Neighbourhood Farm Stock System and Data Analysis

Tawira Valentine
[1]*UWC, Robert Sobukwe Road, Bellville, Cape Town, 7535, South Africa*
Tel: +27 82478617, Email: kudatawira@gmail.com

Abstract: Neighbourhood Farm is initiative committed to the development of community market gardens for the growing of organic crops at schools in the Deep South of Cape Town stretching from Muizenberg through to Kommetjie. This project proposal will attempt the implementation of ICTs to amplify the farming operation of the Neighbourhood farm initiative. This part of the project will attempt to design an ICT based solution to resolve potential problems relating to stock system and some simple data analysis.

Keywords: SMS-Stock Management System, Food Security, URD,MVC

1. Introduction

Food insecurity is a major global challenge in both developing and developed countries with its main effects being heavily felt in poor marginalized populations within these countries. Food security exists when all people, at all times, have physical and economic access to sufficient, safe and nutritious food to meet their dietary needs and food preferences for an active and healthy life [1]. According to some population statistics published by United Nations, of the estimated additional 2.2 billion people who may be added to the world's population between 2017 and 2050, 1.3 billion will be added in Africa [2]. From the previous fact, food security has become an area of interest that requires much attention especially in the sub-Saharan region. Governments and non-governmental organizations across the globe have been pursuing programs and policies aimed at attaining food security in the past and recent years, with much effort being channeled in the developing countries where insufficient food is more a common issue. Despite the efforts, most commonly when such programs and policies are set up and initiated, they face several different challenges that threaten their impact in contributing to food security. Some of the challenges that the programs face can be linked broadly to operations and management. In this project we propose the implementation and use of Information Communication Technologies (ICTs) based solutions as amplifiers in the management of operations at farm level specifically concentrating on the Neighbourhood Farm (NF) initiative, a project that is build up inline with the food security programs. ICTs in the Neighbourhood Farm will be implemented in the areas of stock management, data analysis, payment and pre-order .However this part of the project aims to use ICTs in the areas of stock management and some simple data analysis.

1.1 About Neighbourhood Farm

Neighbourhood Farm is a project committed to the development of community market gardens for the growing of organic crops at schools in the Deep South of Cape Town stretching from Muizenberg through to Kommetjie. The communities in this area represent a full economic spectrum from well-resourced to critically in need. The Neighbourhood Farm project does not just aim to grow vegetables and community spirit, moreover there's an economic heart to the project as well. Beyond showing children how food is grown, the market garden provides a small revenue stream from distributing and selling healthy food to both the school and the local community. This not only makes the gardens sustainable but empowers the community through skills-based training and experience. In turn, these create further economic and entrepreneurial opportunities for marginalized community members [3].

1.2 Problem Statement

Efficiency in the operation of the Neighbourhood Farm initiative is one big stumbling block that can threaten the effective management of the whole initiative. Most projects of similar structure usually fail because of uneconomical management and operations hence a crucial need for the Neighbourhood Farm initiative to operate effectively [1]. With quite a number of players involved in the project operations, complex actions are

likely to be experience in the areas of general management of the farms (schools) that include stock management and the analysis of data.

Stock Management Problem

The movement of produce stock in and out of the farm needs to be carefully managed and monitored. The management of stock (produce) is very significant in the successful operations of each farm and the entire project at large mainly because it influences decisions relating to the quantities that need to be harvested, how much produce to be requested from the other farms as well as how much can be transferred to the other nearby farms among other important reasons. The ability to maintain a smooth flow of produce between the farm ground and the shop floor is a a key element. Currently the movement of produce between the farm ground and the shop floor is being tracked in a piece of paper that can be easily misplaced or damaged, resulting in discrepancies in the movement of produce. Allowing third party producers to supply the farm shop with their produce as well as the option of requesting stock from other farms needs a very effectual stock management system to keep track various forms of produce. Poor stock management system often results wastages and loss of the produce (food) which has be argued to be one of the major contributors to food insecurity.

Data Analysis

Analysing data does not only add operational value through motivation but also goes a long way in making inferences about future performance. Problems are likely to be faced in efforts to determine the performances of produces at farm level as well as at the project at large. The tracking of produce wastage and loss is also one potential solution to improve efficiency and productivity of the Neighbourhood Farm project.

1.3 Related Work

Malawi Agricultural Commodity Exchange Market (MACE)

MACE is a commodity exchange market platform that was introduced in Malawi by the government of Malawi. The ICT based MACE was introduced to attend to the the constraint of poor access to market information by farmers in order to strengthen the liberalised markets [4]. It was initiated to assist smallholder farmers to have access to market related information to search for better markets and prices for their produces. Through the use of MACE, an improved market performances of rice markets in Malawi was recorded. The success of the MACE provides a strong argument for the usefulness of implementing ICTs in farming or agriculture.

Israel Farm Management Department on-farm software package (Ma'ayan)

The software runs on a PC and enables the farmer to record production activities, monitor, quantify and cost them, make on farm decisions and follow product marketing. Ma'ayan was introduced after the Farm Management Department noticed that many farmers hand record their relevant data that relates to agricultural production, cost accounting parameters and other critical information resulting in an inability to efficiently recall, analyse and meaningfully use information when needed [5]. The first version of the software was released in 1991 and received favourable acceptance. Following on-farm visits and farmer training updates to the software were made and within a few years from the first version more than 600 copies of the software were sold.

Open Food Network (OFN)

The Open Food Network is an open source online marketplace for local food. It enables a network of independent online food stores that connect farmers and food hubs (including coops, online farmers' markets, independent food businesses etc); with individuals and local businesses [6]. It aims to provide farmers and food hubs an easier and fairer way to distribute their food. The Network was initially started in Australia and is growing to all the other countries including the recent launch in South Africa. The OFN functionality relates more to the NF. In both initiatives, a network is built between the local producers (third part in NF members), customers and the local farm or store. The network allows the aforementioned entities to connect with each

other. The OFN implements a two faced interface structure, the backend management of the system and the front store for customers. The NF intent to implement the same interface structure.

1.4 Proposed Solution

The implementation of ICTs to counter solve the problems that the Neighbourhood farm project will possibly face is one viable solution. Effective implementation of Information Communication Technologies will not only solve the stipulated problems but will also enhance and improve the farming practices in the communities. Using ICTs to effectively manage the stock system and perform some analysis to the generated will notably amplify the management and operations of the whole initiative.

Stock Management System (SMS)

The proposed stock management system aims to allow effective, productive and flexible stock operations between farmer, seller and third party players. The Stock System is mainly designed to allow for a full control of produce movement. The ability to have a clear view of all the information on the varieties and quantities of available produces enables the administrators to communicate correct information to the farms floor team and or third party members.

Data Analysis

The proposed system aims to extract and analyse data from all parties involved in the operations of the project. The system will view product performances, track wastages and loss in produces among other critical data analysis solutions. Extracting important information from the data generated across all the project activities and presenting it in a more analyzable format can allow the the initiative to make informed business decisions and future plans. The ability to view the produces that are making high orders can be of paramount importance and can allow farms to stock the right produces that will sale quickly without the risk of wastages.

2. User Requirements

The system will be designed to be used by four users who are entitled to different privileges and are exposed to different functionalities. The first user is a Super User who have total control of the system and who is responsible for managing all the farms (schools). An Admin User is privileged to take control of all activities for a specific farm under him. A Third Party supplier is a community member who is registered with any farm who is entitled to supplier a farm with his produces and will have an option to follow up on his produces using the system. The forth user is a Customer who also can be any community member or organization and is entitled to use the system to follow up orders and contributions to the neighbourhood farm. For this part of the project the user requirements that will be defined are that aligned to the interaction of all these different users with the SMS and Data Analysis.

2.1 Requirements Gathering

For this project a number of requirements gathering techniques were performed in order to have a clear set of the user requirements and a concise view of the expected solutions. First a formal meeting was conducted where the founder of the Neighbourhood Farm initiative introduced the background and underlying details of the project to set a clear picture of the project. Also a similar meeting was set up a few weeks from the first meeting but this second meeting aimed at outlining a high level view of the requirements and addressing some questions from the first gathering. A pilot visit to one of the school that will be used as a model farm followed the two formal meetings, also a visit to a farm with similar setup was done and a more enhanced understanding of the farm set up was gained. The last meeting followed a couple of weeks later with the aim of presenting a first version of a prototype and build up feedback as well as suggestions from the project manager.

Table 1: Requirements Gathering Table

Date	Technique	Goal	Identified requirements
9 Feb 2018	formal meeting	Background of the Neighbourhood Farm project Underlying Details	Stock management Data analysis

27 Feb 2018	formal meeting	Highlight problems and requirements Answer questions and clarifying concepts	Produce requisitions Produce transfers
10 Mar 2018	pilot visit	View ground set up View similar farm set up	Record processed stock Track wastages
7 April 2018	formal meeting	Presenting first prototype Get feedback and suggestions	Track produce shelf expiry View demands performances
10 Aug 2018	site visit	observe farm operations visit the farm store	Need for communication medium between the farm floor and store
Weekly Meetings (Group)	Brainstorm	Updates on progress and challenges	Order follow ups Farm performances

3. Requirements Analysis

With the user requirements identified during the gathering process, the requirements analysis proceeds, breaking down the gathered requirements into more compact functional and non-functional requirements.

3.1 Functional Requirements

An overview of the high level functional user requirements that were identified and recorded out through the requirements gathering process are formatted in *Fig 1*. The requirements in the above fig are not organised systematically and are organised according to the user type.

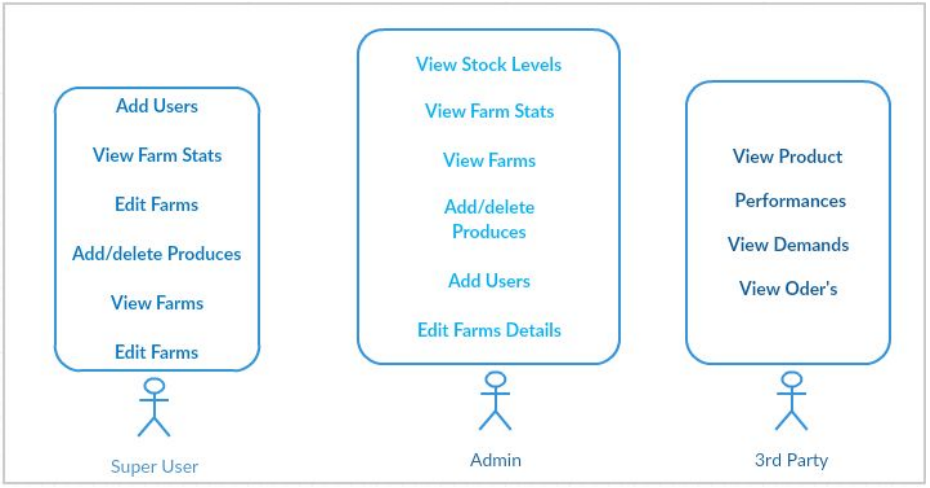

Fig 1: User Requirements Diagram (URD-functional requirements)

3.2 Non-Functional Requirements

A number of non-functional requirements were identified during the data gathering process and some of the requirements include visibility, usability, security and capacity.

Visibility
The aim is to design a simplified system with the most used options clearly visible. Critical notifications should be pushed to the home screen in order to be easily noticed.

Usability
There is a need to design a platform that is easy to use with little or no training needed to operate the system. The main emphasis is to design in a way that allow user to use recognition other than recall

Capacity

There is a need to enable the system especially the SMS to handle large volumes of stock data as large volumes of harvested produces will be pushed into the system.

Security
There is a need to ensure that there is security and data privacy for the users.

4. System Design

The gathered and analysed requirements from the previous chapters provides a good basis to proceed with the designing of the proposed solution. In this section we will transform the analysed requirements into design models that describe the details of the data structures, system architecture, interfaces, and components necessary to implement the proposed solution.

4.1 Data Design

From the requirements gathered it is quite clear that the SMS as well as the Data Analysis (DA) component of the proposed involves some high volumes of data and it is crucial to have some good structures to handle these numerous volumes. The functional requirements highlighted in the URD (*fig 1*) gives a good understanding of the dimension of data that needs to be handled .Data that include the different farms (schools), produces, quantities of produce in stock etc needs to be handled effectively and efficient data designing is required to produce a good product of the proposed solution. Fig 3 below show the dimension of data flow from the major players that will the involved in the proposed solution

4.2 Data Analysis Design

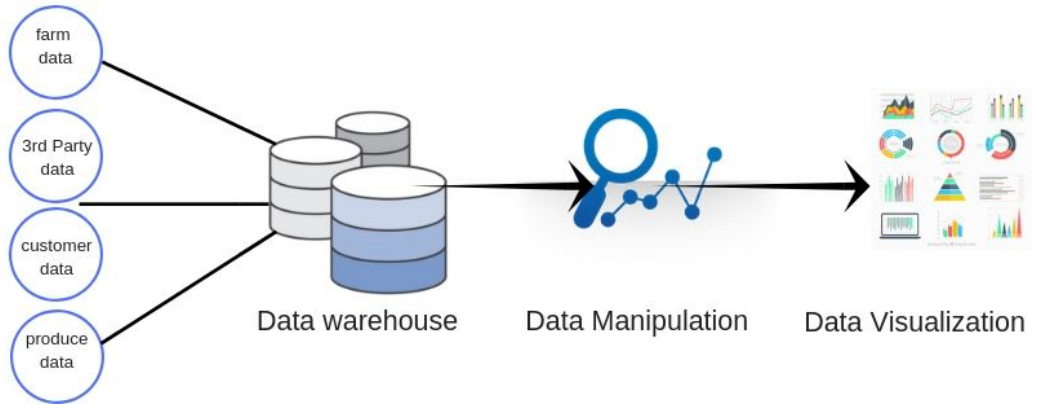

Fig 2: *Data Analysis Implementation Diagram*

The Data Analysis part of the project follows 4 stages that begin with the data being received from different parties that will interact with the proposed system. The data that is generated by the 4 parties highlighted in the data design section is stored in a secondary database. The secondary database is used as a data warehouse medium that stores all the important data that can be extracted from the application's primary database . The main motivation for a secondary database is to allow access to unapdated or altered data for precise analysis. For analysis and visualization, powerful statistical packages and languages can be used , however in this project we resorted to the use of simple sql queries to organise the data for graphical visualization.

4.3 *System Architectural Design pattern*

The MVC paradigm is a way of breaking an application, or even just a piece of an application's interface, into three parts: the model, the view, and the controller [7]. MVC was originally developed to map the traditional input, processing, output roles into the graphical user interface realm. The classes that are described in the Low Level Design section *Table 2* are organised in the MVC design pattern.

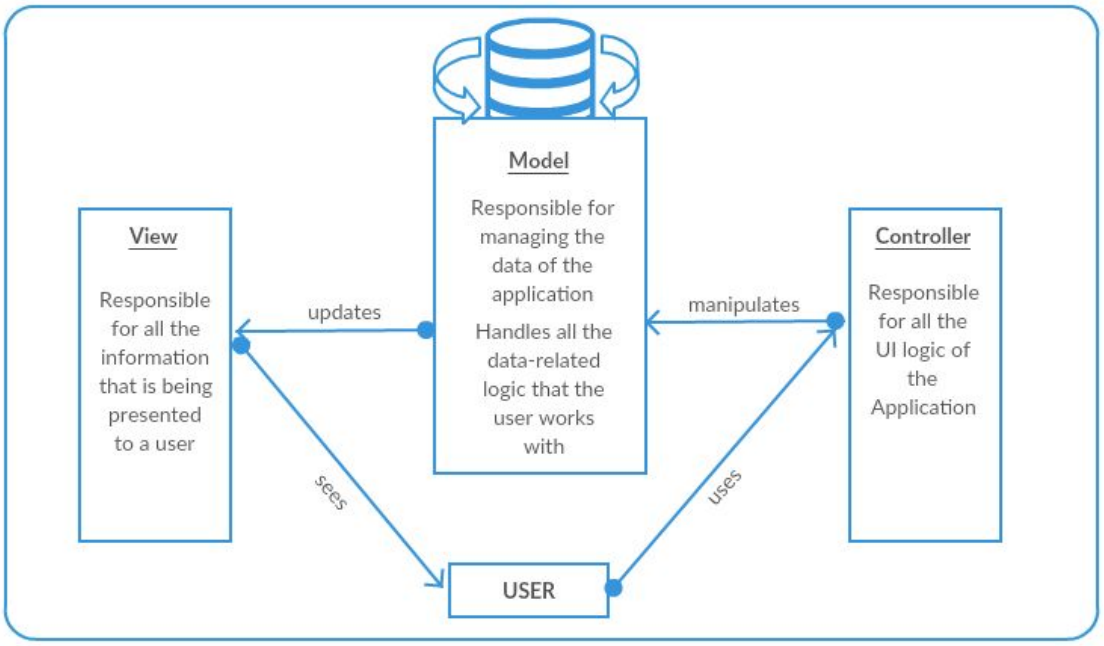

Fig 3: MVC architectural Design pattern

4.4 Prototype Designing

In this section we will develop a typical prototype of the proposed system based on the designs covered in the Design section to depict the concepts, design alternatives, and screen layouts. For the prototype, only the main functionalities of the proposed system are considered and tools such as database and analytical tools are not implemented in the backend.

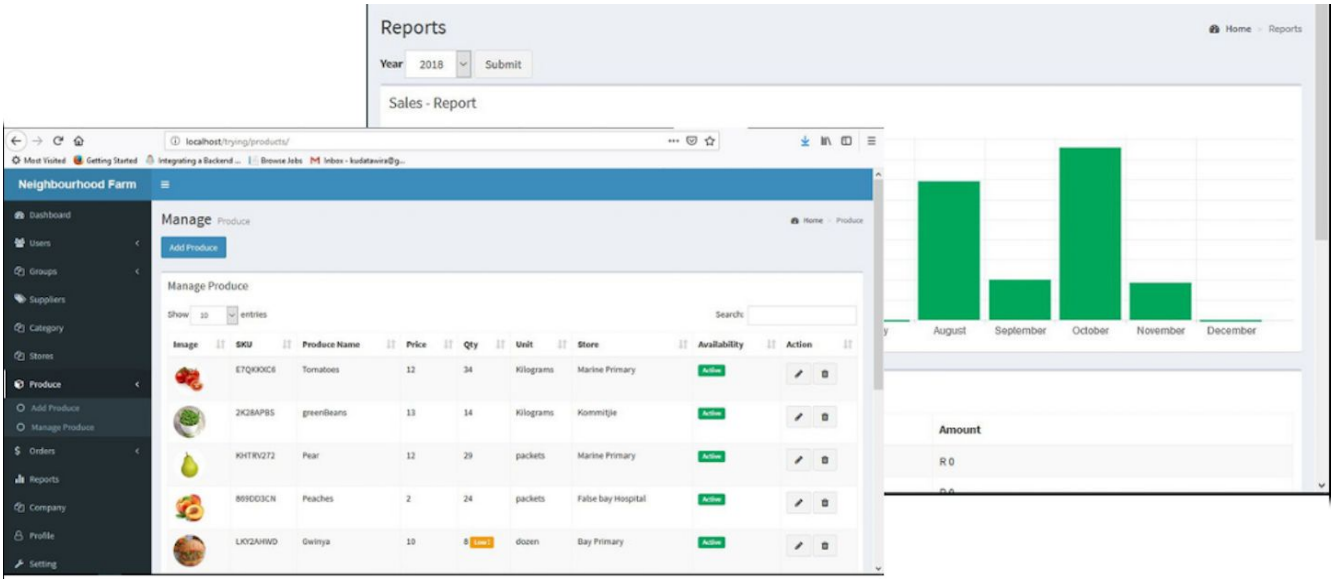

Fig 4: Stocked Produce and Analysis View

The stock tab allows the user to view the stock level of the produces. The stock will be categorized in different types such as vegetables, fruits, herbs etc. and upon selecting a certain type a list of the produces that falls under the category will be displayed with their respective available quantities. The stock gives the user an option to request or send stock from and to other farms respectively as highlighted in *Fig 4*. The Analysis tab allows the user to view some analysis displayed in graphs and tables. The interface displays tables and graphs with general statistics on wastages, demand and harvest.

5. High Level Design

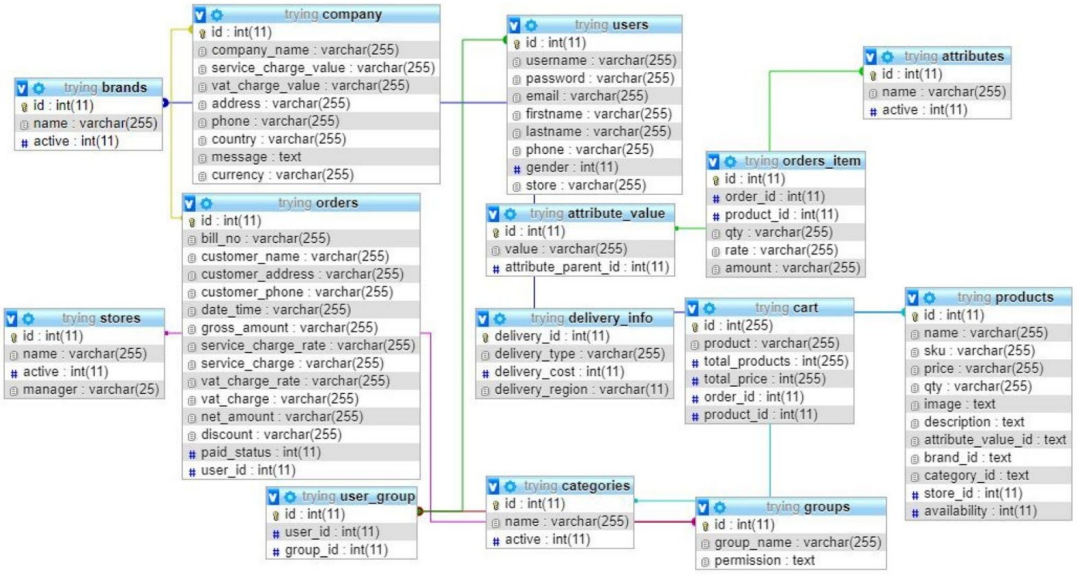

Fig 5: Entity Relation Diagram (ERD)

The high level design of the backend database can be represented by *Fig 5 above.* The design of the database is structured to represent both the backend and frontend entities. The above ERD only describes all the major entities in the database and their relationship with each other .

6. Low Level Design

Table 2: Class Description Table

Component	Name of Classes	Description
Controllers	Auth,Categories Suppliers Dashboard Orders Reports Stores Products Users	A controllers are a means by which the user interacts with the application. A controller accepts input from the user and instructs the model and view to perform actions based on that input. In effect, the controller is responsible for mapping end-user action to application response. The controller translates interactions with the view into actions to be performed by the model.
Models	Model_Auth Model_Categories Model_Suppliers Model_Dashboard Model_Orders Model_Reports Model_Stores Model_Products Model_Users	An object representing data or even activity. The model manages the behavior and data of the application domain, responds to requests for information about its state and responds to instructions to change state.The model is the piece that represents the state and low-level behavior of the component.

Views	Auth Categories Suppliers Dashboard Orders Reports Stores Products Users	Views are responsible for some form of visualisation of the state of the model. The views manages the graphical and/or textual output to the portion of the bitmapped display that is allocated to its application. The views renders the contents of a model. It accesses enterprise data through the model and specifies how that data should be presented. The views are responsible for mapping graphics onto a device. Views typically has a one to one correspondence with a display surface and knows how to render to it. Views attaches to a model and renders its contents to the display surface.

7. Implementation

For the implementation stage of this project, we opted for the quite a number of tools to write the code and followed the MVC design pattern to structure the application.

7.1 *Code*

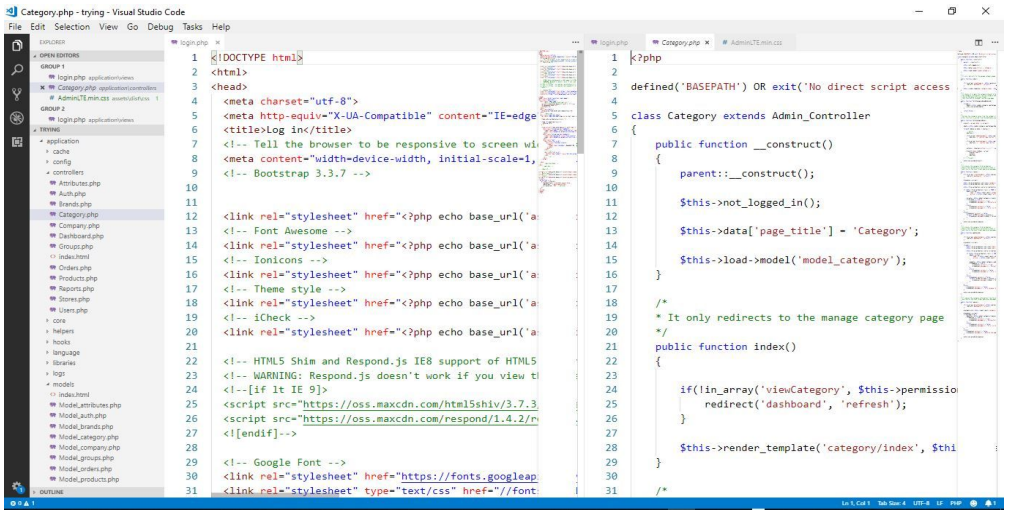

Fig 6:Code Snippet

The above fig shows a snippet of the application's code built during the implementation stage.

7.2 *Tools Used*

Table *3: Tools and Application Table*

Tool	Application
CodeIgniter	development framework
PHP	Server-side scripting
SQL	Database design and management
HTML , CSS	Web structuring and Styling
Xampp	local server

The above Table shows the tools that are used in the implementation stage and their applicability

8. Testing

This chapter brings to detail all the testing that has been performed on the implemented solution to evaluate its correctness against the requirements identified in the Requirements Gathering *Table 1*. We performed quite a number of testing methods that include unit test , integration test ,usability test and functional test to the application and details of the results are explained below.

8.1 *Unit Testing*

Unit test was performed to test different functions of the application separately to determine the correctness of the function output results. The table below outlines the functions tested as well as the results obtained.

Table *4: Unit Testing Results*

	Function	Test Description	Result
Views	editProduce reportsView	Test form input validation Test for correct graph values	Input validation leaked some cases correct graph figures
Models	getProduceData removeProduce	Test retriving from database Test updating database entries	correct database data retriving correct database updating
Controllers	login fetchOrderData	Test for correct Authentication Test for fetching correct data	correct Authentication correct data fetch

8.2 *Integration Testing*

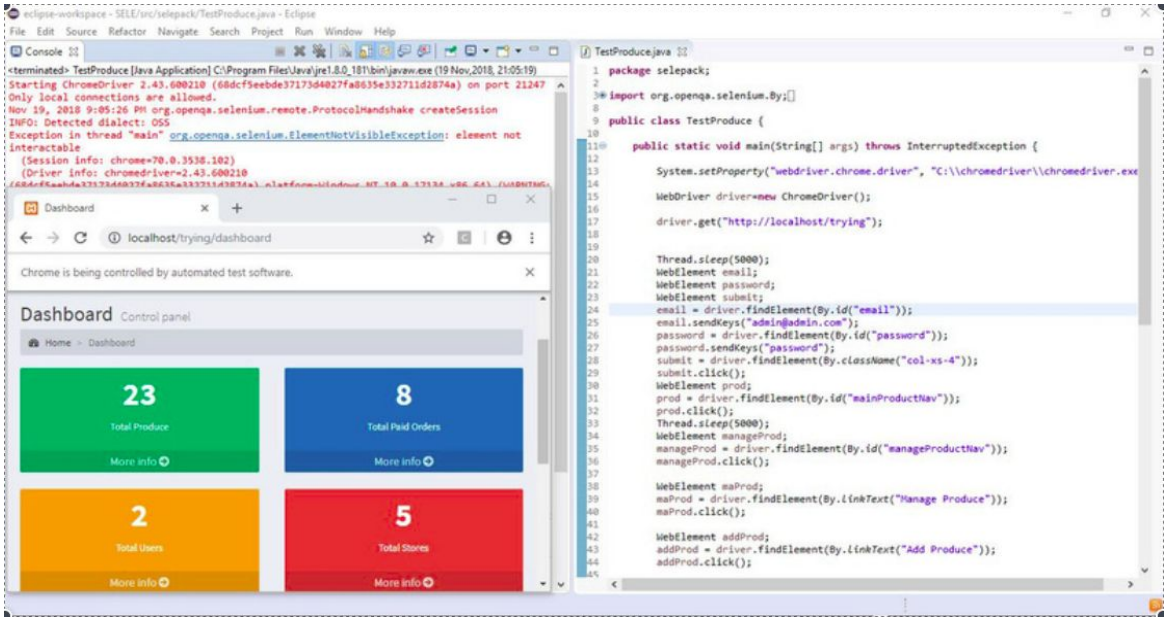

Fig 7 Integration Testing Selenium WebDriver

The aim of integration testing was to combined all the functions of the application and proceed to test them as a single module using selenium Webdriver. We tested all the functions and classes in the models ,views and controllers to evaluate the ability of these three separate modules to connect to each other. A java code was written to run an automated test of the application. The application responded well to the test.

8.3 *Functional and Usability Test*

A functional and usability test was performed to evaluate the ability of the application to meet the required specification. A sample of 5 users with different farming and technological background to perform a usability test was chosen. The main objective of the usability test was to assess the extent to which the users are able to navigate the application using recognition other than recall, thus testing if the application is user friendly. The results of the test are tabulated below.

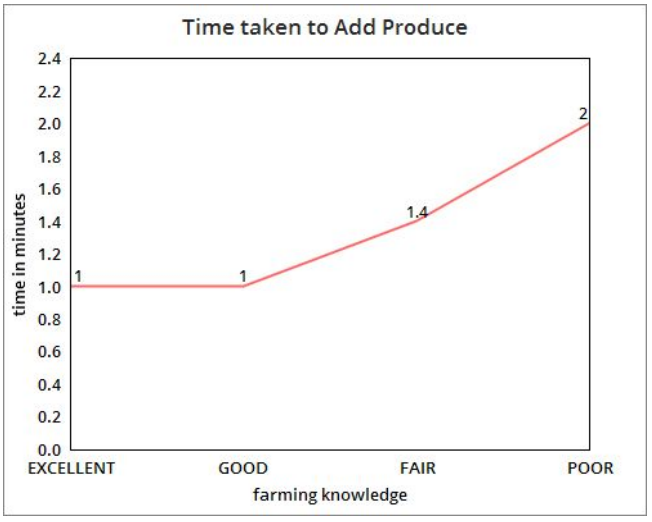

Fig 8 Task Duration

The results of the functional test are recorded in the below table with an indication relating back to the defined user functional requirement.

Table *5: Functional requirements results table*

Test Number	Functional Requirement	Status
1	manage produce	complete
2	view produce levels	complete
3	view stats	complete
4	analyze data	partial complete
5	add farms	complete
6	view orders	complete

9. Conclusion

The ICT's are effective tools for enhancing the management and operations of farms. Information Communication Technology based solutions can indeed be amplifiers in the management of operations at farm level. The proposed ICT based solution for the Neighbourhood Farm initiative was successfully implemented and quite a number of requirements were met.

9.1 *Recommendation and Future work*

Despite the ability to have implemented most specifications, there a few more specifications that were not fully meet because of different constraints that include the time frame. Based on the experience acquired in the project, more progress and success can be achieved if the implementation is carried out at different development stages with more detailed and specific stage goals. This avoids the risk of generalizing functional requirements. It is also recommended that the data analysis part be explored as a separate module as it has quite a lot of requirement and is depended on the other projects facets.

References

[1] Food and Agriculture Organisation of the United Nations. The State of Food Insecurity in the World 2011. Food and Agriculture Organisation; Rome, Italy: 2011. p. 51.

[2] United Nations World Population Prospects The 2017 Revision. Key Findings and Advance Tables. ESA/P/WP/248

[3] J. Bonello, Neighbourhood Farm, Neighbourhood Farm, 2018. [Online]. Available: http://neighbourhoodfarm.org. [Accessed: 03- May- 2018].

[4] Katengeza, S.P.; Mangisoni, J.H. and Okello, J.J. The Role of ICT-based Market Information Services in Spatial Food Market Integration: The Case of Malawi Agricultural Commodity Exchange, Contributed Paper presented at the Joint 3rd African Association of Agricultural Economists (AAAE), September 19-23, 2010.

[5] Bracha Gal, Yoav Gal, Ehud Gelb. Adoption of Information Technology for Farm Management, A Case Study. International Farm Management Congress 20013

[6] "Organisation", Open Food Network, 2018. [Online]. Available: https://openfoodnetwork.org/about/organisation/. [Accessed: 14- Oct- 2018].

[7] T. Marston, "The Model-View-Controller (MVC) Design Pattern for PHP", Tonymarston.net, 2018. [Online]. Available: https://www.tonymarston.net/php-mysql/model-view-controller.html. [Accessed: 14- Oct- 2018].