SMS BASED RURAL AGRICULTURAL MARKETS MONITORING INFORMATION SYSTEM IN KENYA

Prepared By : KIPROTICH CHEMWENO

Supervised by : DR. ROBERT OBOKO

MSC COMPUTER SCIENCE

UNIVERSITY OF NAIROBI

2012

This report is submitted in partial fulfillment of the requirements for the degree of Master of Science Computer Science of the University of Nairobi. This report is the product of my own work except where indicated in the text. The report may be copied and distributed freely in part or whole as long as the source is acknowledged.

This project, as presented in this report is the product of my original work and has not been presented anywhere else for any award whatsoever.

Signature: .... Date ....31/10/2012..........

Kiprotich Chemweno

Reg. No: P58/62236/2010

This project has been submitted as part fulfillment of the requirement for Masters of Science in Computer Science with my approval as the University Supervisor.

Signature: .... Date 2/11/2012..........

Dr. Robert Oboko

School of Computing & Informatics

ABSTRACT

Agricultural market information in Kenya has been inaccessible to rural farmers. The rural farmers have been left to depend on traders who dictate prices of their produce leading to low prices that are unfavorable to farmers. Most of existing solutions that provide market information use SMS technology which is widely available and cost effective in Kenya, however, requirement to adhere to a strict syntax, registration via the web and the fact that most of these systems are offered in English making most rural farmers who cannot effectively communicate in English unable to use the system.

The lack of information has been worsened by a combination of factors that include; low levels of literacy, limited connection to internet and high cost of broadband internet when available, and weak purchasing power among rural farmers. In this research, a framework was developed around the challenges mentioned above. The framework seeks to guide development of an SMS (Short Message Service) system that will act as a platform with which rural farmers and traders can share information regarding price of commodities in various markets. Rural farmers are also able to place offers for produce they have and intend to sell. On their part, traders are able to advertise the produce they wish to buy. These offers to sell and buy are matched in the system with buyers receiving contacts of sellers and sellers receiving contacts of buyers. At this point the two parties can communicate using their phones and strike a deal.

The system was developed using agile development methodology with iterative development and testing done. On completion of development, rural farmers were recruited to use the system. In four out of eight Kenyan provinces, two rural markets were sampled. Five rural farmers and five traders in each of the markets were sampled. The participants were trained on how to use the system and given access to use it for a period of two weeks. Messages sent by users and responses sent to users were used to determine accuracy of the system. The participants were interviewed and their responses used to fill a SUS (System Usability Scale).

It was found that the system accurately translated 95% of all incoming user messages. On the SUS scale of 0 to 100, the system scored 93.6% with feedback from 75 users.

These results show that problem of lack of affordable and easily accessible market information can be solved using a system like the one developed in this research by following a framework that targets solutions to challenges faced in accessing agricultural market information in rural Kenya.

ACKNOWLEDGEMENT

First I would like to express my most sincere thanks to God who has been a constant source of strength, direction and inspiration throughout and particularly during masters study.

Sincere thanks and gratitude to my supervisor Dr. Robert Oboko for guiding me throughout the entire project from inception to the writing of this report. Without his guidance, the scope of this work would have been too wide and un attainable.

I would also like to express my thanks to my family, my wife Drusclla Lorot, my son Eden Cheserek, beloved parents Patrick Chebelieny and Magdalene Teriki. Without your constant support and encouragement this work would not have been possible.

Lastly, thanks to my colleagues at Food and Agriculture Organization (FAO) Somalia for their great ideas on refining my project topic and to my classmates at the University of Nairobi class of 2010 for their positive criticism in this project.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

CCK	Communications Commission of Kenya
COMESA	Common Market for East and Southern Africa
FAMIS	Food And Market Information System
FAO	Food and Agriculture Organization
GSM	Global Systems for Mobile communications
KIPPRA	Kenya Institute of Public Policy and Research Analysis
MIS	Market Information System
QUE	Query Understanding Engine
RAD	Rapid Application Development
RATIN	Regional Agricultural Trade Intelligence Network
SAMIS	SMS Based Agricultural Market Information System
SMS	Short Message Service
SQL	Structured Query Language
SUS.	System Usability Scale

GLOSSARY

Farm Gate Price ....Prices charged by farmers without incurring any transport to market

Offer...Intention to buy or sell by a farmer or trader

Query ....A question on the situation in the market regarding for example price of a commodity

Query Understanding Engine...... A set of algorithms for syntehsizing and defining meaning from an input query

Short Code Number .... A shortened mobile number for ease of rememberance

SQL Query........ A structured query language query, the mechanism for manipulating data stored in a database or manipulating a database structure

System ....A software built to provide a solution to a certain existing problem

1. INTRODUCTION

Kenya is an agriculture based economy with majority of the citizens relying on small holder agriculture production. Most of this small holder agricultural production is practiced in the rural areas. Farmers who produce agricultural commodities and traders who deal in agricultural produce lack a platform for exchanging information. Information such as availability of supply or demand is useful to both parties. Currently, such information does not find its way easily from one party to another especially where the parties are far apart. Media such as Internet are not easy to come by for small holder agriculture farmers and traders. Various factors contribute to ensure that market information is not freely available to the two parties. These include; low literacy level that makes media like newspapers and Internet not easy to use by some traders and farmers, high cost of Internet connectivity, low purchasing power that makes media like television beyond the reach of most rural dwellers.

1.1. Problem Statement

Traders and rural famers experience market information drought, they do not have access to information on market conditions. Information on prices, levels of demand and supply in the markets is hard to come by. With a mechanism that links demand for and supply of agricultural produce, Kenyan rural farmers would be able to earn more money. This is because they get to gauge prices offered by various traders and choose the better option. Also there would be reduced quantities of food perishing in storage. Since farmers would be able to source for market for their produce using the established mechanism, there would not be need to keep the produce in stores for too long. Traders would benefit too since their network for potential suppliers is increased. Their ability to know of future supply means traders can effectively plan and source for future demand well in advance. Having a long list of suppliers will boost the traders' business because they will be able to get best possible prices.

There have been attempts to make market information more readily available. However, most of these solutions have been geared towards providing information to donors or government departments. This information does not become of much use to famers and traders because they do not access it in shortest time possible. Timely information is important because agricultural produce is perishable and therefore markets conditions fluctuate. For market information to be reliable, it ought to be as up to date as possible. Another reason that has made most of the solutions advanced for market information availing to have little impact on farmers and traders has to do with the media used in those solutions. Some solutions like Food and Market Information System (FAMIS) and Regional Agricultural Trade Intelligence Network (RATIN) are web based. Very few if any rural farmers have access to Internet. This could be because of lack of connectivity or low literacy levels.

Viable solutions to providing market information like prices of commodities, demand and supply ought to use more readily available media that do not demand high literacy levels on the individual user. Literacy levels dictate in a big way whether a solution will be viable to the target population or not. Lack of proper understanding of the English language in rural areas of Kenya demands that solutions targeted at rural farmers and traders be flexible enough as to incorporate other widely spoken languages. For this project, Kiswahili as well as English are supported by the solution that is built .

Comparing mobile phones and the internet, not only do mobile phones come at more affordable prices but already majority of Kenyan population own mobile phones, 72 per 100 Kenyans (CCK, 2012). The explosion of mobile phones in Kenyan is phenomenal, this means that it is easier to get in touch with people using mobile phones than using any other media in Kenya. Aker and Mbiti (2010) observe that mobile phones can have a great impact on people's livelihoods by connecting the population to markets.

They describe the phenomenal growth in mobile telecommunication over the years and economic efficiency that mobile phone technology can bring to agricultural markets.. In this project, SMS (Short Message Service) has been chosen as the data transfer medium as opposed to implementing systems that require high end phones such as smart phones. The choice of SMS is largely informed by the target audience of the solution that is advanced by this project, rural farmers and traders. Most rural farmers and traders can hardly afford smart phones and other high end phones. Using SMS ensure that all farmers and traders will be able to access the system without having to have smart phones.

1.2. Overall Aim

The main aim of this project is to develop an SMS based system where farmers can send SMS message to announce the produce they have. On the other hand, traders can also send SMS message to announce produce they are seeking to buy, they could also inquire prices of produce in various locations. Using only SMS messages, a farmers intention to sell and a trader's offer to buy can be matched. These two parties can then contact each other directly using mobile number that either the farmer or trader receives in a reply SMS.

1.3. Specific Objectives

The specific objectives in this research are;

i. To develop a QUE that processes queries received in Kiswahili or English.
ii. To develop an SMS gateway for receiving queries from farmers and traders and sending back requested information to farmers and traders.
iii. To test the Agricultural Market Information System for accuracy in processing and understanding queries sent in Kiswahili or English and for usability.

To achieve the aim of this project, SMS Based Agricultural Market Information System (SAMIS) framework will be followed. This framework has been developed to address the unique challenges affecting agricultural market information gathering and dissemination in rural agricultural markets. The components of this framework are; accessibility of system where SMS is selected as the more accessible medium, language support where English and Kiswahili are advanced as crucial to making the system usable by more rural farmers and traders, query syntax flexibility. Query syntax in other systems has been found to make them less fault tolerant. Any slight variation from the syntax leads to message not being understood. The fourth component in the framework is pricing model, whereas other existing solutions charge users fixed amount for use over a given period of time, SAMIS framework suggests a pay per use payment model. This ensures that users always get value for money spent.

With the framework built, overall system architecture is put in place. The system is composed of five major components; mobile phone that is used by farmers and traders to send messages to the system, SMS gateway, query understanding engine, database for storing messages and prices, a web interface from where system administrator is able to monitor the system and intervene in providing human intervention to queries that were not properly understood by the query understanding engine. With the five components, it is important to highlight two parts that are very crucial to the solution, SMS gateway and query understanding engine. An SMS gateway is composed of hardware (mobile phone or Global Systems for mobile Telecommunications (GSM) modem) and software. It provides services of sending SMS from computer and receiving SMS messages on computer. SMS gateways can be varied and can have more features.

The SMS gateway that is referred to in this paper is one that provides the functions of sending and receiving SMS messages using a GSM mobile phone as a GSM modem, this is the hardware part. The software part is a number of Java programs. The other important component is a query understanding engine. This is purely a software program that takes an SMS message as input and sets out to understand the message with a view of performing an action that will satisfy the sender. The query understanding engine applies a number of things in achieving its task, one is the syntax that is set. For example "[INTENT] [COMMODITY] [MARKET]". INTENT specifies the intention of the sender, for example price inquiry, placing offer to sell or offer to buy. COMMODITY stands for the product that the sender is inquiring about or intending to buy or sell.

MARKET refers to the market or area of interest that the sender is asking about. A sender could be interested in the price of a commodity in a given market, they could be specifying that they have a commodity for sale at the market location. They could also be interested in buying the commodity at that market location.

2. LITERATURE REVIEW

Rural agricultural markets monitoring helps farmers and traders to get timely information on what is on demand and where the demand is. With this information, a farmer is able to make decisions before incurring cost of transporting the produce to the market. Similarly for a trader, knowing the produce that is in supply and prices allows them to make critical decisions of what to buy and in what quantities.

The SAMIS framework with its four components; accessibility, language support, query syntax flexibility and pricing model guide the project. It does so by providing a guideline on how the solution is created so that most of the challenges to be addressed will be solved. Adopting SMS has made sure that the solution is accessible to majority of the rural population who have intentions to inquire for prices, inquire about availability of produce or announce the availability of their own produce. SMS is an easy to use communication mechanism where the SMS sender types in a message using their phone and sends the message to a number f the recipient. The major advantage of using SMS is its availability in all mobile phones even the lowest end. Most mobile phones are SMS enabled. Support for English and Kiswahili are the two major languages spoken in Kenya. In the rural area, farmers and traders who cannot speak one of the languages can most likely speak the other.

Supporting both languages ensures a high proportion of rural farmers and traders are able to use the system. Human error is inevitable in any kind of situation, more so when typing of SMS message is involved. Mobile phones have been designed to fit into the human hand, this means they are not large in size. More often than not, errors are bound to happen during typing the message. The specified syntax can also suffer from users wanting to send query in the same way they would ask a question. For example if the syntax is "[INTENT] [COMMODITY] [MARKET]", a correct message would be one like "Price Maize Wajir". However, a user can miss the syntax and type in "Price of Maize in Wajir". In a syntax strict system, the second query would not be understood. In this project the syntax put in place is flexible enough to tolerate syntax errors as seen in the second message example above. Rules have been built into the query understanding engine in order to handle messages that have syntax errors. This will ensure that users have a better experience while using the system. The pricing model adopted by a solution can determine whether it is used or not.

To ensure that majority of rural farmers and traders who are the main target of this system are able to use it, the pricing model is on use basis. There are unique challenges in the rural Kenya, a user who subscribes to use a system to market his produce for fixed period can lose money. This can happen when the expected harvest is destroyed by elephants or unexpected change in weather patterns resulting in flooding or drought. Paying per use ensures that farmers and traders only pay when they are using the system and not paying to use it in the future.

There have been attempts to provide solutions that seek to provide market information. The systems developed have had varied objectives from providing market price information for national planning to government institutions to providing information for monitoring of household livelihoods to donors funding various interventions. Few of the systems have focused on the farmer and trader with a view to spurring trade and increasing earnings by the two parties. These systems are outlines below.

2.1 FAMIS (Food And Market Information System)

FAMIS is a web based application system that captures trade information for major tradable commodities in Common Market for East and Southern Africa (COMESA) region. FAMIS not only has commodity price information of products, it also has other useful data. This include production data, and a list of major transporters with their contact details. The system helps traders and farmers within COMESA to get prices of various commodities, place offers to sell or buy. The FAMIS user manual indicates that FAMIS has a Regional Trade Platform which is a virtual platform that is used to facilitate trade in commodities between buyers and sellers from the COMESA region. A run of this system in February 2012 indicates that, the data available is not up to date. Most recent prices being of the year 2010. This lack of up to the day data does not serve the farmer and trader well in that they cannot rely on historical data alone. Another major setback with this system is its availability only through the web. FAMIS is inaccessible to rural farmers and traders who have no internet connection. Coupled with these is the availability of the system only in English which locks out Kenyan rural farmers and traders who do not understand or speak English. This system is useful to many users, but rural farmers and traders are not among them.

2.2RATIN (Regional Agricultural Trade Intelligence Network)

An attempt has been made to develop a system where users can register to receive prices of commodities by Short Message Service (SMS). RATIN is one such system. It was developed as a joint effort by three USAID (United States Agency for International Development) projects, among them are Famine Early Warning System Network (FEWS NET) and Regional Agricultural Trade Enhancement Support (RATES). The major task of RATIN is to supply traders with improved early warning marketing and trade information that would lead to more efficient and competitive transactions in food trade in East Africa. As such, traders are the major target group of this system (About RATIN – Regional Agricultural Trade Intelligence Network).

RATIN is not only a web application for presenting agricultural price information, it also has useful statistical analyses that include food balance sheet and regional trade flows. These statistical measures help traders to make critical decisions regarding their trade in agricultural commodities. With prices being presented on a monthly basis, RATIN solves the problem of out of date data that is a bottleneck in FAMIS. Even with the availability of an SMS service, subscription to the SMS alerts is done via the web. This in essence means that to benefit from price alerts, users have to visit the web, something which not many rural traders with little computer knowledge are able to use. The fact that focus of this system is on traders in East Africa clearly shows that it is not the right tool for spurring trade between rural farmers and traders.

Providing information on prices of commodities in various markets without linking farmers who have produce and traders who need the produce limits trade between the two parties. Even with focus given to traders, not all of the traders understand and speak English. Traders who speak and understand English are able to use this system whereas their counterparts who do not understand English cannot use the system. RATIN is only available in English. Lack of support for Kiswahili which is spoken by more Kenyan rural farmers and traders limits the user population of RATIN.

2.3ESOKO

ESOKO, a web and mobile agricultural market information system developed in Accra, Ghana at BusyLab and used in nine African countries (BusyLab, 2012) including Ghana, Nigeria, Ivory Coast and Burkina Faso. Like RATIN, users can set up alerts so that commodity price information can be delivered by SMS to their phones. In addition, ESOKO allows farmers and traders to place offers to buy or sale online. With these offers, buyers and sellers can contact each other to negotiate prices and other terms including transport and delivery time and date. More rural farmers and traders are able to use the system to get information regarding prevailing prices of commodities in the market with the use of SMS messages. As this system is rolled out in more and more countries, more languages have been added to serve more farmers and traders. ESOKO solves most of the challenges encountered earlier in FAMIS and RATIN. It solves the challenge of accessibility by allowing users to set up SMS alerts. These alerts deliver price information and other useful agricultural market information. ESOKO supports English, Portuguese, French, and Arabic. The support of multiple languages allows users spread over a large geographical region who speak these four languages to use the system. Use of codes to refer to market names, shops and commodities helps to shorten SMS message hence allowing more information to be delivered to users in one SMS message.

With all these solutions to earlier challenges, ESOKO still has some shortcomings. As a system geared towards empowering farmers and traders to stimulate trade in agricultural commodities, it is important for the farmer and trader to be able to conduct trade with minimal impediments. Placing of offers to buy by a trader or offers to sell by a farmer should be easy to do. In ESOKO offers are placed via the web. As earlier discussed, majority of rural Kenyan farmers do not have access to internet. They are mostly rural dwellers where Internet is either unaffordable or inaccessible. In order to be effective and serve majority of farmers, placing of orders has to be done in a means that is readily accessible to many farmers and traders, for example mobile through SMS messages.

With the support of English, Portuguese, French and Arab, ESOKO has positioned itself to be of greater use by traders and farmers in most African countries, however majority of traders and farmers in rural Kenya speak and understand Kiswahili. They would be better served using a system that supports Kiswahili as a language. Use of codes to represent market names, commodities and traders means that users have to have knowledge of these codes. With over fifty codes standing for traders, commodities or markets, the task of remembering these codes is huge. This greatly reduces usability of the system in that it necessitates recall rather than recognition. People are better at recognizing things than at recalling them (Lidwell, Holden and Butler, 2003, p.200). A system which insists on use of codes require users to recall instead of recognizing.

2.4MFARM

"M-Farm. is a transparency tool for Kenyan farmers where they simply SMS the number **3535** to get information pertaining to the retail price of their products, buy their farm inputs directly from manufacturers at favourable prices, and find buyers for their produce" (M-Farm, 2012).

Developed in Kenya by M-Farm Ltd, M-Farm offers various services to farmers and traders including; group buying, group selling and price information. These services are helpful to the farmers and traders in that they can raise their volumes by putting their produce together and looking for a buyer.

This increases their bargaining power. Through group buying, rural farmers are able to requisition for inputs like fertilizer as a group. In this manner, they are able to bargain for better prices.

To find out price of maize in Nairobi for instance, a user would send a text message "price maize Nairobi" to short code number 3535. A user would then receive the price of Maize in Nairobi. The query sent is broken down into parts with what the user is asking, in this case a price, what commodity, Maize in this case and the location of interest, in this case Nairobi. M-Farm eliminates the use of codes for markets and commodities as required in E-Soko. This makes M-Farm easier to use since users can send queries without having to memorize codes. A downside of this system is its rigidity in query formats. Queries have to be sent in a specific format, any little alteration, addition or deviation will result in a query that cannot be processed. For example in the earlier query mentioned that is "price maize nairobi", if a user alters it even slightly, it will make it not to be processed since it will not be understood. For example "price of maize nairobi" , the introduction of the word "of" changes the query entirely. Queries which are not understood result in users losing money by paying premium rates with no valuable information received. This rigidity makes it imperative for users to have intense training on query syntax. This inherently affects the system's usability.

M-Farm queries are sent in English only. Users who are not conversant with English cannot use the system. Majority of rural farmers and some traders use Kiswahili more regularly than English, lack of support for Kiswahili shuts out most rural farmers from using this system. The new solution solves these issues by incorporating Kiswahili alongside English as the supported languages which users can use to send queries and receive results. The QUE helps prevent errors that could result from users not following standard syntax. For example, where a user has sent a message like "I want to buy milk in Lamu". This engine uses a set of rules and processing criteria that includes filters in order to understand what the user wants not only by picking words at specific indices of the message but by scanning the whole message and pick the various parts from the message. In this way, the system can respond to queries which have been sent by users who have had limited or no training on query syntax. The allows users to send queries as though they are posing a question to a fellow human. It also allows users who have attained expert levels in using the system to pose more well constructed queries using the syntax given which is [INTENT] [COMMODITY] [LOCATION] for example "buy millet chwele".

Connective words for example "of" when used in a query should not result in queries that cannot be processed. These connectives should aid in understanding of the query as opposed to being stumbling blocks to query understanding.

These challenges warrant a system which allows rural farmers and traders to share information regarding produc prices, their supply and demand. The system utilizes a QUE to allow users to send queries without necessarily having to remember codes for markets or commodities. A trader would issue a query like "bei ya maharagwe katika soko la msambweni?" , the QUE will understand this query and return the price of beans in Msambweni market. A more syntactically fault tolerant system helps users get results even when they have errors in their queries. Queries can be sent by users in both English and Swahili. A farmer in order to advertise his produce will send details in an SMS, for example "sell maize kimilili" . This SMS will be processed by the QUE, the variables will be saved in a database and made available for access by traders. Users do not have to access the system via web to be able to place offers but can use SMS only. This feature helps to make the solution accessible to majority of rural farmers and traders who own low end mobile phones. This system fits as a solution designed for the lower pyramid.

3. DESIGN OF STUDY

3.1 Introduction

In going about this research, a systematic approach was followed. The research was designed to achieve the objectives set. The research aims to develop a market information system for rural agricultural markets in Kenya. To achieve this goal, the major impediments to achieving a usable, affordable and accessible system were analyzed. Some of the factors identified include; low purchasing power among rural farmers, low literacy levels among rural farmers and traders, unavailability of internet connectivity in rural parts of Kenya or high cost of Internet in cases where Internet is available. To design and develop a system that can overcome these challenges, a framework was designed and developed. The framework contains four major parts each part addressing each of the challenges mentioned above. The four parts are; accessibility, language support, pricing model, and query syntax flexibility.

With the framework in place, the MIS was developed, taking into account the specific challenges faced by rural farmers and the framework's specification on overcoming these challenges. The system was developed following the Rapid Application Development (RAD) methodology where development and testing were done iteratively.

The developed system consisted of key functional modules which include Query Understanding Engine (QUE), SMS gateway, web interface, and database. These modules work together in harmony to handle incoming queries in natural language (English or Kiswahili) and respond after performing internal operations. These internal operations involve converting the natural language request received into Structured Query Language (SQL) queries. These are then applied against the database. The SQL queries can be insertion, updating, or selection of data.

Selection of participants was done having completed development of the prototype. The sample design involved selection of provinces from the eight Kenyan provinces, selection of markets in the selected provinces, and selection of farmers and traders in the selected markets. This sampling was done hierarchically where four out of eight provinces were sampled based on convenience. From the four provinces two rural markets were sampled randomly. Five farmers and five traders were systematically selected with every fourth farmer and every fourth trader being recruited.

The System Usability Scale (SUS) questionnaire was presented to the participants after a period of two weeks of using the system. The SUS consists of ten questions with five choices for each question.

Analysis of data collected using the SUS questionnaire was done following the standard specification for SUS. SUS being a measure for usability uses an average. Using the average calculation method, the responses were used to calculate the average to gauge the users' perception of the system and also to measure systems usability. Secondly, queries sent in and responses sent out were fetched from the database. These two sets of information were compared. This was used to calculated the systems accuracy in translating natural language queries into SQL queries. This was used to achieve the third objective of this research.

3.2Framework Formulation

In order to develop a system that addresses the challenges mentioned above, a framework was formulated. SAMIS ((SMS Based Agricultural Market Information System) framework was formulated. Pictorial representation of the framework is given below.

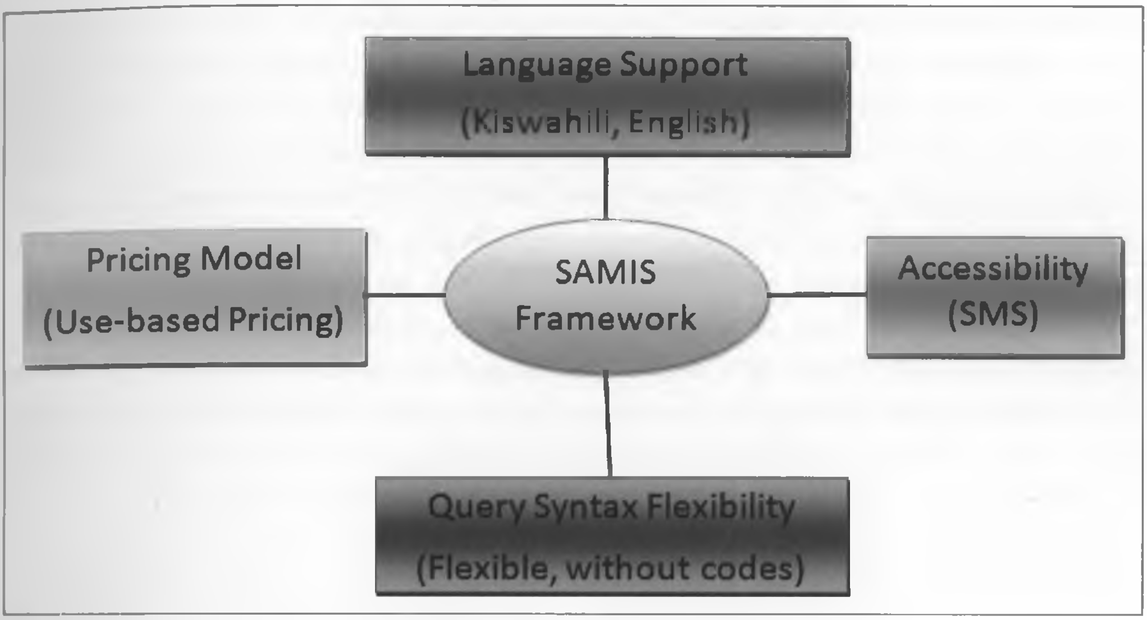


Figure 1 : Proposed Solution Framework

The major components of the framework are;

3.2.1 Accessibility

Accessibility entails the ability of rural population in Kenya to have reach of systems that avail market information to them. It involves availability or lack of media that deliver this data. With vast sections of rural Kenya lacking Internet connectivity, systems that rely on Internet like some of the systems reviewed above will not be ideal. Cost of computers is still inhibitive to many rural dwellers in Kenya. Emphasis should be put on selecting a more ubiquitous media which rural farmers and traders can leverage. There is no better media than mobile phones with more than 72% of Kenyans having access (CCK, 2012). There exist various types of mobile phones ranging from smart phones to low-end phones that have basic SMS and voice call functions. As the features increase, so does the price. Developing for the bottom of the pyramid dictates that a solution has to be affordable and smart phones do not fit in that category. Using media such as General Packet Radio Signal (GPRS) would disadvantage farmers and traders whose phones do not have that capability. They would not be able to use the system. SMS delivery of information to and from the users was chosen. This is due to support of SMS by most phones that are used in the rural areas. Even the most basic phones support SMS therefore choosing SMS will ensure that no rural farmer or trader do not access the system due to the information delivery media.

3.2.2 Language Support

Kenya's two main languages are English and Kiswahili. Majority of the rural population communicate effectively in Kiswahili. Jose (2008) identified language barrier as major obstacle to adoption of information technology. With literacy levels in the rural areas being higher than in urban areas (KIPPRA, 2001), some of the rural farmers are not able to communicate effectively in English. To bring these rural farmers and traders to the system's user population universe, support is added for Kiswahili in the system. Users are able therefore to send in queries in English as well as in Kiswahili. Additional support makes the task of implementing a system with dual language support a challenge. This challenge can be overcome through provision of logic in the database where the QUE can draw data and make sufficiently correct translations from natural language to SQL queries regardless of the natural language. A major challenge with provision of support for Kiswahili in the prototype implementation has to do with the fact that Kiswahili has many versions. Kenyans have continuously used various versions of Kiswahili depending on region. Kiswahili statement like "Mimi ninauza mahindi kerugoya" can be stated other forms for example "Mi nauza mahindi kerugoya". This makes it challenging to be able to recognize the various forms. With thorough training, the system is able to gain knowledge and identify what a user's intentions are. Kenyan's being non-native English speakers have tended to adopt and accept a way of writing and speaking that is not standard English. This coupled with low literacy levels in rural areas make it a challenge to deal with certain words. These words and statements are not always in standard English, for example "I sell cabbages very good". This statement would be understood by a human but for a system to understand it, additional training has to be done.

3.2.3 Syntax Flexibility

Nielsen (1993) in his ten usability heuristics mentioned error prevention as one important usability measure. With many existing systems in this domain depending on a strict SMS syntax, many errors are prone to happen. People are better at recognition than recalling (Nielsen, 1993). To aid recognition rather than recalling, this framework prioritizes a more flexible syntax where users can send in SMS using a preset syntax or using a natural question asking format. For example "Price beans chwele" or "What is the price of beans in chwele?". Both these statements should be pass without errors and the same response be sent to both senders. The only difference is that the first statement which follows a strict syntax gets executed more quickly than the second statement. With a system that follows only a strict syntax, the second statement would have resulted in an error. Syntax flexibility or lack of it is a more important factor in attaining high usability. Some systems like MFarm would not resolve the following statement "buy maize in nakuru". The third word "in" makes this statement unrecognizable by MFarm. With such a strict syntax, accuracy of system in understanding user queries drops significantly. Achieving of syntax flexibility in the developed system was not easy to come by. It involved training and implementation of a module that picks out words that were not helpful in formulating a response to a user sent query.

3.2.4 Pricing Model

In order to ensure that an SMS MIS is developed and is able to help many rural farmers and traders, pricing has to be favorable. Making a system affordable to users and also sustainable to the owners is a delicate balancing act that has to be done. Charging exorbitant rates will prevent users from affording to use it. On the other hand, making a system completely free would make it go offline quickly, this is because effort that goes towards support and maintenance need to be remunerated. Some existing systems have remained largely dormant for a variety of reasons. One of the reasons has been the pricing model adopted. Systems that charge a flat fee for a specific period of time may disadvantage the user. This can happen where a farmer pays for the system but due to their crops getting wiped away by floods or other reason. Before the farmer can grow other crops to sell, the previous payment will have expired. If use for the system had been done on use basis so that users are charged every time they use the system, then they would not incur losses in case such calamities happened. The SAMIS framework has put emphasis on pricing based on system use. A fee can be added on top of the subscriber rate charged by the operator. With local operators charging for SMS messages at below KSh 5.00 per message, a profitable system can be run by adding a small fee that is sustainable.

3.3 Bottom of Pyramid Design Considerations

Rural farmers and traders are in the lower part of the pyramid. In this section of the pyramid, there are unique challenges that need to be overcome for any solution targeted at users in this pyramid part to benefit from the system. Design issues targeting these kind of users have been factored in developing the framework used in this system. See Figure 1. The challenges faced by these users include low literacy level, low purchasing power, and limited internet connection. Adoption of both English and Kiswahili goes a long way in overcoming the problem associated with low literacy levels. Users who do not understand English can use the system in Kiswahili. Use of SMS instead of web in sending and receiving queries mitigates against the limited internet connection and low purchasing power problems. GSM network coverage is wider than Internet connectivity in Kenya, more so in rural areas. Areas where internet connection is available cannot rely on it due to high cost of accessing it. SMS messages are charged lower voice calls. At the time of this writing, cost of voice call per minute is KSh 4.00 per minute whereas cost of text message is KSh 1.00 (Safaricom, 2012). Internet connectivity is even more expensive since it could require other specialized equipment like broadband modems. These equipment cost more money.

3.4 Prototype Development Methodology

The SAMIS framework developed and presented above was used in as a guideline in development of the prototype. RAD methodology was used in development of the prototype. Choosing RAD as opposed to other methodologies is informed by the fact that it is fast and less error prone. Its stability is also an important factor. With limited time for planning and system development, RAD was seen as more suitable since it allowed for more time to be spent in development

	Stage
1	**Requirements Planning** Needs of the final system were assessed. Each of the functionalities which users need to have were considered. The project scope, constraints and requirements were determined during this phase. An SMS based market information system can be of varying scope with varying number and functional modules. In this phase, scope will be determined so that the project can be completed within the allowed time period.
2	**Functional Design** Systems data and processes were modeled in order to build a working prototype of system components that were deemed critical. Important aspects of the system like **usability** and **user satisfaction** were factored in this stage. Design of a system that users are able to use in the language they understand was found to be critical. With the system being SMS based, importance was put in design of a system that is error tolerant. **QUE** was designed to accept a flexible syntax of SMS message. **SMS gateway** was designed to be able to use a mobile phone as a GSM modem. Choice of mobile phone as opposed to industry standard GSM modem was due to mobile phone being more affordable. Database was designed with tables in it. The database was included to provide persistence to learnt knowledge. Everything new that the system learns during its lifetime will be saved in the database. All incoming and outgoing messages were and will be saved in the database.
3	**Prototype construction** **Database design**. Database and tables were designed to support saving of messages and responses. Emphasis was on creation of database design that is scalable to allow new entries to be saved as the system learns new names and words. The database acts as the system's knowledge store. The system gains in intelligence with age.

	Web application development. Writing code and deploying web application to the application server. Emphasis is on the connectivity of web application to database. **On the QUE.** emphasis is on handling of SMS message strings and translation to SQL queries. Messages that are not understood are flagged as un-understood and saved in the un_understood table of the database. **SMS gateway**. The gateway was developed to handle two way (incoming and outgoing) messages. The gateway is based on Haye's AT commands.
4	**Prototype utilization** The developed prototype was tested with test data. Development and testing were done iteratively. Each of the modules were tested while being developed. Training data were presented to the system to make it better at understanding queries sent by users Sample SMS messages were sent to the phone connected to computer where QUE is installed. With the system having attained more than 95% accuracy in query understanding, the system was made available to users to use. Users were trained before using the system.

Table 1 : Prototype Development Methodology

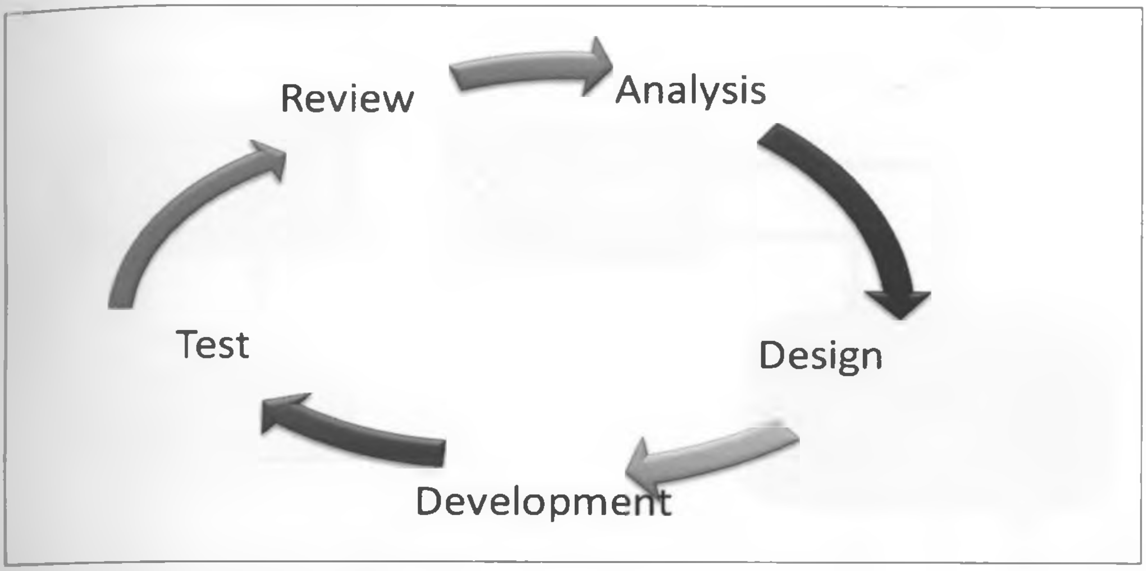


Figure 2 : Rapid Application Development

The process is iterative with analysis design, development, testing and review all happening in succession as shown in Figure 2 above. This iterative process lead to requirements getting continuously being updated. While developing the QUE, a realization was made as to the importance of an unwanted words table in the database. This table was therefore developed to store words that had been found in SMS messages. These words were found to not to be useful in understanding of queries sent by users. They are the words that are saved in the unwanted words table. This led to requirements getting updated. The database design was also updated in the process. New requirements meant more effort put in development. New module or sub module developed leads to testing then review of entire system, and the process starts again.

3.5 Prototype Model

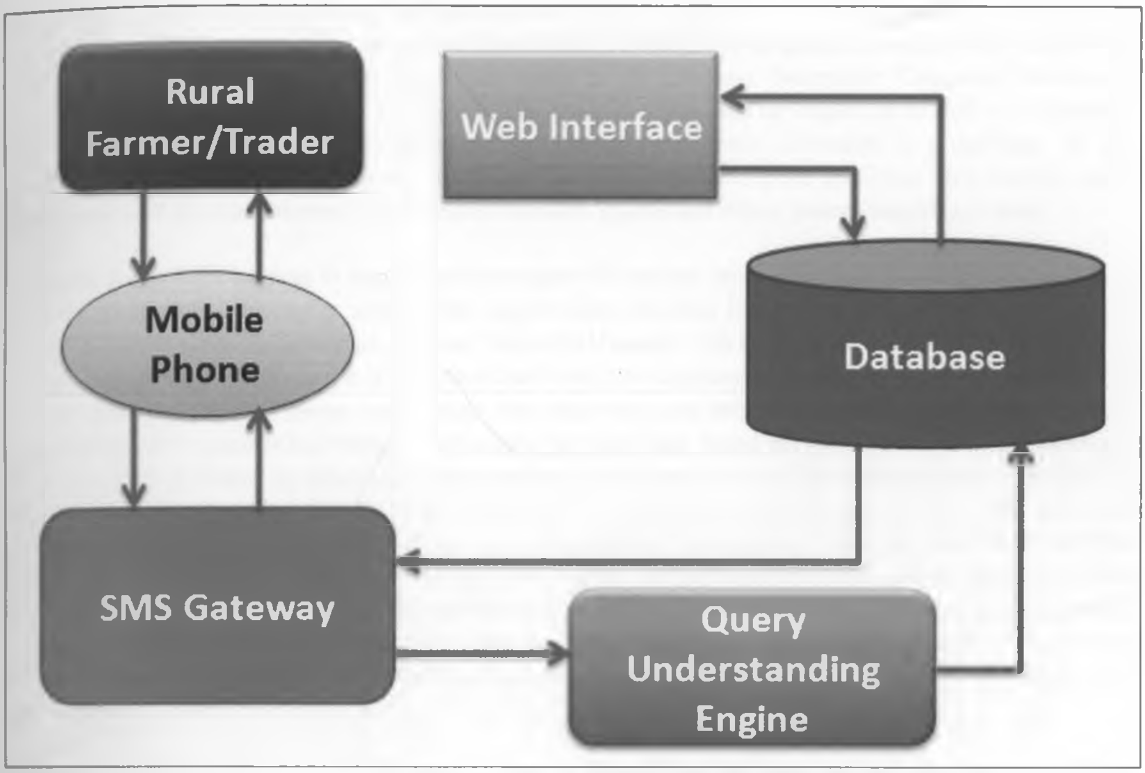


Figure 3 : Prototype Modular Structure

3.5.1 Rural Farmer or Trader

Interaction with system begins when a user (rural farmer or trader) begins by sending a query for price of commodity or an offer to sell or buy a commodity as shown in Figure 3. This happens when a user sends an SMS message to the mobile number selected of the phone used as a GSM modem. The rural farmer or trader is the initiating actor of events that will involve all the modules except the web interface whose actor is the system administrator. Once the SMS is received by the mobile phone, it remains unread till the next time the **SMS Reader** utility that is part of the SMS gateway polls the phone for new unread messages.

3.5.2 SMS Gateway

SMS Reader utility continuously polls the connected mobile phone. When a new unread message is found, it runs a set of Hayes AT (Attention command) commands. These commands read the message and mark it read on the mobile phone. The read messages are saved in the incoming messages table of the database. To distinguish the new messages from other existing SMS messages in the table, a flag (**is_processed**) is set to 'N' which stands for No. This will ensure that next time the incoming messages are processed, only the new (those whose **is_processed** flag is marked **'N'**) messages are processed. The SMS gateway then hands over control to the QUE. These modules work in a conveyor belt manner in that when one finishes working on a dataset, it makes it ready for the next module.

3.5.3 Query Understanding Engine

At the QUE, the message is analyzed to figure out the intention of the user. In order to make understanding of this query possible, the language used will be determined. Currently two languages are supported, English and Kiswahili. This is done by the QUE sending the query to the Language Determiner (Language Determiner) which determines the language used in the query. The language used could be English, Kiswahili or a mixture of both. As noted by Muchemi (2008) distinguishing questions from mere statements is a challenge. In this prototype. a set of rules were used to distinguish between offers and price inquiries. This enables easier understanding of the user intention. To distinguish between queries and offers, pattern matching is done.

Analyzing a new SMS happens in steps. Some messages will undergo more steps than others depending on how much the message has adhered to syntax. First step involves checking if the message has three words and that each of the three words are in correct position. This is for example "bei karati msambweni". This message will only go through the first step and it is understood and ready to be processed by the next module. Messages that have not followed the strict syntax will emerge from step one as not being understood. The next step involves using the system's acquired knowledge of all words that have been found not to add value to understanding a query. The SMS is filtered by removing all the unwanted words from it if any. Some messages will emerge with only the three parts required [INTENT] [COMMODITY] [MARKET]. Such messages would then be passed through the process in step one where all the pars are extracted. The extracted parts are saved in offers table. Some messages will still emerge they way they were before the filtration and some will emerge leaner than before but still not in a format that can be understood. Such messages will discard the filtration and adopt the full unfiltered form. Then each of the words from **index 0** to index **word.lenth – 1** are analyzed. Each of these words are matched against classified words. The classified words are in three groups namely; intent, commodity, and market.

Words in the SMS that meet any of these groups are picked out and once all the three parts [INTENT] [COMMODITY] [MARKET] are found and the message has been fully searched, the parts are saved in an **offers** table of the database. Messages which come through this step without the three parts are saved in the **un-understood** table of the database. These are messages which the system administrator will apply a human touch to resolve. The understood messages are all saved in the offers table of the database and await **Offers** utility which processes all offers that were received in the space of the past seven days. Inquiries are responded to once whereas other offers (sell and buy) can get multiple responses. This depends on how many matching offers are received in the space of the next seven days since the message was received.

3.5.4 Web Interface

In situations where the QUE encounters difficulty in understanding the query, the user is sent a message to clarify their intentions. The SMS message will be saved in the **un-understood** messages table of the database. Other understood queries are saved in the **offers** table as offers and await to be processed and matched with other offers which are less than seven days old. The web interface will be used by system administrator to check transactions taking place. The system administrator will also use their human understanding to figure out messages which are saved in un-understood table. Once system administrator understands these previously un-understood messages, the system admin corrects the message and it become a new message in the incoming messages table. In the un-understood messages table, **is_resolved** flag is set to **'Y'** for Yes and the un-understood table is updated with the new message saved in the **message_resolved _to** column. The system administrator will periodically check the system to see if there are any un-understood messages that need his attention.

3.6 Research Methodology

3.6.1 Sampling Design

A total of 75 participants were randomly selected in two steps, rural farmers and traders. 10 rural farmers and 5 traders were selected randomly from 5 markets. The 5 markets were purposively sampled for ease of logistics and to fit into budget. The markets are distributed geographically across five different counties and cover a Kenyan community in the Rift Valley. The geographical distribution ensures that there is a variety of respondents.

Table 2 below shows the distribution of markets in the four provinces. Number ranges in brackets represent the participants drawn from the market.

Market	Rural Farmers	Traders
Chebara	Farmer 1 – Farmer 10	Trader 1 – Trader 5
Soy	Farmer 11 – Farmer 20	Trader 6 – Trader 10
Nyaru	Farmer 21 – Farmer 30	Trader 11 – Trader 15
Tenges	Farmer 31 – Farmer 40	Trader 16 – Trader 20
Makutano	Farmer 41 – Farmer 50	Trader 21 – Trader 25

Table 2 : Participants Sampling

3.6.2 Data Collection

Participants selected during sampling were given questionnaires to fill in order to gauge their satisfaction and usability of the system. The questions that respondents were asked were based on SUS (System Usability Scale) developed by John Brooke in 1986. The data collection instrument involves a set of ten questions with five possible responses ranging from strongly agree to strongly disagree. The questionnaire is shown in Table 3;

		Strongly Disagree				Strongly Agree
		1	2	3	4	5
1	I think that I would like to use this system frequently	●	○	○	○	○
2	I found the system unnecessarily complex	○	○	○	○	○
3	I thought the system was easy to use	○	○	○	○	○
4	I think that I would need the support of a technical person to be able to use this system	○	○	○	○	○
5	I found the various functions in this system were well integrated	○	○	○	○	○
6	I thought there was too much inconsistency in this system	○	○	○	○	○
7	I would imagine that most people would learn to use this system very quickly	○	○	○	○	○
8	I found the system very cumbersome to use	○	○	○	○	○
9	I felt very confident using the system	○	○	○	○	○
10	I needed to learn a lot of things before I could get going with this system	○	○	○	○	○

Table 3 : System Usability Scale

SUS was chosen as opposed to many other existing usability testing measures because it is technology independent. With this research focusing on various technologies from web to SMS to query understanding to SQL, SUS was found to be better suited. More so, SUS is reliable, valid and is sensitive in that it distinguishes between good and worse systems. Users can provide feedback that touch on both perception and user experience.

Recording was also employed as one other data collection method. Queries sent in and responses sent out were recorded in the database. The data recorded will be used to determine accuracy of the system in understanding and responding to queries.

After participants were selected, the purpose of the study was explained to them. There were no incentives given except airtime of KSh 50 for each of the participants. The airtime was given to allow the participants to send in SMS messages to the system. The participants were trained on how to interact with the system.

Training of participants was done in two days, this is because the markets are widely distributed geographically. After the two days, participants were given access to the system for 7 days. During this period, participants interacted with the system by sending messages and receiving responses. All these messages were recorded in a database. Participants were offered support by the members of the research team during this 7 day period. In the research team, there was a research assistant in addition to the lead researcher.

After the 7 days of using the system and recording of messages sent by users to the system and messages sent by the system to users, participants were given the questionnaire to fill.

3.6.3 Data Collection Instrument

Questionnaires based on the SUS questionnaire were used to collect response from participants. The SUS questionnaire was posed to the participants after a period of using the system. Preference of questionnaires is due to their objective nature. They allow information to be collected in a more standardized manner. In addition, the questions on the questionnaire are standard and have standard set of responses, this makes questionnaires the better data collection method compared to other methods for this research.

Recording using a database of messages sent to the system by users and to users by the system . Choice of database recording was informed by its relative ease in carrying out. Once it has been automated, no further work will be required until when data is ready to be extracted.

3.6.4 Data Analysis

Attainment of the research objectives in this research was measured by analyzing data collected using the questionnaires. Also queries sent by users and responses sent to users by the system were extracted from the database having been earlier recorded. Data collected will be analyzed using the standard SUS formula . Responses that were used to fill the questionnaire were analyzed based on the SUS scale for usability. Analyzing for user satisfaction gave an insight into whether or not the system is successful in bridging the intended gap that this research is about, that is, providing a means for traders and rural farmers to trade in their goods effectively by availing an information exchange platform. Usability was used to test the ease with which the users, in this case traders and rural farmers can trade in their produce using this system. Use of the SUS for analyzing usability encompasses all aspects of a system that determine usability, these include; effectiveness, efficiency, user satisfaction and user perception.

Using the SUS method, responses of various respondents were recorded. For positive worded questions (1,3,5,7, and 9) which are odd numbered, one is subtracted from the response. For negatively worded questions (2,4,6,8, and 10) which are even numbered, the response is subtracted from five. The results of both sets of responses are added up. This makes the scale to be 0, to 4 instead of the original 1, to 5. Next, the summation is multiplied by 2.5. This raises the possible values from 0, to 40 to a new scale of 0, to 100.

SUS method for one respondent

$$2.5\sum((x^{+ve} - 1) + (5 - x^{-ve}))$$

This calculates usability of an individual respondent based on their feedback. Since usability using SUS is suppose to be one value 0n a scale of 0 to 100, an average of all the individual respondents SUS value is calculated. This is found as follows

SUS method for 75 respondents

$$\frac{\sum_{i=1}^{75} 2.5\sum((x^{+ve}-1)+(5-x^{-ve}))}{75}$$

The average acceptable usability is SUS score of **68%** (Sauro, 2011), values over this figure are considered above average, while those less than 68% are considered below average.

In this research, 75 respondents were involved and for each of them there are 10 responses according to the SUS questionnaire. This gives total of 750 responses. Table 4 below shows the frequency distribution of usability scores based on SUS calculation. Detailed SUS results can be found in Appendix C Figure 6. The average SUS was found to be **93.6.**

The arrival at this average SUS is detailed in Table below;

User responses per question.

	Question Response Distribution				
	76 - 80	81- 85	86 - 90	91 - 95	96 - 100
Frequency	3	3	12	29	28

Table 4 : SUS for each participating user

Analyzing for accuracy of system in understanding queries was done by comparing responses of the system to expected responses. Queries sent by users were recorded in the database. For each of the queries, correct responses were determined. These were compared to the system responses. Accuracy of the system in understanding and responding to queries depends on how accurate the QUE translates queries to offers. For a total of 200 messages, the system accurately responded to 190 messages. The other 10 messages were not accurately understood.

$$\frac{190}{200}\, X\, 100\% = 95\%$$

3.7 Results

The results of this research were drawn on how the research objectives were achieved. The results were derived from evaluation of system in functioning and performance towards meeting the set objectives..The objectives of the research were;

i. To develop a QUE that processes queries received in Kiswahili or English.

ii. To develop an SMS gateway for receiving queries from farmers and traders and sending back requested information to farmers and traders.

iii. To test the Agricultural Market Information System for accuracy in processing and understanding queries sent in Kiswahili or English and for usability

There are two ways in which any computer system can fail. First is the functional failure, this involves system being unable to generate correct output for a give input. For example, when a user inquires for a product price it receives the price of a different commodity. The second way in which a user system can fail is in performance. This is where unlike in the functional failure, the system operates correctly but does not do so in a timely and efficient manner.

In evaluating this system, the two approaches; functional and performance evaluation were employed.

3.7.1 Functional Evaluation

Evaluating a system's functionality involves analyzing functional aspects of the system. The system's functionalities include;

i. Send offers to sell or buy

To evaluate system's proper functioning in handling offers to sell or buy. A set of twenty offers , ten offers being buy while other ten being sell. These offers were sent to the system. The offers database table was inspected to check whether new offers were created. Also the matched_offers table was checked. The results showed that all the twenty well formed offers were received and processed correctly. There were no offers that were processed wrongly.

ii. Inquire product prices

Evaluating price inquiry involved sending messages that have an intention of inquiring for prices of different commodities. The prices were first recorded form the database and with prior knowledge of the prices, messages were sent to query the system for these known prices. The results were impressive with all the ten inquiries yielding correct results.

iii. Edit un-understood messages

Message that are not classified to one of {offer to sell, offer to buy, price inquiry} are classified as un-understood and saved in un_understood messages. Editing involves updating the record with a resolution message. A set of ten messages were edited and saved back to the database. All the messages edited were properly saved in the form they were edited.

3.7.2 Performance Evaluation

Being an SMS application, this system's performance depends to some extent on the speed of the mobile operator in delivering SMS messages. On average, with a set of thirty messages, the system took 15 seconds to respond to messages sent to it. Offers and inquiries sent to the system were responded to oin average after 15 seconds. Performance was better with messages sent during the night, after ten past midnight and before seven in the morning. The performance during this period was at 10 seconds. This shows that the system's performance is hampered by message delivery delay by the mobile provider. Performance of editing un-understood messages was better than offers and inquiries. This could be explained by the fact that the web server running the web interface is on a local computer where the system runs. To edit and save an un-understood message, there is a time lapse of three to four seconds depending on the computer's task load.

The system performed reasonably well on both the functional and performance evaluations as detailed above. With further optimization, the system could perform even better . As more data is put to the system, performance could become slower over time.

4. System Requirements Specification

4.1 Product Functions

The four product modules each have one or more functions to perform. These functions are

- Receiving SMS sent by user. This function is performed by the SMS Gateway
- Saving SMS received in the **incoming_messages** table of the database.
- Understanding SMS received in natural language (English or Kiswahili) and saving the message components in the **offers** table of the database. This is performed by the QUE.
- Saving unwanted words in the **unwanted_words** table of the database. This is the learning module of the database.
- Responding to user queries. Information requested by users is prepared by the **response** utility of QUE.
- Sending SMS to users who requested information from the system.
- Interfacing with user to provide a means to manually provide understanding of previously un-understood messages.

4.2 User Classes and Characteristics

There are three classes of users for this system. These are; rural farmers, traders, and system administrators.

The most important users for this system are the rural farmers and traders. They are the users who are the primary system beneficiaries. System administrators are supporting actors whose duty is to make the other user's use of the system worthwhile.

Users characteristics;

i. Rural farmers
   - They seek information on product prices from the system.
   - Their literacy levels is not expected to be very high.
   - They place orders to sell produce.

- Not all of these users can communicate effectively in English, Kiswahili support in the system should bring many to the fold.

ii. Traders

- They are the seekers of information possessed by farmers.
- They seek to buy commodities from farmers.
- They place orders to buy produce.
- They also seek product prices from the system.

iii. System administrators

- They use the web interface to correct un-understood messages sent by users.
- They perform the role of monitors.
- Over time with user increased knowledge of the system and system intelligence improved, their interaction with the system is bound to be less frequent.
- They do not place orders to sell, buy or inquire product prices.
- They are more knowledgeable on the system compared to all other users of the system

4.3 Operating Environment

Hardware Platform.

The system has been tested on Intel processor with two cores. The system performs well on 512MB memory and more than 500MB of hard disk space, this allows the database to grow.

Mobile phone that supports Hayes AT commands (Attention).

Operating System.

The system was implemented in Java and therefore can run on all environments; Windows, Linux and Mac. The system has however been extensively tested on Windows Vista.

Software Environment.

There are some important software utilities and drivers that the system requires, these are;

- MySQL database version 5.1 or higher.
- Java runtime environment version 1.7.
- Java communication API (Application Programming Interface) version 3.0.
- Java JDBC (Java Database Connectivity) driver API version 3.0.

4.4 Design and Implementation Constraints

In the development of this system, all modules have been implemented in Java programming language and database operations are done in SQL (Structured Query Language). A major constraint will be time. There is a specific time period within which the system has to be completed according to the University of Nairobi masters project schedule. All the modules and functionalities described in this document have to be implemented within this period.

4.5 User Documentation

User guide document will be part of the deliverables of this system. There will be additional help available by calling or emailing the developer to provide help that cannot be found in the documentation or in case there are comments regarding further improvement of the system.

4.6 Assumptions and Dependencies

There exist various assumed issue in this system. These assumptions are bound to affect the overall system if they happen to change or are found not to be as first assumed. These assumptions are;

> Samsung S5620 mobile phone supports all the AT commands that will be used in implementing the SMS gateway. This is the mobile phone that will be used as a GSM modem. The system uses Hayes AT commands. These commands are used in the SMS gateway. They are used to interface between the mobile phone and the system. The mobile phone is connected to the computer using a USB (Universal Serial Bus) cable. The computer which this system will be deployed has to have COM ports connected to it.

4.7 External Interface Requirements

4.7.1 User Interfaces

There are two interfaces that uses come into contact with the system. Traders and rural farmers interface with the system using their mobile phones. They do so by sending SMS messages and also receiving feedback through SMS messages from the system. System administrators interact with the system through a web interface. Mobile phone interfaces are varied. Various manufacturers have different interfaces. These interfaces vary also on the features that a phone has with more advanced phones boasting touch screen interfaces. Low end phones largely depend on keypads that users use to type in text messages. A mobile phone interface is shown below.

Figure 4 : Mobile Phone Interface

To send a message, a user has to type in the message and then type in mobile number of the receiver's phone or fetch the number from phone book if it was already saved. Next, the user will choose send from among the available menus. A sample interface of a mobile phone is shown in Figure 4 above.

To receive a message, the user does not have to do anything. They only have to keep the phone powered on then when the message is received, the user needs to read it.

The web interface is composed of the browser window as the top most in a hierarchy. The system loads in the browser and the first interface is the login screen. This screen allows the user to gain access to the system by supplying a username and a password. Figure 5 is the login section of the web interface

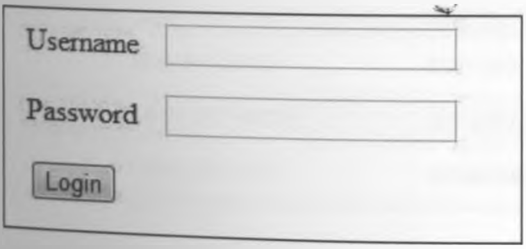
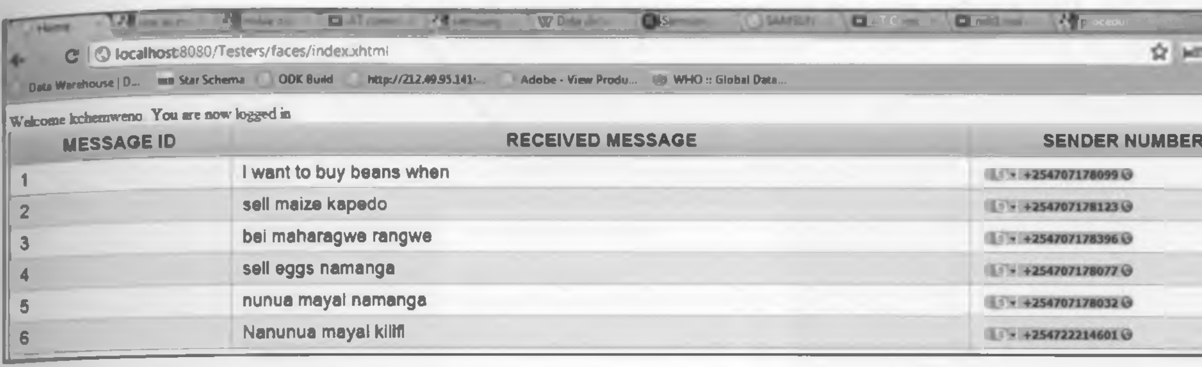


Figure 5 : Web Interface Login

After the user supplies a password and username, the system will grant access to a users if the supplied credentials are valid. In this scenario, the default view messages page is loaded, Figure 6.

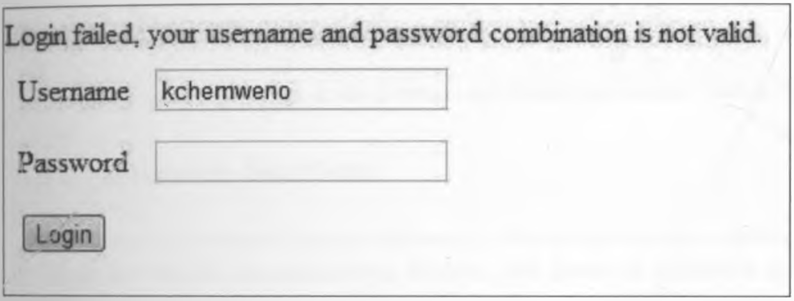


Welcome kchemweno You are now logged in

MESSAGE ID	RECEIVED MESSAGE	SENDER NUMBER
1	I want to buy beans when	+254707178099
2	sell maize kapedo	+254707178123
3	bei maharagwe rangwe	+254707178396
4	sell eggs namanga	+254707178077
5	nunua mayai namanga	+254707178032
6	Nanunua mayai kilifi	+254722214601

Figure 6 : View Incoming Messages

In case the user supplies invalid credentials, the system responds with a message as shown in Figure 7 below.



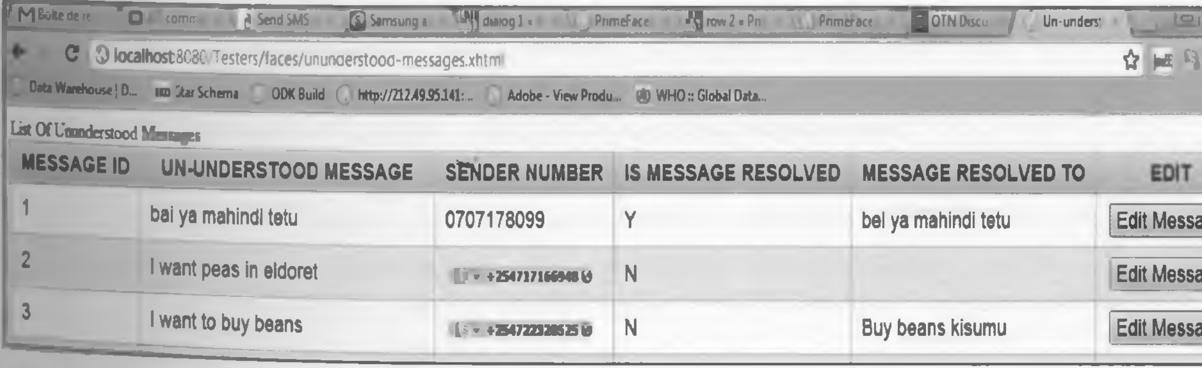
Login failed, your username and password combination is not valid.

Username kchemweno

Password

Figure 7 : Login failed window

From the home page, the system administrator will be able to make changes to un-understood messages. To do this, the system administrator will chose Un Understood messages from the menu. This will lead to the un-understood messages to be loaded on a table as shown. See Figure 8.

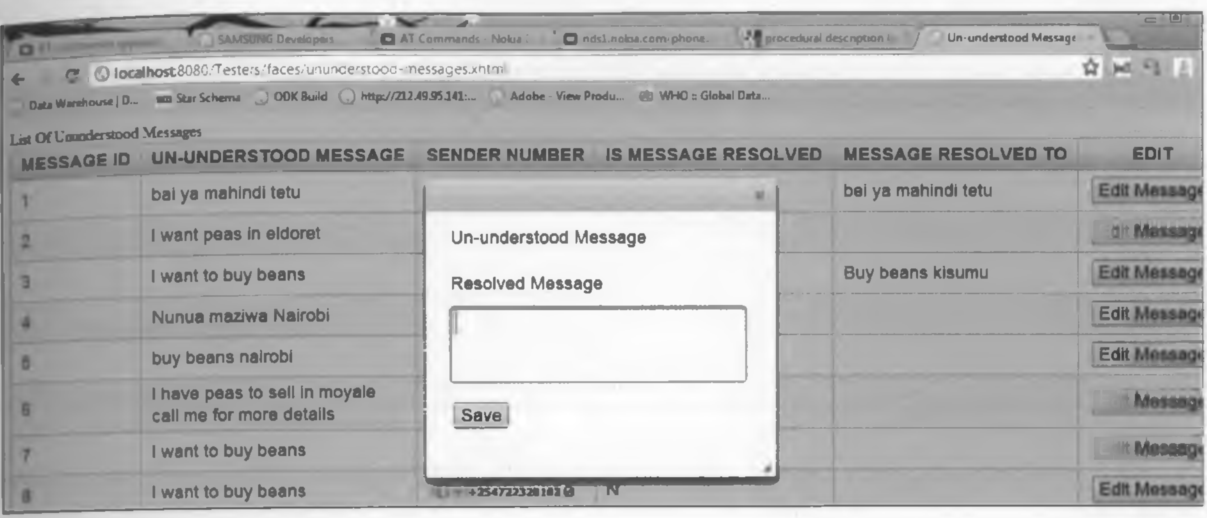


List Of Unnderstood Messages

MESSAGE ID	UN-UNDERSTOOD MESSAGE	SENDER NUMBER	IS MESSAGE RESOLVED	MESSAGE RESOLVED TO	EDIT
1	bai ya mahindi tetu	0707178099	Y	bei ya mahindi tetu	Edit Messa
2	I want peas in eldoret	+254717166948	N		Edit Messa
3	I want to buy beans	+254722328525	N	Buy beans kisumu	Edit Messa

Figure 8 : View Un-understood messages

The system administrator can make changes to each of the messages by clicking on the edit message button on the message row. This will launch a dialog from where the administrator can then edit the messages and save it back to the database, see Figure 9.



List Of Uunderstood Messages

MESSAGE ID	UN-UNDERSTOOD MESSAGE	SENDER NUMBER	IS MESSAGE RESOLVED	MESSAGE RESOLVED TO	EDIT
1	bai ya mahindi tetu			bei ya mahindi tetu	Edit Messag
2	I want peas in eldoret				dit Messag
3	I want to buy beans			Buy beans kisumu	Edit Messag
4	Nunua maziwa Nairobi				Edit Messag
5	buy beans nairobi				Edit Messag
6	I have peas to sell in moyale call me for more details				Messag
7	I want to buy beans				it Messag
8	I want to buy beans	+254722320182	N		Edit Messag

Un-understood Message

Resolved Message

Save

Figure 9 : Editing un-understood message

The user commits the change to the database by clicking on the save button on the dialog.

4.7.2 Hardware Interfaces

The hardware components that are involved in this system include a mobile phone that acts as a GSM (Global Systems for Mobile communication) modem. This phone is connected to the computer where the system is installed. The phone interacts with the SMS gateway to extract SMS messages from the phone and also to write SMS messages in the computer's serial port for onward transmission.

The phone is connected with the mobile phone using a USB cable.

4.7.3 Software Interfaces

Various software interfaces exist that interface with the system. These interfaces include;

i. MySQL database version 5.1. MySQL database will be used to store persistent data that will be required in future sessions. This data include; commodity names, market names, incoming messages, outgoing messages, un-understood messages and wanted and unwanted words.

ii. Java runtime environment version 1.7. This software interface will provide the base on which the system will execute against. Since the system is developed in Java, JRE (Java Runtime Environment) will provide the necessary virtual machine for executing the systems instructions.

iii. JDBC API(Application Programming Interface) version 3.0. To provide interaction between the system and MySQL database, a JDBC driver will be used.

iv. Java Communication API version 3.0. This application programming interface provides means for the system and the computer to communicate. The system can access the computer's serial port by calling on methods provided by the Java communication API.

4.7.4 Communications Interfaces

SMS messages are heavily employed in this system. Text messages are sent by users to the system when querying commodity prices and also the system sends out SMS messages when responding to user requests. In this respect, the users have to have access to GSM network connectivity that is sufficient to send a text message. Hyper Text Transfer Protocol (HTTP) is also used by the system in presenting the web interface, the web server loads the web interface on a browser for the system administrator to interact with the system. Login passwords will be saved in the database after encryption.

4.8 System Features

Major system features include;

SMS gateway, QUE, database and web interface. Each of these features perform at least one function in the system domain.

4.8.1 SMS Gateway

4.8.1.1 Description and Priority

A very important feature that is of high priority to the system's completion. This feature is one of the more expensive features, this is because it relies on a mobile phone unlike other features. The mobile phone will provide the functions of a GSM gateway. This feature uses AT commands to exchange messages back and forth.

4.8.1.2 Stimulus/Response Sequences

A user can initiate the action associated with this feature. This is where a user sends a message to the system. Alternatively, the system can also initiate the action. This occurs when the system responds to a user query by sending an SMS message.

4.8.1.3 Functional Requirements

REQUIREMENT-1: With SMS messages being saved in the database, the database connection has to be already initiated.

REQUIREMENT-2: The gateway is intended to be continuously executing, this therefore demands that a scheduling program be in place in the computer where the system feature is executed.

4.8.2 Query Understanding Engine

4.8.2.1 Description and Priority

The query understanding engine feature of the system is another high priority feature. This module provides the functions of creating meaning from received SMS messages. It helps to also provide intelligence to the system. It is in this module that the system gains in knowledge with age and experience.

4.8.2.2 Stimulus/Response Sequences

QUE receives its input from the database. Completion of execution of the SMS gateway on a round of execution will trigger the QUE to execute. It will fetch and process all unprocessed messages in the **incoming_messages** table of the database. The QUE will upon processing of an incoming message save a record in the **offers** table of the database.

4.8.2.3 Functional Requirements

REQUIREMENT-1: The QUE relies on the database being connected to the system and an active connection being available. The database is where input data and output data is fetched and saved to.

4.8.3 Database

4.8.3.1 Description and Priority

Persistent data is stored in the database. All SMS messages are saved in the database. The database also functions as the systems knowledge store where all unwanted messages are stored each time they are found in an SMS message. This feature is of high priority being the data store around which most systems input and output are stored.

4.8.3.2 Stimulus/Response Sequences

Data saved in the database are derived from the other features, GSM modem and QUE. These modules trigger the database to provide, store or update the data new or existing.

4.8.3.3 Functional Requirements

REQUIREMENT-1: The system has to have the capability to run MySQL database server.

4.8.4 Web Interface

4.8.4.1 Description and Priority

System administrators interact with the system using web interface. The web interface helps the system administrator to monitor the system's functioning including editing un understood messages.

This feature is of medium priority, the major users of the system are able to continue using the system even if this feature was not available.

4.8.4.2 Stimulus/Response Sequences

The system administrator is the initiator of actions in the web interface. The system administrator accesses the system using a web browser. The administrator issues commands using the menus available and the system responds with the requested data.

4.8.4.3 Functional Requirements

REQUIREMENT-1: The web interface depends on Glassfish server version 3.1. The web server has to be running to allow the web interface to be served to the requesting system administrator.

REQUIREMENT-2: Database server has to be running and a connection should be able to be initiated.

3.1.1 Other Non-functional Requirements

3.1.1.1 Performance Requirements

Speed with which users can receive responses from the system depends on the GSM carrier. Some carrier's have better networks that are efficient in transmitting SMS messages.

3.1.1.2 Safety Requirements

User's safety could be hampered if user's mobile numbers are fetched from the system and misused. People who do not mean well are capable of using user's private phone numbers to threaten and even extort money from the owners of those phones. Proper security in the form of requiring usernames and passwords to access the web interface. The passwords have to be encrypted before being saved in the database.

3.1.1.3 Security Requirements

With users sending in messages to the system using their private mobile phones, access to the systems database has to be restricted. Without doing this, these mobile numbers may be used for other purposes which may put safety, security and privacy of the users in the line. To guard against this misuse of information, access to the system has to be restricted and controlled by requiring a username and password to access the system's web interface.

3.1.1.4 Software Quality Attributes

There exist quality attributes for this system. These are features that make the system more useful to the users.

These attributes are;

i. Usability
The system's usability is a key factor of the system. Developing for the bottom of the pyramid entails taking into account the needs of the users in that bottom section of the pyramid.

ii. Robustness
Robustness will be attained by ensuring that the system is able to grow and still maintain its high performance levels even with growth in data.

iii. Adaptability
Learning and improvement of the system with age and experience ensures that the system is able to adapt to new messages coming in. The system should be able to continue performing well even with different messages being received.

iv. Portability
Running of the system on multiple platforms is important. Emphasis should be put on ensuring that all development of the modules is done with portability in mind.

3.1.1.5 Business Rules

System administrators are the only users who interact with the web interface. Their [illegible] intervention in understanding messages which were not understood by the system.

Rural farmers send in messages to inquire for prices or to place orders to sell.

Traders send messages to inquire product prices and to place offers to buy.

Traders and farmers can send messages to place any kind of offers, buying or selling.

3.1.2 Other Requirements

Key requirement that has not been covered anywhere in this document this far include [illegible] in the system. Users should be able to send in messages in English or Kiswahili and the sy[illegible] respond with a great degree of accuracy. The QUE using a language component should [illegible] function. This component should be easy to extend in adding more languages as and when [illegible]

Acute Food Insecurity Reference Table for Area Classification

Phase Name and Description		Phase 1 Minimal	Phase 2 Stressed	Phase 3 Crisis	Phase 4 Emergency	Phase 5 Famine *(evidence for all three criteria of food consumption, wasting, and CDR is required to classify Famine)*
Area Outcomes	Food Consumption & Livelihood Change	More than 80% of households in the area are comfortably able to meet basic food needs without atypical coping strategies & livelihoods are stable	Based on the IPC Household Group Reference Table, at least 20% of the households in the area are in Phase 2, 3, 4, or 5	Based on the IPC Household Group Reference Table, at least 20% of the households in the area are in Phase 3, 4, or 5	Based on the IPC Household Group Reference Table, at least 20% of the households in the area are in Phase 4 or 5	Based on the IPC Household Group Reference Table at least 20% of the households in the area are in Phase 5
	Nutritional Status	**Wasting Prevalence: <3%** **BMI <18.5 Prevalence: <10%**	**Wasting Prevalence: 3-10%,** unstable **BMI <18.5 Prevalence: 10-20%**	**Wasting Prevalence: 10-15% OR >** usual & increasing **BMI <18.5 Prevalence: 20-40%, 1.5x** greater than reference	**Wasting Prevalence: 15 – 30% OR** > usual & increasing **BMI <18.5 Prevalence: >40%**	**Wasting Prevalence: >30%** **BMI <18.5 Prevalence: far > 40%**
	Mortality	**CDR: <0.5/10,000/day** **U5DR: <1/10,000/day**	**CDR: <0.5/10,000/day** **U5DR: ≤1/10,000/day**	**CDR: 0.5-1/10,000/day** **U5DR: 1-2/10,000/day**	**CDR: 1-2/10,000/day OR >2x** reference **U5DR: 2-4/10,000/day**	**CDR: >2/10,000/day** **U5DR: >4/10,000/day**
General Response Objectives		***Cross-cutting Objectives*** (1) mitigate immediate outcomes, (2) support livelihoods, (3) address underlying causes and chronic food insecurity if it exists, and (4) monitoring				
		Priority: Build Resilience, Disaster Risk Reduction	***Priority:*** Disaster Risk Reduction, Protect Livelihoods	***Priority:*** Protect Livelihoods, prevent malnutrition and prevent loss of life	***Priority:*** Save Lives & Livelihoods	***Priority:*** Prevent widespread death and total collapse of livelihoods

5. System Design Specification

5.1 System Overview

To achieve its objectives, the system employs a number of modules that perform specific tasks. The functions performed by the system are implemented in modules. These functionalities include;

i. Interfacing between mobile phone and computer system.

   This interface is provided by SMS gateway. The gateway uses a software modem to bridge between the computer COM port and mobile phone. This interfaces helps during issuing commands to phone from computer. These commands are a set of Haye's AT (Attention) commands. They are used to perform various tasks including reading of messages in phone memory, sending and deleting messages.

ii. Reading SMS messages from mobile phone.

   This function is performed by query understanding engine module. It is the first process that runs when the query engine starts. Incoming messages in connected mobile phone will be read and saved in database table. These messages will be flagged so that they are in a ready state for processing.

iii. Understanding messages sent by users

   This involves processing messages that were flagged ready for processing. It is a function performed by query understanding engine. Messages are fetched from database table, they undergo a processing stage. During this processing stage, the output is an offer to sale or buy, it could also be price inquiry. Messages that do not fit any of these three outputs becomes an un-understood message. Such messages are saved and await human intervention. Messages that fall in the three output categories are saved in the database and flagged in a ready state for further processing.

iv. Responding to user queries.

   From the above process, offers and price inquiries that were saved in the database will be acted on. This function is performed by the query understanding engine. It utilizes input from the above process and outputs message to be sent to users. An offer to sell will be processed and all matching offers to buy that were sent in the last seven days will be sent in the response message. An offer to buy will be responded to with all matching offers to sell that are not more than seven days old. Inquiries for price will be responded to with price of the product in the market specified. Other queries that did not have an existing offer or price will be responded with a message specifying that the requested price does not exist.

v. Editing un-understood messages.

Messages that were not properly understood by the system will be saved in the database as un-understood messages. These will be acted on by a system administrator who will apply human touch to provide their understanding of user's intention. This will help to lower the messages that go un-answered.

5.2 System Architectural

5.2.1 Architectural Design

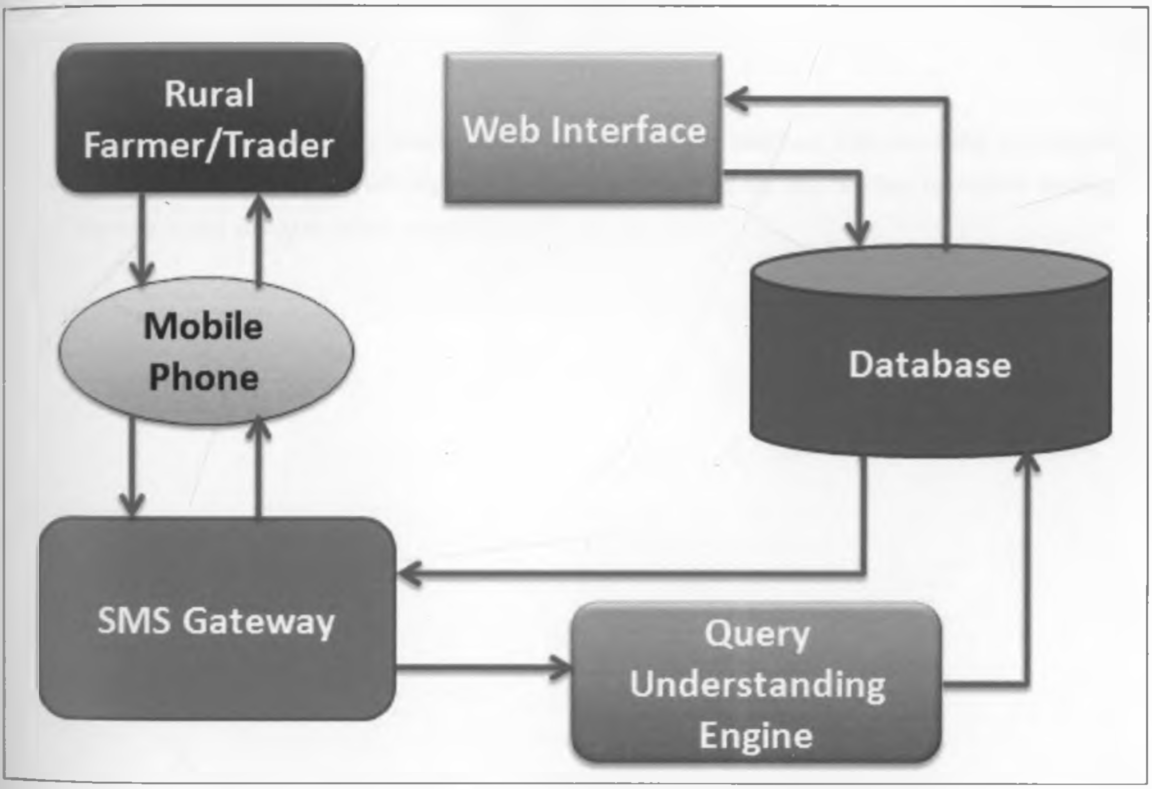


Figure 10 : System Modules

The diagram above (Figure 10) show the interaction of the main modules of the system. These are;

i. SMS gateway

This provides interface functionality between the system and the mobile phone which functions as GSM modem. Messages are read from the mobile phone by running a set of AT commands. The data read is saved in a database.

ii. Query understanding engine

Messages read from phone and saved in database is now processed. The messages will be processed and classified as either offers to sell, offers to buy, or price inquiry. Those that do not fall into any of these categories are classified as un-understood and saved in database. Once all the messages have been processed, this process waits for the next run to process any messages that could have been received.

iii. Database

This is the central repository for data. This system is database driven and therefore all messages read from phone will be saved in it. All other processes that follow will store or read data into or from the database. Messages received or sent by the system are saved in the database. Other supporting data like names of commodities and markets are also saved in this here.

iv. Web interface

Un-understood messages are viewed and edited using a web interface. Also incoming and outgoing messages are viewed using this interface. System administrators use this interface to monitor messages coming in and going out of the system.

5.2.2 Decomposition Description

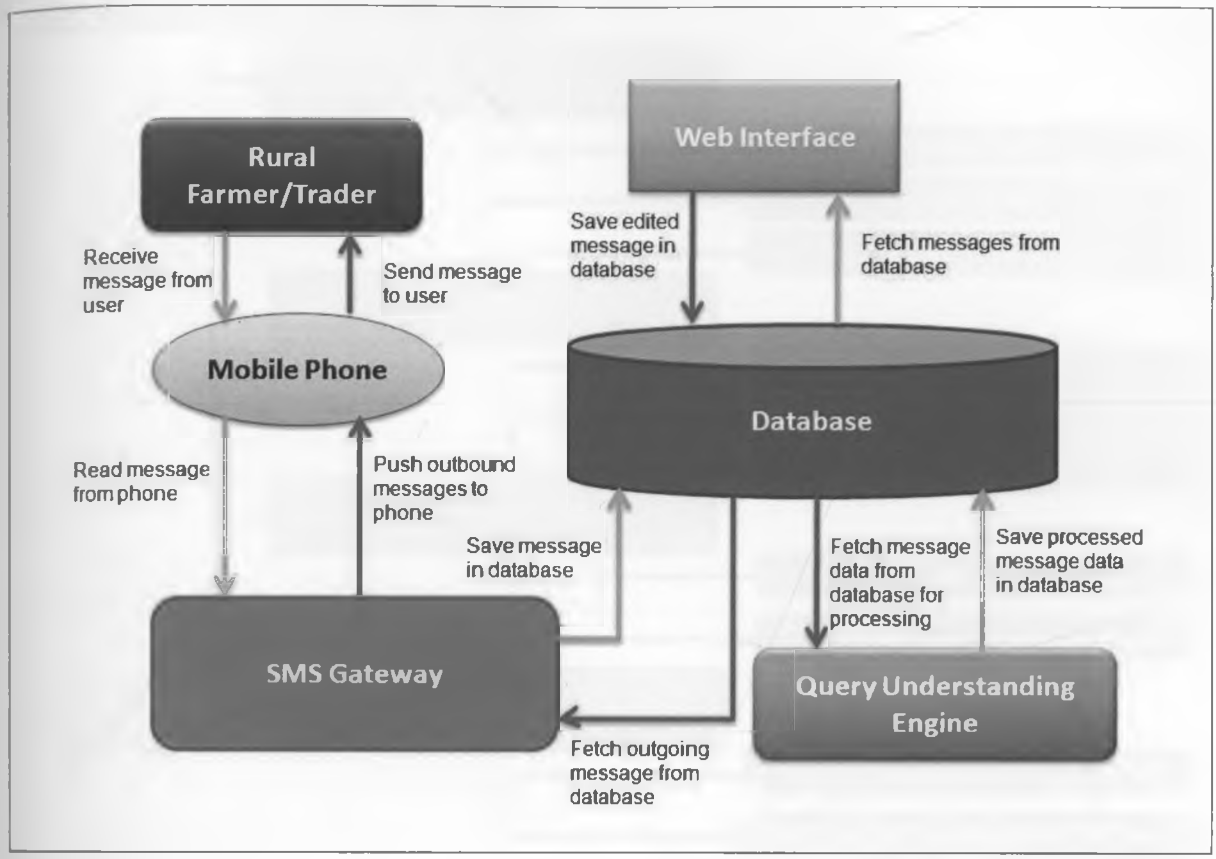

Rural Farmer/Trader
Receive message from user
Send message to user
Mobile Phone
Read message from phone
Push outbound messages to phone
SMS Gateway
Web Interface
Save edited message in database
Fetch messages from database
Database
Save message in database
Fetch message data from database for processing
Save processed message data in database
Query Understanding Engine
Fetch outgoing message from database

Figure 11 : Data Flow Diagram

The diagram (Figure 11) above describes the flow of data from one module to the next. This diagram differs from Figure 1 in the manner in which the modules are depicted. In Figure 10, the modules are shown according to how processed flow without necessarily showing data flow. In Figure 11, emphasis is placed on data flow from one module to the other. The central role played by the database is also laid out.

Below is the system's functional decomposition diagram, Figure 12. The manner in which the functions are decomposed into processes and sub processes is shown.

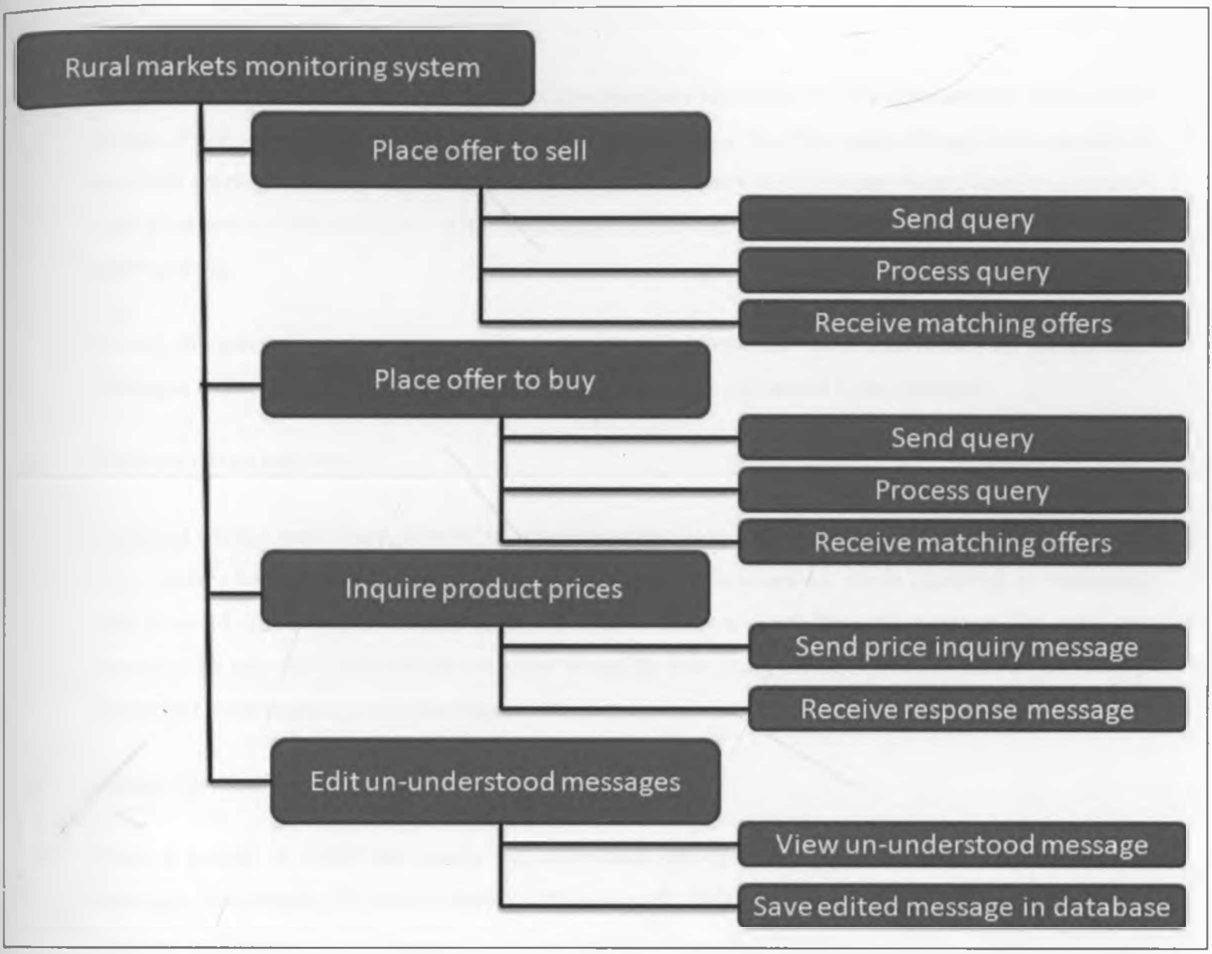


Figure 12 : Functional decomposition diagrams

5.3 Design Rationale

Architecture depicted in Figure10 was chosen in consideration of some factors. These factors are;

i. Central role of query understanding engine in the application.

The query understanding engine plays a pivotal role in translating natural language queries to SQL The syntax flexibility aspect of this system which is a key feature is enabled by the query understanding engine. Making the query understanding engine a central module ensures that understanding meaning of incoming messages, determining language used in a message and also determining response message to be sent to users is as efficient as possible.

ii. SMS gateway interface

The position of the SMS gateway in the system architecture is because of its role as an entry point to the system. Rural farmers and traders send messages to the system. The first point of entry to the system is the SMS gateway. Making any other module the point of entry to the system would lead to a system that is not lean and less efficient. After a message is received, it is read from the mobile phone by the SMS gateway.

Having the gateway as close as possible to the phone ensures that there's efficiency in the system. Messages received by the phone can be pulled by the gateway and stored in the database.

iii. Database driven operations

Persistent storage was chosen in order to ensure that new knowledge acquired by the system is not lost. The system's learning ability is enabled by the database. It is where all words classified as unwanted will be saved. These words are later used in a filter to remove words from the message. The message remains with only few words which are easier to handle than many words, some of which do not add to the overall understanding of the message.

iv. Human intervention using web interface

There is a limit to which the system can understand incoming messages. Some improperly formed messages, for example "is maize expensive these days?". This message contains only one of the three required parameters for a message to be understood. The other parameters being [MARKET] and [INTENTION]. The intention in this message is implied, it can be understood to be price inquiry. Such messages will require human intervention in the web interface to be able to offer a human touch in resolving what the user intended.

This approach improves the system's usability. By intervening to avert errors or unresponsiveness to user request, the user's can always get feedback on the state of their requests.

5.4 Data Design

5.4.1 Data Description

The system described in this design document will be highly data driven. From the moment a user sends a message to the system to the point when they receive a response, many operations occur along the way. All these operations handle data in one way or another, either fetching, saving, or updating. There is a database that will be the central repository of all messages, incoming and outgoing.

There will be one database to store all the data involved in the system. In this database, there will be a total of 12 tables. Each of the tables will perform specific functions. Figure 13 shows tables and their relationship to each other.

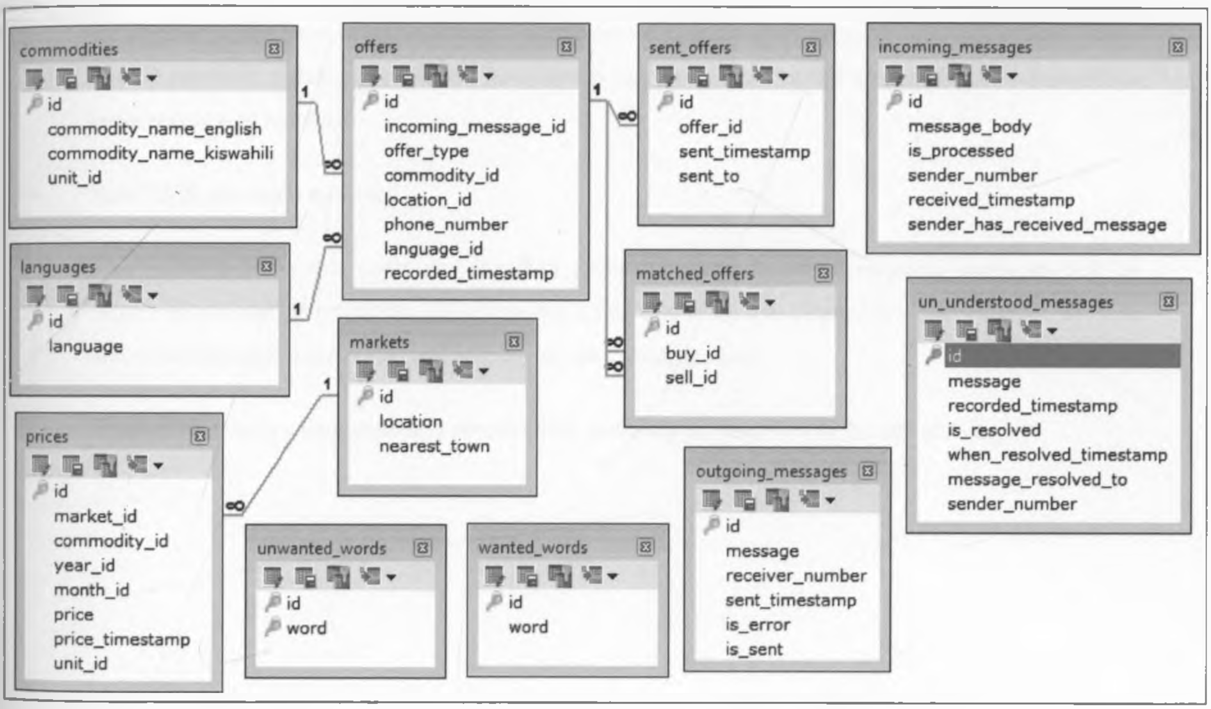


Figure 13 : Database tables

A typical flow of data in the system is as follows;

i. SMS message is received by the system from a user and SMS reader process in SMS gateway module runs.

   This message could be either offer to sell, offer to buy, price inquiry or an un-understood message. This message, regardless of the category it falls in, will be saved in the incoming_messages table.

ii. Query understanding engine process runs.

During this run, all messages in incoming_messages table that were received in the last seven days will be processed. These messages will be subjected to a series of rules in the query understanding engine. One of these rules will be the language determiner. After the language is figured out, the meaning of the message is then figured out. This is done using other sets of rules. Messages that are understood are saved in the offers table. Those that are not understood are saved in the un_understood_messages table.

iii. The offers process executes.

Offers saved in the offers table in (ii) above will be processed. During this phase, each offer is matched to all other offers in the offers table. Offers to sell are matched with offers to buy and vice versa. Price inquiry messages on the other hand are processed by getting the requested price. After offers have been matched or prices have been determined, a new record is inserted to outgoing_messages table. Offers that do not have matching offers will have a response constructed to the user so that they know their request was well received.

iv. Send SMS process is executed.

This is the process that completes the data cycle. Messages saved in outgoing_messages will be processed and sent to recipients. This table has a flag that is used to determine what messages wo send and what messages are already sent and do not need sending again.

Figure 14 below is a data transition process flow summary as described in the text above.

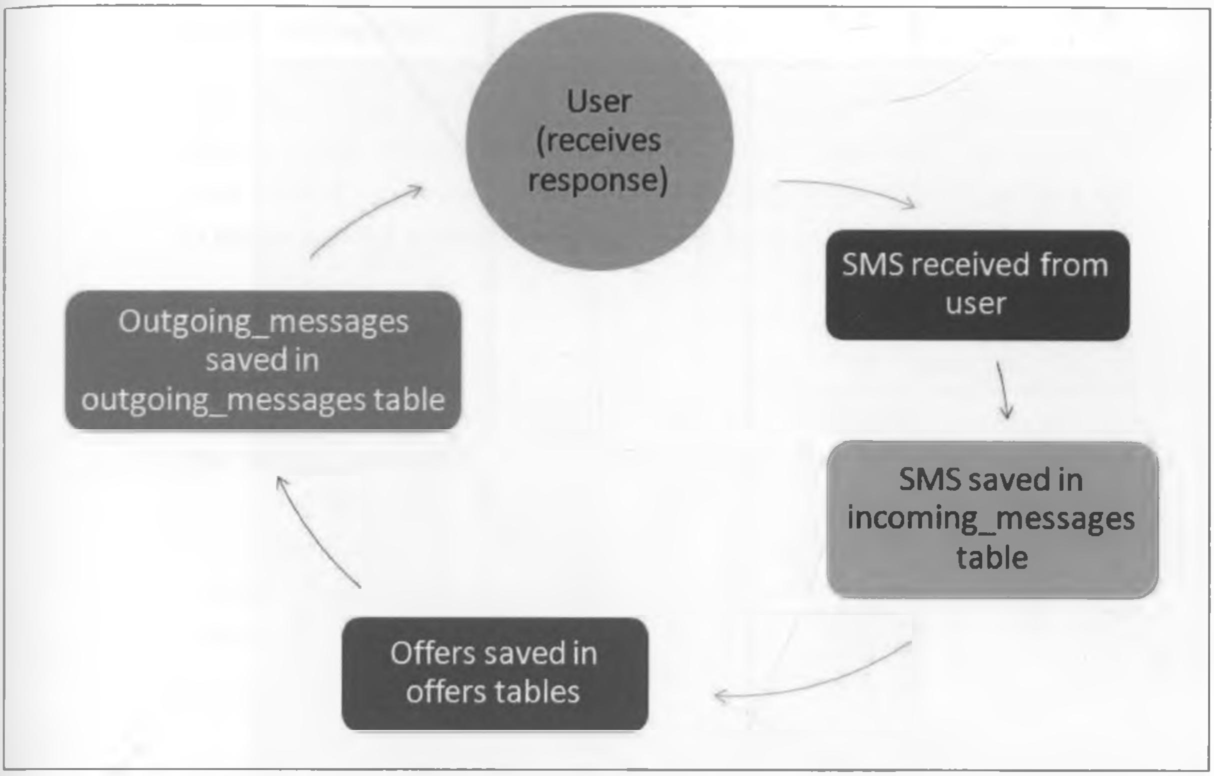

User (receives response)
SMS received from user
SMS saved in incoming_messages table
Offers saved in offers tables
Outgoing_messages saved in outgoing_messages table

Figure 14 : Data transition flow process

5.4.2 Data Dictionary

To describe the various data objects involved in the system, a functional approach will be used. Functional parameters will be described since they are the data objects going in or out of the system's functional components.

i. Place offer to buy or sell

Both offers to sell and buy involve similar steps with similar data involved. Placing an offer involves three major steps. These are;

a. Send query

The query sent is an SMS message in a natural language, either English or Kiswahili. This SMS message is a String and will be saved in the incoming_messages table as a String. The associated sender's number will be saved also as a String

b. Process query

Message saved in the incoming_messages table will be processed. The input to this process is a data item (message) of type String. This message will be processed to yield offer details. These are ; incoming message id (integer), offer type (String) commodity id (integer), market id (integer) and language id (integer). These data items will be saved in the offers table.

c. Receive matching offers

With offers saved in offers table, the system will match these offers to come up with pairs of offers to sell and buy. Also price inquiries will have matching product prices. The output of offers matching is a response message of type String. This message will contain product prices or phone numbers of traders or farmers whose offers are matched the current offer.

ii. Inquire product price

a. Send price inquiry message

Price inquiry message sent by users is of type String. The associated sender's number is also saved as a String. Message received is saved as a String in incoming_messages table. Processing of this message will yield same results as in (b) above. With incoming message id (integer), offer type (String), commodity id (integer), market id (integer) and language id (integer).

b. Receive response message

Since the offer type for this message is inquiry, the response will include the price of the commodity if the price is in the system database. The response message is a String send as an SMS to the recipient.

iii. Edit un-understood message

Human intervention occurs through the web interface. This will help to correct some of the un-understood messages. The messages are stored in the database's un_understood_messages table.

a. View un-understood message

Process involves fetching messages from the database. The message details fetched include message (String), sender number (String), is resolved flag (character (Y/N)), time when resolved (timestamp). When the message is fetched, these details are displayed to the user on the screen from where they can edit.

Acute Food Insecurity Reference Table for Area Classification

		Phase 1 Minimal	Phase 2 Stressed	Phase 3 Crisis	Phase 4 Emergency	Phase 5 Famine *(evidence for all three criteria of food consumption, wasting, and CDR is required to classify Famine)*
Phase Name and Description						
Area Outcomes	Food Consumption & Livelihood Change	More than 80% of households in the area are comfortably able to meet basic food needs without atypical coping strategies & livelihoods are stable	Based on the IPC Household Group Reference Table, at least 20% of the households in the area are in Phase 2, 3, 4, or 5	Based on the IPC Household Group Reference Table, at least 20% of the households in the area are in Phase 3, 4 or 5	Based on the IPC Household Group Reference Table, at least 20% of the households in the area are in Phase 4 or 5	Based on the IPC Household Group Reference Table, at least 20% of the households in the area are in Phase 5
	Nutritional Status	**Wasting Prevalence**: <3% **BMI <18.5 Prevalence**: <10%	**Wasting Prevalence**: 3-10%, unstable **BMI <18.5 Prevalence**: 10-20%	**Wasting Prevalence**: 10- 15% OR > usual & increasing **BMI <18.5 Prevalence**: 20-40%, 1.5 x greater than reference	**Wasting Prevalence**: 15 – 30%; OR > usual & increasing **BMI <18.5 Prevalence**: >40%	**Wasting Prevalence**: >30% **BMI <18.5 Prevalence**: far > 40%
	Mortality	**CDR**: <0.5/10,000/day **U5DR**: <1/10,000/day	**CDR**: <0.5/10,000/day **U5DR**: <1/10,000/day	**CDR**: 0.5-1/10,000/day **U5DR**: 1-2/10,000/day	**CDR**: 1-2/10,000/day OR >2x reference **U5DR**: 2-4/10,000/day	**CDR**: >2/10,000/day **U5DR**: >4/10,000/day
General Response Objectives		***Cross-Cutting Objectives***: (1) mitigate immediate outcomes, (2) support livelihoods, (3) address underlying causes and chronic food insecurity if it exists, and (4) monitoring				
		Priority: Build Resilience, Disaster Risk Reduction	*Priority:* Disaster Risk Reduction. Protect Livelihoods	*Priority:* Protect Livelihoods, prevent malnutrition, and prevent loss of life	*Priority:* Save Lives & Livelihoods	*Priority:* Prevent widespread death and total collapse of livelihoods

b. Save edited message

Un-understood message is displayed on a table with its details. The message can be edited by clicking on a button on the table row. A dialog is displayed where the user can write a new message and save in the system. The saving process will return an integer, this integer shows the number of records created. It should typically be 1. Less than one means saving process did not succeed.

5.5 Component Design

System components will be described using algorithm for each function listed above (3.2 Decomposition description). Each of the components will perform specific number of operations in specific order. This will make the system to behave in a predictable manner. The functional components with their algorithms are as follows;

i. Place offers (sell and buy)

a. Send query algoritm

```
1. Compose SMS message
2. Enter phone number of phone connected to system
3. Send message
4. End
```

b. Process query algorithm

```
1. Read unread messages in phone  memory
2. Save all new messages in incoming_messages table
3. Fetch all unprocessed messages incoming_messages table
4. For each message do 5,6,7
5. Extract INTENTION, MARKET and COMMODITY from the message
6. Get market id from markets table
7. Get commodity id from commodities table
8. Save offer details {offer type, commodity id, market id} in offers table
9. End
```

c. Receive matching offers algorithm

```
1. Fetch all offers that are < 7 days old in the offers table
2. For each offer do 3,4,5,6,7,8,9
3. Compare each offer to all other offers that are < 7 days old
4. If offer's market id, commodity id are equal and current offer type is buy and matched offer type is sell
 or
        If offer's market id, commodity id are equal and current offer type is sell and matched offer type is buy
            Then
                5. Construct response message
                6. Save response message in outgoing_messages table
                7. Save offer ids in matched_offers table
                8. Process next offer
  else process next offer
 9. Send response messages as SMS
10. End
```

ii. Inquire product prices

a. Send price inquiry message algorithm

```
1. Compose SMS message
2. Enter phone number of phone connected to system
3. Send message
```

b. Receive response message algorithm

```
1. Fetch all offers that are < 7 days old in the offers table
2. For each offer, do 3,4,5,6,7
3. If offer type is inquiry
        Else
        4. Go to next offer
5. Fetch price for matching market id and commodity id in prices table
6. Construct response message
7. Send response messages as SMS
8. End
```

iii. Edit un-understood messages

a. View un-understood message algorithm

```
1. Login to system
2. Load all un-understood messages on table
```

b. Save edited message

```
1. Login to system
2. Load all un-understood messages on table
3. Load particular message on dialog with button click
4. Enter new message in message resolved to text area
5. Update un_understood messages table with new message
6. If record is saved successfully return 1
        else
        return -1;
7. End
```

5.6 Human Interface Design

5.6.1 Overview of User Interface and Screen images

In the design of this system, user centeredness ensure that all actions that a user can do are well laid out. The functions of the system as explained in 5 above details how each action that can be completed by the user is translated into an algorithm. These algorithms when followed will lead users to complete all the intended functions of the system. Following are user actions in graphical view

i. Place offer to sell

User composes a text message then inserts the phone number of the recipient (phone connected to the system), see Figure 15 below.

Figure 15: Composing SMS message

User sends message.

Message example: "SELL MILK LKAPEDO"

Depending on various phone models and user settings, confirmation may be received when message is sent. If user does not have sufficient funds, the message will not be sent. User will receive appropriate information depending on phone model.

Feedback Example: "Message sent" or "Sending message failed"

ii. Place offer to buy

This action involves similar interface but slightly different message sent as in i. above. Users composes messages with intention being a buy offer.

Message example: "BUY MAIZE LOKICHAR"

Then user sends message to the phone number of phone connected to system.

Feedback Example: "Message sent" or "Sending message failed"

iii. Inquire product price

Action involves user sending message with intention of inquiring product price.

Interface is a mobile phone SMS compose window, this is different in different phone models. An example is shown in Figure 6.

Message example: "BEI MAHARAGWE CHWELE"

Next user sends message to phone connected to system. Phone responds according to phone model, carrier settings or phone settings. Also depends on whether or not user has enough units to complete the message sending. Availability of network coverage is also a factor.

Feedback Example: "Message not set"

iv. Edit un-understood messages

User has to login to the system to complete this action. User enters username, password and clicks Login button as shown on Figure 16.

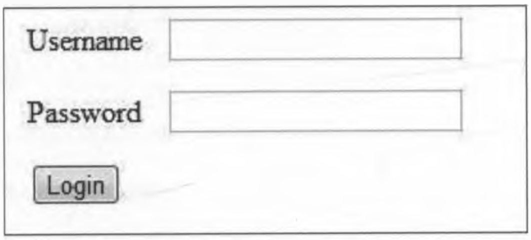
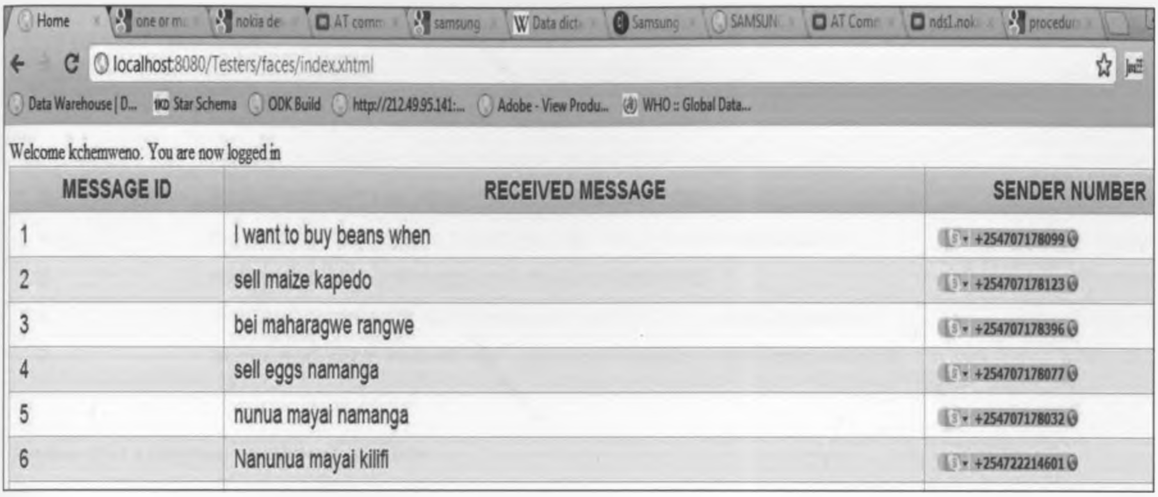


Figure 16 : System login

If user credentials are valid, user is presented with home screen (Figure 17) which defaults to incoming messages window.

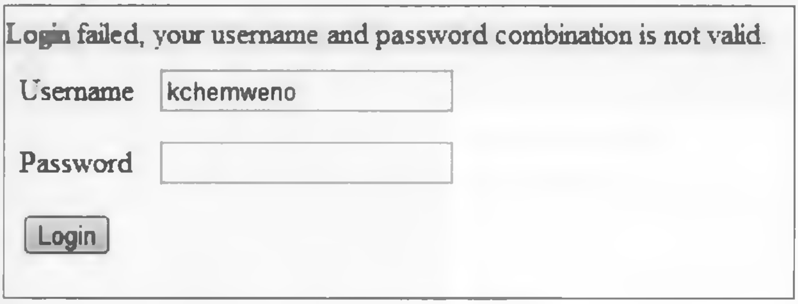


Welcome kchemweno. You are now logged in

MESSAGE ID	RECEIVED MESSAGE	SENDER NUMBER
1	I want to buy beans when	+254707178099
2	sell maize kapedo	+254707178123
3	bei maharagwe rangwe	+254707178396
4	sell eggs namanga	+254707178077
5	nunua mayai namanga	+254707178032
6	Nanunua mayai kilifi	+254722214601

Figure 17 : Incoming messages window

If the credentials are invalid as shown in Figure 18, the user remains on the login page with information that attempt to login was not successful.



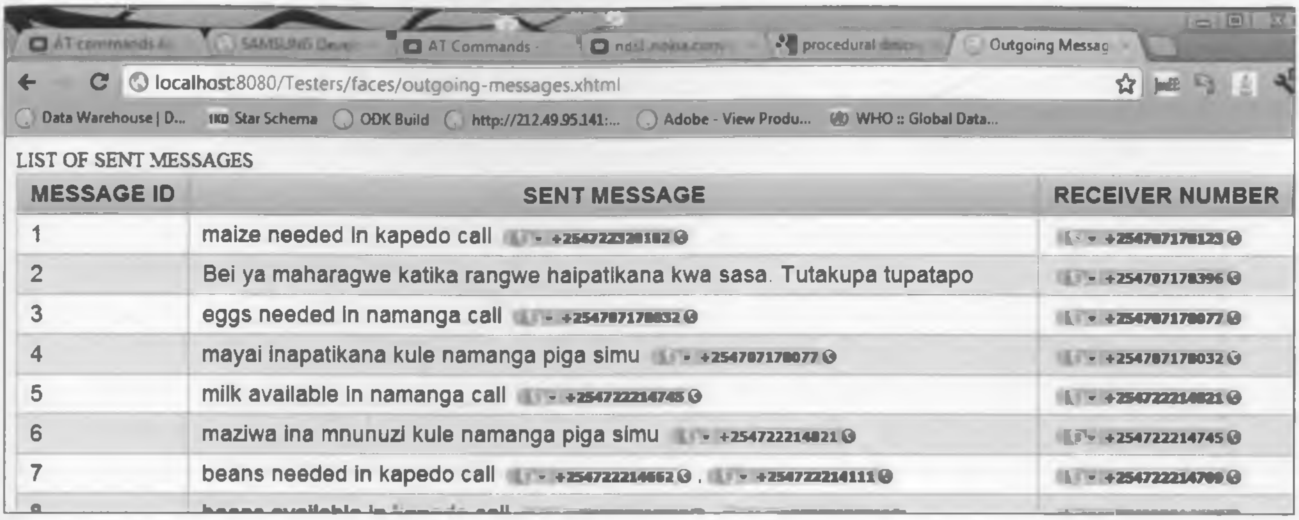
Login failed, your username and password combination is not valid.

Username kchemweno

Password

Figure 18 : Login failed

With user successfully logged in, other windows he/she can access include the outgoing messages window shown below on Figure 19.



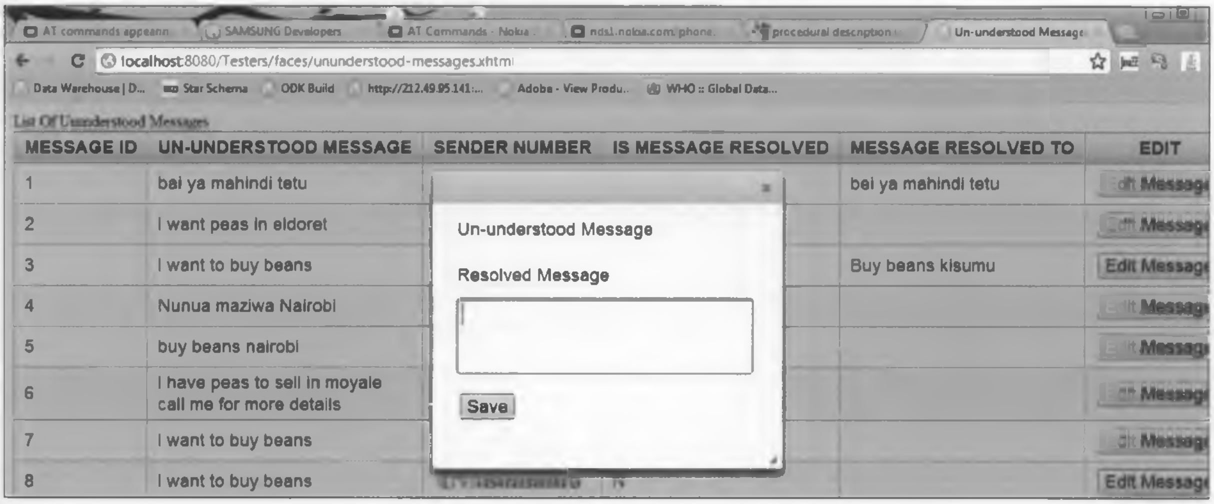
localhost:8080/Testers/faces/outgoing-messages.xhtml

LIST OF SENT MESSAGES

MESSAGE ID	SENT MESSAGE	RECEIVER NUMBER
1	maize needed in kapedo call +254722320182	+254707178123
2	Bei ya maharagwe katika rangwe haipatikana kwa sasa. Tutakupa tupatapo	+254707178396
3	eggs needed in namanga call +254707178032	+254707178077
4	mayai inapatikana kule namanga piga simu +254707178077	+254707178032
5	milk available in namanga call +254722214745	+254722214821
6	maziwa ina mnunuzi kule namanga piga simu +254722214821	+254722214745
7	beans needed in kapedo call +254722214662 , +254722214111	+254722214709
8	beans available in kapedo call [illegible]	[illegible]

Figure 19 : Outgoing messages window

To complete the editing un-understood messages, user will access the un-understood messages window. This window lists all messages that were not understood by the system when the messages was first processed.



localhost:8080/Testers/faces/ununderstood-messages.xhtml

List Of Ununderstood Messages

MESSAGE ID	UN-UNDERSTOOD MESSAGE	SENDER NUMBER	IS MESSAGE RESOLVED	MESSAGE RESOLVED TO	EDIT
1	bai ya mahindi tetu			bei ya mahindi tetu	Edit Message
2	I want peas in eldoret				Edit Message
3	I want to buy beans			Buy beans kisumu	Edit Message
4	Nunua maziwa Nairobi				Edit Message
5	buy beans nairobi				Edit Message
6	I have peas to sell in moyale call me for more details				Edit Message
7	I want to buy beans				Edit Message
8	I want to buy beans	[illegible]	N		Edit Message

Un-understood Message

Resolved Message

Save

Figure 20 : Un-understood messages

A mouse click on the Edit Message button on a row will launch a dialog. The dialog helps a user to edit un-understood message. After typing in the resolution message, the user will click on save to commit the data to the database.

5.6.2 Screen Objects and Actions

Most of the actions that occur in this system are internal. The system performs many tasks for the good of the user. The on-screen actions objects and actions can be categorized under;

i. Message compose window.

This is a mobile phone interface. Depending on the phone type, the screen can be of varied size. Also the screen can have a touch enabled keypad or a legacy keypad. User types messages and phone number, then sends the message using the keypad.

ii. Message reading window

To read an incoming message, a user will access the message inbox and use available command on the phone to bring the message to view. This view is the interface with which a user reads incoming messages.

iii. Login window

To access the web interface, system administrators will login. They will enter a username on the username field, a password on the password field. Next the user will click on Login button. This button will execute commands to authenticate user. If user supplied valid credentials, then they are granted access to the system, otherwise, an error message will be displayed.

iv. View messages window

The view messages window has a table and various components depending on what messages are on view. In the un-understood messages, the last column of the table has a command button for launching dialog for editing un-understood messages. The other message type views do not have this buttons.

5.6.3 **Requirements Matrix**

Following is the tabular view of high level system components alongside their SRS functional requirements. This table depicts a clear match between system requirements and system design.

System Component	SRS Functional Requirement
SMS Gateway	SMS Gateway REQUIREMENT 1
SMS Gateway	SMS Gateway REQUIREMENT 2
Query Understanding Engine	Query Understanding Engine REQUIREMENT 1
Database	Database REQUIREMENT 1
Web Interface	Web Interface REQUIREMENT 1
Web Interface	Web Interface REQUIREMENT 2

Table 5:Systems Requirements Matrix

5.7 Usability Design

A key component in this system in ensuring that users achieve higher satisfaction and better experience while using is factored in usability. Various aspects incorporated in the system have the objective of enhancing usability of the system.

5.7.1 Error prevention

Occurrence of errors is part of any software. However effort has to be put to minimize and prevent them from occurring. Those that occur should be handled well to ensure that user experience and use of the system is not hampered a great deal. In the system, use of a flexible SMS message syntax ensures that messages sent in by users are processed regardless of position of parameters in the message. Messages which have more than the required words will also be able to be processed. The system does more work for the user in filtering out unwanted words and remaining with only those words that are useful in understanding the message. Without this, the system would go into an error state whenever a message is read that has more words that necessary. This is a clear departure from other systems like MFarm.

5.7.2 Feedback

Whenever users send offers to the system either to sell, buy or inquire product prices, the system will respond. If there are matching offers, the system will respond with all the matching offers in as few messages as possible, otherwise the system will alert the user that there are no matching offers currently. This information feedback allows users to be in the know regarding the status of their inquiry.

5.7.3 Minimal user memory load

Recognition rather than recall still is a key usability aspect. The system does not require the user to memorize codes of markets, commodities or people. The system helps the user in performing the functions of recall by using past experience in processing incoming messages. Use of a flexible syntax that can be in a natural question format will yield same results with a message that has only the required number of parameters. This way users are able to interact with the system in a natural dialog as much as possible.

5.7.4 User Language

Support for both English and Kiswahili languages allows more users to be able to use the system without language restriction. User's who send in queries in English will receive responses in English, those that send queries in Kiswahili will receive responses in Kiswahili. This makes users experience with the system a pleasant one.

5.8 Testing

The developed system is for the purpose of attaining the three objectives set in this research. [illegible] are;

To develop a QUE that processes queries received in Kiswahili or English.

- To develop an SMS gateway for receiving queries from farmers and traders [illegible] requested information to farmers and traders.

- To test the Agricultural Market Information System for accuracy in processing [illegible] queries sent in Kiswahili or English and for usability.

In the requirements specification document for this system, various key requirements were [illegible] done on this system were done around the requirements specified and also the objectives [illegible] the system to be successful, it ought to aid in attainment of the objectives set above and [illegible] requirements set in the requirements specification document.

Acute Food Insecurity Reference Table for Area Classification

	Phase Name and Description	Phase 1 Minimal	Phase 2 Stressed	Phase 3 Crisis	Phase 4 Emergency	Phase 5 Famine *(evidence for all three criteria of food consumption, wasting, and CDR is required to classify Famine)*
Area Outcomes	Food Consumption & Livelihood Change	More than 80% of households in the area are comfortably able to meet basic food needs without atypical coping strategies & livelihoods are stable	Based on the IPC Household Group Reference Table, at least 20% of the households in the area are in Phase 2, 3, 4, or 5	Based on the IPC Household Group Reference Table, at least 20% of the households in the area are in Phase 3, 4, or 5	Based on the IPC Household Group Reference Table, at least 20% of the households in the area are in Phase 4 or 5	Based on the IPC Household Group Reference Table, at least 20% of the households in the area are in Phase 5
	Nutritional Status	**Wasting Prevalence**: <3% **BMI <18.5 Prevalence**: <10%	**Wasting Prevalence**: 3-10%, unstable **BMI <18.5 Prevalence**: 10-20%	**Wasting Prevalence**: 10-15% OR > usual & increasing **BMI <18.5 Prevalence**: 20-40%, 1.5 x greater than reference	**Wasting Prevalence**: 15 – 30%; OR > usual & increasing **BMI <18.5 Prevalence**: >40%	**Wasting Prevalence**: >30% **BMI <18.5 Prevalence**: far > 40%
	Mortality	**CDR**: <0.5/10,000/day **U5DR**: <1/10,000/day	**CDR**: <0.5/10,000/day **U5DR**: ≤1/10,000/day	**CDR**: 0.5-1/10,000/day **U5DR**: 1-2/10,000/day	**CDR**: 1-2/10,000/day OR >2x reference **U5DR**: 2-4/10,000/day	**CDR**: >2/10,000/day **U5DR**: >4/10,000/day
General Response Objectives		***Cross-Cutting Objectives***: (1) mitigate immediate outcomes, (2) support livelihoods, (3) address underlying causes and chronic food insecurity if it exists, and (4) monitoring				
		Priority: Build Resilience, Disaster Risk Reduction	***Priority***: Disaster Risk Reduction, Protect Livelihoods	***Priority***: Protect Livelihoods, prevent malnutrition, and prevent loss of life	***Priority***: Save Lives & Livelihoods	***Priority***: Prevent widespread death and total collapse of livelihoods

All the tests done that involved messages being sent, received or processed involved 40 messages during each test.

5.8.1 SMS Gateway Testing

Interface between users (traders and farmers) and the system is provided by the SMS gateway. These users rely on the gateway to deliver their messages to the system and also to deliver the system's messages to these users. The gateway interacts with the mobile phone used as GSM modem to either pull or push messages. The SMS basis in this system would not have been attained without the SMS gateway.

5.8.1.1 Requirements

A high priority feature that is more expensive than other features. This is because it relies on a mobile phone that acts as a GSM modem. The gateway is implemented using Hayes AT commands. The system reads from or writes to a serial port depending on the direction of message movement, in or out of system.

5.8.1.2 Tests

The test for this module was done both during development and also after completion of development. Errors found during development included some messages sent from the system containing AT command text in them. Message intended to be "maize is available" contained unwanted text for example "AT+CMGS maize is available". These errors were rectified and were not experienced by the time testes after development were done.

Apart from testing for accuracy in messages coming in and going out of the system, that is, messages sent to the system is received in the same format, length and wording. On the same note, messages sent by the system are received in the same format, length, and wording by intended recipients.

5.8.1.3 Outcomes

Tests for accurate handling of SMS messages that were done discovered some bugs. Some include messages sent containing some unwanted and intended text, as mentioned above. These errors were resolved. Other errors encountered involved messages slipping through the gateway when attempts were made to read. This scenario was caused by the phone's primary memory getting filled up. Solution was to always delete messages after they are read by the gateway. Storage of the read messages was done in the **incoming_messages** table of the database.

At the time of completing system development, tests done on the SMS gateway did not detect any errors on the messages, or any other errors. Messages sent from the system and messages sent to system were found to be 100% successfully delivered.

5.8.2 QUE Testing

Testing QUE entailed thorough checks on incoming messages and classifications made to offers. Each understood message results in an offer that can be a sell, buy or inquiry for product price. Messages that are not understood were also checked to ensure whether the QUE was at fault or the message was properly classified as not understood. The system's intelligence and learning sub module is built into this module. Newly received messages are processed and unwanted words feed into the **unwanted_words** table. These words act as filters for future messages received. The leaning ability of the system is also a feature that was subjected to test.

5.8.2.1 Requirements

QUE is another high priority module in the system. It provides language and learning functions that are in the system. The system is able to process both English and Kiswahili messages into offers via SQL statements. Learning is included to make the system improve in performance (speed and accuracy) with age and experience.

5.8.2.2 Tests

Proper functioning of the QUE is a key part that builds to the overall success of this system. The two major sub-modules of the QUE mentioned above were subjected to tests both during and after development. Messages were sent in and the output in the **offers** table or **un_understood_messages** table were compared. During developed and before the system had undergone training, the system's accuracy in classifying messages stood at 52%. Messages that were sent in a strict syntax of [INTENT] [COMMODITY] [MARKET] were classified to an accuracy of 88%. The remaining 12% were messages whose commodities were not already in the system. This improved to 97% when the messages contained commodities that are already in the system.

5.8.2.3 Outcomes

Bugs were found in the module that makes meaning from messages (QUE). The errors included messages being classified wrongly for before the system was trained. After the system had been trained with 200 messages, there QUE performed better, classifying 96% of all the messages correctly. The sub-module that handles determination of language used in a message performed well with 99% accuracy rate.

5.8.3 Database Testing

Database tests were done to check saving, updating, retrieving and processing data in the system. All SMS messages received by the system or sent by the system are saved in the database. Various tables are used in this task, these tables are; incoming_messages, out_going_messages, and un_understood_messages. The data stroed by each table is described by its name. The comparison between messages sent to the system and the messages in the incoming_messages and un_understood_messages. During the test, the counts of these tables had to be equal. Messages sent by system and received were compared with the out_going_messages table. The counts in this table had to equal the messages sent by the system and received.

5.8.3.1 Requirements

The database server must allow connections to be made to it. The various modules like the QUE, and the SMS gateway rely on the database to provide persistent storage of information. This is why the database server has to keep running as long as there is a user logged in.

5.8.3.2 Tests

Saving, retrieving, updating and processing of data in the database was tested. For saving data, new messages were sent to the system using SMS. The incoming_messages, offers and out_going_messages table of the database were first truncated (all data was deleted). Then ten new SMS messages were sent to the system.

The SMS gateway was then executed to fetch all the messages from the phone attached to the system. The incoming_messages table was then inspected and its contents in wording and number were compared. This was done repeatedly for four times. The findings were impressive with all the messages sent to the phone being saved in the database.

Retrieving messages function of the database was also tested. Retrieval is performed by offers and QUE modules. The database in this case accepts these modules to connect to it and fetch data in them. The tests done on fetching also were successful with all the data in incoming_messages and offers tables of the database. All the data in this database that were meant to be fetched based on the SELECT query criteria were fetched. This was verified by inspecting the messages in the respective table and the changes it had on other tables where the source data selected was meant to update.

5.8.3.3 Outcomes

Tests done for database module were found to confirm successful completion of saving, updating and retrieval processes. Validation was done on the data before saving in each of the modules. This has helped ensure that issues like saving of blank phone numbers do not arise.

5.8.4 Web Interface Testing

System administrator functions will be done via the web interface. These functions include editing un-understood messages to ensure that the system understands them next time during the next pass. The system administrator will also monitor the system to ensure that all modules are properly functioning.

5.8.4.1 Requirements

This is a medium priority feature. Main users (farmers and traders) can still continue to use the system even without the web interface being in place. To aid the system in messages that it is not able to identify all the important parts, the system administrator uses the web interface. The web interface will depend on a web server. The web server should start and be ready to accept connections as long as there is a user logged on.

5.8.4.2 Tests

Messages loaded on the web interface were verified to ensure that they are of the right classification. When the system administrator is loading in-coming messages, the messages should be those in the incoming_messages table in the database. All the other message types should be loaded from their respective tables. That is, outgoing messages from the out_going_messages table and un understood messages from the un_understood_messages table. The tests involved comparing messages by wording and number or count.

Login tests were done to check whether users who do not have an account in the system can access the system. Also, tests were done to check whether users can register using previously registered usernames.

5.8.4.3 Outcomes

First five tests revealed incoming messages being loaded on the web interface were less in number than those in the incoming_messages table of the database. This was found to be caused by an erroneous filter in the SQL string for fetching incoming messages. The SQL statement was omitting some valid incoming messages from being loaded. Thirty other tests confirmed successful fetching and displaying of the incoming messages.

On the Login tests, security checks were done to verify whether non-registered users could have access to the system. This tests which involved using random usernames and passwords proved that only registered users are able to access the system. Tests on whether registered users can register more than once were negative, showing that only users who have not previously registered are able to register.

6. Discussion and Conclusion

Achieving an affordable, and usable SMS system for use by rural farmers and traders is a huge boost. Not only to the farmers and traders but also to the wider economy and society in general.

6.1 Implications

Successful implementation of an SMS based agricultural monitoring system will result in the following implications;

i. Better farm gate prices for farmers' produce

With farmers having the capacity to inquire prices prevailing in other markets, they are able to determine better prices to sell their produce. They are able to make decisions on whether to sell at the farm gate, and if they do, they do so at better prices. Exploitation by middle men will be greatly reduced.

ii. Wider market access by rural farmers
Using their mobile phones, users can virtually visit most markets and find out prices and levels of demand in those markets. With this information, the farmers are able to deliver to markets with better prices and higher demand.

iii. Better decision making by traders
Traders can consume the vast amount of information availed at their disposal by this system. They are able to know where to buy produce and at what prices. They are able to combine many offers and utilize single transportation for all nearby offers.

iv. Less produce going to waste in farms or markets
Farmers are now in a position to place offers to sell produce even before harvesting. Produce like tomatoes that perish easily can be sold before harvesting hence minimizing losses.

v. Food security
Improved agricultural income for farmers will make agricultural production a profitable venture. Attracting more citizens to practice agriculture will lead to better food stocks in the economy hence food security.

6.2 Limitations

In conducting this research, some aspects proposed brought about challenges along the way. These make the limitations of this research, they include;

i. Data collection method.
Use of questionnaires to collect data ran into some challenges regarding participants who were unable to read. Some of the rural farmers had challenges reading both the Kiswahili and the English version of the questionnaires. To mitigate this challenge, the questionnaire was read to those participants.

ii. Limited data available on rural markets
The markets selected for this study were the rural markets. Data regarding prices, commodities sold in most of the markets was found to be limited. Most data that exist are on major urban markets in Kenya. Each of the rural markets differ in terms of products sold, buying habits of the population and even prices of commodities, extrapolation of nearby major markets may not have shed a clear picture on the rural markets.

iii. Multiple versions of Kiswahili language spoken in Kenya
There were many versions of Kiswahili encountered during this research. Some participants would use non-standard versions of the language with a heavy influence on their mother tongue. Bringing these versions on board would be a huge boost to the usefulness of any research similar to this.

iv. Budget constraints
With adequate budget, a GSM modem would be bought to replace mobile phone that performs similar function. This would make the system better in terms of performance. A GSM modem can handle huge amounts of data at one go compared to a mobile phone. More funding would also help to increase the number of participating markets and farmers in order to get a better feasibility of the system.

6.3 Recommendations for further research and practice

There is need for more research in this area. The ultimate goal of providing a system which rural farmers and traders can use to exchange information is one that ought to be reached. The following are the recommendations for further research on this work.

i. Inclusion of additional languages

   English and Kiswahili are enables in this research. There is need for more Kenyan local languages to be included in the system. This would ensure that the bottom of the pyramid who do not understand English, Kiswahili or those not comfortable using the two languages can use. More languages will also mean more versions of Kiswahili and English spoken around Kenya are included in the system

ii. Addition of market analysis functionalities

   Statistical information would be a boost to farmers and traders. With quick statistical information, they are able to make predictions or plan ahead. Statistics like trends of prices of commodities help to predict future behavior of the market.

iii. Enhancement of query understanding engine

   The learning ability of the system can be further improved. This can be done by not only allowing the system to learn with unwanted words in messages but also with wanted words. This way the system is able to learn from mistakes and from correct information. It currently learns on mistakes only.

iv. Increasing information available to users

   Currently users are not able to get information on quantities. They can access prices, and products available in various locations. Including quantity in products available will be of importance to users since they can do group buying or group selling.

6.4 Conclusion

An SMS based agricultural markets monitoring system built using a framework that addressed specific challenges experienced Kenyan rural areas is possible. This system will heavily be guided on a framework like the once developed during the course of this research, refer to Figure 1.

To access market information, rural farmers and traders can send SMS messages without adhering to a rigid syntax as long as they supply all required parameters. The system will then respond with requested information. This system can act a platform that users can use either in English or Kiswahili and be able to receive timely information regarding market situation.

With a 95% accuracy in processing received queries and responding to users, the system with its learning ability is able to improve with time and be more useful with time. With time, more data is available to the system and also the filter that builds up with each learning experience makes it more useful to users.

7. BIBLIOGRAPHY

1. Aker, J.C. and Mbiti, M.M.,2010. Mobile Phones and Economic Development in Africa. *Journal of Economic Perspectives,* 24(3) , pp.207-232.

2. BusyLabs, 2012. *BusyLabs|ICTWorks.*[online], Available at: <http://www.ictworks.org/tags/busylab> [Accessed 2 Feb 2012].

3. Common Market for East and Southern Africa, 2007.*FAMIS Users' Manual* . Lusaka: COMESA Secretariat.

4. Communications Commission of Kenya, 2012. *Quarterly Sector Statistics Report 1st Quarter July-September 2011/2012*. Nairobi: CCK

5. Eastern Africa Grain Council, 2012. *About RATIN – Regional Agricultural Trade Intelligence Network.* [online], Available at: <http://www.ratin.net/section.asp?id=3> [Accessed 2 Feb 2012].

6. Famine Early Warning System Network, 2011. *Using ICT To Provide Agriculture Market Price Information In Africa* . New York: FEWSNET.

7. Jose, A.I., 2008. Private Extension and Cyber Extension in India. In: B.S. Hansra et al, ed. 2008. *Agricultural Extension Systems.* New Delhi: Concept Publishers. Ch.5.

8. Kenya Institute of Public Policy Research and Analysis, 2001. *Working Paper Number 4 May 2001.* Nairobi: KIPPRA

9. Lidwell, W. Holden, K. and Butler, J., 2003, *Universal Principles of Design.* Gloucester Massachusetts : Rockport .

10. M-Farm Ltd, 2012. *About Us.* [online] Available at: < http://mfarm.co.ke/about > [Accessed 29 February 2012]

11. Muchemi, L. 2008. Towards Full Comprehension of Swahili Natural Language Statements for Database Querying. In: Aisbett, J. et al, *Strengthening the Role of ICT in Development.* Kampala, Uganda, 25 Nov 2008, Kampala: Fountain Publishers.

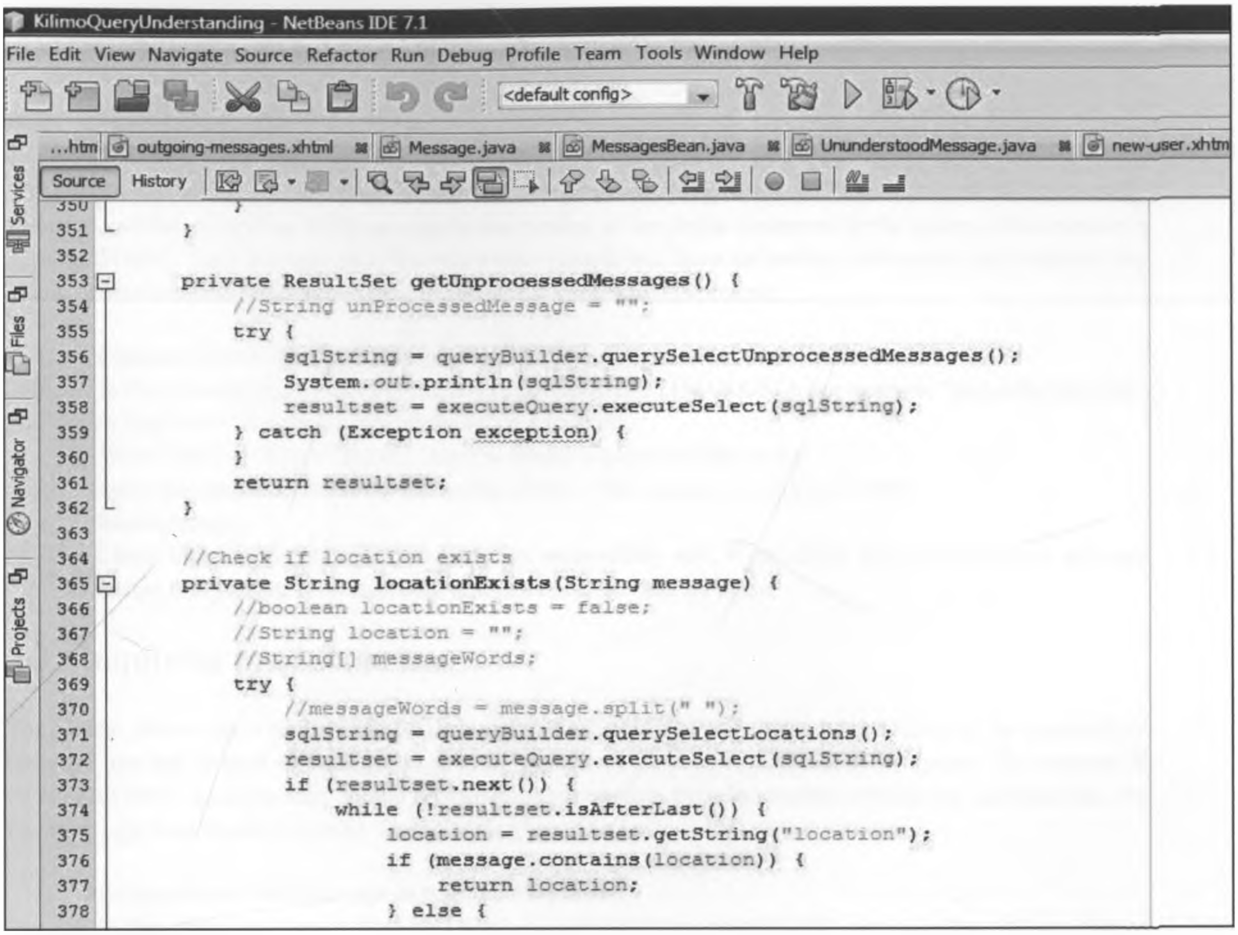
12. Nielsen, J.,1993. *Usability Engineering.* San Francisco: Morgan Kaufmann Publishers.

13. Safaricom, 2012. *Safaricom Prepaid.*[online] Available at : <http://safaricom.co.ke/index.php?id=146> [Accessed 13 March 2012].

14. Sauro, J., 2011. *A practical Guide to the System Usability Scale.* Denver: Createspace

8. APPENDICES

APPENDIX A : Sample Code



```
        }
    }

    private ResultSet getUnprocessedMessages() {
        //String unProcessedMessage = "";
        try {
            sqlString = queryBuilder.querySelectUnprocessedMessages();
            System.out.println(sqlString);
            resultset = executeQuery.executeSelect(sqlString);
        } catch (Exception exception) {
        }
        return resultset;
    }

    //Check if location exists
    private String locationExists(String message) {
        //boolean locationExists = false;
        //String location = "";
        //String[] messageWords;
        try {
            //messageWords = message.split(" ");
            sqlString = queryBuilder.querySelectLocations();
            resultset = executeQuery.executeSelect(sqlString);
            if (resultset.next()) {
                while (!resultset.isAfterLast()) {
                    location = resultset.getString("location");
                    if (message.contains(location)) {
                        return location;
                    } else {
```

Figure 21 : Sample code

APPENDIX B : User Guide

There are two types of users for this system. The major users who include rural farmers and traders. They use the system to place offers to sell or buy. Another type of users are system administrators, they provide a human touch in intervening with un-understood messages. They edit these messages so that they are understandable to the system. This user guide is divided into two parts and is presented in a functional manner.

8.1 Placing an offer to Sell

The system allows users to use both English and Kiswahili to query it. For an offer to sell to be successfully placed, a user has to send an SMS message to the number of the phone connected to the system. This number is +254722214690. Each message should contain three parameters, these are market, commodity and intention, the intention here is to sell. The following are the steps to complete this process;

i. Compose new SMS message in your mobile phone

ii. In the message type in [INTENTION] [COMMODITY] [MARKET]. For example "sell peas marigat" in English or "uza mbaazi marigat" in Kiswahili.
Word "sell" or "uza" signify that you intend to place an offer to sell.
iii. Input the number of the recipient mobile phone. This number is +254722214690
iv. Send message
v. Check that confirmation that message was successfully sent, if not check that you have units and also ensure that you are in an area with network coverage and try again.

8.2 Placing an offer to buy

The system allows users to use both English and Kiswahili to query it. For an offer to buy to be successfully placed, a user has to send an SMS message to the number of the phone connected to the system. This number is +254722214690. Each message should contain three parameters, these are market, commodity and intention, the intention here is to buy. The following are the steps to complete this process;

i. Compose new SMS message in your mobile phone
ii. In the message type in [INTENTION] [COMMODITY] [MARKET]. For example "buy milk kimende" in English or "nunua maziwa kimende" in Kiswahili.
Word "buy" or "nunua" signify that you intend to place an offer to buy.
iii. Input the number of the recipient mobile phone. This number is +254722214690
iv. Send message
v. Check that confirmation that message was successfully sent, if not check that you have units and also ensure that you are in an area with network coverage and try again.

8.3 Inquiring product price

The system allows users to use both English and Kiswahili to query it. For a price inquiry to be successfully placed, a user has to send an SMS message to the number of the phone connected to the system. This number is +254722214690. Each message should contain three parameters, these are market, commodity and intention, the intention here is to inquire for price. The following are the steps to complete this process;

i. Compose new SMS message in your mobile phone
ii. In the message type in [INTENTION] [COMMODITY] [MARKET]. For example "price chicken matuu" in English or "bei kuku matuu" in Kiswahili.
Word "price" or "bei" signify that you intend to inquire about price of a product.
iii. Input the number of the recipient mobile phone. This number is +254722214690
iv. Send message
v. Check that confirmation that message was successfully sent, if not check that you have units and also ensure that you are in an area with network coverage and try again.

8.4 Receiving matching offers to sell

After you have sent an offer to sell, the system will process it and respond to you with matching offers. All offers that match yours will be sent to you in SMS.

For example an offer like "sell beans moiben" will receive offers like these "beans needed in moiben call +254712326197,+254714521902" . This offer received two matching offers. Once you open the SMS, you will be able to see the phone numbers of the persons whose offer has matched yours. Use these numbers to call the potential buyers.

8.5 Receiving matching offers to buy

After you have sent an offer to buy, the system will process it and respond to you with matching offers. All offers that match yours will be sent to you in SMS.

For example an offer like “nunua viazi nakuru” will receive offers like these “viazi inapatikana kule nakuru piga simu +254733908456” . This offer received one matching offers. Once you open the SMS, you will be able to see the phone number of the seller. Use these numbers to call the potential sellers.

8.6 Editing un-understood messages

All un-understood messages are logged by the system. As a system administrator you are able to edit these message so that they are processed when the system runs next. Follow these steps to edit un-understood message.

i. Access the web interface using the url ***http://localhost:8080/Testers***

ii. Provide your valid username and password and click Login as shown on Figure 22

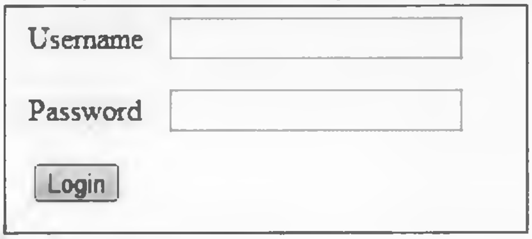
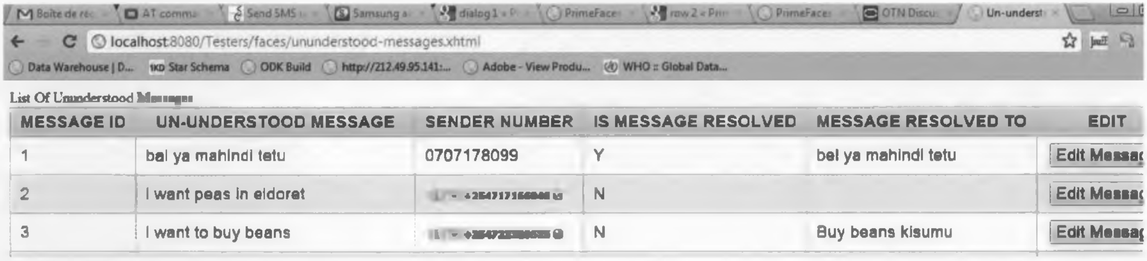


Figure 22 : Login window

iii. Open the un-understood messages window, see Figure 23 below



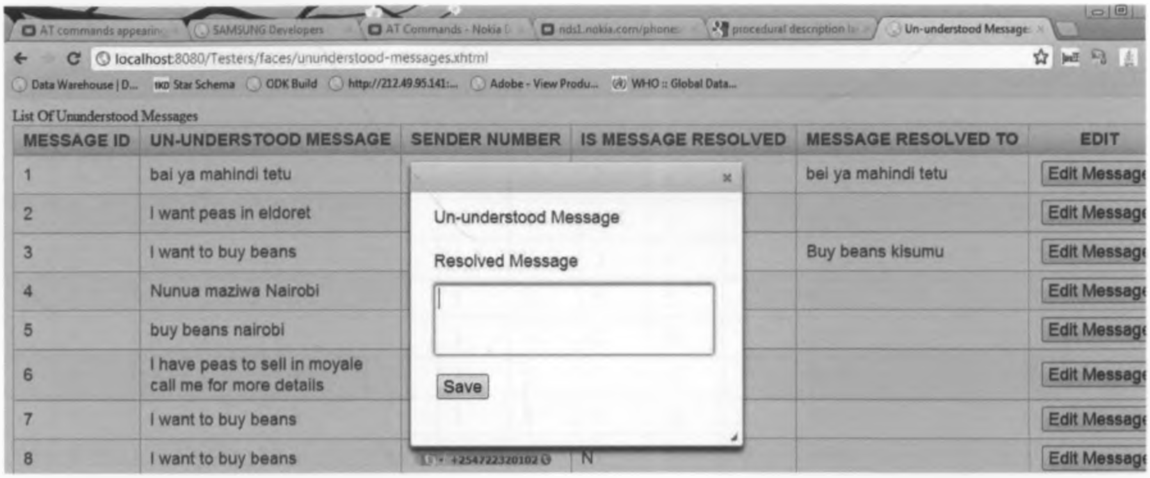
List Of Ununderstood Messages

MESSAGE ID	UN-UNDERSTOOD MESSAGE	SENDER NUMBER	IS MESSAGE RESOLVED	MESSAGE RESOLVED TO	EDIT
1	bal ya mahindi tetu	0707178099	Y	bei ya mahindi tetu	Edit Messag
2	I want peas in eldoret	[illegible]	N		Edit Messag
3	I want to buy beans	[illegible]	N	Buy beans kisumu	Edit Messag

Figure 23 : Un-understood messages window

iv. Click on a message you want to edit. Click on the row

v. Click on the Edit button, see Figure 24



List Of Ununderstood Messages

MESSAGE ID	UN-UNDERSTOOD MESSAGE	SENDER NUMBER	IS MESSAGE RESOLVED	MESSAGE RESOLVED TO	EDIT
1	bai ya mahindi tetu			bei ya mahindi tetu	Edit Messag
2	I want peas in eldoret				Edit Messag
3	I want to buy beans			Buy beans kisumu	Edit Messag
4	Nunua maziwa Nairobi				Edit Messag
5	buy beans nairobi				Edit Messag
6	I have peas to sell in moyale call me for more details				Edit Messag
7	I want to buy beans				Edit Messag
8	I want to buy beans	+254722320102	N		Edit Messag

Un-understood Message

Resolved Message

Save

Figure 24 : Editing un-understood messages

vi. Input the new message

vii. Click Save

APPENDIX C : Detailed SUS Table

	Question Response										
	1	2	3	4	5	6	7	8	9	10	SUS Score
User 1	5	1	4	1	5	1	5	1	4	1	95.0
User 2	5	2	5	1	5	1	4	1	5	1	95.0
User 3	4	1	5	1	5	1	4	1	5	1	95.0
User 4	4	1	5	1	5	1	5	1	5	1	97.5
User 5	5	2	5	1	5	1	5	1	5	1	97.5
User 6	5	1	4	1	5	2	5	1	4	1	92.5
User 7	5	1	5	1	5	1	5	1	5	2	97.5
User 8	5	2	4	1	5	1	5	1	5	1	95.0
User 9	5	1	5	1	4	1	5	1	5	1	97.5
User 10	5	1	4	1	5	2	3	1	5	1	90.0
User 11	5	1	4	1	5	1	5	1	5	1	97.5
User 12	5	1	5	1	5	1	4	1	5	1	97.5
User 13	4	1	5	1	5	1	5	1	5	1	97.5
User 14	4	2	5	1	5	2	3	3	4	2	77.5
User 15	5	1	5	1	5	1	5	1	5	2	97.5
User 16	5	1	5	1	5	2	5	1	5	1	97.5
User 17	5	3	5	1	5	1	5	1	5	2	92.5
User 18	5	1	4	1	5	1	5	1	5	1	97.5
User 19	5	1	5	1	5	1	4	1	5	1	97.5
User 20	5	2	5	1	5	1	5	1	5	1	97.5
User 21	5	1	2	1	5	1	5	2	5	2	87.5
User 22	5	1	3	1	4	1	5	1	5	1	92.5
User 23	5	1	5	2	5	1	5	1	5	1	97.5
User 24	5	1	5	1	4	1	5	1	5	1	97.5
User 25	4	1	5	1	5	1	3	2	5	2	87.5
User 26	3	1	4	1	4	1	5	2	4	2	82.5
User 27	5	2	5	1	4	2	5	1	4	1	90.0
User 28	4	1	5	1	5	1	5	1	5	1	97.5
User 29	4	1	4	1	5	1	4	1	2	1	85.0
User 30	5	2	5	2	5	2	5	2	5	1	90.0
User 31	3	1	4	1	4	1	5	1	4	2	85.0
User 32	5	2	5	2	5	1	4	1	5	1	92.5
User 33	5	2	4	1	5	2	4	2	3	2	80.0
User 34	5	1	4	1	5	1	5	1	4	1	95.0
User 35	5	1	5	1	3	1	5	1	3	1	90.0
User 36	4	2	5	1	4	4	5	2	5	3	77.5
User 37	4	1	5	1	5	1	5	1	5	1	97.5
User 38	4	1	5	1	5	1	5	1	5	1	97.5
User 39	3	1	5	1	5	1	5	1	4	1	92.5
User 40	4	1	4	1	3	1	5	1	5	2	87.5
User 41	5	1	5	2	5	1	5	1	5	1	97.5

User 42	4	1	4	1	5	1	4	1	4	1	90.0
User 43	5	1	5	1	5	2	5	1	5	3	92.5
User 44	5	2	5	2	5	1	5	1	5	1	95.0
User 45	5	1	5	1	4	1	5	1	4	1	95.0
User 46	5	2	5	2	5	1	5	1	5	1	95.0
User 47	5	1	5	1	5	1	4	1	5	1	97.5
User 48	4	2	4	1	4	1	5	1	5	2	87.5
User 49	5	1	5	1	5	1	5	1	5	3	95.0
User 50	5	1	5	1	5	1	5	2	4	1	95.0
User 51	5	2	4	3	4	1	5	1	5	1	87.5
User 52	5	1	5	1	5	3	4	1	5	2	90.0
User 53	5	2	5	1	5	1	5	1	5	1	97.5
User 54	4	1	5	1	5	1	5	1	5	1	97.5
User 55	5	3	5	1	5	1	5	1	5	1	95.0
User 56	5	1	5	1	4	1	5	1	4	1	95.0
User 57	5	1	4	1	4	1	5	2	5	1	92.5
User 58	5	1	5	1	5	1	4	2	5	2	92.5
User 59	5	1	5	2	5	1	5	1	5	2	95.0
User 60	5	1	5	1	5	1	5	1	4	1	97.5
User 61	5	1	5	1	4	1	5	1	5	1	97.5
User 62	5	2	4	1	4	1	5	1	5	2	90.0
User 63	5	2	5	1	5	1	4	1	5	1	95.0
User 64	5	1	5	2	5	1	5	1	5	2	95.0
User 65	5	1	5	1	4	1	5	1	5	1	97.5
User 66	4	1	5	1	5	1	5	1	4	1	95.0
User 67	5	1	4	1	5	1	5	1	5	1	97.5
User 68	5	1	5	1	5	1	5	1	4	1	97.5
User 69	4	2	5	1	5	1	5	1	5	1	95.0
User 70	5	1	5	1	5	1	4	1	5	1	97.5
User 71	5	1	4	2	5	1	5	1	5	1	95.0
User 72	4	1	5	1	5	1	5	1	5	1	97.5
User 73	5	1	5	1	5	2	5	1	4	1	95.0
User 74	5	1	5	1	5	1	5	1	4	2	95.0
User 75	5	1	3	1	5	1	5	1	5	1	95.0
								Average SUS			93.6

Table 6 : Detailed SUS table