2016-04

An integrated mobile application for enhancing management of nutrition information in Tanzania

Mduma, Neema

NM-AIST

https://doi.org/10.58694/20.500.12479/49

AN INTEGRATED MOBILE APPLICATION FOR ENHANCING MANAGEMENT OF NUTRITION INFORMATION IN TANZANIA

Neema Mduma

A Dissertation Submitted in Partial Fulfillment of the Requirements for the Degree of Master's in Information and Communication Science and Engineering of the Nelson Mandela African Institution of Science and Technology

Arusha, Tanzania

April, 2016

ABSTRACT

Malnutrition contributes to over one half of the deaths of children under age of five years in developing countries and is the single greatest cause of child mortality in Tanzania. Studies reveal that, the issue of malnutrition is aggravated by lack of nutrition information especially in rural communities. Absence of proper tools makes collection, management and access to nutrition information very difficult. This study improves accessibility of nutrition information by taking advantage of the advanced mobile technologies and develops a system for managing nutrition information. The system was implemented using a mixed approach involving qualitative techniques whereby the requirements and fact finding was done through interviews and literature review. Unified Modelling Language (UML) technique was used to design and model the user requirements and system specification. PHP, MySQL, XML and Java were used to complement the development of this system.

The developed mobile-based nutrition information management system was then integrated with existing Health Centre System and is able to provide a platform that gives mothers instant access to nutritional tips, allow them to interact with nutrition practitioners and help in record keeping. The results demonstrate the potential of using mobile technology for collection and delivering nutrition information in various sectors. In particular, this system could be adopted to improve prenatal and postnatal health in Tanzania and therefore help in bringing down the number of deaths of children under age of five.

DECLARATION

I, **NEEMA MDUMA** do hereby declare to the Senate of Nelson Mandela African Institute of Science and Technology that this dissertation is my own original work and that it has neither been submitted nor being concurrently submitted for degree award in any other institution.

Neema Mduma **13th April 2016**

Name and signature of candidate **Date**

The above declaration is confirmed

Dr. Khamisi Kalegele 13/04/2016

Name and signature of supervisor **Date**

COPYRIGHT

CERTIFICATION

The undersigned certify that has read and found the dissertation acceptable by the Nelson Mandela Institution of Science and Technology.

Dr. Khamisi Kalegele 	13/04/2016
Name and signature of supervisor	**Date**

ACKNOWLEDGEMENT

First and foremost, I would like to thank God the Almighty for He is good and has always been a source of courage and inspiration in my life.

I would also like to acknowledge the support of the government of Tanzania for granting me a scholarship to pursue the masters' degree studies at The Nelson Mandela African Institution of Science and Technology.

Special thanks go to my supervisor, Dr Khamisi Kalegele, for his great support and dedication in my research. It has been a great learning and working experience under the guidance of my supervisor.

I offer my sincerest gratitude to the Vice Chancellor of The Nelson Mandela African Institution of Science and Technology, Prof. Burton Mwamila, for encouraging me during my research work and allowing me to grow as a research scientist.

I would also like to give my special thanks to all my lecturers for their dedication in teaching us. We had a wonderful time and we have learned a lot from them.

I would like to sincerely appreciate the management of Green Hope Clinic at Arusha for having granted me a chance to conduct my study there. Without them, I wouldn't have succeeded to collect assessment data for my study.

I extend my special thanks to the management of The University of Iringa for giving me permission to study the master's degree and for supporting me during the whole period of my studies.

I acknowledge the support of my very best friend, Kanyika Mtaita, together with my colleagues, especially Ms Erica Kimei, Prisilla Ishabakaki, Sara Nyanjara and Luba Pascoe, for being cooperative from the start to end of my research.

Without forgetting, I would like to give special thanks to my parents, Rev and Mrs Mathias Mduma, together with Eng. and Mrs Simon Sumari, for all their support and for always being there to counsel and for their love.

DEDICATION

I dedicate this dissertation to my father and greatest friend, Rev Mathias Mduma. It always feels great to have you in my life. I dedicate this to you as a symbol of togetherness and infinite love.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF APPENDICES

LIST OF ABBREVIATIONS

ANC	Antenatal Care
CARE	Cooperative for Assistance and Relief Everywhere
DFD	Data Flow Diagram
ECD	Early Childhood Development
FSNA	Food Security and Nutrition Assessment
HMIS	Health Management Information System
HPHB	Healthy Pregnancy Healthy Baby
MAMA	Mobile Alliance for Maternal Action
MDG	Millennium Development Goals
MTUHA	Mfumo wa Taarifa za Uendeshaji Huduma za Afya
PHP	Hypertext Preprocessor
RAD	Rapid Application Development
RVA	Rapid Vulnerability Assessment
SSH	Secure Shell
SSL	Secure Sockets Layer
SDLC	Software Development Life Cycle
TCRA	Tanzania Communications Regulatory Authority
TDHS	Tanzania Demographic and Health Survey
UNICEF	The United Nations Children's Fund
USSD	Unstructured Supplementary Service Data
XML	Extensible Markup Language

CHAPTER ONE

Introduction

This chapter presents a general introduction of the study which mainly focuses on background information, research problem and justification of study, general and specific objectives of the research, research questions and significance of the research.

1.1 Background information

Mobile communication is one of the fastest growing technologies in use today (Cole-Lewis & Kershaw, 2010). This technology is perceived as a powerful and potential tool for information dissemination due to its affordability and convenience (Muthee & Mhando, 2006). Mobile phones support work in health education, disease prevention, treatment of disease, health care, and could potentially be used to overcome some of the traditional barriers in accessing the necessary information and services for social benefits (Khan, Yang, & Kahn, 2010).

Despite the improvements under Millennium Development Goals (MDGs), many developing countries are still facing some challenges. Malnutrition is one of the greatest challenges facing Tanzania. Over the past decade, over 600,000 children aged below 5 years are estimated to have died as a result of inadequate nutrition. In 2010 alone, 43,000 children died prematurely because of malnutrition. That averages to one child dying every 12 minutes; this involves the combined factors including lack of nutrition knowledge and understanding of the association between food and health, inadequate feeding practices, poor health care and hygiene and to a lesser extent food insecurity (Tanzania Bureau of Statistics and Macro International & (Measure DHS), 2010).

The issue of malnutrition is aggravated by lack of nutritional information especially in rural communities. Lack of appropriate tools makes the collection, management and access to nutritional information very difficult. It has been argued that there is huge potential for mobile-health interventions to have beneficial effects on health and health service delivery processes, especially in developing countries such as Tanzania where there are poor resource settings (Aker & Mbiti, 2010).

Taking advantage of the growing mobile technology, this research proposed and developed a mobile-based information management platform which is integrated with existing health information systems that provide opportunities to improve accessibility of nutrition information. The platform will give mothers instant access to nutritional tips, allow them to interact with nutrition practitioners and help in record keeping.

1.1.1 Nutrition situation in Tanzania

Nutrition is the process of providing nutrients for growth and health through eating, and subsequent absorption and utilization by the body. Like other developing countries, the major nutrition problem in Tanzania is under nutrition. Approximately 22% of children are underweight (low weight-for-age), which is a composite measure of long and short term under nutrition. This last indicator is one of the Millennium Development Goals' (MDG) indicators, which aims to halve the prevalence of underweight children under five years of age in Tanzania from 28.8% (1990 baseline) to 14.4% by 2015 (United Nations, 2006).

Despite improvements over the past years, the rate of malnutrition among Tanzanian children remains high. Malnutrition is also one of the largest causes of death of children under five years. It was estimated that, there are 130 children deaths every day (UNICEF Tanzania, 2011) and half of these deaths are attributable to malnutrition (Caulfield Laura, de Onis Mercedes, Blössner Monika, & Black Robert, 2004). It was reported that, there is a lack of nutrition information especially in rural communities and strategies to improve situation must be done (Leach & Kilama, 2009). Nutrition information must be accurately provided and feedback must be well received so as to make communication effective.

The use of appropriate technologies such mobile phones, which are used by many people, can address the issue of malnutrition in supporting individuals and communities throughout the country to meet nutritional needs by providing timely and accurate information (Ministry of Health and Social Welfare, 2011). This research intends to use an integrated mobile application for enhancing management of nutrition information in terms of collection and dissemination of nutrition information to improve prenatal and postnatal health in Tanzania.

1.1.2 Mobile technology in Tanzania

Mobile technology is the technology used for cellular communication. In the context of this study, it refers to the tool used for enhancing management of nutrition information. In Tanzania, there have been a wide spread use of mobile technology in various sectors including health and it has proven to be a vital medium for accessing and dissemination of information.

According to TCRA, The number of Internet users rose 22 percent to 11.35 million from 2013 and the number of mobile phone subscribers in Tanzania rose by 16 percent in 2014 to 31.86 million which marking further growth in the communications sector as the fastest

expanding in the economy (Tanzanian mobile phone subscribers jump by 16 pct in 2014 | Reuters,2015).

The rapidly growing number of internet users is attributed to the technological revolutions and improvement. Cost reductions and the increasing number of internet enabled devices like mobile phones have also attracted more subscribers to the service. Although computer and internet penetration in most African countries like Tanzania is still very low, the mobile phone is increasingly becoming key entry point for internet adoption in which majority of people currently access internet for the first time in their mobile phones (Mtweve, 2014).

It was reported that mobile phones appear mostly compatible to work in health and there is a growing body of evidence that demonstrates the potential of mobile communications to radically improve healthcare in some of the most remote and resource-poor environments (Vital Wave Consulting, 2009). This study, therefore uses this technology to develop an integrated mobile application for enhancing management of nutrition information in Tanzania.

1.2 Research problem and justification of study

Currently, the issue of malnutrition in Tanzania is aggravated by poor access to appropriate nutrition information especially in rural communities. Access is poor because most information is stored offline and not in digital form. And there are no means for online access to available nutrition information, which is largely not integrated. Mobile technology can be used to tackle these challenges.

However, it was difficult to use the technology in the past because of poor mobile phone penetration and readiness. As a result, there has been very few mobile technology based nutrition information management systems, which match our context. Recently, mobile phone penetration and readiness have significantly increased and therefore it is anticipated that nutrition information management systems that are designed according to the current socio-economic (affordability) and physical environment (connectivity) realities have the potential to bring about real change. This research proposes the use of an integrated mobile application for enhancing management of nutrition information in terms of collection and dissemination of nutrition information in order to improve prenatal and postnatal health in Tanzania.

Nutrition information needs to be easily accessed and shared to improve service delivery and influence change in nutritional status. Access to different information contributes to poverty reduction and improved quality of life, especially for people living in rural areas (Sife, Kiondo, & Lyimo-Macha, 2010). The difficulty of access to information about nutrition greatly affects mothers and children particularly those living in remote areas. Nutrition information is provided verbally only with no emphasis and there is no nutrition information handling which is done. Developing a mechanism that will contribute towards achieving accessibility of nutrition information is therefore a necessity. In response to this, this research proposed and developed an integrated mobile application for enhancing management of nutrition information in Tanzania.

1.3 Objectives

1.3.1 General objective

The main objective of this research is to enhance collection and delivery of nutrition information by using an integrated mobile application that will facilitate an interactive flow of nutrition information between nutrition practitioners, researchers and patients especially mothers.

1.3.2 Specific objectives

1. To identify and establish collection and information delivering requirements between nutrition practitioners, researchers and patients especially mothers.

2. To design the mobile application for collection and information delivering among actors.

3. To implement and integrate the developed system with a health centre system.

1.4 Research questions

1. What are the requirements for developing an integrated mobile application for nutrition information management?

2. What are the design requirements of an integrated mobile application for management of nutrition information?

3. What is the implementation process and requirements for an integrated nutrition information management mobile application?

1.5 Significance of the research

The goal of this research is to address effective collection and management of nutrition information by ensuring timely access and better delivery of nutrition information. The mobile application will improve access to nutrition information by facilitating an interactive flow of nutrition information. And it will bridge the information gap by simplifying report process which will lead to effective planning, decision making and policy formulation with regard to nutrition information management using mobile phones as appropriate tools for simplifying collection, management and access to nutritional information. This research provides a possible solution for minimizing the death of children below five years, which are aggregated by poor management of nutrition information.

CHAPTER TWO

ENHANCING MANAGEMENT OF NUTRITION INFORMATION USING MOBILE APPLICATION: PRENATAL AND POSTNATAL REQUIREMENTS

Neema Mduma, Khamisi Kalegele

School of Computation and Communication Science & Engineering

Nelson Mandela African Institution of Science and Technology

Arusha, Tanzania

mduman@nm-aist.ac.tz, khamisi.kalegele@nm-aist.ac.tz

Abstract— *Malnutrition contributes to over one half of the deaths of children under age of five years in developing countries and is the single greatest cause of child mortality in Tanzania. Investigations reveal that, the issue of malnutrition is aggravated by lack of nutritional information especially in rural communities. Absence of proper tools makes collection, management and access to nutrition information very difficult. The aim of this study is to improve accessibility of nutritional information by taking advantage of the advanced mobile technologies to integrate a mobile-based information management platform with existing Health Information Systems. The platform will give mothers instant access to nutritional tips, allow them to interact with nutrition practitioners and help in record keeping. In this paper, we present the requirements and initial analysis and design of a mobile application for managing prenatal and postnatal nutritional information. The requirements have been established from interviews with the various stakeholders and literature review. The established requirements become a necessary input towards development of a complete mobile-based nutrition information management platform, which is to be integrated with existing health information systems.*

Keywords— *Mobile application, nutrition information, requirement specification, DFD*

2.1 Introduction

Nutrition information provides general knowledge for understanding the associations between food and health. Managing this information plays a critical role in improving health care provision at facilities. However, most information in Tanzania is stored offline and there are no means for online access to available nutrition information which is largely not integrated. Hence, designing and developing an integrated mobile application to enhance management of nutrition information was inevitable. Designing and developing a system requires determination and analysis of the system's requirements.

Requirements specification process highly depends on the knowledge to allow communication between the user and system developer to develop the right system (Cafer & Misra, 2009); it serves as the basis for the design, implementation, testing and maintenance of the target system. When developing the mobile application for managing nutrition information, requirement analysis as the first stage in the system development must be well documented to generate the desired specifications. Unfortunately, data models of most systems are poorly designed. As a result, users are flooded with information when using the system. In our study, we considered this challenge to design a more effective data model by categorizing nutrition information into tips and recommendations.

This paper presents functional and non-functional requirements as well as design for developing an integrated mobile application for enhancing management of nutrition information in Tanzania. The analysis is based on a combination of both object oriented and structured analysis.

2.2 Management of nutrition information

Information about intake of food in relation to the body's dietary needs must be well managed to maintain good health. Management of this information involves a diet of what people eat, which is largely determined by the availability, processing and palatability of foods (World Health Organization, 2013). Furthermore, managing the nutrition information plays a critical role in collecting, maintaining, and analysing the information to improve health care services delivery (Sharon Silow-Carroll, Jennifer N. Edwards, & Diana Rodin, 2012).

2.2.1 Literature review

Various studies have been done to collect basic information about this study. Review of different studies related to mobile technology for enhancing management of nutrition information was conducted. Likewise, the empirical literature helped scrutinizing other alternatives to solve the problem of inefficient management of nutrition information.

MTUHA/HMIS, TDHS, FSNA/RVA and MUCHALI are some of the information systems that are used for collecting and managing nutrition information. These systems collect only a subset of the key nutrition indicators. Major nutrition indicators such as optimal infant feeding indicators have not been considered. Moreover, there are weaknesses regarding collection, management and use of nutrition data, including limited access, dissemination and sharing of nutrition information (Ministry of Health and Social Welfare, 2013).

2.2.2 The challenges

In Tanzania, nutrition information is normally verbally provided and it is not stored in digital form. This makes it difficult for users especially mothers who get nutrition information from clinics. Furthermore, most nutrition information is not categorized in such a way that sometimes users especially mothers are flooded with unnecessary information. During the clinic visits, nutrition information is provided to the group of mothers without considering the fact that some mothers may need special attention, such as having high blood pressure, and thus require special kind of nutrition attention related to their situation.

These challenges can be overcome by designing and developing an integrated mobile application for collection and dissemination of nutrition information. In this application, nutrition information is categorized into tips and recommendations to support analysis.

2.2.3 Other related work

CARE developed mobile phone applications for early childhood development (ECD) as part of their efforts to tackle malnutrition. This application was designed to connect people in isolated communities and with information and skills to assess, protect and improve the health of children (CARE, 2014). However, the limitation of this study is that there is no single magic bullet for nutrition and early childhood development.

Mobile Alliance for Maternal Action (MAMA) uses technology to improve health and nutrition outcomes among pregnant women and new mothers and their infants in resource poor settings. It empowers women in low resource settings to improve and protect their own

health and that of their children and families. MAMA uses mobile technology to deliver time sensitive, stage based information on critical health issues directly to expectant and new mothers (Mobile Alliance for Maternal Action, 2012).

The Tanzania Healthy Pregnancy, Healthy Baby Text Messaging Service (HPHB), otherwise known as Wazazi Nipendeni, offers free maternal and early childcare health information to subscribers of all networks. The service seeks to assist health professionals in the dissemination of information typically shared during antenatal care (ANC) visits (GSMA, 2015). This study uses SMS as delivery channel and SMS recipients cannot respond to messages, which would be beneficial for answering questions and collecting data.

2.3 Methods for requirements analysis

The study used qualitative research methods such as interviews where by casual talks were conducted for the collection of information. This method provided us with more useful information for our study. We interacted with the nutrition practitioners together with mothers, asking some questions and noting down important data for system requirements specification. Qualitative research consists of an investigation that seeks answers to a question (Silva, 2008). Other methods like literature review were also used and we found various works related to our study.

2.3.1 Requirement specification

In this study, the requirement specification involves functional and non-functional requirements. Functional requirements capture the intended behaviour of the system which may be expressed as services, tasks or functions the system is required to perform (Bredemeyer, 2001). Non-functional requirements capture required properties of the system and show how structural aspects of the system should be accomplished. A tip is a set of nutritional information concerning nutrition improvements added by nutrition practitioners for the user. The key items which make up a tip are shown in Table 1 below.

Table 1: Tip key items

ITEM	**DATA TYPE**
Title/Subject	Text
Date posted	Date/Time
Tip detail	Text
Author	Text
Category	Text
Rating	Text

A recommendation is a set of nutritional information that is suggested by nutrition practitioners to users based on the user's described information. The key items which make up a recommendation are shown in Table 2 below.

Table 2: Recommendation key items

ITEM	**DATA TYPE**
Title/Subject	Text
Recommendation detail	Text
Nutrition practitioner	Text
Date replied	Date/Time

The requirements for developing an integrated mobile application for enhancing management of nutrition information have been specified. Table 3 below shows functional requirements and Table 4 shows non-functional requirements of the system.

Table 3: Functional requirements for nutrition management information system

REQUIREMENTS	**DESCRIPTION**	**ACTOR**
Add recommendation	The nutrition practitioner is the one who can add recommendations for the user based on nutrition status.	Nutrition practitioner
Add nutrition tip	Nutrition tips will be added to the system by nutrition practitioners based on request from a user and normal addition without request from the user.	Nutrition practitioner, Researcher
Edit recommendation	Recommendations will be edited by the specific nutrition practitioner who provided it to the user.	Nutrition practitioner
Edit nutrition tip	The nutrition practitioner is the one who can edit the specific provided nutrition tips to the system.	Nutrition practitioner
Search user	The nutrition practitioners will be able to search users based on user's ID.	Nutrition practitioner
Generate report	The researcher is the one who can generate reports based on necessary nutrition information provided by the system.	Researcher
View nutrition tip	The user of the system will be able to view nutrition tips provided to the system.	User
Request new tip	In case specific nutrition tip is not available in the system, the user can request it.	User
View recommendation history	Recommendations history will be viewed by the user and nutrition practitioner.	User, Nutrition practitioner
Give reminder	The user is the one who will receive reminders as notifications based on required situation.	User
Register user	The issue of registration will involve all actors of the system.	User, nutrition practitioner, researcher, system administrator
Approve user	The system administrator is the one who will approve user of the system	System administrator

Table 4: Non-functional Requirements for Nutrition Management Information System

REQUIREMENT	DESCRIPTION
Maintainability	• The system will allow upgrading to smartphone usage operation when needed. This can be done by other researchers.
Operability	• The system will interface with SQL database. • The system will be written in PHP. • The system will be supported by android, which is easily accessed.
Performance	• The system will provide enough time for the user to access and interact with the system.
Security	• The system will allow authorized users to access their confidential information. These users will provide username and password to login to the system.

2.3.2 Use case modelling

Table 3 above shows the interactions between the actors (nutrition practitioner, user, researcher, system administrator) and the nutrition management information system. As part of the modelling process, the functionality of the system has been defined using use cases shown in Table 4 above. Use case describes the proposed functionality of the system and interactions between external actors and the system under consideration (Sparx systems, 2004). A complete set of use cases specifies all different ways of using the system, and thus defining all behaviours required by the system. At a high level of abstraction, the entire functionality of the nutrition management information system is captured in the conceptual use case diagram shown in Figure. 1 below.

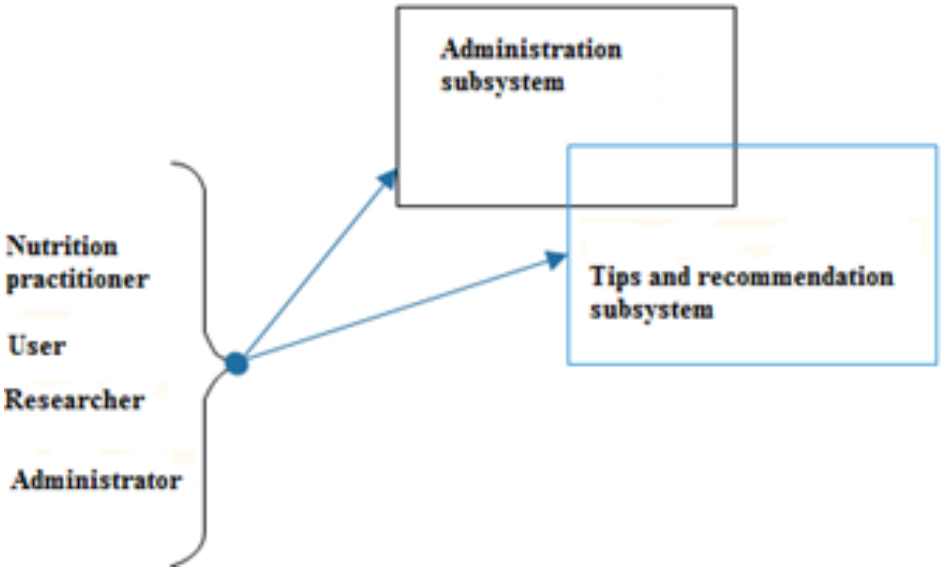

Figure 1: Conceptual use case diagram shows functional requirements

2.3.3 Domain modelling

A model of the domain is mapped to requirements that prescribe further system development (Thalheim Bernhard, Schewe Klaus-Dieter, & Ma Hui, 2009). This makes it important for us to describe key aspects of the system and how it fits within the domain of its operation.

The envisioned application is expected to be operating in Tanzania. Any person with a mobile phone and who wishes to get access to nutrition information will be allowed to get access. This application will be integrated with existing health centre systems to facilitate an interactive flow of information by supporting access, dissemination and management of nutrition information.

We have set our focus on maternal health and so the nature of end users can easily be anticipated. Generally, this application is expected to be used in an environment that accommodates users of different levels of literacy, sex and financial status. According to National Telecommunication reports, more than 70% of people in Tanzania have access to mobile phones making mobile phones the best tool for issuing nutrition information to majority of people in the country.

For this application, it is obvious that a genuine source of nutrition information needs to be modelled. We have modelled a nutrition practitioner to be the right source of information. This is planned this way to make sure that end users of the system get access to correct information. Both the end users and nutrition practitioners are modelled as not having expertise in information technology. We have also modelled the researcher to generate reports based on nutrition information and system administrator to monitor overall activities of the system including user approval. Figure. 2 shows the conceptual domain model of nutrition information system.

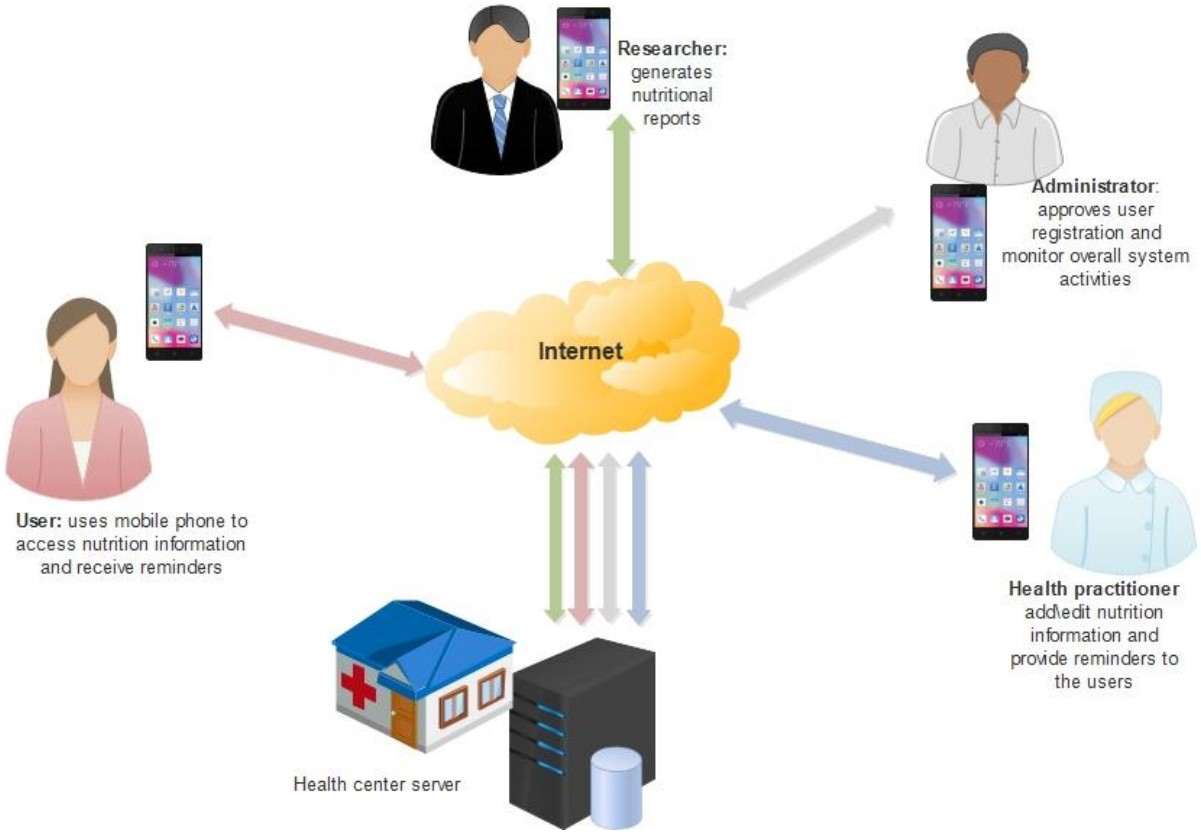

Figure 2: Conceptual domain model of the system

2.3.4 Data flow diagram

Data flow diagrams (DFD) reveal relationships among and between the various components in a system. Entity, process, data store, and data flow are basic components that illustrate how data flows in a system (Donald S. & Le Vie, 2000) . DFDs are an important technique for modelling a system's high-level detail by showing how input data is transformed to output results through a sequence of functional transformations. Figure. 3 shows administration management data flow diagram and Figure. 4 shows tips and recommendation management data flow diagram.

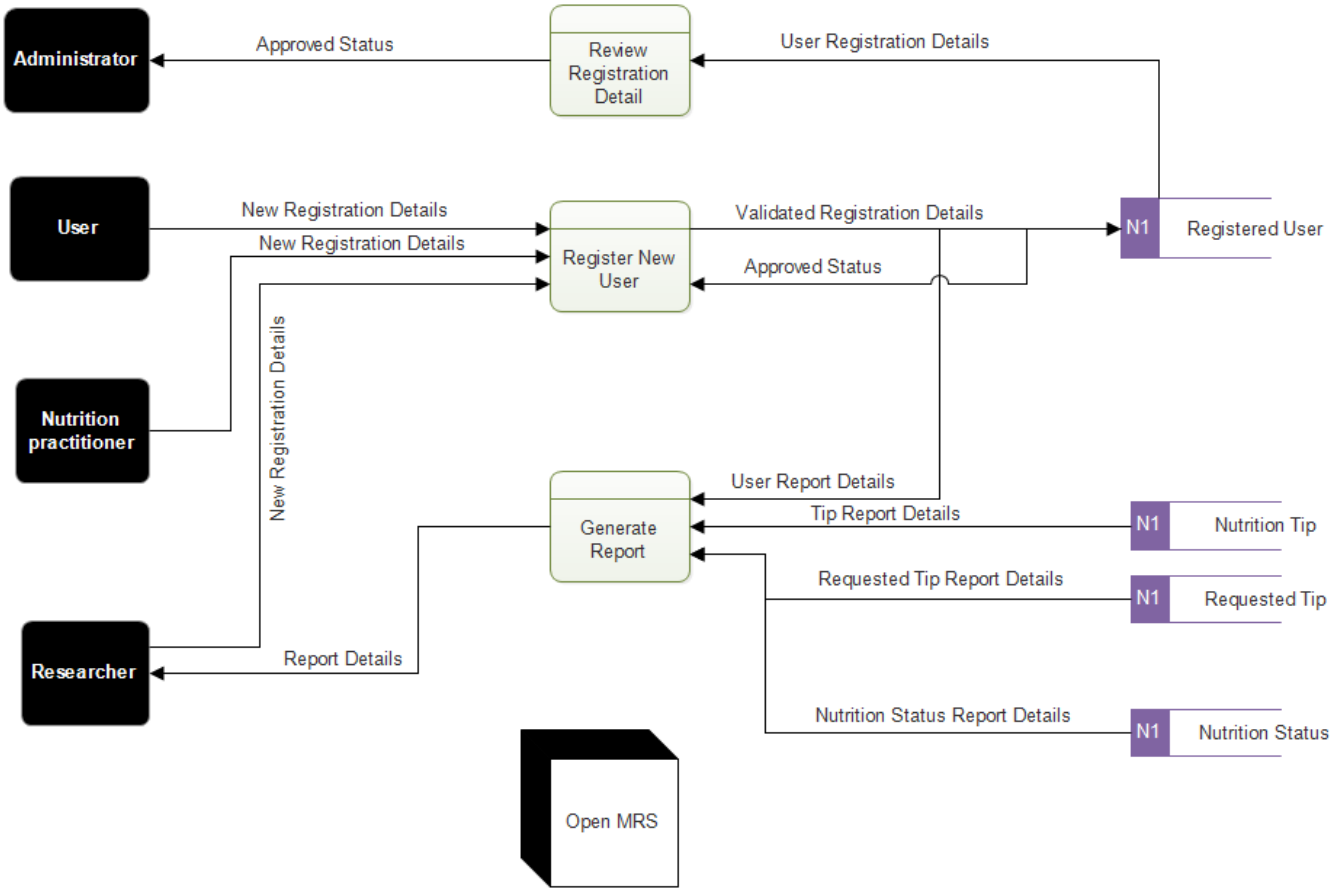

Figure 3: Administration data flow diagram

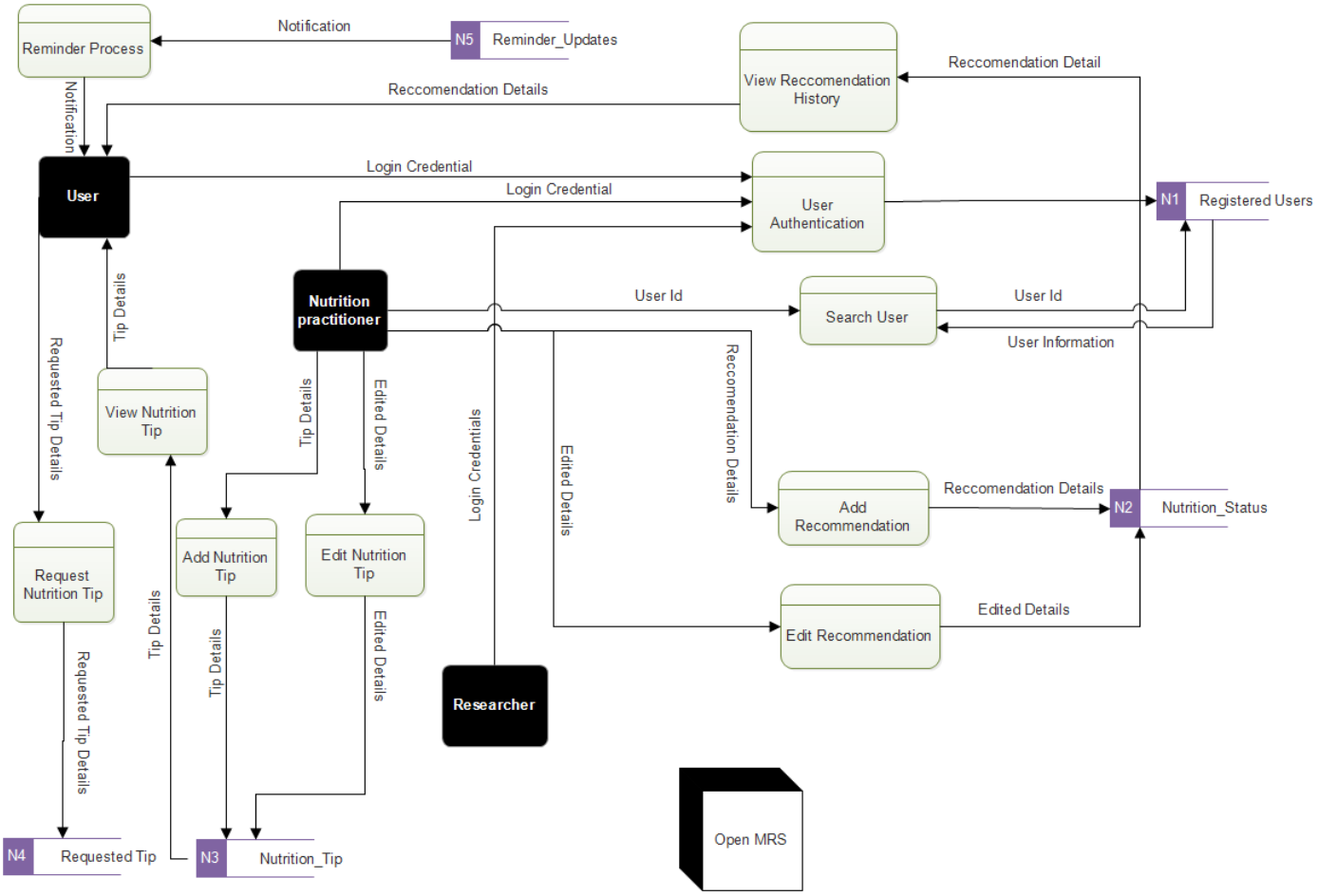

Figure 4: Tips and Recommendation data flow diagram

2.4 Discussion

From the survey, we discovered that nutrition information in Tanzania is only verbally provided and there is high possibility for the mothers to forget what they have been recommended once they leave the clinic. There is also the problem of mothers forgetting important events for clinic visit; this is because there is no mechanism of reminding them about important events such as clinic visit for vitamin A supplements. The problem of getting information and research findings by researchers when conducting their studies on nutrition was another identified problem. As a result, a good system for management of nutrition information needs to be restructured.

2.5 Conclusion

The mobile application for management of nutrition information whose specifications have been presented in this paper provides simple but efficient means of managing nutrition information. We also see how system requirements gathering and design are very critical in

any software development to make sure that the project solves the right problem using the right approach. Findings show that, there is poor access to nutrition information which is mostly stored offline. Results presented are a foundation for the design and development of an integrated mobile application for enhancing management of nutrition information, which will also be improved through the feedback from the end users during testing.

CHAPTER THREE

AN INTEGRATED MOBILE APPLICATION FOR EHNANCING MANAGEMENT OF NUTRITION INFORMATION IN ARUSHA TANZANIA[1]

Neema Mduma, Khamisi Kalegele

School of Computation and Communication Science & Engineering

Nelson Mandela African Institution of Science and Technology

Arusha, Tanzania

mduman@nm-aist.ac.tz, khamisi.kalegele@nm-aist.ac.tz

Abstract *— Management of nutrition information is still a problem in many developing countries including Tanzania because most information is stored offline and there are no means for online access to available nutrition information, which is largely not integrated. This paper discusses the development of an integrated mobile application for enhancing management of nutrition information based on the context of northern Tanzania, basically Arusha region. In this region, there is poor access to appropriate nutrition information especially in rural communities. The system used the PHP technique to build the application logic and MySQL technology for developing the back-end database. Using XML and Java, we have built an application interface that provides easy interactive view.*

Keywords- *Nutrition information, MySQL, XML, Java, PHP, Mobile Application*

[1] Published in the International Journal of Computer Science and Information Security (IJCSIS) Vol. 13 No. 7, July 2015

3.1 Introduction

Nutrition information provides the general knowledge about intake of food in relation to the body's dietary needs. This information delivers understanding of the associations between food and health. Therefore, managing nutrition information plays a critical role in improving health care provision at facilities.

The use of mobile technology in Tanzania health sector has been significantly increasing over the recent years, especially mHealth. This technology directly aims at the general public through engaging users in health related activities, and thereby improving accessibility to quality health information, health services, and encouraging user behaviour that involves seeking preventive health solutions (Rick van Genuchten, Wouter Haring, Daan van Kassel, & Kaoutar Yakubi, 2012). The widespread use of mobile phones has led to significance increase in mobile applications for providing access to various information that are needed by the community.

This paper presents the implementation process of the developed system. The system allows user interaction and supports an interactive flow of nutrition information. The proposed system is in the form of an integrated mobile application, which is designed for enhancing management of nutrition information in Tanzania.

3.2 Requirements of the system

The functional requirements for developing this mobile application covers the issues of recommendations as a set of nutritional information that are suggested by nutrition practitioners to the user based on the user's described information, and nutrition tips as the set of nutritional information concerning nutrition improvements added by nutrition practitioners for the user. The functional requirements also include a reminder as notification provided to users based on necessary nutrition events and reports that are generated by the researcher based on the nutrition information provided. The non-functional requirements of the system cover the issues of maintainability, operability, performance and security of the system. The system is expected to provide an integrated solution for management of nutrition information in Tanzania, which is experiencing the challenge of poor access to appropriate nutrition information because there is no digital form of storing the information, and no means for online access to available nutrition information, which is largely no integrated.

3.3 The proposed system design

The proposed system allows nutrition practitioners to send information to the targeted user. In this aspect, the users access nutrition information and request any other nutrition related details when necessary. The system provides reminders so as to notify users on necessary events such as clinic visits for vitamin A supplements. Moreover, this system involves the researcher who generates nutrition reports based on provided information, and the administrator for monitoring the overall activities of the system.

To accomplish the implementation part of the proposed system, a model adopted from SDLC has been chosen for developing a successful information system. The software development life cycle (SDLC) is a framework that defines the tasks performed at each step in the software development process. It consists of a meticulous plan that describes the processes for developing, maintaining, replacing and altering the specific software. The SDLC defines the method for software quality enhancement and the overall development process (Schwaber, 2001). To make the complete product to deliver faster, we decided to use the Rapid Application Development (RAD) model.

The RAD is a model designed to facilitate much faster software development and provides higher quality results compared to the traditional lifecycle; this model delivers faster and high quality product (Hirschberg, 2015). In this study, we preferred to use RAD as it proved to be a successful tool for developing our mobile application. Figure. 5 shows the Rapid Application Development model of our system.

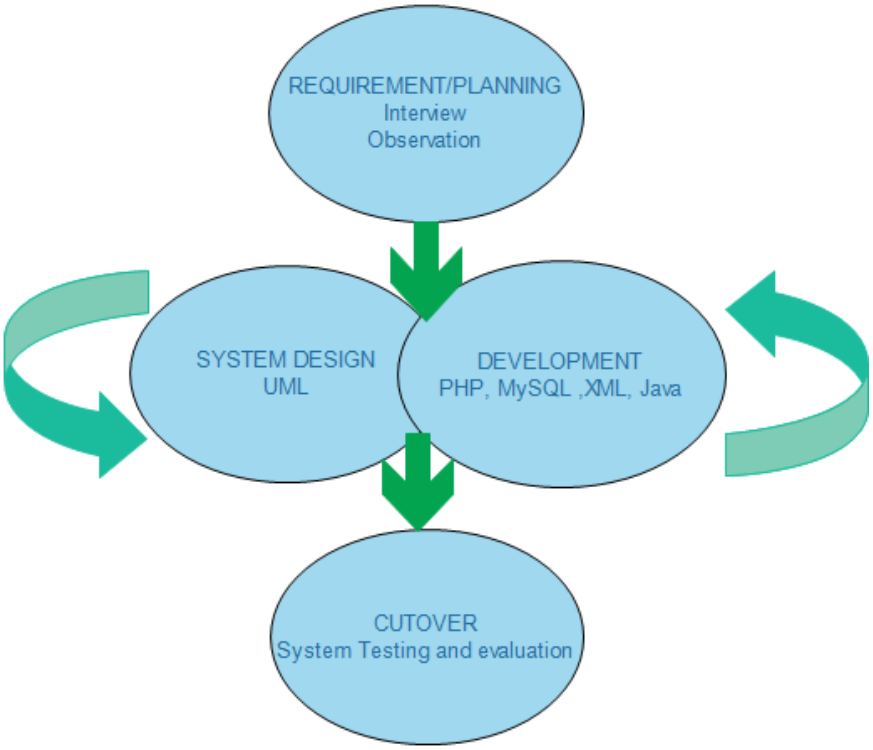

Figure 5: The system's rapid application development model

3.3.1 Android

Android is one of the most powerful and flexible open source platforms and it's increasingly becoming popular. There are no licensing fees; this increases preference by many developers. In this study, we preferred developing the mobile application supported by this operating system with consideration of market terms. The growth of mobile devices such as mobile phones is a worldwide phenomenon with mobile phone ownership outstripping computer ownership in many countries. Also, there is an increase in smart phones, which created a growth market for advanced mobile applications (Meier, 2010).

3.3.2 PHP

In developing the mobile application, we used the Hypertext Pre-processor (PHP) because this is one of the server-sided languages widely-used in software development and is an open source scripting language that we found appropriate for developing our system. PHP was preferred in this system development study because it is simple and thus easy to learn. It efficiently runs on the server side and its codes runs faster due to the fact that it runs in its own memory space so it has a fast loading time. The PHP has tools that are open source software, and thus are freely available for use. Furthermore, it is flexible for database connectivity and it supports a wide range of databases. Additionally, the PHP can connect to

a number of databases, but MySQL is the most commonly used as it can also be used at no cost (Pitts Monica, 2015).

In addition, PHP is compatible with almost all servers and its security features allow many functions to protect users against certain attacks. This language runs on various platforms such as Android, Windows and many others.

3.3.3 MySQL

MySQL is one of the database systems that run on a server and uses the standard Structured Query Language (SQL). It is easy to use, reliable and it runs very fast. In this study, we used MySQL Database so as to enable the cost-effective delivery of reliable and high-performance application. The data in a MySQL database are stored in tables and offers a flexible programming environment (Heng Christopher, 2010). Database systems are vital in computing and can be used as standalone utilities or as part of other applications.

The MySQL database server provides the ability to handle applications that are deeply embedded and offers platform flexibility; this is a MySQL stalwart feature. It allows customization so it is easy for a programmer to improve the database server by adding unique features. MySQL has been used by many database professionals due to the unique storage-engine architecture that allows configuration of the database server remarkable end results performance in particular applications. Apart from that, MySQL offers a variety of unique high-availability database server options ranging from high-speed master/slave replication configurations, specialized cluster servers offering instant failover, to third party vendors. So it provides high availability for programmers to rely on it.

MySQL protects data through its outstanding security features; it has powerful mechanisms, which ensures that access to the database server is possible only to authorized users and other users are limited to the client machine level. MySQL also has granular object privilege framework, which ensures that users can only see what they are supposed to see. Another important feature is that it has powerful data encryption and decryption functions, which protects sensitive data from unauthorized users. Secure Shell (SSH) and Secure Sockets Layer (SSL) are provided to ensure safe and secure connections. It also provides backup and recovery utilities so as to allow complete logical and physical backup, and also full and point-in-time recovery. MySQL offers full support needed for development of applications and

developers can get all they required for developing information systems that are based on databases (MySQL, 2015).

3.3.4 XML

Extensible Markup Language (XML) is designed to describe data (Hall Marty & Brown Larry, 2001). This language is used as a medium for carrying information independently from the involved software and hardware of the information system. Through the XML, you can create information formats and structured data can be shared electronically. XML data is self-describing, which means the data and its structure are embedded replacing the need for pre-building the structure for storing the data when it arrives. XML allows sharing of information in a consistent way due to its simpler format (Rouse Margaret, 2001).

XML has good features for storing and transmitting information, which simplifies data storage and sharing. This language is useful in accurately describing and identifying information without mistake so as to allow information to be understood (Tutorialspoint, 2014). Standardized description and control of particular types of document structure is possible in XML. It provides messaging systems' common syntax to facilitate information exchange between applications. In this study, we decided to use XML because it is free so we don't need to pay and it is easier to upgrade without losing data.

3.3.5 Java

Java is a programming language and computing platform that is designed to support many applications to work. This language is fast, secure, and reliable so as to assure developers about performance, stability and security of the developed application (Eck David, Hobart, & Colleges William, 2014). In this study, we decided to use Java because it is platform independent so applications can run on many different types of devices such as computers and even mobile phones. Java is essentially made up of objects, which are programming elements, and therefore it is object-oriented (Lowe Doug & Burd Barry, 2007). This language is very simple, so it is easier for the developer to engage it in application development.

3.4 Mobile Application

Mobile application is a type of application software that takes advantage of the mobile technology, and it can be used with any other technology apart from mobile phones (Muthee & Mhando, 2006). The numerous functions and services offered prompt the extensive use of the mobile applications. In this paper, we use android mobile application in order to

distinguish with other Unstructured Supplementary Service Data (USSD) applications that provide limited information and don't support storage of provided information. The reason is to provide an interactive flow of information by supporting interaction and allow access of large amount of information.

Designing an interface is described as the process of developing a method in a system to connect and communicate so as to allow exchange of information. This acts as a channel of communication between users and the application. Interface design focuses on anticipating what users might need to do and ensuring that the interface has elements that are easy to access, understand, and use to facilitate those actions (Usability, 2014).

First, the system administrator will register users by approving their registration requests as no one can use the system without registration. Users will be using mobile phones to access this application. The mobile application user interface is presented in the Figure. 6 below.

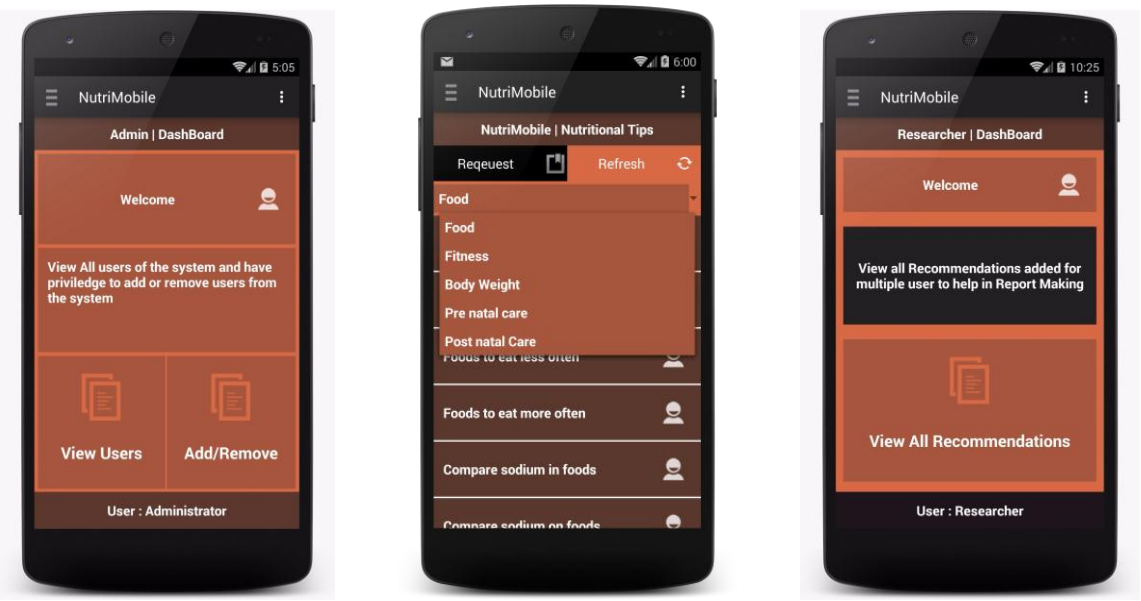

Figure 6: Mobile application user interface

3.5 Discussion

Implementation of an integrated mobile nutrition information management system was undertaken in this study. The key to the success of this implementation relied largely on backend. The backend was implemented using PHP and MySQL database both of which are free software, making it a great choice for future maintenance. The system interface was developed by using XML and Java so as to allow user interaction with the system. We used

the mobile technology, which is accessed by majority of people making it a reliable way of delivering nutrition information.

3.6 Conclusion

This study was on developing an integrated mobile application for enhancing management of nutrition information in Tanzania. The system development used various methods and materials, which were determined after the design process discussed in this paper. The developed system uses mobile phones as the tool to manage nutrition information by allowing interaction without time and place limitations due to ownership issues. As the result, this culminated into the development of a mobile application that will solve the issue of poor access to nutrition information by facilitating collection and delivering of those information.

CHAPTER FOUR

TESTING AND EVALUATION OF AN INTEGRATED MOBILE APPLICATION FOR ENHANCING MANAGEMENT OF NUTRITION INFORMATION

Neema Mduma, Khamisi Kalegele

School of Computation and Communication Science & Engineering

Nelson Mandela African Institution of Science and Technology

Arusha, Tanzania

mduman@nm-aist.ac.tz, khamisi.kalegele@nm-aist.ac.tz

***Abstract*—** *This paper presents a systematic testing and evaluation of a mobile application system for managing nutrition information. The system was developed for the purpose of enhancing and facilitating exchange of nutrition information between nutrition practitioners and mothers. The targeted population for the system are small towns where internet connectivity is normally in low bandwidth. In this paper, we present how the system was tested and evaluated for use under such environments. Three different kinds of tests were adopted. In one, we tested each part of the system individually, and then integrated modules were tested to determine if they are working and interacted successfully. The last method used was conducted to determine if the system functionalities are working successfully. Thereafter, the assessment was conducted with 50 participants who were randomly selected based on their clinic attendance to determine the effectiveness of the system. Questionnaires administered to the chosen group of users to have their opinion on how they perceive the system, and later we analysed the data to determine the findings of the study. The testing and evaluation results depicted that the system enhances management of nutrition information by facilitating collection and dissemination of nutrition information.*

***Keywords*—** *Unit testing, Integration testing, System testing, System evaluation, Results*

4.1 Introduction

The process of performing system testing usually occurs during development and after software version increments. A variety of testing methods are used in order to explore functionality of the system as well as to identify problems. When testing, a series of systematic procedures are referred to determine the system performance and where common mistakes may be found. In order to accomplish this, three different techniques are performed.

4.1.1 Unit testing

Unit testing is a software development process in which the smallest testable parts of an application called units are individually and independently scrutinized for proper operation (Margaret Rouse, 2007). Each part of the system was tested before integration to determine if is the parts are working correctly.

4.1.2 Integration testing

Integration testing is a logical extension of unit testing. In its simplest form, two units that have already been tested are combined into a component and the interface between them is tested. A component, in this sense, refers to an integrated aggregate of more than one unit (Microsoft, 2003). This idea identifies problems that occur when units are combined. In our study, we tested and fixed bugs that were identified while integrating the subsystems. This was performed to determine if all integrated parts are appropriately working and interacting with each other.

4.1.3 System testing

System testing is the testing of behaviour of a complete and fully integrated software product based on requirement specifications (Software Testing Class, 2014). System testing is most often the final test to verify that the system to be delivered meets the specification and its purpose. This testing should investigate both functional and non-functional requirements of the testing. The system testing was performed to evaluate system requirements if they match with the developed system. The testing was done to check if the data are successfully stored in the system and updated when some changes are made. We tested the complete system to ensure that the whole system runs as designed. This was done by deploying the system to users and allowing them to test if the system performs according to their expectations.

4.2 System Evaluation

In this study, evaluation process aimed at evaluating the acceptance and usability of the developed nutrition information management system. The evaluation process also helped to pinpoint some of the design variations that could be implemented to improve the developed system in the future.

4.3 Test Report Details

Every test of the system conducted including unit testing, integration testing and system testing is documented in a test report. Table 5 to 9 shows the report details of the conducted testing.

Table 5: Login testing

LOGIN TESTING		
Checked	**Test**	**Result**
✓	Grant permission to the user who provides a correct username and password	PASS
✓	Deny permission to the wrong username or password	PASS

Table 6: Database testing

DATABASE TESTING		
Checked	**Test**	**Result**
✓	When user registered to the system, database updated	PASS
✓	When send requests to the system, database updated	PASS
✓	When send information to the system, database updated	PASS

Table 7: Reminding testing

REMINDING TESTING		
Checked	**Test**	**Result**
✓	The system remind user on important events for clinic visit	PASS

Table 8: Reporting testing

REPORTING TESTING		
Checked	**Test**	**Result**
✔	The system allows user to view reports based on the provided recommendations.	PASS

Table 9: Testing all system modules

TESTING ALL SYSTEM MODULES		
Checked	**Test**	**Result**
✔	The system allows registered user to login, send requests, view recommendations, view reports, and receive reminders, and the data integrated successfully.	PASS

4.4 Methodology

This section discussed various methods used in the study. It contains and explains sample size, sampling techniques, study tools, ethical consideration, data gathering procedures and data presentation. Evaluation of the developed nutrition information management system is considered a critical element in determining whether the developed system will be able to enhance management of nutrition information and improve user acceptance level.

The study was conducted at Green Hope Clinic in Arusha by selecting a group of users to test the system and providing their opinions on how do they perceive the system. This group of users included both professionals and normal users whereby each user was first introduced to the system and how it works and were allowed to register and access the system. After completing accessing the system, the evaluation method used was questionnaire survey which was designed and reviewed before distribution to the sampled participants.

4.4.1 Sample size

The study involved 50 participants who were selected randomly based on their clinic visits at the time we were doing the assessment. Evaluation was conducted for a one week period by involving two types of groups. The first group comprised of nutrition practitioners and the second group involved prenatal and postnatal mothers from Green Hope Clinic and some of NM-AIST students. The participants were all registered to the system. The reason for

selecting nutrition practitioners is because the system is for nutrition and they are responsible for providing nutrition information. The prenatal and postnatal mothers were selected because they are the targeted users of the system. Age was not a criteria of the study and an assumption was made that everyone who attended the clinic would benefit from using the system.

4.4.2 Sampling technique

We used a random sampling technique to get all the participants. We left it open for anyone to participate. The sampling process was dynamic in the sense that whoever attended the clinic had a chance of participating in the study; the only requirement was that this person must be willing to do so.

4.4.3 Study tools

The study relied on the working prototype of the integrated nutrition information management system and on semi-structured questionnaire. The prototype was a necessary tool as it gave user access to the system during the assessment. Questionnaires were used as a tool of gathering users' feedback.

4.4.4 Ethical consideration

Any kind of information gathering process must put consideration of ethical issues. Studies involving human beings require ethical practices in handling information that is gathered and in management of the information. Our study involved user nutrition information based on their conditions, and we only tested the use of our system. Users were free to use their registered credentials, which are username and password to access the system. All users were informed about how the system will record the information, and only those who were willing to continue with the exercise accessed the system. In addition, the researcher ensured that the clinic authorities were consulted and permission was granted for us to work during the data collection. The researcher considered moral justification by ensuring safety, social and psychological aspects, privacy, confidentiality, disclosure and security of the data collected and respondents involved.

4.4.5 Data gathering procedures

Before data collection commenced, we requested for permission from the clinic authorities of which we had to explain about the idea of the study we wanted to do. During the assessment

of the system, we allowed users to register into the system and allowed them to request for nutrition information and interact with the overall system. After using the system, each user was given a questionnaire to provide feedback on the system.

4.4.6 Data presentation

Tables have been used to present the results of the assessment. Responses in the questionnaires were fed into a digital form made using Google forms service; this way it was easy to get the summary of the findings for further analysis because Google forms assists in compiling the data.

4.5 System Analysis and Results

This section describes the statistical analyses and validation performed on data collected through the questionnaire. Data were exported to Google forms service for analysis and results are summarized in the tables as depicted in Tables 10 and 11 below.

Table 10: Usability testing

	Strongly Agree (%)	Agree (%)	Disagree (%)	Not sure (%)
I found the system simple to use.	61	34	3	2
The system is interactive.	40	55	0	5
I think the system will need support from technical person to user.	28	57	3	12
I would imagine that most people will learn to use system very quickly.	34	56	3	7

Table 11: User acceptance test

	Strongly Agree (%)	Agree (%)	Disagree (%)	Not sure (%)
I think that I would like to continue using this system.	31	68	0	1
I think the system will assist me in record keeping.	75	23	1	1
The system will assist me in reminding on important clinic visits.	43	57	0	0
The system will provide easy interaction between nutrition practitioners and prenatal and postnatal mothers.	69	31	0	0
The system will facilitate management of nutrition information.	27	69	1	3
The system will be useful to me.	77	23	0	0

As shown in Tables 10 and 11 above, both groups of participants were extremely satisfied with the system. The questioned nutrition practitioners and users were strongly agreed with the developed nutrition information management system.

4.6 Discussion and conclusion

This paper demonstrates different testing methods of the system. The testing included unit, integration and system testing which were done for the nutrition information management system and passed successfully. Each part of the system was tested to make sure that it is working effectively. Thereafter, different parts of the system were combined and tested to realize how they interact with each other; lastly, the whole system was tested. Subsequently, the system was taken out to selected users to assess the performance and effectiveness in order to determine if the objective of the study were achieved or not. Tables 10 and 11 demonstrate the views from users of the system who altogether strongly agreed with the developed nutrition information management system.

CHAPTER FIVE
General Discussion, Conclusion and Recommendations

5.1 General discussion

The main objective of this research was to enhance collection and delivery of nutrition information by using an integrated mobile application that will facilitate an interactive flow of nutrition information between nutrition practitioners, researchers and patients especially mothers. In the study, the system concept presented in this dissertation was implemented and tested. The developed system has a great potential of being used as a reliable tool for collecting and disseminating nutrition information to the general public in Tanzania while reaching out to all groups of people.

The empirical contribution of this study is the automation of the nutrition information management system, which involves different services including generating reports, providing reminders, keeping nutrition information and enabling access to nutrition information categorized into tips and recommendations. We have established the key requirements for an effective nutrition management system and systematically demonstrated how to design such system. We have developed a working application which can easily go to production after enhancing some features like security and compiled a dissertation detailing all of the above.

5.2 Conclusion

A preliminary study conducted before this study showed that there is lack of nutrition information and hence poor management of nutrition information in Tanzania. Absence of proper tools makes collection, management and access to nutrition information very difficult. Despite many efforts directed towards enhancing the management of nutrition information, the issue of malnutrition continues to contribute to the deaths of children under age of five years. The nutrition information provided to pre-natal and post-natal mothers is verbal and delivered in groups without consideration of personal conditions, which differ from one another. Furthermore, keeping records of the provided nutrition recommendations is a challenge. Hence, this study proposed and developed an integrated mobile application for enhancing management of nutrition information.

With results from the survey and critical literature review, we analysed different systems developed by others for enhancing management of nutrition information. In this regard, the study came up with a solution that matches with the environment of Tanzania using Arusha as the case area. The study proposed the use of mobile application with android technology for enhancing collection and dissemination of nutrition information. The reason for proposing the android technology was based on the nature of this technology, which is very user friendly and easy to access. This technology is also cost effective compared to other technologies such as USSD. Through critical analysis of the requirements, the nutrition information mobile application was designed and developed. The system is a mobile application based for management of nutrition information. The application allows users to login to the system, access nutrition tips and request for any relevant information concerning nutrition. Putting it altogether, the advantage of this application is that nutrition information will be well managed and users' satisfaction will be remarkable.

The system was tested to determine its performance as per requirements using three methods of testing, namely; unit testing, integration testing and system testing. The testing achieved success and the system allowed registered users to login through username and password. Nutrition information were sent and saved to database successfully. Equally, users received notification messages to remind them about clinic visits. Nutrition practitioners could login and respond to user requests by sending the requested information.

5.3 Recommendations

Although the findings of this study show that users have accepted the system and stated that it will enhance management of nutrition information, we suggest further empirical validation of the system to test if it has any effect in user attitude towards nutrition. This can be achieved by providing the tool to one group and other group continues with their traditional system and compare the selected measurement criteria between the two groups over time. There is also room for implementing the developed system on devices running other operating systems. It is desirable to operate in other operating systems apart from Android and should support other mobile devices such as basic phones.

This study was conducted to enhance management of nutrition information. The aim of developing the system was to supplement the verbal means of providing nutrition information with an automated mobile application. Since the developed application requires users to visit

the clinic to allow nutritional practitioners to provide the recommendations through the developed mobile application, it is better to have the system that can record the nutrition information without relying on the users' visits to the clinic. However, further studies are needed to enable the improvement of the developed application. This can be achieved by strengthening network coverage, especially in rural areas.

REFERENCES

Aker, J. C., & Mbiti, I. M. (2010). Mobile Phones and Economic Development in Africa Working Paper 211 June 2010 Mobile Phones and Economic Development in Africa. *Center for Global Development*, (June 2010), 1–44.

Bredemeyer. (2001). FunctReq.PDF - functreq.pdf. Retrieved May 20, 2015, from http://www.bredemeyer.com/pdf_files/functreq.pdf

Cafer, F., & Misra, S. (2009). A cognitive requirement specification model. In *2009 24th International Symposium on Computer and Information Sciences, ISCIS 2009* (pp. 518–521). http://doi.org/10.1109/ISCIS.2009.5291868

CARE. (2014). CH-Using-Mobile-Health-Technology-to-Improve-Nutrition-Early Childhood-Development.pdf. Retrieved May 20, 2015, from http://www.care.org/sites/default/files/documents/CH-Using-Mobile-Health-Technology-to-Improve-Nutrition-Early Childhood-Development.pdf

Caulfield Laura, de Onis Mercedes, Blössner Monika, & Black Robert. (2004). Undernutr-underlying-cause-diarrhoea,-pneum,malaria,-measles-2004.pdf. Retrieved May 20, 2015, from http://www.cmamforum.org/Pool/Resources/Undernutr-underlying-cause-diarrhoea,-pneum,malaria,-measles-2004.pdf

Cole-Lewis, H., & Kershaw, T. (2010). Text messaging as a tool for behavior change in disease prevention and management. *Epidemiologic Reviews*, *32*(1), 56–69. http://doi.org/10.1093/epirev/mxq004

Donald S., & Le Vie, J. (2000). Understanding Data Flow Diagrams - DFD_over_Flowcharts.pdf. Retrieved May 20, 2015, from http://ratandon.mysite.syr.edu/cis453/notes/DFD_over_Flowcharts.pdf

Eck David, Hobart, & Colleges William. (2014). () - javanotes6-linked.pdf. Retrieved May 17, 2015, from http://math.hws.edu/eck/cs124/downloads/javanotes6-linked.pdf

GSMA. (2015). GSMA_mHealth_Country_Feasibility_Report_Tanzania_2015.pdf. Retrieved May 20, 2015, from http://tanzania.gsmamhealthfeasibility.com/GSMA_mHealth_Country_Feasibility_Report_Tanzania_2015.pdf

Hall Marty, & Brown Larry. (2001). Microsoft PowerPoint - XML.ppt - xml.pdf. Retrieved May 17, 2015, from http://www.cs.toronto.edu/~mlou/csc309/xml.pdf

Heng Christopher. (2010). What is MySQL? What is a Database? What is SQL? (thesitewizard.com). Retrieved May 16, 2015, from http://www.thesitewizard.com/faqs/what-is-mysql-database.shtml

Hirschberg, M. A. (2015). STN 2-1 Topic: Rapid Application Development (RAD): A Brief Overview. Retrieved from https://www.mendeley.com/library/viewer/?fileId=6f1c4f30-d7da-50d9-67a7-2c02747d1f83

Khan, J. G., Yang, J. S., & Kahn, J. S. (2010). "Mobile" health needs and opportunities in developing countries. *Health Affairs*.

Leach, V., & Kilama, B. (2009). *Institutional analysis of nutrition in Tanzania*. Retrieved from http://www.repoa.or.tz/documents/Special_Paper_09.31_.pdf

Lowe Doug, & Burd Barry. (2007). What Is Java, and Why Is It So Great? - For Dummies. Retrieved May 17, 2015, from http://www.dummies.com/how-to/content/what-is-java-and-why-is-it-so-great.html

Margaret Rouse. (2007). What is unit testing? - Definition from WhatIs.com. Retrieved August 13, 2015, from http://searchsoftwarequality.techtarget.com/definition/unit-testing

Meier, R. (2010). *Professional Android 2 Application Development*. *Development*. Retrieved from http://www.amazon.co.uk/Professional-Android-Application-Development-Programmer/dp/0470565527

Microsoft. (2003). Integration Testing. Retrieved August 13, 2015, from https://msdn.microsoft.com/en-us/library/aa292128(v=vs.71).aspx

Ministry of Health and Social Welfare. (2011). National Nutrition Strategy 2012-2015, (June).

Ministry of Health and Social Welfare. (2013). NATIONAL NUTRITION SOCIAL AND BEHAVIOR CHANGE COMMUNICATION STRATEGY July 2013 – June 2018, (July).

Mobile Alliance for Maternal Action. (2012). MAMA_Global_MEPlan_FINAL_all.pdf. Retrieved May 20, 2015, from http://www.mobilemamaalliance.org/sites/default/files/MAMA_Global_MEPlan_FINAL_all.pdf

Mtweve, S. (2014). Tanzania's Internet users hit 9m - Business | The Citizen. Retrieved March 19, 2016, from http://www.thecitizen.co.tz/Business/Tanzania-s-Internet-users-hit-9m/-/1840414/2254676/-/dgt0ps/-/index.html

Muthee, J., & Mhando, N. (2006). AMDI-BBC-summary-report. *African Media Development Initiative*.

MySQL. (2015). MySQL :: Top Reasons to Use MySQL. Retrieved May 16, 2015, from https://www.mysql.com/why-mysql/topreasons.html

Pitts Monica. (2015). 5 Reasons Why PHP Is a Great Programming Language | Mayecreate Design. Retrieved May 16, 2015, from http://www.mayecreate.com/2013/12/5-reasons-php-great-programming-language/

Rick van Genuchten, Wouter Haring, Daan van Kassel, & Kaoutar Yakubi. (2012). Mobile phone use in Tanzania. *TAN2012 Market Research*.

Rouse Margaret. (2001). What is XML (Extensible Markup Language)? - Definition from WhatIs.com. Retrieved May 17, 2015, from http://searchsoa.techtarget.com/definition/XML

Schwaber, K. (2001). Software Development Life Cycle (SDLC). *Tutorials Point Simple Easy Learning*. Retrieved from http://www.tutorialspoint.com/sdlc/sdlc_tutorial.pdf

Sharon Silow-Carroll, Jennifer N. Edwards, & Diana Rodin. (2012). Using Electronic Health Records to Improve Quality and Efficiency: The Experiences of Leading Hospitals - 1608_SilowCarroll_using_EHRs_improve_quality.pdf. Retrieved August 21, 2015, from http://www.commonwealthfund.org/~/media/Files/Publications/Issue Brief/2012/Jul/1608_SilowCarroll_using_EHRs_improve_quality.pdf

Sife, A. S., Kiondo, E., & Lyimo-Macha, J. G. (2010). CONTRIBUTION OF MOBILE PHONES TO RURAL LIVELIHOODS AND POVERTY REDUCTION IN MOROGORO REGION, TANZANIA. *EJISDC*, *42*(3), 1–15.

Silva, C. N. (2008). Designing Qualitative Research. *Forum Qualitative Sozialforschung/Forum: Qualitative Social Research*, *9*(3).

http://doi.org/10.1057/omj.2011.23

Software Testing Class. (2014). System Testing: What? Why? & How? | Software Testing Class. Retrieved August 13, 2015, from http://www.softwaretestingclass.com/system-testing-what-why-how/

Sparx systems. (2004). The_Use_Case_Model.pdf. Retrieved May 20, 2015, from http://www.sparxsystems.com/downloads/whitepapers/The_Use_Case_Model.pdf

Tanzania Bureau of Statistics and Macro International, & (Measure DHS). (2010). *Tanzania Demographic and Health Survey*. *National Bureau of Statistics Dar es Salaam, Tanzania ICF Macro Calverton, Maryland, USA*. Retrieved from http://www.measuredhs.com/pubs/pdf/FR243/FR243[24June2011].pdf

Tanzanian mobile phone subscribers jump by 16 pct in 2014 | Reuters. (2015). Retrieved March 7, 2016, from http://www.reuters.com/article/tanzania-telecomunications-idUSL5N0XD37P20150416

Thalheim Bernhard, Schewe Klaus-Dieter, & Ma Hui. (2009). Conceptual Application Domain Modelling. Retrieved May 20, 2015, from http://crpit.com/confpapers/CRPITV96Thalheim.pdf

Tutorialspoint. (2014). xml_tutorial.pdf. Retrieved May 17, 2015, from http://www.tutorialspoint.com/xml/xml_tutorial.pdf

UNICEF Tanzania. (2011). Children and Women in Tanzania Volume I Mainland, 1–202.

United Nations. (2006). MDGReport2006.pdf. Retrieved May 20, 2015, from http://www.un.org/zh/millenniumgoals/pdf/MDGReport2006.pdf

Usability. (2014). User Interface Design Basics. Retrieved from http://www.usability.gov/what-and-why/user-interface-design.html

Vital Wave Consulting. (2009). *mHealth for Development: The Opportunity of Mobile Technology for Healthcare in the Developing World*. *Technology* (Vol. 46). http://doi.org/10.1145/602421.602423

World Health Organization. (2013). WHO | Nutrition. Retrieved August 21, 2015, from http://www.who.int/topics/nutrition/en/

APPENDICES

Appendix 1: Interview guide questions

REQUIREMENTS GATHERING

PRE-NATAL AND POST NATAL MOTHERS INTERVIEW QUESTIONS

Introduction

I am Neema Mduma, a master's candidate from NM-AIST Arusha. I am currently doing research on developing an integrated mobile application for enhancing management of nutrition information in Arusha Tanzania. These interview questions aim at gathering requirements of the proposed system. Thank you for your willingness to participate in this exercise.

---

1. How old are you?
2. Is this your first pregnancy?
3. Do you have a mobile phone?
4. For what purpose do you use your phone (e.g. sending messages, calling, browsing the internet etc.)?
5. Do you keep your information about nutrition?
6. If NO in question 5, why (e.g. I don't see its importance, I have no tools for keeping those information etc.)?
7. Do you think mobile phone will help you to access nutrition information, keeping your records and reminding on necessary events for clinic visit?
8. How do you get information about nutrition (e.g. Radio/TV, nutrition practitioners, mobile etc.)?
9. How do you communicate with nutrition practitioners (e.g. face to face, phone etc.)?
10. How often do you communicate with nutrition practitioners (e.g. weekly, monthly, during emergencies etc.)?
11. What way would you prefer to receive information from nutrition practitioners (e.g. computer, face to face, phone etc.)?
12. Will you be willing to request for nutrition assistance to nutrition practitioners?
13. Will you be willing to incur communication cost in order to receive nutrition information?
14. Whom do you consult when you need nutrition assistance (e.g. neighbours, nutrition practitioners etc.)?
15. After having mobile application what way would you like to receive your nutrition information (e.g. android notification, SMS, chart, image etc.)?

NUTRITION PRACTITIONERS INTERVIEW QUESTIONS

Introduction

I am Neema Mduma a master's candidate from NM-AIST Arusha. I am currently doing research on developing an integrated mobile application for enhancing management of nutrition information in Arusha Tanzania. These interview questions aim at gathering requirements of the proposed system. Thank you for your willingness to participate in this exercise.

---

1. Your gender
2. How old are you?
3. Your experience
4. How do you receive information from prenatal and postnatal mothers (e.g. phone, clinic visit, other practitioners etc.)?
5. How do you get new tips concerning nutrition (e.g. email, phone, browsing etc.)?
6. Are there available information systems to keep nutrition information?
7. If YES in question 6, please mention them

   …….
8. Do you think it will benefit you to have a mobile application for nutrition information?
9. Are you comfortable working with mobile application system?
10. What nutrition information do prenatal and postnatal mothers ask from you?

    ……

Appendix 2: Questionnaire for system evaluation

SYSTEM EVALUATION

Introduction

I am Neema Mduma a master's candidate from NM-AIST Arusha. I am currently doing research on developing an integrated mobile application for enhancing management of nutrition information in Arusha Tanzania. This questionnaire is aimed at evaluating the developed system by exposing users to the system and gives us the feedback by filling this questionnaire.

Personal Information				
Name:				
Sex:				
Professional:				
	Strongly agree	Agree	Disagree	Not sure
a) I think that I would like to continue using this system				
b) I found the system simple to use				
c) The system is interactive				
d) I think the system will need a support of technical person to user				
e) I would imagine that most people will learn to use system very quickly				
f) I think the system will assist me in record keeping				
g) The system will assist me in reminding on important clinic visits				
h) The system will provide easily interaction between nutrition practitioners and prenatal and postnatal mothers				
i) The system will facilitate management of nutrition information				
j) The system will be useful to me				

Appendix 3: User registration

```
package com.neemamduma.nutrimobile;

import java.io.BufferedReader;
import java.io.InputStream;

import java.io.InputStreamReader;

import java.util.ArrayList;

import java.util.Random;

import org.apache.http.HttpResponse;
import org.apache.http.NameValuePair;

import org.apache.http.client.entity.UrlEncodedFormEntity;

import org.apache.http.client.methods.HttpPost;

import org.apache.http.impl.client.DefaultHttpClient;

import org.apache.http.message.BasicNameValuePair;

import android.app.Activity;
import android.app.ProgressDialog;

import android.content.Intent;

import android.os.AsyncTask;

import android.os.Bundle;

import android.view.View;

import android.view.View.OnClickListener;

import android.widget.AdapterView;

import android.widget.ArrayAdapter;

import android.widget.Button;

import android.widget.EditText;

import android.widget.RadioButton;

import android.widget.RadioGroup;

import android.widget.Spinner;

import android.widget.Toast;

public class SignUp extends Activity { EditText edt_fullname, edt_email, edt_password;
   String fullname, email, password, userid, user_type;
```

```
    RadioGroup rg_usertype;
    RadioButton usertype;
    Button signup;
    int id;
    // db operations global variable declarations
    String conn_state;
    String myresult;
    ProgressDialog pDialog;
    String insert_url = "http://www.mayombo.com/nutrimobilescripts/signup.php";
    //change to this
    //String insert_url = "http://phone_ip_address/nutrimobilescripts/signup.php";
    String[] users = {"User", "Nutritionalist", "Researcher", "Administrator"};
    String selecteduser;
    Spinner sp_users;
    ArrayAdapter<String> myadapter;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        // TODO Auto-generated method stub
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_signup);
        sp_users = (Spinner) findViewById(R.id.spusers);
        edt_fullname = (EditText) findViewById(R.id.edtfullname);
        edt_email = (EditText) findViewById(R.id.edtemail);
        edt_password = (EditText) findViewById(R.id.edtpassword);
        myadapter = new ArrayAdapter<String>(getApplicationContext(),
R.layout.spinnerdesign, R.id.spinner_text, users);
        sp_users.setAdapter(myadapter);
        sp_users.setOnItemSelectedListener(new AdapterView.OnItemSelectedListener() {
            @Override public void onItemSelected(AdapterView<?> parent, View view, int
position, long id) {
                selecteduser = users[position];
```

```
        }
        @Override public void onNothingSelected(AdapterView<?> parent) {
            selecteduser = users[0];
        }
    });
    signup = (Button) findViewById(R.id.btnsignup);
    signup.setOnClickListener(new OnClickListener() {
        @Override public void onClick(View v) {
            // TODO Auto-generated method stub
            Random rn = new Random();
            id = rn.nextInt(1000);
            patientid = "nutriP" + id;
            new RegisterUser().execute();
        }
    });
}
class RegisterUser extends AsyncTask<String, String, String> {
    @Override protected void onPreExecute() {
        // TODO Auto-generated method stub
        super.onPreExecute();
        // Asynch variables
        pDialog = new ProgressDialog(SignUp.this);
        pDialog.setMessage("Registering User...Please Wait...");
        pDialog.setIndeterminate(false);
        pDialog.setCancelable(false);
        pDialog.show();
    }
    @Override protected String doInBackground(String... arg0) {
        // TODO Auto-generated method stub
        fullname = edt_fullname.getText().toString();
```

```
email = edt_email.getText().toString();
password = edt_password.getText().toString();
try {
                        /* seting up the connection and send data with url */
   // create a http default client - initialize the HTTp client
   DefaultHttpClient httpclient = new DefaultHttpClient();
   // Create a HTTp post object to hold our data - url
   // the IP is used when testing locally but for online server you
   // write the actual server name
   HttpPost httppost = new HttpPost(insert_url);
   // use HTTPClient to execute the HTTPPost
   // Execute HTTP Post Request
   // encode URL
        ArrayList<NameValuePair> nameValuePairs = new
ArrayList<NameValuePair>(3);
   // must be in pairs like localhost.....php?id=username.getText()
   nameValuePairs
      .add(new BasicNameValuePair("usertype", selecteduser));
   nameValuePairs.add(new BasicNameValuePair("email", email));
   nameValuePairs.add(new BasicNameValuePair("password", password));
   nameValuePairs
      .add(new BasicNameValuePair("fullname", fullname));
   nameValuePairs.add(new BasicNameValuePair("patientid", patientid));
   httppost.setEntity(new UrlEncodedFormEntity(nameValuePairs));
   HttpResponse response = httpclient.execute(httppost);
                        /* receiving response from the database */
   // use Input stream to read the http client response
   InputStream inputStream = response.getEntity().getContent();
   // use buffered reader and InputStreamReader to read the input // stream
BufferedReader rd = new BufferedReader(new InputStreamReader( inputStream), 4096);
   String line;
```

```
        // initialize StringBuilder
        StringBuilder sb = new StringBuilder();
        // read everything from the Buffered reader and append the to
        // the
        // string builder
        while ((line = rd.readLine()) != null) {
            sb.append(line);
        }
        rd.close();
        // our result
        myresult = sb.toString();
        inputStream.close();
    } catch (Exception e) {
        Toast.makeText(getApplicationContext(),
                "Error inside set:" + e.toString(), Toast.LENGTH_LONG)  .show();
    }
    return myresult;
}
@Override protected void onPostExecute(String result) {
    // TODO Auto-generated method stub
    super.onPostExecute(result);
    pDialog.dismiss();
    // check if response is 4
    if (myresult.equals("1")) {
        Toast.makeText(getApplicationContext(),
                "User Registered Sucessfully", Toast.LENGTH_LONG)
                .show();
        startActivity(new Intent(getApplicationContext(), Login.class));
    }
    // /check if response is 5
```

```
        else if (myresult.equals("0")) {
            Toast.makeText(getApplicationContext(),
                "User Registration Failed.Please Try Again!",
                Toast.LENGTH_LONG).show();
        }
      }
   }
}
```

Appendix 4: Login interface

```
package com.neemamduma.nutrimobile;

import java.io.BufferedReader;
import java.io.InputStream;
import java.io.InputStreamReader;
import java.util.ArrayList;

import org.apache.http.HttpResponse;
import org.apache.http.NameValuePair;
import org.apache.http.client.entity.UrlEncodedFormEntity;
import org.apache.http.client.methods.HttpPost;
import org.apache.http.impl.client.DefaultHttpClient;
import org.apache.http.message.BasicNameValuePair;

import android.app.Activity;
import android.app.ProgressDialog;
import android.content.Intent;
import android.net.ConnectivityManager;
import android.net.NetworkInfo;
import android.os.AsyncTask;
import android.os.Bundle;
import android.view.View;
import android.widget.ArrayAdapter;
import android.widget.Button;
```

```
import android.widget.EditText;
import android.widget.RadioButton;
import android.widget.RadioGroup;
import android.widget.Spinner;
import android.widget.Toast;

public class Login extends Activity {
   String[] user_type = {"Nutritionist", "User"};
   ArrayAdapter<String> myadapter;
   Spinner spusertype;
   String userselected, email, password;
   EditText edt_email, edt_password;
   // db operations variable declarations
   String conn_state;
   String myresult;
   ProgressDialog pDialog;
   RadioGroup rg_usertype;
   RadioButton usertype;
   Button signup;
   int selectedid;
   String[] datareceived;
   @Override protected void onCreate(Bundle savedInstanceState) {
      // TODO Auto-generated method stub
      super.onCreate(savedInstanceState);
      setContentView(R.layout.activity_login);
      edt_email = (EditText) findViewById(R.id.edtemail);
      edt_password = (EditText) findViewById(R.id.edtpassword);
      //rg_usertype = (RadioGroup) findViewById(R.id.rgusertype);
   }
   // method for checking network state of the device
   public String checkConnectionState() {
```

```
        // check Internet or connection availability
        ConnectivityManager conMgr = (ConnectivityManager)
getSystemService(Activity.CONNECTIVITY_SERVICE);
        if (conMgr.getNetworkInfo(0).getState() == NetworkInfo.State.CONNECTED
                || conMgr.getNetworkInfo(1).getState() == NetworkInfo.State.CONNECTING) {
            // notify user you are online
            conn_state = "connected";
        } else if (conMgr.getNetworkInfo(0).getState() ==
NetworkInfo.State.DISCONNECTED
                || conMgr.getNetworkInfo(1).getState() == NetworkInfo.State.DISCONNECTED)
{
            // notify user you are not online
            conn_state = "not_connected";
        }
        return conn_state;
    }
    public void buttonClicked(View v) {
        switch (v.getId()) {
            case R.id.btnlogin:
                email = edt_email.getText().toString();
                password = edt_password.getText().toString();
                new UserLogin().execute();
                break;
            case R.id.btnsignup:
                startActivity(new Intent(getApplicationContext(), SignUp.class));
                break;
            case R.id.btnforgetpass:
                break;
            default:
                break;
        }
```

```
    }
    class UserLogin extends AsyncTask<String, String, String> {
        @Override protected void onPreExecute() {
            // TODO Auto-generated method stub
            super.onPreExecute();
            // Asynch variables
            pDialog = new ProgressDialog(Login.this);
            pDialog.setMessage("Authenticating..Please Wait...");
            pDialog.setIndeterminate(false);
            pDialog.setCancelable(false);
            pDialog.show();
        }
        @Override
        protected String doInBackground(String... arg0) {
            // TODO Auto-generated method stub
            try {
                DefaultHttpClient httpclient = new DefaultHttpClient();
                HttpPost httppost = new HttpPost(
                        "http://www.mayombo.com/nutrimobilescripts/userlogin.php"
                        // "http://phone_ip_address/nutrimobilescripts/userlogin.php";
                );
                // Add your data
ArrayList<NameValuePair> nameValuePairs = new ArrayList<NameValuePair>( 1);
//                                  nameValuePairs.add(new BasicNameValuePair("usertype",
//                                                          userselected));
                nameValuePairs.add(new BasicNameValuePair("email", email));
                nameValuePairs
                        .add(new BasicNameValuePair("password", password));
                httppost.setEntity(new UrlEncodedFormEntity(nameValuePairs));
                // Execute HTTP Post Request
```

```
        HttpResponse response = httpclient.execute(httppost);
        InputStream inputStream = response.getEntity().getContent();
BufferedReader rd = new BufferedReader(new InputStreamReader( inputStream), 4096);
        String line;
        StringBuilder sb = new StringBuilder();
        while ((line = rd.readLine()) != null) {
           sb.append(line);
        }
        rd.close();
        myresult = sb.toString();
        inputStream.close();
    } catch (Exception e) {
        Toast.makeText(getApplicationContext(),
              "Error inside set:" + e.toString(), Toast.LENGTH_LONG)
              .show();
    }
    return myresult;
 }
 @Override protected void onPostExecute(String result) {
    // TODO Auto-generated method stub
    super.onPostExecute(result);
    pDialog.dismiss();
    if (myresult.equals("0")) {
        Toast.makeText(getApplicationContext(), "Invalid Login",
              Toast.LENGTH_LONG).show();
    } else {
        datareceived = myresult.split("#");
        if (datareceived[1].equals("User")) {
           Intent a = new Intent(getApplicationContext(), UserMainActivity.class);
           Bundle bundle = new Bundle();
```

```
            bundle.putString("useremail", datareceived[0]);
            a.putExtras(bundle);
            startActivity(a);
        } else {
            Intent a = new Intent(getApplicationContext(),
                    MainActivity.class);
            Bundle bundle = new Bundle();
            bundle.putString("useremail", datareceived[0]);
            bundle.putString("usertype", datareceived[1]);
            a.putExtras(bundle);
            startActivity(a);
        }
    }
}
class DoctorLogin extends AsyncTask<String, String, String> {
    @Override protected void onPreExecute() {
        // TODO Auto-generated method stub
        super.onPreExecute();
        // Asynch variables
        pDialog = new ProgressDialog(Login.this);
        pDialog.setMessage("Authenticating..Please Wait...");
        pDialog.setIndeterminate(false);
        pDialog.setCancelable(false);
        pDialog.show();
    }
    @Override protected String doInBackground(String... arg0) {
        // TODO Auto-generated method stub
        try {
            DefaultHttpClient httpclient = new DefaultHttpClient();
            HttpPost httppost = new HttpPost(
```

```
                "http://www.mayombo.com/nutrimobilescripts/nutritionistlogin.php"
                // change to this to test direct to a phone
                // String "http://phone_ip_address/nutrimobilescripts/nutritionistlogin.php";
            );
            // Add your data
            ArrayList<NameValuePair> nameValuePairs = new
ArrayList<NameValuePair>( 1);
            nameValuePairs.add(new BasicNameValuePair("usertype",userselected));
            nameValuePairs.add(new BasicNameValuePair("email", email));
            nameValuePairs.add(new BasicNameValuePair("password", password));
            httppost.setEntity(new UrlEncodedFormEntity(nameValuePairs));
            // Execute HTTP Post Request
            HttpResponse response = httpclient.execute(httppost);
            InputStream inputStream = response.getEntity().getContent();
    BufferedReader rd = new BufferedReader(new InputStreamReader(inputStream), 4096);
            String line;
            StringBuilder sb = new StringBuilder();
            while ((line = rd.readLine()) != null) {
               sb.append(line);
            }
            rd.close();
            myresult = sb.toString();
            inputStream.close();
          } catch (Exception e) {
            Toast.makeText(getApplicationContext(),
  "Error inside set:" + e.toString(), Toast.LENGTH_LONG) .show();
          }
          return myresult;
        }
        @Override protected void onPostExecute(String result) {
```

```
            // TODO Auto-generated method stub
            super.onPostExecute(result);
            pDialog.dismiss();
            // check if response is 4
            if (myresult.equals("0")) {
               Toast.makeText(getApplicationContext(), "Invalid Login",
                     Toast.LENGTH_LONG).show();
            } else {
               String useremail = myresult;
               Intent a = new Intent(getApplicationContext(),
                     MainActivity.class);
               Bundle bundle = new Bundle();
               bundle.putString("useremail", useremail);
               a.putExtras(bundle);
               startActivity(a);
            }
         }
      }
   }
}
```

Appendix 5: Tips details

```
package com.neemamduma.nutrimobile;

import java.io.BufferedReader;
import java.io.InputStream;
import java.io.InputStreamReader;
import java.util.ArrayList;
import java.util.Random;

import org.apache.http.HttpResponse;
import org.apache.http.NameValuePair;
import org.apache.http.client.entity.UrlEncodedFormEntity;
```

```
import org.apache.http.client.methods.HttpPost;
import org.apache.http.impl.client.DefaultHttpClient;
import org.apache.http.message.BasicNameValuePair;

import android.app.Dialog;
import android.app.ProgressDialog;
import android.os.AsyncTask;
import android.os.Bundle;
import android.support.annotation.Nullable;
import android.support.v4.app.Fragment;
import android.support.v4.app.FragmentManager;
import android.support.v4.app.FragmentTransaction;
import android.view.LayoutInflater;
import android.view.View;
import android.view.View.OnClickListener;
import android.view.ViewGroup;
import android.widget.AdapterView;
import android.widget.AdapterView.OnItemClickListener;
import android.widget.ArrayAdapter;
import android.widget.Button;
import android.widget.EditText;
import android.widget.ListView;
import android.widget.Spinner;
import android.widget.TextView;
import android.widget.Toast;

public class Fragment_Users_TipsList extends Fragment implements
      OnClickListener {
   View view;
   // db operations variable declarations
   String conn_state;
   String myresult;
```

```
    ProgressDialog pDialog;
    ListView lview;
    ArrayAdapter<String> adapter;
    String[] splittedresult;
    String retrieve_url = "http://www.mayombo.com/nutrimobilescripts/nutriiontips.php";
    FragmentManager fm;
    FragmentTransaction ft;
    Fragment myfragment = null;
    String tipselected, receivedemail, title, details, reqid;
    Button refresh, request;
    TextView tvmail;
    EditText edt_tittle, edt_details;
    int id;
    Spinner spcategories;
    String[] tips_categories = {"Food", "Fitness", "Body Weight", "Pre natal care", "Post natal
Care"};
    @Override public View onCreateView(LayoutInflater inflater,
                        @Nullable ViewGroup container, @Nullable Bundle savedInstanceState) {
        // TODO Auto-generated method stub
        view = inflater.inflate(R.layout.fragment_user_tips, container, false);
        Bundle b = getArguments();
        receivedemail = b.getString("useremail");
        refresh = (Button) view.findViewById(R.id.btnrefresh);
        request = (Button) view.findViewById(R.id.btnrequest);
        spcategories = (Spinner) view.findViewById(R.id.spcategories);
        adapter = new ArrayAdapter<String>(getActivity(), R.layout.spinnerdesign,
R.id.spinner_text, tips_categories);
        spcategories.setAdapter(adapter);
        refresh.setOnClickListener(this);
        request.setOnClickListener(this);
        new RetrieveNutritionTips().execute();
```

```
    return view;
}
@Override public void onClick(View v) {
    // TODO Auto-generated method stub
    switch (v.getId()) {
        case R.id.btnrefresh:
            new RetrieveNutritionTips().execute();
            break;
        case R.id.btnrequest:
            final Dialog dialog = new Dialog(getActivity());
            dialog.setContentView(R.layout.dialog_requesttip);
            dialog.setTitle("Request Nutrition Tip");
            tvmail = (TextView) dialog.findViewById(R.id.tvemail);
            edt_tittle = (EditText) dialog.findViewById(R.id.edttitle);
            edt_details = (EditText) dialog
                    .findViewById(R.id.edtrequestdetails);
            Button submit = (Button) dialog.findViewById(R.id.btnsubmit);
            // if button is clicked, close the custom dialog
            submit.setOnClickListener(new OnClickListener() {
                @Override public void onClick(View v) {
                    title = edt_tittle.getText().toString();
                    details = edt_details.getText().toString();
                    Random rn = new Random();
                    id = rn.nextInt(10000);
                    reqid = "reqid:" + id;
                    new InsertReqeuest().execute();
                    dialog.dismiss();
                }
            });
            dialog.show();
```

```
            break;
        default:
            break;
    }
}
class RetrieveNutritionTips extends AsyncTask<String, String, String> {
    @Override protected void onPreExecute() {
        // TODO Auto-generated method stub
        super.onPreExecute();
        // Asynch variables
        pDialog = new ProgressDialog(getActivity());
        pDialog.setMessage("Retrieving data...");
        pDialog.setIndeterminate(false);
        pDialog.setCancelable(false);
        pDialog.show();
    }
    @Override protected String doInBackground(String... arg0) {
        // TODO Auto-generated method stub
        try {
            DefaultHttpClient httpclient = new DefaultHttpClient();
            HttpPost httppost = new HttpPost(retrieve_url);
            // Execute HTTP Post Request
            HttpResponse response = httpclient.execute(httppost);
            InputStream inputStream = response.getEntity().getContent();
BufferedReader rd = new BufferedReader(new InputStreamReader(inputStream), 4096);
            String line;
            StringBuilder sb = new StringBuilder();
            while ((line = rd.readLine()) != null) {
               sb.append(line);
            }
```

```
            rd.close();
            myresult = sb.toString();
            inputStream.close();
        } catch (Exception e) {
            Toast.makeText(getActivity(),
                    "Error inside set:" + e.toString(), Toast.LENGTH_LONG) .show();
        }
        return myresult;
    }
    @Override protected void onPostExecute(String result) {
        // TODO Auto-generated method stub
        super.onPostExecute(result);
        pDialog.dismiss();
        // received data from the server split them and store on String
        // array
        final String[] splittedresult = myresult.split("#");
        // use data inside string array to generate list of items
        ListView lview = (ListView) view.findViewById(R.id.lvtips);
        ArrayAdapter<String> adapter = new ArrayAdapter<String>(
                getActivity(), R.layout.list_data, R.id.listtext, splittedresult);
        lview.setAdapter(adapter);
        lview.setOnItemClickListener(new OnItemClickListener() {
            @Override public void onItemClick(AdapterView<?> parent, View view,
                                int position, long id) {
                // TODO Auto-generated method stub
                tipselected = splittedresult[position];
                myfragment = new Fragment_User_TipsDetails();
                Bundle mybundle = new Bundle();
                mybundle.putString("tipselected", tipselected);
                myfragment.setArguments(mybundle);
```

```
            fm = getFragmentManager();
            ft = fm.beginTransaction();
            ft.replace(R.id.maincontent, myfragment);
            ft.commit();
        }
    });
  }
}
class InsertReqeuest extends AsyncTask<String, String, String> {
  @Override protected void onPreExecute() {
    // TODO Auto-generated method stub
    super.onPreExecute();
    // Asynch variables
    pDialog = new ProgressDialog(getActivity());
    pDialog.setMessage("Please Wait...");
    pDialog.setIndeterminate(false);
    pDialog.setCancelable(false);
    pDialog.show();
  }
  @Override protected String doInBackground(String... arg0) {
    // TODO Auto-generated method stub
    try { /* seting up the connection and send data with url */
      // create a http default client - initialize the HTTp client
      DefaultHttpClient httpclient = new DefaultHttpClient();
      // Create a HTTp post object to hold our data - url
      // the IP is used when testing locally but for online server you
      // write the actual server name
      // e.g "http://maasaimarket.site88.net/maasaimarket/insert.php"
      HttpPost httppost = new HttpPost(
          "http://www.mayombo.com/nutrimobilescripts/inserttiprequest.php");
```

```
        // use HTTPClient to execute the HTTPPost
        // Execute HTTP Post Request
        // encode URL
        ArrayList<NameValuePair> nameValuePairs = new
ArrayList<NameValuePair>(3);
        // must be in pairs like localhost.....php?id=username.getText()
        nameValuePairs.add(new BasicNameValuePair("reqid", reqid));
        nameValuePairs.add(new BasicNameValuePair("useremail",
              receivedemail));
        nameValuePairs.add(new BasicNameValuePair("tittle", title));
        nameValuePairs.add(new BasicNameValuePair("details", details));
        httppost.setEntity(new UrlEncodedFormEntity(nameValuePairs));
        HttpResponse response = httpclient.execute(httppost);
                              /* receiving response from the database */
        // use Input stream to read the http client response
        InputStream inputStream = response.getEntity().getContent();
        // use buffered reader and InputStreamReader to read the input
        // stream
        BufferedReader rd = new BufferedReader(new InputStreamReader(
              inputStream), 4096);
        String line;
        // initialize StringBuilder
        StringBuilder sb = new StringBuilder();
        // read everything from the Buffered reader and append the to
        // the
        // string builder
        while ((line = rd.readLine()) != null) {
           sb.append(line);
        }
        rd.close();
```

```
            // our result
            myresult = sb.toString();
            inputStream.close();
        } catch (Exception e) {
            Toast.makeText(getActivity(),
                    "Error inside set:" + e.toString(), Toast.LENGTH_LONG)
                    .show();
        }
        return myresult;
    }
    @Override protected void onPostExecute(String result) {
        // TODO Auto-generated method stub
        super.onPostExecute(result);
        pDialog.dismiss();
        // check if response is 4
        if (myresult.equals("1")) {
            Toast.makeText(getActivity(), "Request Sent Sucessfully",
                    Toast.LENGTH_LONG).show();
        }
        // /check if response is 5
        else if (myresult.equals("0")) {
            Toast.makeText(getActivity(), "Please Try Again!",
                    Toast.LENGTH_LONG).show();
        }
    }
  }
}
```

Appendix 6: Home dashboard

```
package com.neemamduma.nutrimobile;

import android.app.AlertDialog;
```

```
import android.content.DialogInterface;
import android.os.Bundle;
import android.support.annotation.Nullable;
import android.support.v4.app.Fragment;
import android.support.v4.app.FragmentManager;
import android.support.v4.app.FragmentTransaction;
import android.view.LayoutInflater;
import android.view.View;
import android.view.View.OnClickListener;
import android.view.ViewGroup;
import android.widget.Button;
import android.widget.TextView;

public class Fragment_HomeDashBoard extends Fragment implements OnClickListener {
   Button viewall, search;
   FragmentManager fm;
   FragmentTransaction ft;
   Fragment myfragment = null;
   String receivedemail, receivedusertype;
   Bundle mybundle;
   TextView tvusertype;
   @Override public View onCreateView(LayoutInflater inflater,
                  @Nullable ViewGroup container, @Nullable Bundle savedInstanceState) {
      // TODO Auto-generated method stub
      View v = inflater.inflate(R.layout.fragment_doctordashboard, container,
            false);
      Bundle b = getArguments();
      receivedemail = b.getString("useremail");
      receivedusertype = b.getString("usertype");
      tvusertype = (TextView) v.findViewById(R.id.tvusertype);
      tvusertype.setText("User : " + receivedusertype);
```

```
    viewall = (Button) v.findViewById(R.id.btnviewall);
    search = (Button) v.findViewById(R.id.btnsearch);
    viewall.setOnClickListener(this);
    search.setOnClickListener(this);
    return v;
}
@Override public void onClick(View v) {
    // TODO Auto-generated method stub
    switch (v.getId()) {
        case R.id.btnviewall:
            myfragment = new Fragment_PatientListByID();
            fm = getFragmentManager();
            ft = fm.beginTransaction();
            ft.replace(R.id.maincontent, myfragment);
            ft.addToBackStack(null);
            ft.commit();
            break;
        case R.id.btnsearch:
            AlertDialog.Builder myalert = new AlertDialog.Builder(getActivity());
            myalert.setTitle("NutriMobile Options");
            myalert.setMessage("Select an Action to be performed");
            // icon from drawable folder
            // myalert.setIcon(R.drawable.about);
            // icon from android library
            myalert.setIcon(getResources().getDrawable(
                    android.R.drawable.ic_dialog_info));
            myalert.setCancelable(false);
            myalert.setPositiveButton("View Requested Tips",
                    new DialogInterface.OnClickListener() {
                        @Override public void onClick(DialogInterface arg0, int arg1) {
```

```
            // TODO Auto-generated method stub
            myfragment = new Fragment_Nutritionist_RequestedTips();
            fm = getFragmentManager();
            ft = fm.beginTransaction();
            ft.replace(R.id.maincontent, myfragment);
            ft.addToBackStack(null);
            ft.commit();
         }
      });
myalert.setNegativeButton("Add New Tip",
      new DialogInterface.OnClickListener() {
         @Override public void onClick(DialogInterface arg0, int arg1) {
            myfragment = new Fragment_Nutritionist_NewTip();
            mybundle = new Bundle();
            mybundle.putString("useremail", receivedemail);
            myfragment.setArguments(mybundle);
            fm = getFragmentManager();
            ft = fm.beginTransaction();
            ft.replace(R.id.maincontent, myfragment);
            ft.addToBackStack(null);
            ft.commit();
         }
      });
myalert.setNeutralButton("Cancel",
      new DialogInterface.OnClickListener() {
         @Override public void onClick(DialogInterface dialog, int arg1) {
            // TODO Auto-generated method stub
            dialog.cancel();
         }
      });
```

```
            myalert.show();
            break;
        default:
            break;
    }
  }
}
```

Appendix 7: Recommendation details

```
package com.neemamduma.nutrimobile;

import java.io.BufferedReader;

import java.io.InputStream;

import java.io.InputStreamReader;

import java.util.ArrayList;

import org.apache.http.HttpResponse;
import org.apache.http.NameValuePair;

import org.apache.http.client.entity.UrlEncodedFormEntity;

import org.apache.http.client.methods.HttpPost;

import org.apache.http.impl.client.DefaultHttpClient;

import org.apache.http.message.BasicNameValuePair;

import android.app.ProgressDialog;
import android.os.AsyncTask;

import android.os.Bundle;

import android.support.annotation.Nullable;

import android.support.v4.app.Fragment;

import android.view.LayoutInflater;

import android.view.View;

import android.view.ViewGroup;

import android.widget.TextView;

import android.widget.Toast;

public class Fragment_NutritionistRecco_Nutritionist extends Fragment {
```

```
        TextView nutritionistname,username, dates, recco;
        String receievedemail, reveivedname, receiveddate;
        String[] splittedresult;
        // db operations variable declarations
        String conn_state;
        String myresult;
        ProgressDialog pDialog;
        @Override public View onCreateView(LayoutInflater inflater,
                        @Nullable ViewGroup container, @Nullable Bundle
savedInstanceState) {
                // TODO Auto-generated method stub
View v = inflater.inflate(R.layout.fragment_nutritionistrecco_nutritionist, container, false);
                Bundle b = getArguments();
                receievedemail = b.getString("receivedemail");
                reveivedname = b.getString("username");
                receiveddate = b.getString("dateselected");
                doctorname = (TextView) v.findViewById(R.id.tvdoctorname);
                patientname = (TextView) v.findViewById(R.id.tvpatientname);
                dates = (TextView) v.findViewById(R.id.tvdates);
                recco = (TextView) v.findViewById(R.id.tvrecco);
                patientname.setText("Patient Name: "+reveivedname);
                new RetrieveRecco().execute();
                return v;
        }
        class RetrieveRecco extends AsyncTask<String, String, String> {
                @Override protected void onPreExecute() {
                        // TODO Auto-generated method stub
                        super.onPreExecute();
                        // Asynch variables
                        pDialog = new ProgressDialog(getActivity());
```

```
                pDialog.setMessage("Retrieving Reccomendations..Please Wait..");
                pDialog.setIndeterminate(false);
                pDialog.setCancelable(false);
                pDialog.show();
        }
        @Override protected String doInBackground(String... arg0) {
                // TODO Auto-generated method stub
                try {
                        DefaultHttpClient httpclient = new DefaultHttpClient();
                        HttpPost httppost = new HttpPost(
                "http://www.mayombo.com/nutrimobilescripts/retrieverecco.php"
                                        // change to this to test direct to a phone
                                        //
"http://phone_ip_address/nutrimobilescripts/retrieverecco.php"; );
                        // Add your data
                        ArrayList<NameValuePair> nameValuePairs = new
ArrayList<NameValuePair>(1);
nameValuePairs .add(new BasicNameValuePair("name", reveivedname));
nameValuePairs.add(new BasicNameValuePair("email",receievedemail));
nameValuePairs.add(new BasicNameValuePair("dateselected",receiveddate));
                httppost.setEntity(new UrlEncodedFormEntity(nameValuePairs));
                        // Execute HTTP Post Request
                        HttpResponse response = httpclient.execute(httppost);
                        InputStream inputStream = response.getEntity().getContent();
                BufferedReader rd = new BufferedReader(new InputStreamReader(
                                        inputStream), 4096);
                        String line;
                        StringBuilder sb = new StringBuilder();
                        while ((line = rd.readLine()) != null) {
                                sb.append(line);
                        }
```

```
                    rd.close();
                    myresult = sb.toString();
                    inputStream.close();
                } catch (Exception e) {
                    Toast.makeText(getActivity(),
                            "Error inside set:" + e.toString(),
Toast.LENGTH_LONG).show();
                }
                return myresult;
            }
            @Override protected void onPostExecute(String result) {
                // TODO Auto-generated method stub
                super.onPostExecute(result);
                pDialog.dismiss();
                splittedresult = myresult.split("#");
                recco.setText(splittedresult[0]);
                nutritionistname.setText(splittedresult[1]);
                dates.setText("Dated : "+splittedresult[2]);
            }
        }
}
```