYSTEM

Poultry record keeping system has been computerized to run the farm easily. Records are to be maintained in a database and any record related to farm is generated and edited easily. The system is also able to automatically calculate the total income, expenses and profits and later generate a report summary for the farmer. Our proposed system has several advantages

- User friendly interface
- Fast access to database
- Search facility
- Quick transaction
- More storage capacity
- Less error

All the book keeping difficulties in managing the farm activities has been rectified by implementing computerization.

2.0 REQUIREMENTS ANALYSIS

Requirements Analysis refers also called requirements engineering is 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.

2.1 USER REQUIREMENTS

Specifies 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;

- Provide user friendly interfaces for easy interactions.
- Allow the user enter and read data on a daily basis

- Restrict unauthorized login attempt to the system

2.2 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

2.3 FUNCTIONAL REQUIREMENTS

This outlines how the system serves the users by ensuring efficient and effective functionality.

Functional requirement 1

ID: FR1

Title: record summary.

Description: the system shall be able to display records of previous entries according to houses/batch numbers and monthly summary. Under monthly summary, the system shall be able to calculate and display profit made per month.

Functional requirement 2

ID: FR2

Title: record expenses

Description: the system shall be able to accept and record daily entries from the user, find their total and display it accordingly. Expenses may include amount spent on vaccine, feeds, chicken, salaries etc.

Functional requirement 3

ID: FR3

Title: record income

Description: the system shall be able to accept and record daily entries from the user, find their total and display it accordingly. Income are activities that generate money for the farmer. These may include; selling of eggs, chicken etc.

Functional requirement 4

ID: FR4

Title: stock

Description: the system shall be able to accept and record daily entries of available stock and update available information about the farmer's stock in terms of number of chicken, number of eggs produced, quantity of feeds etc per house. Under stock, user shall record losses and consumptions; losses may include damaged eggs, dead chicken etc and consumption may include quantity of feeds and water used which all in all updates stock.

2.4 NON-FUNCTIONAL REQUIREMENTS

Non-Functional requirement 1

ID: NFR1

TITLE: User registration - Mobile application

Description: Given that a user has downloaded the mobile application, then the user should be able to register through the mobile application. The user must provide user-name, password and e-mail address. The user can choose to provide a regularly used phone number.

Non-Functional requirement 2

ID: NFR 2

TITLE: User log-in - Mobile application

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

Non-Functional requirement 3

ID: NFR3

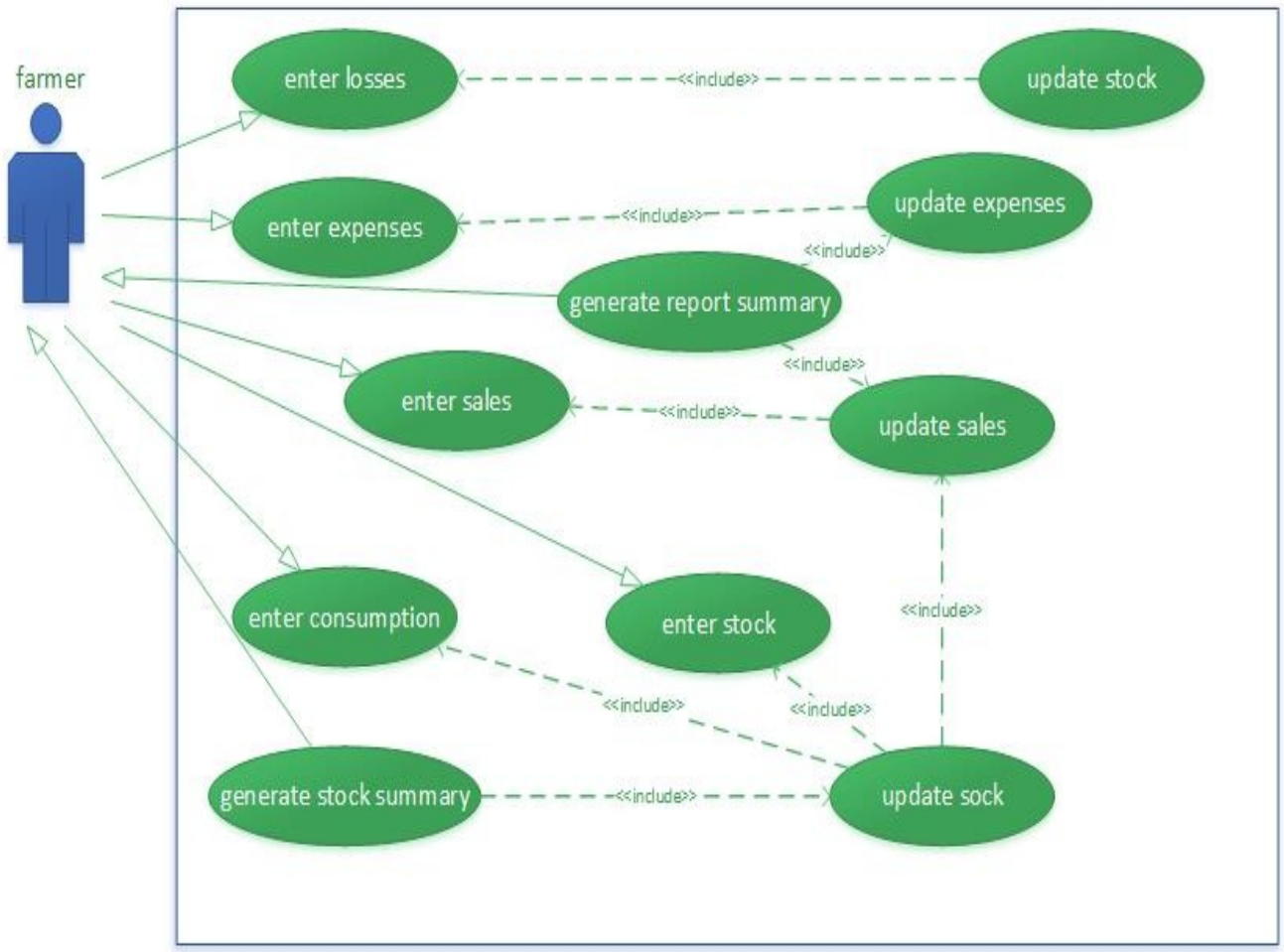
TITLE: Retrieve password

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

3.0 USE CASE DIAGRAMS

A use case defines a goal-oriented set of interactions between external actors and the system under consideration. Actors are parties outside the system that interact with the system.

Use case diagram



3.1 USE CASE DESCRIPTION

Use case name **:** enter expenses

Requirements **:** FR2

Player(actor) **:** farmer

Preconditions **:** The farmer has spent on the farm

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

Use case name **:** update expenses.

Requirements **:** FR2.

Player(actor) **:** system.

Preconditions : the user submitted expenses.

Main course of action **:** the system records and stores new information.

Use case name : enter sales.

Requirements **:** FR3.

Player(actor) **:** farmer.

Preconditions **:** the farmer has sold from the farm

Main course of action : the farmer **rec**ords the items and prices they have sold

Use case name **:** update sales

Requirements **:** FR3

Player(actor) **:** system

Preconditions **:** the user entered sales

Main course of action **:** the system records and stores new information

Use case name **:** enter consumptions.

Requirements : FR2.

Player(actor) **:** farmer.

Preconditions **:** 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 **:** FR4

Player(actor) **:** system

Preconditions **:** either the farmer enters losses or consumptions or system updates sales

Main course of action **:** system records and stores new information of the stock.

Use case name **:** Generate stock summary

Requirements **:** FR4

Player(actor) **:** system

Preconditions **:** Stock has been updated.

Main course of action **:** System displays the stock summary.

Use case name **:** Generate Record Summary

Requirements **:** FR1

Player(actor) **:** system

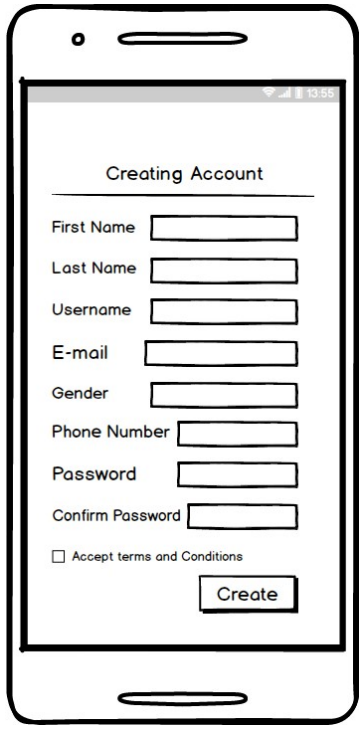
Preconditions **:** either the system updates sales or expenses.

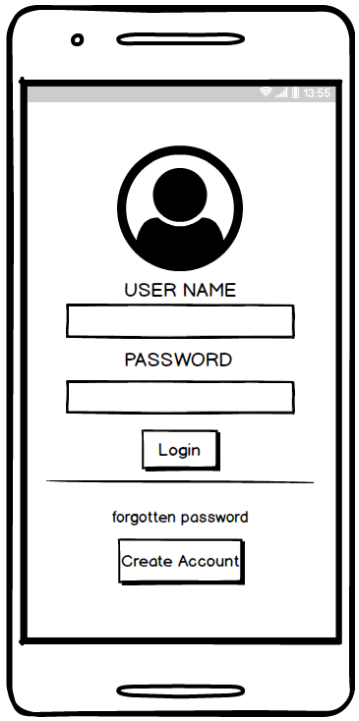
Main course of action **:** System displays the summary of the previous records.

4 USER INTERFACES

A First-time user of the mobile application should see the login page when he/she opens the application, if user is not registered he/she should be able to register from the log in page.

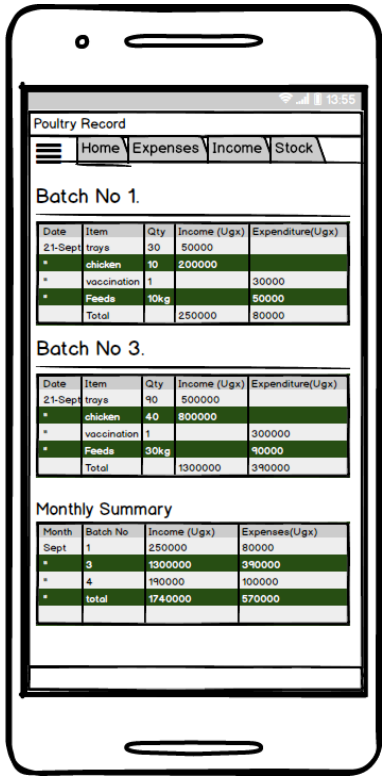
If they choose to stay logged in, then the next time they open the app they should be able to go to the home page first.



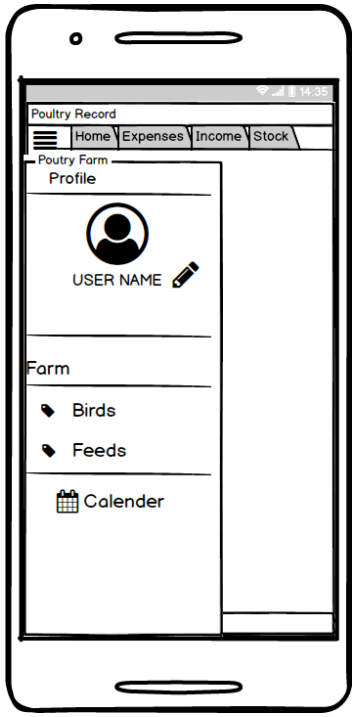


The home page contains previous records of entries and their monthly summary as shown below.

reference FR1

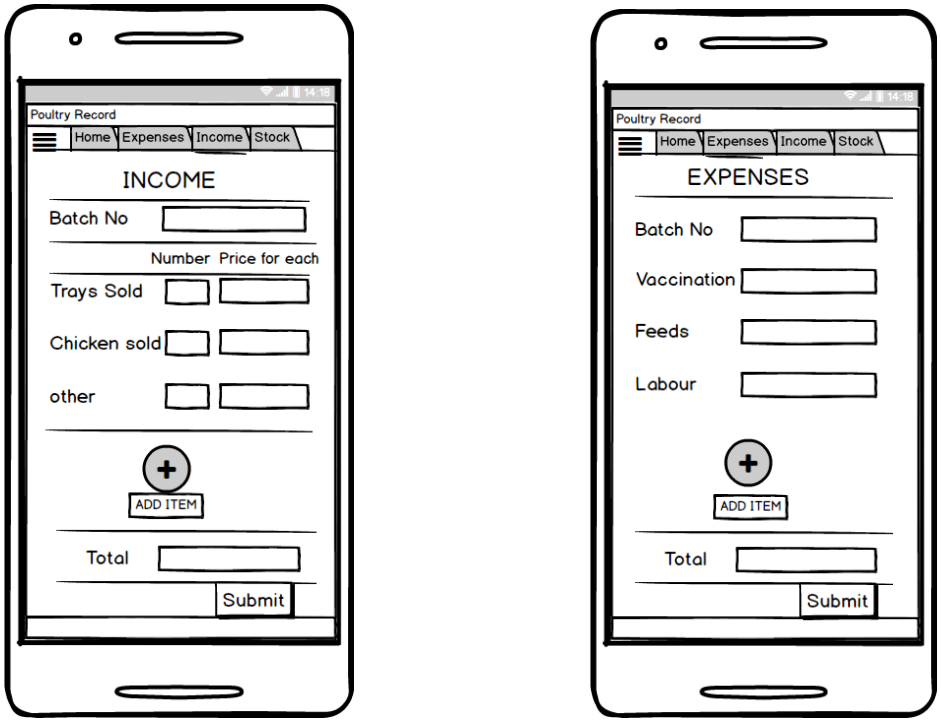


A side bar is also available and has the profile information of the user, here user can view and edit their profile information. There is also information about chicken and how to feed them etc. that may be helpful to the user.



When the user wants to record his/her expenses or income, they tap on the expenses or income tab, enter records and submit them.

Reference FR2 and FR3



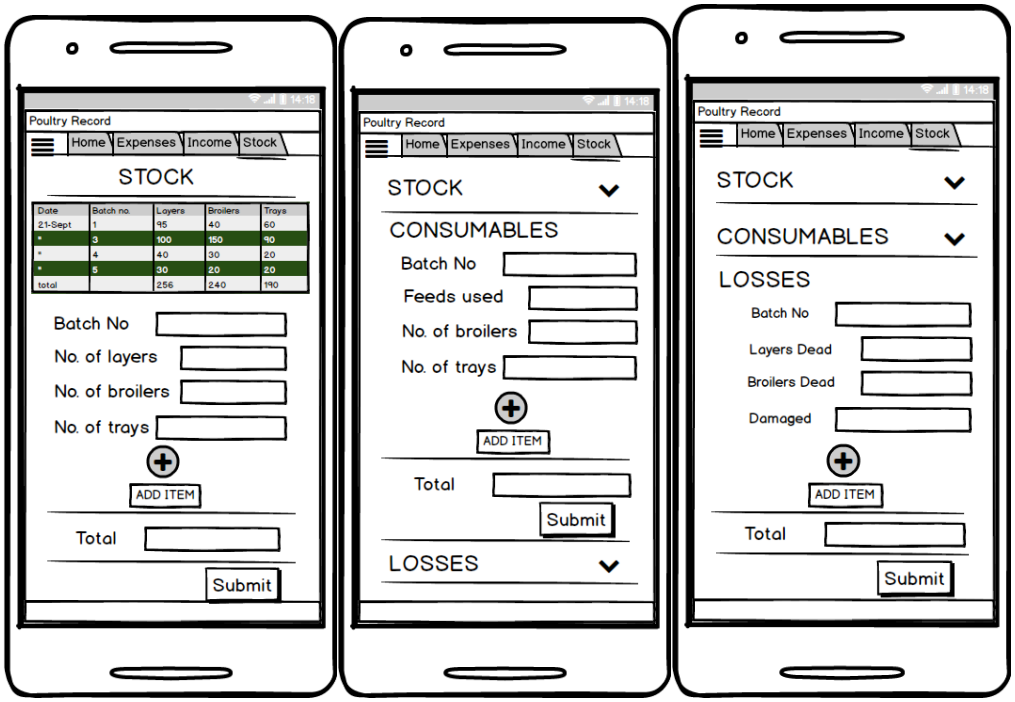
The stock tab contains a drop-down menu for stock, consumables and losses.

Reference FR4

Stock shows the users available stock for example of birds, eggs etc.,

consumables allow the user to enter daily consumption of materials on the farm e.g. feeds, water, vaccine etc. and losses allows user to enter materials that have gotten damaged or stolen like damaged eggs, chicken that's dead etc. as shown below;

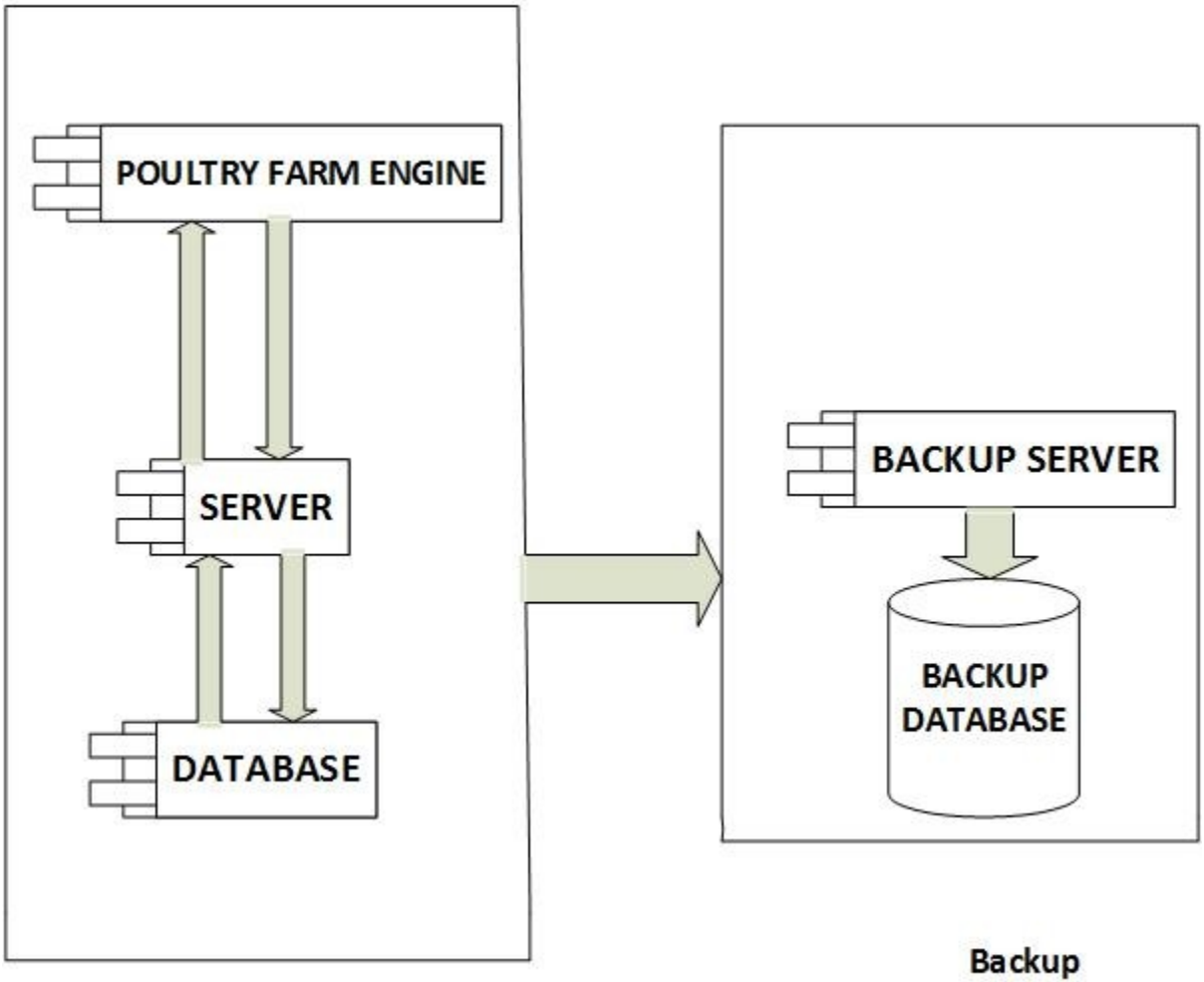
Reference FR4



STOCK CONSUMABLES LOSSES

5 COMPONENT DESIGN

Component diagram

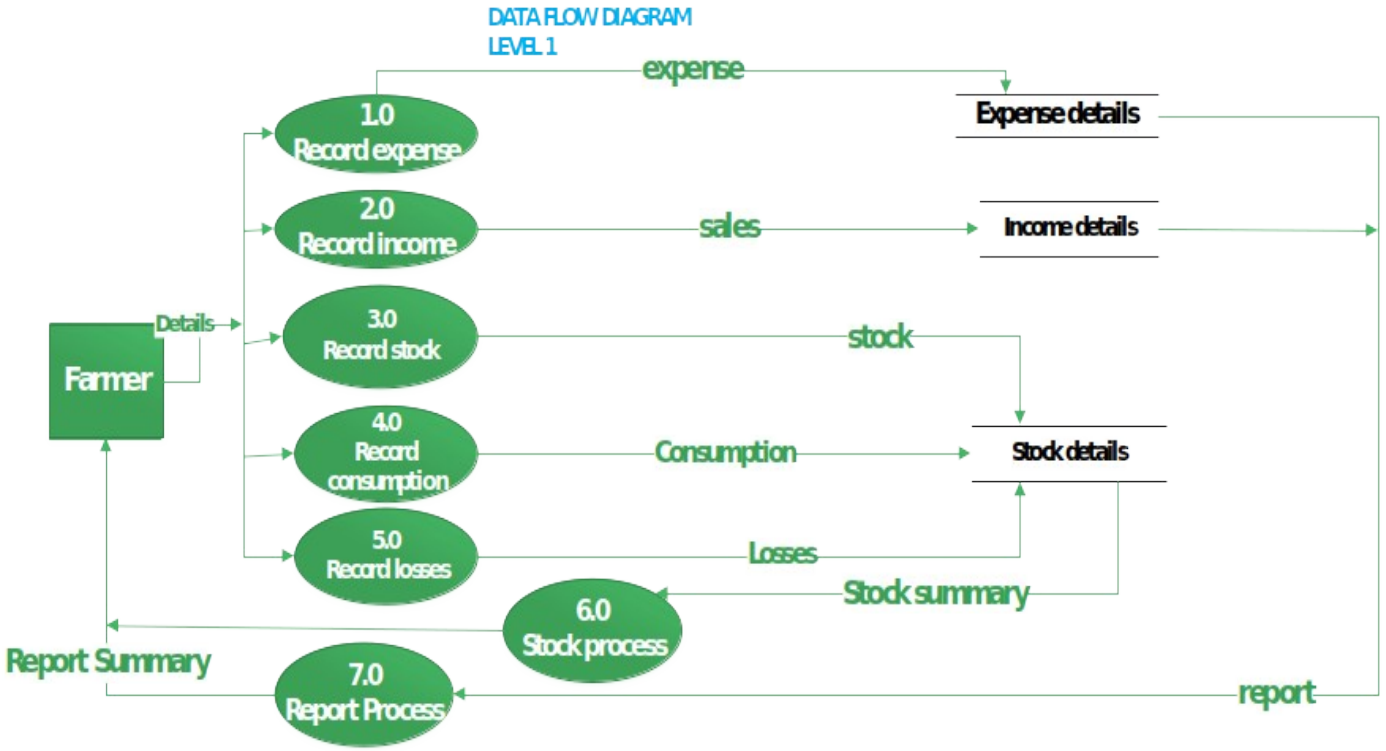


Application

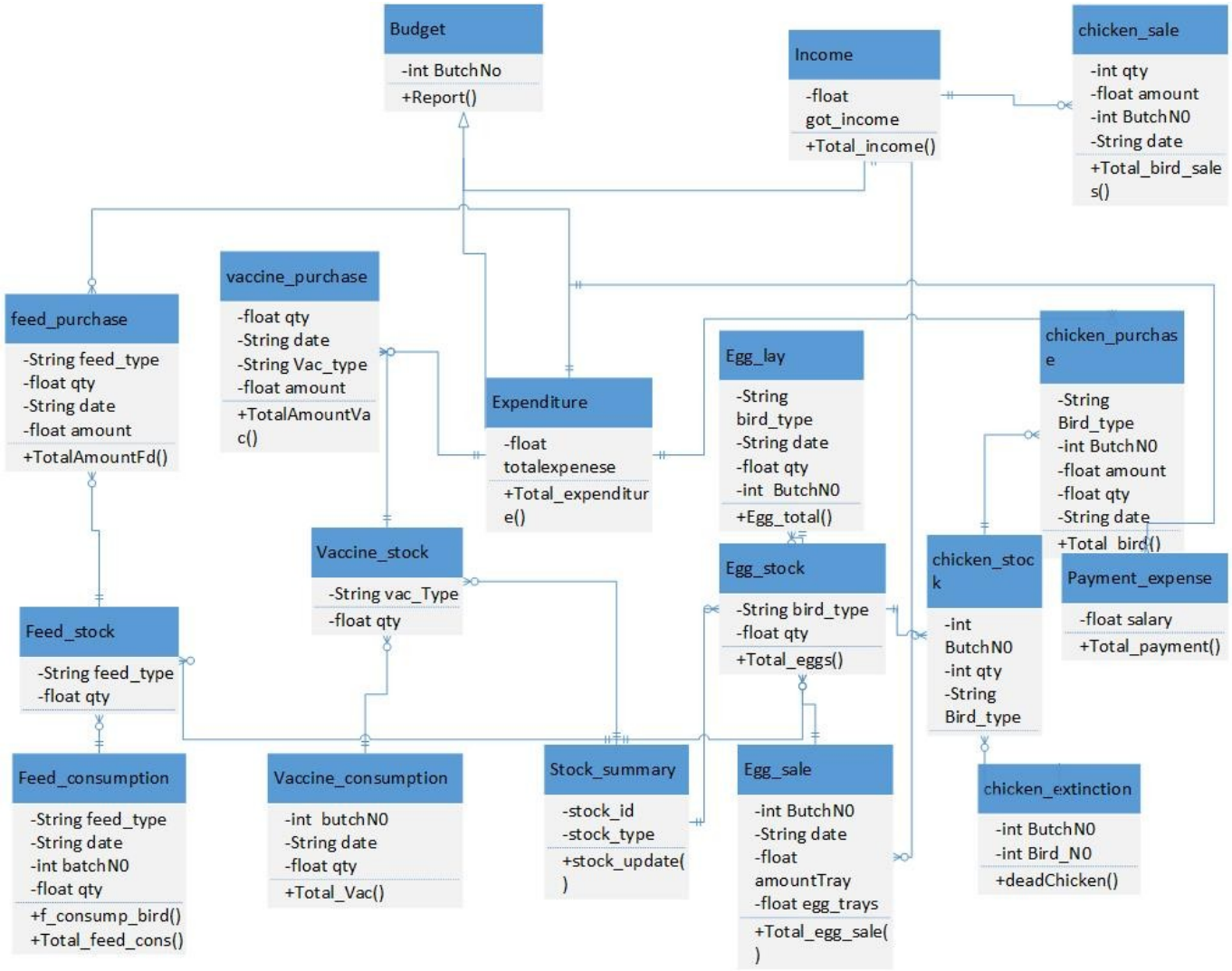
Backup

6 ARCHITECTURAL DESIGN

DATA FLOW DIAGRAM



CLASS DIAGRAM OF POULTRY RECORD KEEPING SYSTEM


Budget
-int ButchNo
+Report()
Income
-float got_income
+Total_income()
chicken_sale
-int qty
-float amount
-int ButchN0
-String date
+Total_bird_sales()
feed_purchase
-String feed_type
-float qty
-String date
-float amount
+TotalAmountFd()
vaccine_purchase
-float qty
-String date
-String Vac_type
-float amount
+TotalAmountVac()
Expenditure
-float totalexpenese
+Total_expenditure()
Egg_lay
-String bird_type
-String date
-float qty
-int ButchN0
+Egg_total()
chicken_purchase
-String Bird_type
-int ButchN0
-float amount
-float qty
-String date
+Total_bird()
Vaccine_stock
-String vac_Type
-float qty
Egg_stock
-String bird_type
-float qty
+Total_eggs()
chicken_stock
-int ButchN0
-int qty
-String Bird_type
Payment_expense
-float salary
+Total_payment()
Feed_stock
-String feed_type
-float qty
Feed_consumption
-String feed_type
-String date
-int batchN0
-float qty
+f_consump_bird()
+Total_feed_cons()
Vaccine_consumption
-int butchN0
-String date
-float qty
+Total_Vac()
Stock_summary
-stock_id
-stock_type
+stock_update()
Egg_sale
-int ButchN0
-String date
-float amountTray
-float egg_trays
+Total_egg_sale()
chicken_extinction
-int ButchN0
-int Bird_N0
+deadChicken()

APPENDIX

Requirement Analysis Concepts

Requirements Analysis refers also called requirements engineering is the process of determining user expectations for a new or modified product.

It can be divided into the following

User Requirements. Refer to expectations of user from the system to be developed

Function Requirements. Refer to how the system serves the users by ensuring efficient and effective functionality.

Non- Functional Requirements. Refer to constraints under which the system will operate

System Requirements. These are requirements that are needed to incorporate the desired functionality in the system.

Use case diagram. This shows user interactions with the system

Use cases. Different processes in the system

Data flow diagram. Indicates movement/flow of data within the system

Class diagram. This shows the relationship between structural components

Architectural design. This shows the overall structure of the system development. It is based on dataflow diagrams and class diagrams.

Fact Finding Techniques. We used Questionnaire and Interviews as methods to obtain requirements for system

REFERENCES

1. CS605-Software Engineering Practitioner's Approach by Roger S. Pressman
2. MGH.Software. Engineering. A. Practitioners. Approach. 8th.Edition.0078022126
3. Software Engineering for Game Developers by J.P. Flynt, Salem
4. Software Engineering a Practitioner's Approach 8th Edition
5. Gary_B._Shelly, _Harry_J._Rosenblatt] _Systems Anal(BookFi.org)
6. Software_Engineering_9th_Edition