ADDIS ABABA UNIVERSITY SCHOOL OF INFORMATION SCIENCE AND SCHOOL OF PUBLIC HEALTH

MSc in HEALTH INFORMATICS PROGRAM

Automating an Integrated Management of Newborn and Childhood Illness (e-IMNCI): Optimize real-time clinical decision support system

By:

TUHA MUSTEFA AHMED

JUNE 2014

ADDIS ABABA, ETHIOPIA

ADDIS ABABA UNIVERSITY SCHOOL OF INFORMATION SCIENCE AND SCHOOL OF PUBLIC HEALTH

Automating an Integrated Management of Newborn and Childhood Illness (e-IMNCI): Optimize real-time clinical decision support system

A Project Submitted to the School of Information Science and Public Health of Addis Ababa University in Partial Fulfillment of the Requirements for Degree of Master of Science in Health Informatics

By:
TUHA MUSTEFA AHMED

JUNE 2014

ADDIS ABABA, ETHIOPIA

ADDIS ABABA UNIVERSITY SCHOOL OF INFORMATION SCIENCE AND SCHOOL OF PUBLIC HEALTH

Automating an Integrated Management of Newborn and Childhood Illness (e-IMNCI): Optimize real-time clinical decision support system

By:
TUHA MUSTEFA AHMED

Names and Signature of Members of the Examining Board

1.	Examiner	Signature	Date
2.	Examiner	Signature	Date
3.	Advisor	Signature	Date
4.	Advisor	Signature	Date

DEDICATION

To my beloved brother Rufael Gebeyehu (Babi)

ACKNOWLEDGEMENTS

My special thanks goes to my advisors Dr. Martha Yifru and Dr. Ayele Belachew for their commitment, encouragement, valuable comments, stimulating advice, and constructive suggestions who helped me to pass the difficulties I faced in the course of conducting the project and finish my project. My grateful thanks go to all staff members of Addis Ababa University, School of Public health and Information science for their coordination and facilitation efforts toward my achievement.

I would like to extend my thanks to Dr. Aderajew Sendek (MD in Hiwot-Fana Hospital, Harar), who freely gave his time and valuable information necessary for the project work. My special thanks are also due to all my friends for their all rounded contribution and support in the course of conducting this project work. To all others who helped me in completing this project, I am really grateful.

I would like also to extend my thanks and appreciation to health workers in Janmed and Yeka Worda 4 for providing me with necessary valuable information

Last but not least, I owe my sincere gratitude to my family members. Especially, to my beloved brother Babi for your inspiration to join this department.

.

Table of Contents

List of Figures

List of Tables

List of Acronyms

ADT	Adaptive Turnaround Document
Aka	aka also known as
CDS	Clinical Decision Support
CDSS	Clinical Decision Support System
ECG	Electro Cardio Graph
e-IMCI	Electronic Integrated Management of Childhood Illness
e-IMNCI	Electronic Integrated Management of Newborn and Childhood Illness
EMR	EMR Electronic Medical Record
ICD-10	International Classification of Disease, 10th Version
IMCI	Integrated Management of Childhood Illness
IMNCI	Integrated Management of Newborn and Childhood Illness
MRI	Magnetic Resonance Imaging
OPD	Out Patient Department
QMR	QMR Quick Medical Reference
SDLC	Software Development Life Cycle
SQL	Structured Query Language
UML	Unified Modeling Language
UNICEF	United Nations International Children's Fund
WHO	World Health Organization
WWW	World Wide Web

ABSTRACT

Background: Every year almost 10 million children die before reaching the age of five despite the fact that two-thirds of these deaths could be prevented by effective low-cost interventions. To combat this, the World Health Organization (WHO) and UNICEF developed the Integrated Management of Childhood Illness (IMCI) treatment guidelines. In Ethiopia, IMCI is used as a guideline for the treatment of childhood illness. Currently, health centers use paper based guideline of IMNCI. The e-IMNCI is the electronic version of IMNCI – a computer-based Decision Support System based on IMNCI. The eIMNCI follows the same protocol with the current paper based guidelines used. The e-IMNCI will serve as clinical decision supporting system, the aim of CDS is to assist, rather than to replacing the health professionals.

Objective: The objective of this project is to automate IMNCI to improve the clinical decision making process.

Methodology: To achieve the objective of this project, requirements was identified and collected by applying interview, observation and relevant document review techniques. Object oriented system analysis and design methodology is used to develop the system. Analysis, design and implementation of the system were performed by using the unified modeling language tools for analysis and design, and C# programming language and Microsoft Structured Query Language for implementation stage of the system. At the end of system development phase, pilot testing was done to check the performance and usability of the prototype system by using Morae TechSmith software.

Discussion of Results: System analysis done by using functional, object and dynamic models. The functional model is described by use case diagrams, object model is described by class diagram, and the dynamic model is described by sequence diagrams. Both performance and usability test was conducted to measure the performance of the system and to describe to what extent the system is usable. The result of system usability testing shows 83.4% willingness to use the system easily. And also the accuracy of the system's clinical decision supports module is 94.44%.

Conclusion: Generally, the prototype system serve as a guideline, patient information management system and diagnosis assistant. The initial feedback from health works has been extremely positive. Hence the prototype system achieves a good performance and meets the objectives of the project. However, in order to make the system applicable in the domain area, some adjustments like including more disease and symptoms information in the rules of knowledge base of the system, and changing its passive mode to active mode of information delivery will be made to handle the limitation of the system.

CHAPTER ONE

INTRODUCTION

1.1 Background

The World Health Organization (WHO) and United Nations International Children's Fund (UNICEF) developed Integrated Management of Childhood Illness guidelines (IMCI) in the mid-1990s to enable a holistic approach to the care of children presenting on developing-country healthcare facilities with symptoms of common childhood illnesses [1].

The IMCI guidelines describe how to care for a child who is brought to a clinic with an illness, or for a scheduled follow-up visit to check the child's progress. The guidelines give instructions on how to routinely assess a child for general danger signs (or possible bacterial infection in a young infant), common illnesses, malnutrition and anemia, and to look for other problems. In addition to treatment, the guidelines incorporate basic activities for illness prevention [1].

The IMCI process can be used by doctors, nurses and other health professionals who see sick infants and children aged from one week up to five years. It is a case management process for a first-level facility such as a clinic, a health center or an out-patient department of a hospital [2].

The IMNCI Module, together with Antenatal, Labor, Delivery and Postnatal Care, provides a knowledge-base for many of the critical health services that Health Extension Workers and Health Extension Practitioners deliver for infants and children and their families. You will find that some of the concepts and competencies in the IMNCI Module are also covered in other Modules in the Level IV Health Extension Practitioner Program. This is because IMNCI is based on international guidelines and protocols and therefore a Module on the IMNCI strategy needs to have all elements of the approach included. The complete IMCI case management process involves the following elements [3]:

1. **Assess** a child by checking first for danger signs (or possible bacterial infection in a young infant), asking questions about common conditions, examining the child, and checking nutrition and immunization status. Assessment includes checking the child for other health problems.

2. **Classify** a child's illnesses using a color-coded triage system. Because many children have more than one condition, each illness is classified according to whether it requires:

    - Urgent pre-referral treatment and referral (red), or
    - Specific medical treatment and advice (yellow), or
    - Simple advice on home management (green).

3. After classifying all conditions, **identify** specific treatments for the child. If a child requires urgent referral, give essential treatment before the patient is transferred. If a child needs treatment at home, develop an integrated treatment plan for the child and give the first dose of drugs in the clinic. If a child should be immunized, give immunizations.

4. Provide practical **treatment** instructions, including teaching the care taker how to give oral drugs, how to feed and give fluids during illness, and how to treat local infections at home. Ask the caretaker to return for follow-up on a specific date, and teach her how to recognize signs that indicate the child should return immediately to the health facility.

5. Assess feeding, including assessment of breastfeeding practices, and **counsel** to solve any feeding problems found. Then counsel the mother about her own health.

6. When a child is brought back to the clinic as requested, give **follow-up** care and, if necessary, reassess the child for new problems.

For example, if nurses observe/found a problem which is related to malnutrition and anemia then figure1 is used to classify the observed problems.

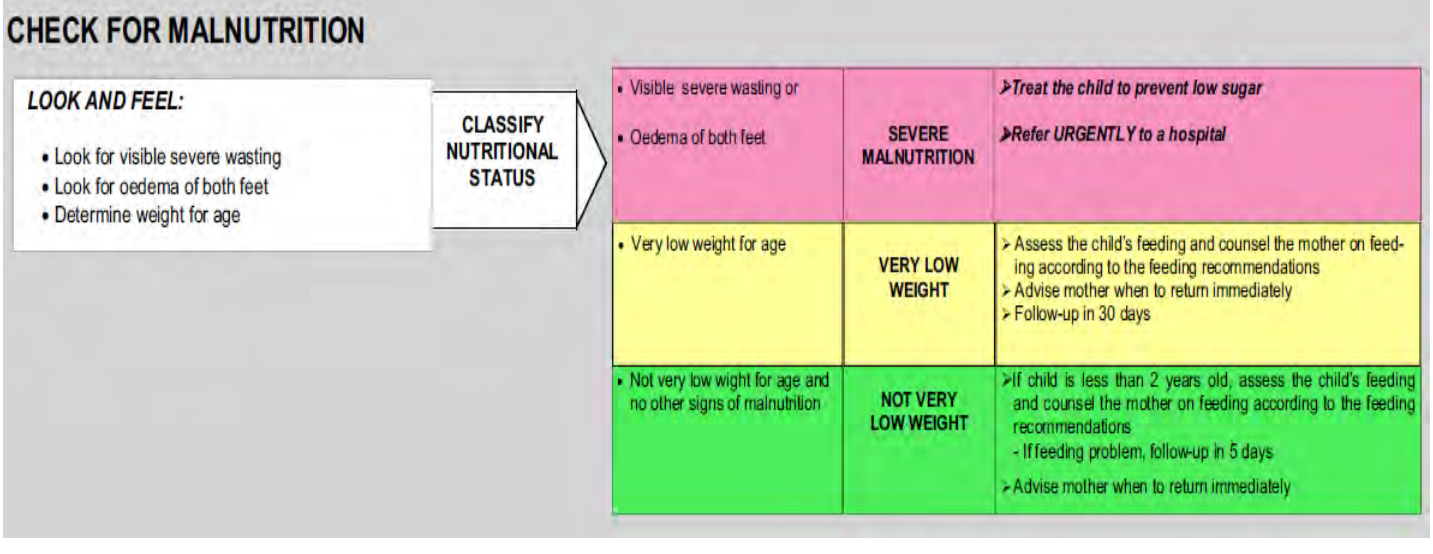


Figure 1.1: A guideline to check and classify for malnutrition and Anemia [2].

1.1.1 Electronic Integrated Management of Newborn and Childhood Illness (e-IMNCI)

The e-IMNCI is the electronic version of IMNCI – a computer-based Decision Support system based on IMNCI [5]. This project is developed under the international guidelines and protocols which are developed by World Health Organization (WHO). The electronic IMNCI can be developed for cell phone, PDA, tablets and computer to guide health workers through the full IMNCI assessment, classification and treatment plan in order to deliver a service with quality. It can also be used to improve communication between health works and child caretaker about how to treat the child and watch for danger signs in children aged one week to five years old.

The e-IMNCI can reduce skipped steps, branching-logic errors, and miscalculations. Second, training time can be reduced because the algorithm itself does not need to be as rigorously taught. Since the software automatically navigates through the IMCI chart, they expected it to be more efficient than paper-based methods where the clinician must determine the next question. Additionally, more sophisticated protocols can be deployed, as the design of IMCI was constrained by what could be practically included in paper flipcharts. Similarly, updating electronic protocols is dramatically easier than on paper. Finally, the data from e-IMCI can be collected to assist with clinic supervision and to provide program managers and policy makers with a wealth of population health data. There are many potential benefits of e-IMCI compared to the current paper-based approach.

1.1.2 Clinical Decision Support System (CDSS)

Clinical decision support (CDS) systems provide clinicians, staff, patients, and other individuals with knowledge and person-specific information, intelligently filtered and presented at appropriate times, to enhance health and health care. The most common use of CDS is for addressing clinical needs, such as ensuring accurate diagnoses, screening in a timely manner for preventable diseases, or averting adverse drug events. However, CDS can also potentially lower costs, improve efficiency, and reduce patient inconvenience. In fact, CDS can sometimes address all three of these areas simultaneously—for example, by alerting clinicians to potentially duplicative testing. For more complex cognitive tasks, such as diagnostic decision making, the aim of CDS is to assist, rather than to replace, the clinician, whereas for other tasks (such as presentation of a predefined order set) the CDS may relieve the clinician of the burden of reconstructing orders for each encounter [7]

The CDS may offer suggestions, but the clinician must filter the information, review the suggestions, and decide whether to take action or what action to take. Table 1 below provides examples of CDS that address a range of target areas. For more examples of how various types of CDS can be applied to address specific improvement objectives see the work of Osheroff and his colleagues [7].

Target Area of Care	**Example**
Preventive care	Immunization, screening, disease management guidelines for secondary prevention
Diagnosis	Suggestions for possible diagnoses that match a patient's signs and symptoms
Planning or implementing treatment	Treatment guidelines for specific diagnoses, drug dosage recommendations, alerts for drug-drug interactions
Follow-up management	Corollary orders, reminders for drug adverse event monitoring
Hospital, provider efficiency	Care plans to minimize length of stay, order sets
Cost reductions and improved patient convenience	Duplicate testing alerts, drug formulary guidelines

Table 1.1.Examples of CDS interventions by target area of care.

1.2 Statement of the Problem

Every year, about 10 million children die before reaching their fifth birthday. Over 70% of these deaths, the vast majority occurring in the developing world are due to the respiratory infections, diarrheal diseases, malarial, measles and malnutrition, often in combination. Nevertheless, WHO and UNICEF have developed Integrated Management of Childhood Illness guidelines (IMCI) in 1995 to reduce the mortality and morbidity associated with the major cause of the childhood illness. The implementation of the strategy and guidelines has achieved impressive results both in reducing child mortality and in improving the quality of life of children all over the world. The IMCI guidelines describe how to care for a child who is brought to a clinic with an illness, or for a scheduled follow-up visit to check the child's progress. The guidelines give instructions on how to routinely assess a child for general danger signs (or possible bacterial infection in a young infant), common illnesses, malnutrition and anemia, and to look for other problems. In addition to treatment, the guidelines incorporate basic activities for illness prevention.

Currently, all health centers in Ethiopia are using a paper based IMNCI guideline as a reference and OPD register book as data recording tool that is used as an input for clinical decision making for the services delivery of infant and child health. However, the current system is really challenging to access patient's diagnosis history and to generate a report for further decision making process. For example it is difficult to generate reports like percentage of vaccination level, prevalence of epidemic disease among children and so on.

Some of the problems in the existing manual system are:

- Health professionals rarely follow the recommendation of using the paper chart booklet during encounters because it is perceived as taking too long, and instead rely on their memory.
- Difficulty to generate a report from the under 5 years out patient department register.
- Duplication of patient's data.
- Missing essential information when nurse registers patient data.

As a solution for the problems related with paper based IMCI, some researchers and developers worked towards the development of the electronic version of the IMCI and adaption of it. In Tanzania, the electronic IMCI is developed that runs on personal digital assistant (PDA) to guide health worker step-by-step through the IMCI treatment algorithm that is capable to reduce skipped steps, branching-logic errors, and miscalculations. The eIMCI tool provided classification, treatment and communication recommendations in accordance to the IMCI protocol [8]. In another work, WHO and Novartis foundation for sustainable development developed IMCI Computerized Adaption Training Tool (ICATT) which is used for the adaption of generic IMCI guidelines at national and sub national levels, and to develop training courses that fit into various training approaches [9]. ICATT allows adapting and updating IMCI guidelines, to develop training courses, and to be translated into different languages and used in a range of environments and settings.

Even if they developed the electronic version of IMNCI they didn't include new born care treatment and data record management. The purpose of this project is, therefore, to develop the electronic version of IMNCI guideline that runs on computers. This project includes all the things in IMNCI protocol to guide health workers through the full IMNCI assessment, classification and treatment plan. In addition to the guideline the system also includes patient information management and diagnosis assistant to provide support for clinical decision making process.

1.3 Objectives of the Project

1.3.1 General Objective

The general objective of this project is to automate an integrated management of newborn and childhood illness (IMNCI) to improve the clinical decision making process of health centers.

1.3.2 Specific Objectives

The specific objectives of the project are:

- To collect and analyze the required data for the development of the system;
- To design the e-IMNCI and the decision support system;
- To write and test a code;
- Testing the system at different perspectives;
- Performing a pilot test to check the performance and usability of the system;
- Maintain the system if there is performance and usability related problems after pilot testing session is conducted.

1.4 Scope of the project and Limitations

1.4.1 Scope of the project

This system will be used as record management system (only for patients under 5 year's children), clinical decision support system, report generating system, and as a guideline. The e-IMNCI protocol follows the same protocol in the current paper based system, guiding healthcare workers step by step for child's problem assessment, classification, treatment, and communication of instructions to the caregiver. Some of the modules of the system are:

1. **Registration Module:**

   - The system **registers a patient's detail information**. It gives a unique medical record number for every
   - After the patient is registered in the system, the system will provide a page to **register the diagnosis and treatment result**. It registers one patient's many diagnoses and treatment result in to one account.

2. **Record Tracking Module:**

   - Managing of information such as storing, updating, searching, information efficiently.
   - Generate various types of report.
   - Follow-up

3. **Decision Support Module:**

   - Suggestions for possible diagnoses that match the observed patient's signs and symptoms.
   - Treatment guidelines for specific diagnoses and drug dosage recommendations.

4. **Guideline Module:**

   - Dynamic IMNCI guideline(no need to see the whole guideline to find out a guide for a single problem rather the system displays a guide for a specific disease depending on user selection).

1.4.2 Limitations of the Project

Through the course of conducting this project, it was learned that health information management systems plays a great role towards the enhancement of health care service quality. Furthermore, clinical decision support systems help health professionals to make patient care decisions based on patient specific data and latest research findings. Even though, the initial feedback from health workers has been positive, the system had some limitations especially in clinical decision support module, which can affect the applicability of the system in the domain area. Among these limitations, rule of the knowledge base is developed in static way, and the system is using passive mode of information transmission.

1.5 Significance of the Project

Despite, the listed problems in the statements of the problem part will be addressed by developing a system that has features like registering patient information; generate reports and assisting the diagnosis procedure and treatment plan. To minimize infrequent use of the guideline, the system provides a link to open the corresponding guideline booklet chart when the user classifies the observed problem. The system simplifies the existing working process, allowing the users to use the guideline consistently, minimizing the risk of diagnosis and treatment error, the users can learn the system easily and it provides a dynamic diagnosis assistance whether the user accept it or not.

The prototype system serves as a guideline, patient information record and as CDSS by generating reports, provide diagnosis and treatment plan suggestions for possible diagnosis that match a patient's signs and symptoms. It has a referencing tool by simplifying the guideline style which they used before. The prototype system will have a positive effect on giving health professionals more time to engage on other tasks. At the successful completion of this project the health centers' way of operation will be improved. The overall activities will be easier; working process will be fast and effective than it was before. Some of the benefits of using the system at different perspective described as follows:

For health centers:

- The service delivery will be fast and effective by adopting the system's consistent and simple data management features.
- Unlike the current manual system, the system creates a uniformed working process and data recording format to replace the current recording style on the OPD register.

For health center managers:

- To generate different reports with in short period of time. Based on the selected report parameter the system generates a report like monthly IMNCI report, case report, and other reports.
- To get the right information at the right time.

For health professionals:

- To minimize the work load of health professionals.
- The system allows the users to access patient information easily.
- The system reduces diagnosis and treatment errors made by health professionals by providing diagnosis assistant, data validation and verification techniques.

.

For patients

- Patients can get correct and reliable information from the health center through health professions within a short period of time.
- Get better service, by presenting the entire patient's previous diagnosis history, the health workers uses the history to diagnose the current problems. This will help to give a better service for the patients.

CHAPTER TWO

LITERATURE REVIEW

2.1 GENERAL LITERATURE

Introduction

Patient data must be adequate to make a valid decision. The problem arises, when the clinician is met with an overwhelming amount of specific and unspecific data, which he/she cannot satisfactorily process. Therefore it is important to assess when additional facts will confuse rather than clarify the patient's case. For example, usual settings for such a problem are intensive-care units, where practitioners must absorb large amounts of data from various monitors, be aware of the clinical status, patient history, accompanying chronic illnesses, patient's medication and adverse drug interactions, etc. - and on top of that make an appropriate decision about the course of action.

The quality of available data is of equal importance. Measuring instruments and monitors should be as accurate as technologically possible, since erroneous data could have serious adverse effect on patient-care decisions. Knowledge used in decision-making process must be accurate and current. It is of major importance that the deciding clinician has a broad spectrum of medical knowledge and access to information resources, where it is possible to constantly revise and validate that knowledge. For a patient to receive appropriate care, the clinician must be aware of the latest evidence-based guidelines and developments in the area of the case in question. It is in clinician's hands to bring proven therapies from research papers to the bedside. CDSS analogously needs an extensive, well-structured and current source of knowledge to appropriately serve the clinician. By incorporating patient-specific data and evidence-based guidelines or applicable knowledge base, the CDSS can improve quality of care with enhancing the clinical decision making process. In order to be able to construct applicable CDSS, it is imperative to have a broader-based understanding of medical decision-making as it occurs in the natural setting. Designing CDSSs without understanding the cognitive processes underlying medical reasoning and decision analysis is pliable for ineffectiveness and failure for implementation into everyday clinical workflow [17].

2.1.1 Integrated Management of Newborn and Childhood Illness

Since the 1970s, the estimated annual number of deaths among children less than 5 years old has decreased by almost a third. This reduction, however, has been very uneven. And in some countries rates of childhood mortality are increasing. In 1998, more than 50 countries still had childhood mortality rates of over 100 per 1000 live births [1].

Altogether more than 10 million children die each year in developing countries before they reach their fifth birthday. Seven in ten of these deaths are due to acute respiratory infections (mostly pneumonia), diarrhea, measles, malaria, or malnutrition—and often to a combination of these conditions

Providing quality care to sick children in these conditions is a serious challenge. In response to this challenge, WHO and UNICEF developed a strategy known as Integrated Management of Childhood Illness (IMCI). Although the major stimulus for IMCI came from the needs of curative care, the strategy combines improved management of childhood illness with aspects of nutrition, immunization, and other important disease prevention and health promotion elements. The objectives are to reduce deaths and the frequency and severity of illness and disability and to contribute to improved growth and development.

The strategy includes three main components:

- Improvements in the case-management skills of health staff through the provision of locally adapted guidelines on IMCI and through activities to promote their use
- Improvements in the health system required for effective management of childhood illness
- Improvements in family and community practices

The core of the IMCI strategy is integrated case management of the most common childhood problems, with a focus on the most important causes of death. The generic guidelines, however, are not designed for immediate use. A guided process of adaptation ensures that the guidelines, and the learning materials that go with them, reflect the epidemiology within a country and are tailored to fit the needs, resources and capacity of a country's health system.

The clinical guidelines, which are based on expert clinical opinion and research results, are designed for the management of sick children aged 1 week up to 5 years. They promote evidence-based assessment and management, using a syndromic approach that supports the rational, effective and affordable use of drugs. They include methods for assessing signs that indicate severe disease; assessing a child's nutrition, immunization, and feeding; teaching parents how to care for a child at home; counseling parents to solve feeding problems; and advising parents about when to return to a health facility. The guidelines also include recommendations for checking the parents' understanding of the advice given and for showing them how to administer the first dose of treatment.

When assessing a sick child, a combination of individual signs leads to one or more classifications, rather than to a diagnosis. IMCI classifications are action oriented and allow a health care provider to determine if a child should be urgently referred to another health facility, if the child can be treated at the first-level facility (e.g. with oral antibiotic, antimalarial, ORS, etc.), or if the child can be safely managed at home.

Integrated case management relies on case detection using simple clinical signs and empirical treatment. As few clinical signs as possible are used. The signs are based on expert clinical opinion and research results, and strike a careful balance between sensitivity and specificity. The treatments are developed

according to action oriented classifications rather than exact diagnosis. They cover the most likely diseases represented by each classification.

The IMCI guidelines describe how to care for a child who is brought to a clinic with an illness, or for a scheduled follow-up visit to check the child's progress. The guidelines give instructions for how to routinely assess a child for general danger signs (or possible bacterial infection in a young infant), common illnesses, malnutrition and anemia, and to look for other problems. In addition to treatment, the guidelines incorporate basic activities for illness prevention.

The IMCI case management process is presented on a series of charts that show the sequence of steps and provide information for performing them. This series of charts has also been transformed into an IMCI chart booklet designed to help you carry out the case management process. The IMCI chart booklet contains three charts for managing sick children age 2 months up to 5 years, and a separate chart for managing sick young infant's age 1 week up to 2 months. The case management process of IMCI is summarized as below in figure 2.1.

The case management process for sick children age 2 months up to 5 years is presented on three charts titled:

- Assess and classify the sick child
- Treat the child
- Counsel the mother

If the child is ***not yet* 2 months of age**, the child is considered a young infant. Management of the young infant age 1 week up to 2 months is somewhat different from older infants and children. It is described on a different chart titled:

- Assess, Classify and Treat The Sick Young Infant.

2.1.2 Clinical Decision Support System

A CDSS is a computerized system that uses different kind of techniques and method of reasoning to assist clinicians in assessing disease status, in making a diagnosis, in selecting appropriate therapy or in making other clinical decisions. An effective CDSS can reduce the variation of clinician's practice plans that plagues the process of healthcare delivery. The dynamic environment surrounding patient diagnosis complicates its diagnostic process due to numerous variables in play; for example, individual patient circumstances, the location, time and physician's prior experiences. An effective CDSS reduces variation by reducing the impacts of these variables on the quality of patient care [18]. CDSS have been recognized as promising tools for influencing healthcare provider performance to improve and streamline the quality of healthcare delivery [19]. Goals of implementing advanced CDSS are to decrease errors and improve patient safety, improve quality through adoption of best practices, increase cost-effectiveness and optimize the management of chronic diseases [20].

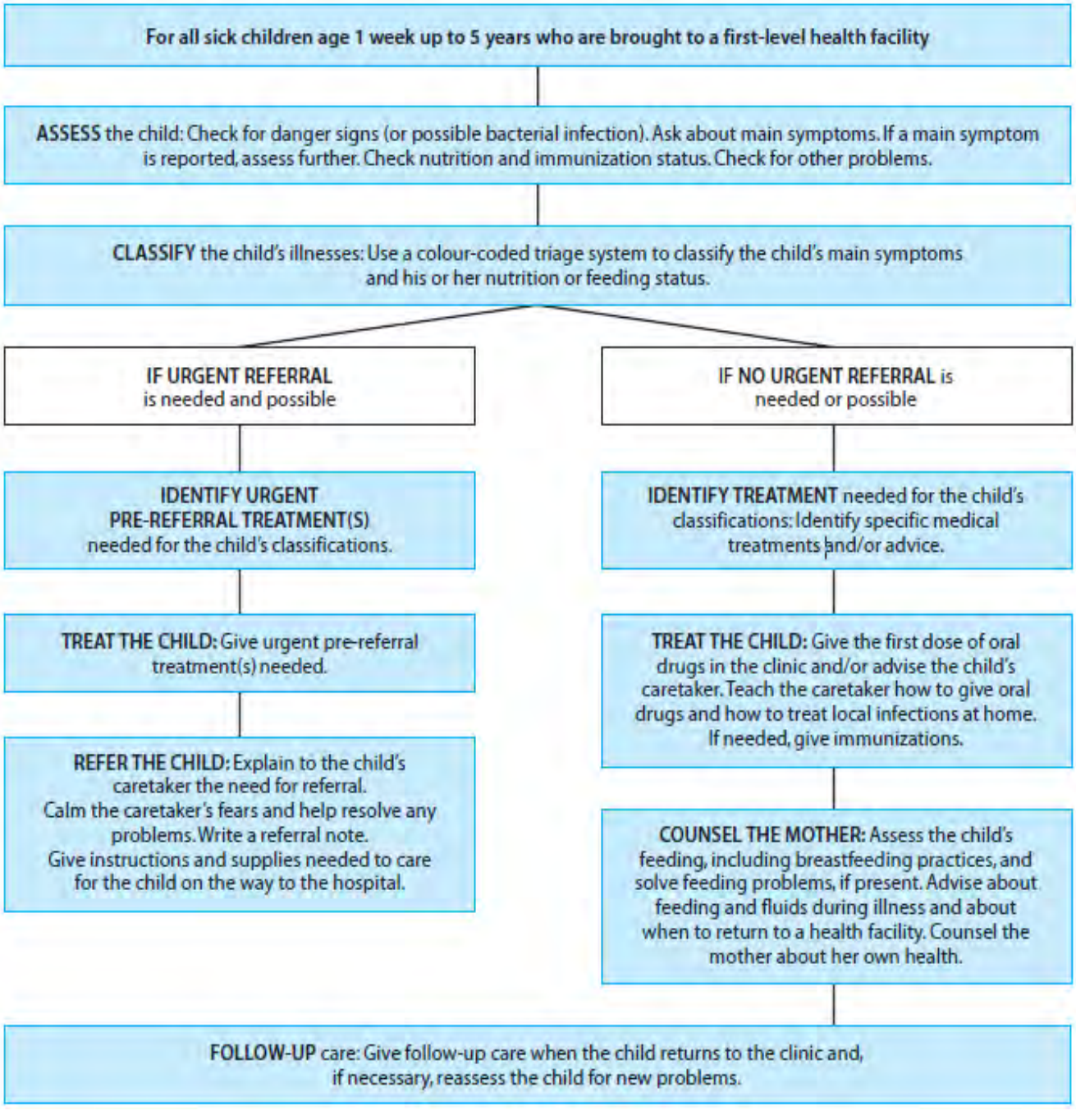


Figure 2.1: summary of the integrated case management process

Computerized physician order entry refers to a variety of computer-based systems that share the common features of automating the medication ordering process and that ensure standardized, legible, and complete orders. Clinical decision support systems are built into almost all computerized physician order entry (CPOE) systems to varying degrees. Basic clinical decision support provides computerized advice regarding drug doses, routes, and frequencies, and more sophisticated CDSSs can perform drug allergy checks, drug–laboratory value checks, and drug-drug interaction checks and can provide reminders about

corollary orders (e.g., prompting the user to order glucose checks after ordering insulin) or drug guidelines. Clinical decision support systems may also be implemented without CPOE. Basic CDSSs often assist in tasks such as drug selection, dosing and duration, and more refined CDSSs can incorporate patient- or pathogen-specific information. The ordering physician may view such advice and then proceed with a conventional handwritten medication order [21].

2.1.3 Structure of clinical decision support systems

CDSSs vary very much in their design. The basic principles of structure and design had also changed substantially in the last decade. Various traits of CDSSs are related to or have a direct effect on clinical effectiveness, functionality, error prevention, potential for acceptance in the clinical world, system portability, cost-effectiveness etc. It is thus important to characterize CDSSs in a way to best understand the diversity of CDSS.

2.1.3.1 Categories of CDSS

By combining several resources to create a comprehensive classification that capture key elements of CDSS design and function. The CDSS can be categorized along 5 axes [17].

A. **Context axes**

1. Clinical setting.

- Inpatient setting
- Outpatient setting

2. Clinical Task.

- **Diagnostic assistance**. Based on the patient's data and the system's knowledge base, the CDSS provides likely diagnoses. Diagnostic assistance can be coupled with complex data-retrieval systems, like ECG. It seeks to identify "what is true" for a specific patient.
- **Therapy critiquing and consulting**. This function can for example be incorporated into clinician's order-entry. It assesses the therapy, looks for inconsistencies, errors, cross-references for drug interactions and prevents prescribing of allergenic drugs. It has been shown that the necessity of a clinician to provide a statement of an appropriate reason, if he is or she is not to follow the recommendations, significantly increases the clinical importance of CDSSs. The CDSS can use protocols and evidence-based guidelines, combined with patient's facts, acquired from EMR, to provide an optimal treatment plan and help following it. These kinds of CDSSs address the question "what to do" with a patient and are often combined with recommendations about further diagnostic processing (i.e. which tests to order, X-ray, CT etc.). Such software can generate additional questions as to provide even more specific advice about further therapy (and diagnosis).

- **Drug dosing or prescribing**: CDSS have the power to reduce toxic drug levels, reduce medical errors, change prescribing in accordance to guideline recommendations and reduce time to achieving therapeutic control. If connected to EMR, the system can prevent prescription of drugs that cause allergic reactions. Such systems have been widely accepted, since they are well integrated into a routine part of the clinician's workflow and they provide automated order-entry forms and electronic transmission to pharmacies. Overall prescription of medication is one of the commonest tasks of a physician and also one of the commonest clinical tasks where CDSSs are being applied.
- **Test selection**.
- **Alerts and reminders**: An expert system that is integrated into a monitoring device or health-care information system (e.g. laboratory information system, EMR) can provide real-time sound, visual or tactile alerts thru various communication tools (e.g. e-mail, SMS, pager). Reminder systems are designed to remind the clinician of crucial tasks that need to be done before a certain event (e.g. fasting before endoscopy, no anticoagulants before abdominal surgery).
- **Information retrieval:** Relevant information retrieval through World Wide Web or comprehensive knowledge bases.
- **Image recognition and interpretation:** Clinical images from CT, MRI, angiograms etc. can nowadays be partially automatically interpreted. More importantly CDSS can function as a mass screening tool, where software flags critical images, which require clinician's special attention.
- **Prevention.**
- **Screening.**
- **Expert laboratory system.**
- **Chronic disease management.**

B. Knowledge axes

This axes deals with the sources, quality and customization of the CDSS's knowledge and data.

1. **Clinical knowledge source:** It can be derived from high-quality sources (e.g. randomized-controlled trials, systematic reviews, national or professional society guidelines) and/or from participation of clinicians that will eventually use the system.

2. **Data source**: The patient-specific data can be retrieved from computerized order entry, medical instrument (e.g. blood pressure measuring device), EMR or other data repository. The data may also be gathered from a paper chart or a person. In that case data must be entered in the system using a data input intermediary. This characteristic affects the likelihood of CDSS adoption in practice to a high extent. It has been shown, that automatic computerized provision of data to the system (e.g. from EMR) is preferred.

3**. Data source intermediary** is a clinician who inputs data into data source. Intermediaries could also be patients themselves.

4. **Data coding**: For various reasons (e.g. funding, epidemiology) it is desirable to use a widely use coding schema, i.e. ICD-10, SNOMED. Obviously the data could also be in plain text.

5. **Data customization**: The more CDSS produces patient-specific targeted recommendations, customized to age, gender, concomitant diagnoses etc., more it has a probability of clinical relevance and effectiveness.

6. **Update mechanism:** As stated above, knowledge base ought to be current and constantly up-to-date. CDSS can divide into knowledge based and non-knowledge based systems. The knowledge-based systems (aka expert systems) mostly consist of three parts – the knowledge base, inference engine and the mechanism to communicate. They contain expert clinical knowledge about very specific facts and tasks and are able to reason with the input of data from all the various sources stated above. These systems usually use knowledge in form of IF-THEN rules and probabilistic associations between compiled data. The inference engine lies at the core of the artificial intelligence part of knowledge-based systems – it combines and correlates knowledge base rules with the patient's data. Basically the inference engine reasons with the given information to form new conclusions.

The non-knowledge-based systems rather use principle of machine learning in the form of i.e. neural networks or genetic algorithms, where computers learn from past experience and/or find patterns in clinical data of an individual.

C. **Decision support axes**

Addressing a suitable decision-making process is probably the most important dimension of CDSS.

1. Reasoning method. Some of CDSS reasoning engines are:

- **Rule-based systems**: A rule-based system uses different expert knowledge bases in form of expressions that can be evaluated as IF-THEN rules (production rules). Such a system is an example of heuristic approach in which individual logical statements in the form of production rules are obtained by observing human experts, or interviewing and debriefing them, and then combined in an attempt to emulate the reasoning processes of experts.
- **Neural networks**: Artificial neural network is a non-knowledge-based adaptive CDSS that uses machine learning to learn from experiences and recognize patterns in clinical information.
- **Bayesian network**: A typical knowledge-based decision-making system is the Bayesian network (aka belief network or causal probabilistic network) that shows probabilistic relationships between sets of variables – diseases and symptoms, based on conditional probability according to Bayes theorem. It is a network with an explicit requirement, that the relationships be causal. Such a network helps to model the progression of a disease over time and the interaction between diseases; a big road-block however is that medical knowledge sometimes finds it difficult to directly specify what the effect is and what the cause.
- **Model based systems**: The latest achievement is patient specific modeling.

- **Logical condition:** Logical reasoning makes decisions according to the value of a given variable. The results of a decision-making process are different if the value is within or outside of the set boundaries.
- **Data mining and machine learning:** These methods are based on probabilistic decision-making according to the system's database. The ideal databases should be large and well-constructed, so that they allow precise retrieval of patients similar to a current patient. Analysis of responses of those patients to various treatments is used to decide upon the best treatment for the current patient.
- **Genetic algorithm:** As a non-knowledge-based method it uses iterative processes to rearrange itself and provide an optimal solution based on the patient data.

2. Clinical urgency.

It is the provision of decision support to decisions that have to be made urgently. CDSS should first provide its function to issues with clear clinical priority, according to the principle "treat first what kills first". This characteristic leads to better patient outcomes and physician's performance.

3. Recommendation explicitness.

Users are more likely to follow explicit recommendations, providing concrete course of action.

4. Response requirement.

The using clinician may be required to provide justification for the way he/she responded to the recommendation provided by the CDSS. This could be done in a form of acknowledgement of recommendation, with a statement of what alternative action was taken and with an explanation for non-compliance.

D. Information delivery axes

This axes deals with the delivery of newly produced information to the user.

1. **Delivery format**: it can be paper-based, online (via internet or integrated into EMR), via other technology – phone, pager, e-mail etc.

2. **Delivery mode**: The recommendations can be delivered after the decision-maker has requested so or on the other hand it could be delivered without the consent in form of an alert, reminder or optimization request. In the first case the clinician has to make an additional effort, recognize when the advice would be useful, "go to the program" and enter data to request diagnostic or therapeutic assessment, thus the program is passive. The so-called "push systems" that automatically provide recommendations may be more effective and substantially more used. They play an active role with providing decision support as a byproduct of data-managing activities (e.g. monitoring, EMR supervision). System decision logic is in a way integrated within a patient's database that is already being gathered from various sources and provides results of its decision analysis without an additional effort of the clinician. A valid point to

consider here is how to avoid so-called “alarm fatigue“, where the clinician is over-warned to minor discrepancies which are otherwise noted and commonly understood.

3. **Action integration:** It is imperative, that the CDSS provides the ability to the decision-maker to exert the recommended actions with ease. For example, the software can, while providing prompts on therapy-critique also provide direct links to order-entry forms and therapy planning section of EMR. The action of therapy change should be completed within a range of few clicks, for example with checking a mark. Action integration unambiguously adds to wider acceptance and usability of CDSS.

4. **Explanation availability:** It is a function, where the system provides an explanation of its recommendations, through e.g. links to evidence-based articles, books or directly from the knowledge base.

E. Workflow axes

CDSS can be seen as a process, however at the point of care it is virtually an intervention of technology, which could also act as an interruption. Systems that are synergistic with the institution's workflow are likely to experience higher usage and prove to be more effective in optimizing practitioner's performance.

2.2 RELATED WORKS

A. e-IMNCI

In Tanzania, an electronic version of IMCI was developed to run on a personal digital assistant (PDA). The software was built using C# on windows mobile 5.0. Currently, the data is output to CSV log files. The e-IMCI protocol followed the same protocol in the paper based IMCI guideline. The eIMCI tool provided classification, treatment and communication recommendations in accordance to the IMCI protocol. For symptoms that required additional follow up to assess severity, the algorithm prompted the necessary questions as per the IMCI protocol. For steps that didn't require data gathering, for example the communication of an instruction to a mother, the healthcare worker selected the next button in the PDA to indicate that the step was completed. All eIMCI instructions were available in English and Swahili [6, 8].

B. ICATT

Integrated Management of Childhood Illness Computerized Adaption Training Tool (ICATT) is a software application that supports the implementation of WHO/UNICEF strategy of IMCI. The tool provides the possibility to adapt the generic IMCI guidelines at national and sub national levels, to develop training courses that fit into various training approaches, and to be translated into different languages and used in a range of environments and settings. This software used as a teaching tool, self-learning, reference, test for students, and to update the guideline. WHO estimates that in total, 12 countries completed adaption through ICATT (Cambodia, Peru, Fiji, Indonesia, Kiribati, Malaysia, Mozambique, Philippines, Solomon Islands, South Africa, Tanzania, and Vietnam). The program can either be installed in computer or from a CD ROM or memory stick. The most recent version of ICATT released in August 2011 is equipped to run on intranets and internet sites as well as the World Wide Web, creating global access to the training [9].

C. Isabel

Isabel is a web-based diagnosis decision support system created in 2001 by physicians to offer diagnosis decision support at the point of care. Isabel has been extensively validated and been shown to enhance clinician's cognitive skills and thereby improve patient safety and the quality of patient care. Isabel now covers all ages (neonates to geriatrics) and all major specialties and sub-specialties in Internal Medicine, Surgery, Gynecology & Obstetrics, Pediatrics, Geriatrics, Oncology, Toxicology and Bioterrorism. Isabel is fast and easy to use & gives the clinician an instant list of likely diagnoses for a given set of clinical features (symptoms, signs, results of tests and investigations etc). Following on from history taking and clinical examination Isabel assists the provider ["learned intermediary"] by reconciling [concept matching] patient data sets with data sets as described in established medical literature (textbooks and journals). Isabel also has the ability to suggest causative drugs for clinical features entered and allows clinicians to follow their hunches by related diagnoses and restricting searches to specific body systems. Isabel has been interfaced with electronic medical record systems [EMR]. Isabel is able to extract pre-assigned data from an EMR - on a single click delivers diagnoses and knowledge to the EMR user [no data entry into Isabel required]. Isabel not only assists in making the right diagnosis but

helps answer clinical questions with up to date knowledge from textbooks and journals

Isabel uses Autonomy's natural language processing software as opposed to standard key word searches. Isabel consists of a proprietary database of medical content and a tutored taxonomy of over 11,000 diagnoses and 4,000 drugs and heuristics. Each diagnosis / drug entity has a kernel of knowledge and heuristics (age, region, gender, pregnancy). So, when a clinician enters clinical features into Isabel he/she is given a list of likely diagnoses or causative drugs for consideration. The pattern of the clinical features entered is concept-matched with kernels of knowledge and the best matched kernels of knowledge (diagnoses) are returned for consideration. Autonomy technology is based on advanced pattern-matching techniques (non-linear adaptive digital signal processing) rooted in the theories of Bayesian Inference and Claude Shannon's Principles of Information. These enable identification of the patterns that naturally occur in text, based on the usage and frequency of words or terms that correspond to specific concepts. Based on the preponderance of one pattern over another in a piece of unstructured information, Autonomy technology enables computers to understand that there is a particular probability that a document in question is about a specific subject. In this way, Autonomy technology is able to extract a document's digital essence, encode the unique "signature" of the key concepts, then enable a host of operations to be performed on that text, automatically.

Isabel starts at an earlier point in the clinical journey with clinical features. Isabel helps the clinician reach a diagnosis first, and using the same proprietary technology mobilizes medical up to date knowledge from textbooks and journals [10].

D. QMR

Developed out of INTERNIST-I, Quick Medical Reference (QMR) is an in-depth information resource that helps physicians to diagnose adult diseases. It provides electronic access to more than 750 diseases representing the vast majority of the disorders seen by internists in daily practice as well a compendium of less common diseases. QMR uses more than 5,000 clinical findings to describe the features of diseases in the QMR knowledge base. Findings include medical history, symptoms, physical signs, and laboratory test results. Laboratory test results are subdivided into three categories based on increasing levels of cost and invasiveness. QMR findings represent abnormal conditions, e.g., "Abdomen Pain Severe" or "Blood Hepatitis B Virus by Polymerase Chain Reaction." Every disease profile included in the QMR knowledge base is the result of an extensive review of the primary medical literature. Consultation with experts is used to resolve any inconsistencies or deficiencies found in published reports. QMR is used in in-hospital and office practice. Windows versions of QMR were available for single and multiple users; a hand-held version was under development and may have been released [11].

E. HELP

HELP is a complete knowledge based hospital information system. It supports not only the routine applications of an HIS including ADT, Order Entry/Charge Capture, Pharmacy, Radiology, Nursing documentation, ICU Monitoring, but also supports a robust decision support function. The decision support system has been actively incorporated into the functions of the routine HIS applications. Decision support has been used to provide alerts/reminders, data interpretation, patient diagnosis, patient

management suggestions and clinical protocols. Activation of the decision support is provided interactively within the applications and asynchronously through data and time drive mechanisms. The data driven activations is instantiated as clinical data is stored in the patient's computerized medical record. Time driven activation of medical logic is triggered at defined time periods. The HELP system supports an integrated database structure which facilitates the decision support functions of HELP. The database structure also lends itself to design of application independent patient reports [14].

F. TheraSim CS

TheraSim CS (Clinical Simulator) is clinical simulation technology to support the training of physicians, nurses, medical students and pharmacists in the diagnosis and treatment of chronic and infectious diseases. The first to be implementation of the technology for clinical care is TheraSim CS-HIV for the diagnosis and management of HIV. Implementations of TheraSim technology covering other clinical domains are planned. These include Diabetes mellitus (2004), malaria, TB, hepatitis, cardiovascular and neurological disease. TheraSim knowledge bases for individual disease modules comprise clinical data, patient simulation data and best practice guidelines [12].

G. CHICA

CHICA is a computer-based decision support system to improve the quality of pediatric primary care. The system uses standard care guidelines, encoded in logic rules (Medial Logic Modules, MLM) to analyze data stored in an electronic medical record. The system generates questions for patients and advice for physicians. This information is provided on tailored forms (Adaptive Turnaround Documents, ADT) on which patients can record answers and physicians can document care. The ADTs are optically scanned so the data collected from patients and physicians are automatically entered into the electronic medical record. CHICA is an extension of the Regenstrief Medical Records System (RMRS), a computer-based inpatient and outpatient information system that contains more than 30 years of patient data and more than 300 million patient observations - primarily on adults. The RMRS includes an internationally respected physician reminder system that offers suggestions on appropriate diagnosis, tests and treatment management for each patient. CHICA currently uses a subset of Arden Syntax for guideline modeling. At present the system has 290 "rules". More rules will be added based on pediatric preventive care needs and as the disease management paradigm continues to be addressed [15].

CHAPTER THREE

METHODOLOGY

3.1 Target Population

Effectively gathering user requirements is a critical first step of any project and perhaps one of the most challenging project management skills. It is vitally important to build the project on well-formed and verifiable user requirements to avoid cost overruns, unsatisfied users, or even project cancellation.

Three health centers were purposively chosen for the data collection and testing of the system in due consideration of their better ICT infrastructure, organizational functionality and the availability of skilled human resource. In addition to the health centers, MOH's under 5 years children case team were also another primary source of data. Data collection was conducted to collect all the available resources and requirements which were needed for the development of the system. These requirements and resources include working guideline, OPD registration book reporting forms and work flow of the existing system.

3.2 System Development Methodology

3.2.1 Spiral Software Development Life Cycle

Collecting all the requirements from the user at the beginning of project development is very difficult. Users typically have difficulty explaining what they need, and the problems increase when developers fail to translate requirements into working programs. The users may change the requirements depending on their needs through time. So to manage those upcoming requirements is very necessary. Hence, to handle such kind of uncertainties spiral SDLC methodology was used for the development of the project. A software development life cycle promotes sustainable development by using different methods which attempts to minimize risk by developing software in short time boxes called iterations. The spiral model has four phases. A software project repeatedly passes through these phases in iterations called Spirals.

A. **Requirement Identification and Analysis**

This phase starts with gathering the business requirements in the baseline spiral. In the subsequent spirals as the product matures, identification of system requirements, subsystem requirements and unit requirements are all done in this phase. This also includes understanding the system requirements by continuous communication between the customer and the system analyst. At the end of the spiral the product is deployed in the identified market. In this phase, requirements were identified and collected for analysis of the system. To analyze the collected requirements Object Oriented (OO) methodology was used to develop the e-IMNC and UML notations was applied for the analysis and design model of the system. It was an appropriate descriptive notation to develop the system hence there are different objects interacting within the system like patient, health worker, diagnosis, testes and

medications.

The object model has influenced even earlier phases of the software development life cycle. Object-oriented analysis (or OOA, as it is sometimes called) emphasizes the building of real-world models, using an object-oriented view of the world. Object-oriented analysis is a method of analysis that examines requirements from the perspective of the classes and objects found in the vocabulary of the problem domain [22].

Before collecting any kind of requirements first I must understand 'how the system works?' and 'what the problems are?' in the existing system to minimize the risk of handling some uncertainties. To obtain all the required information, I was engaged to collect and see the work flow, working forms (registration, report), and handbooks and guidelines of health institutions. I applied different kind of requirement elicitation techniques and tools to get those required data. The techniques which were used for requirement collection are listed below.

1. Interview

Interview was first made with nurses and health professionals in health centers and ministry of health who work on under 5 years children's health. The interview was selected because it helps to get the required information genuinely, and some information better described in interview rather than other data collection techniques. I asked the stakeholders about the system that they use (if any), stakeholder needs, organizational standards, domain information and the system to be developed. All the interviews were conducted face to face in all offices of the health centers which deliver a service for under 5 year's children. At the time of interview health professionals have had difficulties to express the real requirements, so a paper prototyping and mock-up of the system's UI prototyping was applied to provide insight to the new functionalities of the new system.The interview used interview checklist (described in appendix A), prototyping and discussion on the prototypes was applied. A paper prototyping and mock-up user interface of the system was done at the end of each phase of the project deliverables.

2. Observation

In addition to the interview, observation was applied to make sure the respondents did what they responded during the interview and the reported tools are available in the health center in order to gather all requirements needed for further analysis and design. Observation checklist was used to gather the information about work flow, business process, recording formats, and organizational needs and standards of the health center.

3. Document Analysis

Document review and analysis was applied to collect the forms and documents they used in the office. Document review was conducted from each office of the health center which gives a service for under 5 years children (record office, diagnosis room, vaccination room) included in the study with regard to routine data recording, compilation, processing and reporting. The data collected by this technique were used as a raw material for analysis and design of the system.

Analysis of the interview, document review and observation was started immediately after having all the required requirements. The results of these techniques were presented being summarized from the notes taken during the collection. The responses were first categorized into different themes according to the type of existing system in use, tools for input and reporting. The collected data was analyzed in organized and proper way to identify actors, scenarios, use cases, relationships among use cases, participating objects, and nonfunctional requirements.

The collected requirements were dissected and grouped into different interrelated classes and objects. By identifying the interaction between objects with the system, user and any other external system with which the system interacts, and by using current system business scenario both functional and nonfunctional requirements of the system were identified. Use cases and business scenarios produced during requirements identification and collection time were analyzed to Identifying entity objects, modeling interactions between objects, associations among objects, object attributes, modeling the nontrivial behavior of individual objects, modeling generalization relationships between objects, and reviewing the analysis model.

The requirement analysis results are described by a set of nonfunctional requirements and constraints, use case modeling (describing the functionality of the system from the actors' point of view), object model (describing the entities manipulated by the system) and sequence diagram for each use case, showing the sequence of interactions among objects participating in the use case (*see* Chapter 4, *System Analysis*).

B. **Design**

Design phase starts with the conceptual design in the baseline spiral and involves architectural design, logical design of modules, physical product design and final design in the subsequent spirals. Object oriented design also used to design the system features. The emphasis in programming methods is primarily on the proper and effective use of particular language mechanisms. By contrast, design methods emphasize the proper and effective structuring of a complex system. Object-oriented design is a method of design encompassing the process of object-oriented decomposition and a notation for depicting logical and physical as well as static and dynamic models of the system under design [22]. There are two important parts to this definition: object-oriented design (1) leads to an object oriented decomposition and (2) uses different notations to express different models of the logical (class and object structure) and physical (module and process architecture) design of a system, in addition to the static and dynamic aspects of the system.

The analysis model, however, does not contain information about the internal structure of the system, its hardware configuration, or, more generally, how the system should be realized. Based on the deliverable of the previous phase, the system design was started immediately. System design describes software architecture of the system *(*subsystem decomposition in terms of subsystem responsibilities, dependencies among subsystems, subsystem mapping to hardware, and major policy decisions such as control flow, access control, and data storage). The design goals are derived from the nonfunctional requirements (*see* Chapter 4, *System Design*).

C. **Construct or Build**

Construct phase refers to production of the actual software product at every spiral. In the baseline spiral when the product is just thought of and the design is being developed a POC (Proof of Concept) is developed in this phase to get customer feedback. Then in the subsequent spirals with higher clarity on requirements and design details a working model of the software called build is produced with a version number. These builds are sent to customer for feedback. Construction/development of the system was developed based on the result of the previous phase's deliverable. Ms Visual Studio 2010 and MsSQL 2008 were used to construct and build the system.

D. **Evaluation and Risk Analysis**

Risk Analysis includes identifying, estimating, and monitoring technical feasibility and management risks, such as schedule slippage and cost overrun. After testing the build, at the end of first iteration, the customer evaluates the software and provides feedback. Paper based high fidelity prototyping and the system itself were used for evaluation and risk analysis of the system. End users were involved to get feedback about the evaluation of the system's performance and usability.

3.3.2 Knowledge Representation Method

Knowledge representation used in the clinical decision support is done using the if-then rules. The systems which use a set of if-then statements to declare the rules are also called rule based systems or specifically rule based expert systems. The Fig. 3.1 describes the components of the expert system.

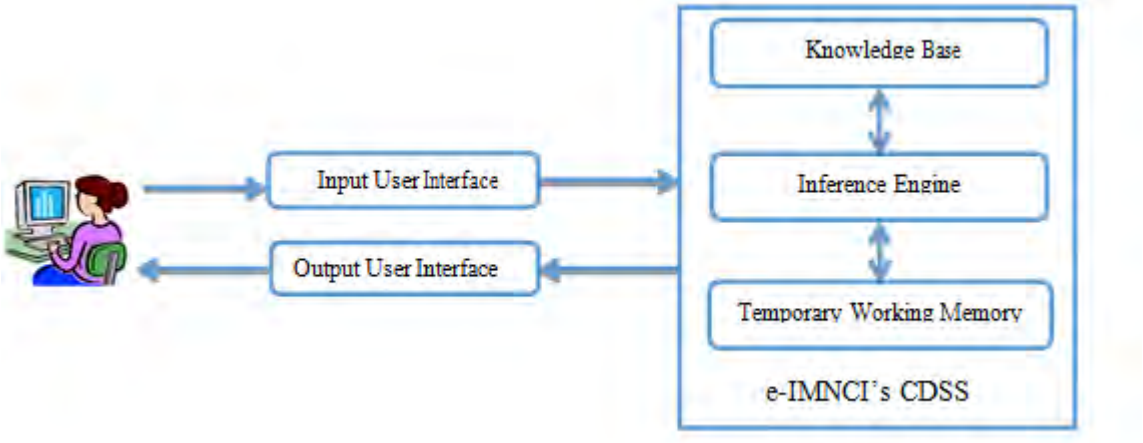


Figure 3.1: architecture of CDS module of the e-IMNCI system

The knowledge base is a collection of rules, so we call it rule base. The construction of the knowledge base is very important process in developing an expert system and hence generally requires a lot of care and many aspects are to be considered. The data for the knowledge base was collected from IMNCI booklet chart; this data was processed and converted into the patterns to define rules. A rule is nothing but an expression having a set of conditions on the left hand side, which when evaluated to TRUE can derived to an action on the right hand side. These set of rules completes the knowledge base creation.

The other component of the rule based expert system is the inference engine. It infers the information or takes the action based on the input and the rule base. The inference engine works in two phases – looking for a rule–match and performing action. This component is tightly coupled with other two components User Interface and Temporary Working Memory. Forward chaining inference method was used to infer the rules of the knowledge base.

The input to act upon is taken by the User Interface, which may be any application's client. The User Interface acts as a communication medium between the outside world and the expert system. The rule collection from knowledge base, rule matching as per the inputs and deriving the action as a conclusion is performed in the Temporary Working Memory. Thus the result is passed on to the user using the User Interface. The detail of knowledge representation and inference engine is described in chapter four.

The passive mode for giving advice was used for the clinical decision support module of the system; this mode allows the physician uses the system when advice needed. A passive system allows the user to require advice, analyses the advice and accepts/Rejects the advice. Some of the characteristics of passive system are stand alone, system initiative data entry and consulting mode of consultation style.

3.3.3 System Design and Development Tools

UML modeling technique, MS Visio & MS SQL tools were used to develop the software architecture of the system. Microsoft visual studio 2010 with C#.NET and Microsoft SQL server 2008 was used. C#.NET was employed for the development the system's component like classes, user interface and whereas MS V was employed for the design and development of the system's database. C#.NET was selected because of its features, some of the features are it's an excellent language for .Net Framework development, extremely versatile, robust, well-designed language and easy to design a report from different datasets.. The SQL also selected due to its important features such as: it is language independent supporting .Net Framework which allows using your favorite .Net language, ease of deploying, security, it has replication & massaging capabilities and it has easy setup options.

3.4 System Testing and Evaluation

A usability testing software "Morae TechSmith" was used to conduct the user satisfaction and usability of the system. The usability evaluation was conducted with the following objectives: to determine whether users can use the software equally well, to assess how easily and successfully users make use of the software, to determine whether the common tasks are easy to perform, to identify the problems users will face when they using the system, to identify the kind of help features that might be required and to assess user satisfaction of the system.

The usability tests were conducted from April 8 to April 21, 2014. This test was conducted in Janmeda, Yeka woreda-4 and shiromeda health centers. Before the final testing schedule, high fidelity prototyping test was done at end of each deliverable of the project by using paper based user interface and database tables & diagram. I used windows 7 operating system with Microsoft 3.5.NET framework to conduct the test. In addition to this I used Morae TechSmith usability testing software to record video and audio files when the testing session is going on and also it's capable of analyzing the recorded data. To conduct the system's usability testing, user satisfaction survey was used (appendix A).

Both user-based and expert-based evaluation methods were employed in the course of conducting the usability study. A brief account of both methods is presented below.

3.4.1 Expert-Based Evaluation

The cognitive walkthrough method was used during expert-based evaluation. First, exact sequences of correct task performance were outlined for some of the functions of the system. Estimates were then made on the likely success or failure of the user in performing the sequence of tasks. The expert-based evaluation concentrated more on the difficulties users may experience in learning to perform a given task. It was also used to prepare the tasks/scenarios for user-based evaluation as well as to estimate the average time it should take to complete each task.

3.4.2 User-Based Evaluation

With this method, test participants were asked to perform a set of tasks of the software. Complete interaction was video recorded for subsequent analysis of transactions, navigation and problem handling. Observations were also made while they were performing the given tasks by sitting next to them. Test participants were also asked to 'think aloud' during the usability test. They were encouraged to describe their reactions as well as their thoughts while completing the requested tasks. This information helped to determine what participants were thinking when they clicked on a particular menu or what information

might be confusing to them. More informal tips and unstructured tests were also used which involved participants and the evaluator jointly interacting with the system in order to check on what works and what is problematic for the participants. Finally, after completing the given tasks the participants were asked to provide data on their likes and dislikes of the system at the end of testing session.

Once the testing session is completed, the recorded data was analyzed using Morea TechSmith usability testing software. The result of the analysis includes "how long the users took to accomplish the given tasks?", "what was the difference between the actual time and expected average time?" and the number of wrong clicks done by users.

3.4 Method of Dissemination of Results

Being an academic project, the output of this project will be delivered to AAU as partial fulfillment of MSc in health informatics. Furthermore the result of this project will be disseminated to the concerned bodies like MoH, Addis Ababa Health bureau and NGO's working on child health.

3.5 Operational Definition

e-IMNCI: an electronic version of Integrated management of newborn and childhood illness, a computer-based clinical decision support system based on paper based IMNCI booklet chart and guideline.

CDSS: Clinical Decision Support System is computer program that's designed to help health care professionals to make clinical decisions. It provides clinician staffs, patients and other individuals with knowledge and person specific information, intelligently filtered and presented at appropriate times, to enhance health and health care services.

Knowledge Base: it stores extensive knowledge gathered from experts, historical data books regarding the application in the form of rules.

Inference Engine: is a mechanism to infer the information or takes the action based on the input and the rule base.

SDLC: a software development life cycle model represents all the activities and work products necessary to develop a software system.

Spiral Methodology: it is a type of software development methodology which focuses on early identification and reduction of project risks. A spiral project starts on a small scale, explores risks, makes a plan to handle the risks, and then decides whether to take the next step of the project - to do the next iteration of the spiral.

3.6 Ethical Clearance

The project was carried out after getting permission from the ethical clearance committee of Addis Ababa university school of public health. Official letter of cooperation was written to all respective health centers and permission to conduct the project was secured accordingly.

CHAPTER FOUR

DISCUSSION OF RESULTS

Introduction

In the previous chapter, the methodology used for development of the system is described and discussed. As I mentioned before C#.NET and MsSQL was used for development of the system. In this section, the system analysis, design, development and pilot testing result of e-IMNCI system will be discussed in detail.

4.1 System Analysis

4.1.1 Current System

Currently, all health centers in Ethiopia are using a manual system for their service delivery of each departments of the health center. The IMNCI outpatient department of the health centers uses a paper based IMNCI guideline as a reference and OPD register book as data recording tool that is used as an input for clinical decision making for the services deliveries of infant and child health. Health professionals in the health centers use the IMNCI guideline for child's problem assessment, classification, treatment, and communication of instructions to the caregiver. The OPD registration book is used to register patient information, observed sign and symptoms, and diagnosis result. This registration book also used as an input for clinical report by counting the recorded data on it.

The overall business process description of the IMNCI is presented as follows:

- The patient presents in the card room/ registration office.
- The data clerk sends the patient to IMNCI department after confirming the child is under 5 years old.
- The nurse/ health officer registers the patient demographic information on OPD register. The OPD register book is used to register the patient detail demographic information and diagnosis result .
- By using the IMNCI guideline the nurse or health officer ask about the presenting complaints.
- After assessing patient's sign and symptoms, the nurse or health officer write or mark the observed sign and symptoms on OPD register.
- If the child was vaccinated in the health center before, the nurse or health officer checks the immunization status on patient's chart. However, if the patient is new for the health center or previous patient's chart is not available, the nurse or health officer asks the mother about the vaccination that the child has taken before.
- The nurse or health officer registers diagnosis and treatment plan and follow up information results on OPD register. In addition to this they also register the diagnosis result on the patient's chart.
- When the patient comes to the health center for another visit, they check the previous history on patient's chart. However, sometimes they checks the previous patients detail history on the OPD register book.

4.1.2 Proposed System

The proposed system is only applicable for IMNCI department of the health center to facilitate the services of the department. This system will be used as record management system (only for patients under 5 year's children), clinical decision support system, report generating system, and as a guideline. The e-IMNCI protocol follows the same protocol in the current paper based system, guiding healthcare workers step by step for child's problem assessment, classification, treatment, and communication of instructions to the caregiver. Some of the modules of the system are:

Registration Module:

- The system registers a patient's detail information. It gives a unique medical record number for every patient. After the patient is registered in the system, the system will provide a page to register the diagnosis and treatment result. It registers one patient's many diagnoses and treatment result in to one account. Once the patient information is saved to the database, for next visit the patient's vital sign and diagnosis result is updated. The registration module also used to register follow up appointment date and follow-up outcome of the patient.

Record Tracking Module:

- The record tracking module of the system is used to manage the recorded information by storing, updating, searching, and deleting the information efficiently. By using different searching parameters the system generates different kind of reports from the recorded data. The system's record tracking module also used to see the patient's previous diagnosis history, follow up date and outcome, immunization status, and previous referral history.

Decision Support Module:

- The decision support module of the system is used to counsel or advice the health professionals about diagnosis and treatment plan for the observed sign and symptoms. The system provides a suggestion for possible diagnoses that match the observed patient's signs and symptoms, treatment guidelines for specific diagnoses, and provide information about drugs side effect, contraindication, drug interaction and dosage recommendations.

Guideline Module:

- The system provides guideline and reference for common diseases in child health. The IMNCI guideline is developed under the paper based WHO IMNCI guideline and protocols in a dynamic way. Unlike the paper based guideline, no need to see the whole guideline to find out a guide for a single problem rather the system displays a guide for a specific disease depending on the user's selection.

4.1.3 Functional and Data Requirements

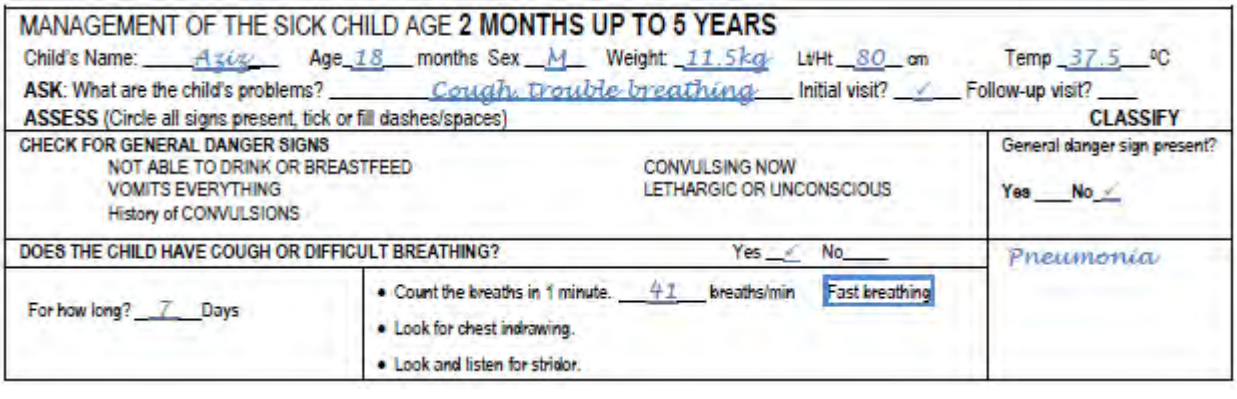
4.1.3.1 Functional and Data Requirements

Functional requirements describe the interactions between the system and its environment independent of its implementation. The environment includes the user and any other external system with which the system interacts. The e-IMNCI includes the following basic functionalities:

- The system registers patient information.
- The system registers patient's observed sign and symptoms, and diagnosis result
- Patient's data is validated before the data is saved to database.
- The system provides dynamic IMNCI guideline
- The system assists the clinicians when they need a help about diagnosis and treatment plan.
- The system used to search patient information.
- It generates different report based on the selected report parameter.
- The system also is designed to create, update and remove user account.

4.1.3.2 Data Requirements

Data requirements are prescribed directives or consequential agreements that define the content and/or structure that constitute high quality data instances and values. During requirement elicitation phase stakeholders were asked about; what data should be input into the system? What data should be output by the system? How will they use the system? By using interview, observation, and document review technique, system inputs and outputs were identified. In current system, patient's demographic information, sign and symptoms, and diagnosis and treatment results are considered as an input of the system.



MANAGEMENT OF THE SICK CHILD AGE **2 MONTHS UP TO 5 YEARS**

Child's Name: Aziz Age 18 months Sex M Weight: 11.5kg Lt/Ht 80 cm Temp 37.5 °C

ASK: What are the child's problems? Cough, trouble breathing Initial visit? ✓ Follow-up visit? ___

ASSESS (Circle all signs present, tick or fill dashes/spaces) **CLASSIFY**

CHECK FOR GENERAL DANGER SIGNS
NOT ABLE TO DRINK OR BREASTFEED
VOMITS EVERYTHING
History of CONVULSIONS
CONVULSING NOW
LETHARGIC OR UNCONSCIOUS

General danger sign present? Yes ___ No ✓

DOES THE CHILD HAVE COUGH OR DIFFICULT BREATHING? Yes ✓ No ___

Pneumonia

For how long? 7 Days

- Count the breaths in 1 minute. 41 breaths/min Fast breathing
- Look for chest indrawing.
- Look and listen for stridor.

Figure 4.1: Recording format in the current system.

The inputs are processed to make the outputs of the system like; number of patients who caught by a specific disease, number of referred patients, patient's previous diagnosis history, and so on. Data storage, security, movement and quality will be assured by using different mapping algorithm and normalization. The identified data elements will be discussed in conceptual and logical model of the database.

Name	Data Type	Description
MRN	Integer	To uniquely identify each patient
Name, Sex	String	To describe demographic information of a patient.
Date of birth	Date	
Age, Kebele, House Number, Phone Number, Woreda	Integer	
Weight, Temperature	Float	To describe the vital signs of a patient.
Height	Integer	
Signs, Symptoms, classified_as, Treatment, Visit, Follow-up, Referred_to, immunization_status	String	Describes the diagnosis and treatment results.
Follow-up date,	Date	

Table 4.1: Data elements in the current system.

4.1.4 Nonfunctional Requirements

The non-functional requirement concerned with the system that must be carried out in addition to functional requirements stated. These requirements include security, performance consideration, error handling, reliability, availability and efficiency.

User interface: The system will have an interactive, dynamic and consistent graphical user interface. Most of the system's interfaces are checkboxes and radio buttons to minimize the access time of the users and to make it user friendly.

Performance consideration: The system shall have acceptable performance with regard to response time. At the same time, the system must provide accurate, consistent and reliable data to the system users. User interface screens shall respond with speed so as to ensure fast data entry to the system**.** The system shall have flexible and convenient provision for data import/export with other application.

Reliability: The system shall show appropriate messages to user's terminal when system is down. The system shall generate error messages when the user attempts to enter invalid data.

Availability: The system is not restricted by dates and time, it will be available whenever the user want to access the needed data.

Error Handling: It should be able to respond descriptive error messages. Appropriate error message will be displayed when the users inserts unexpected/wrong data. With this features, users will be informed about the errors they have committed and what to do to rectify the problem in easy way.

Security: the system is used to store and retrieve patient's information, so the recorded data must be protected from unauthorized users and access. The system must check the user privilege whether it is limited or full access to use the system by using login form of the system. Based on the privilege of the users, the systems functionalities of the system will be changed according to their level of access.

4.1.5 System Analysis Models

To produce a model of the system that captures all functionalities and that eliminates ambiguous requirements, we need to construct the analysis model based on the requirements of the system. System analysis contains three models: functional, object and dynamic models. The functional model is described by use case diagrams, object model is described by class diagram, and the dynamic model is described by sequence diagrams. In this project the analysis model is described with use case, sequence, class, and class diagram.

4.1.5.1 Use case Model

A use case can be described as a specific way of using the system from a user's (actor's) perspective. Use cases provide a means to capture system requirements, communicate with the end users and domain experts, and test the system. Use cases are best discovered by examining the actors and defining what the actor will be able to do with the system.

Use Cases: The following use cases have been identified from the system specification.

1. Login
2. Register patient
3. Register diagnosis result
4. Search patient
5. Update patient information
6. Generate report
7. Check guideline
8. Diagnosis assistant
9. Manage accounts
10. Log out

List of Actors: The identified actors that will be participating in the system are:

1. System Administrator:
   A user who is administers the system like creating, updating and removing user account.

2. Health Center manager:
   A user who is responsible for generating and accessing reports to make a decision.

3. Nurse or Health Officer:
   A user who is responsible for registering patient information and diagnosis result in the IMNCI department.

Use Case Diagram

UML use case diagram and description provides descriptions of the functionality of the system from the user's point of view. The use cases for electronic version of integrated management of newborn and childhood illness are used to represent the basic functionalities of the system as use cases focus on the behavior of the system from an external point of view. Figure 4.1 depicts the use case diagram of the eIMNCI system.

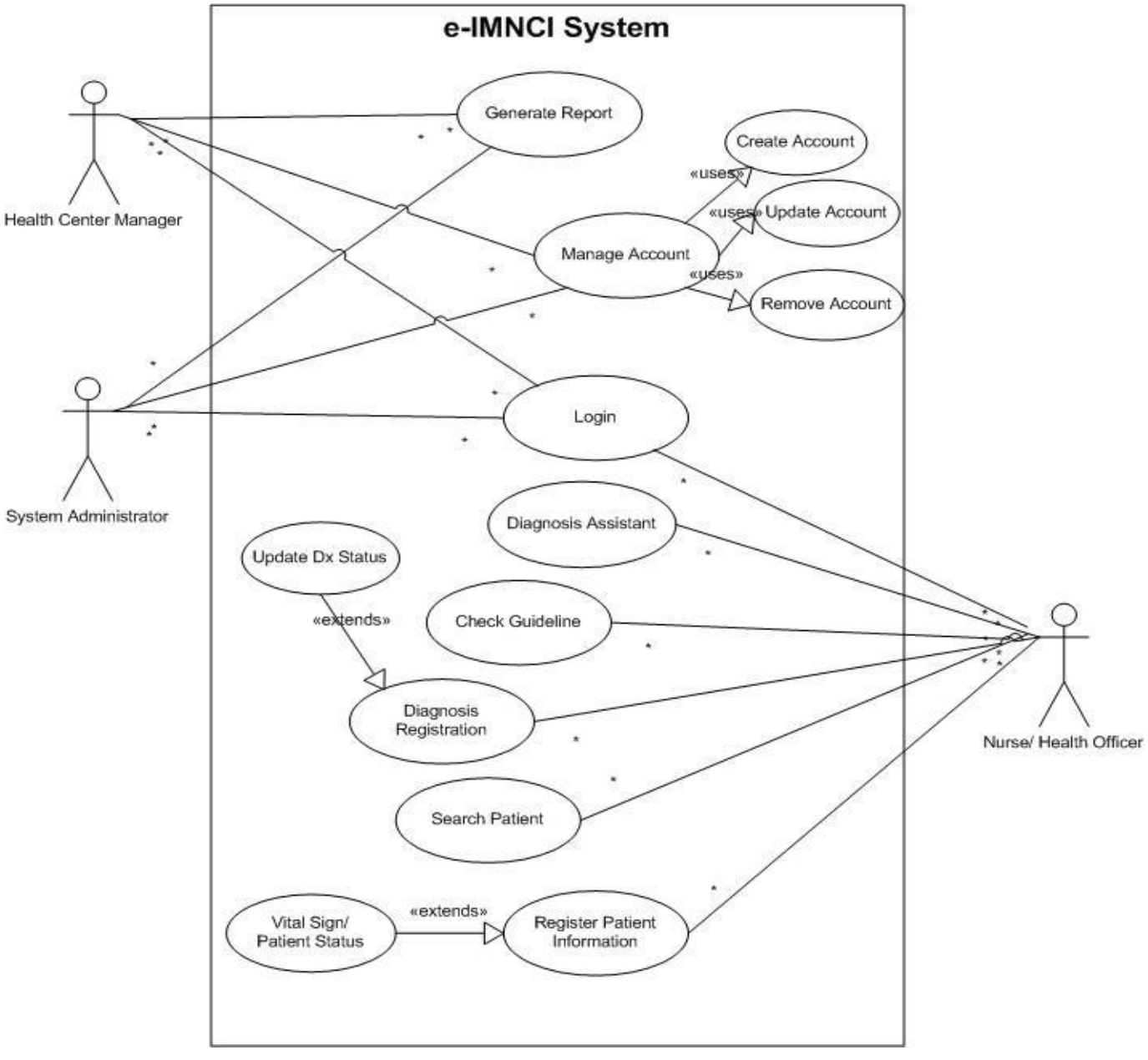


Figure 4.1: Use case diagram for e-IMNCI system.

A. Login

Use case ID	UC-1
Use case Name	Login
Priority	High
Actor	Manger, System Administrator, Nurse and Health officer
Description	Describes how the users will logged into the system
Precondition	User has user account or creates new account.
Post condition	The user logged in to the system.
Main success scenario	1. Open the e-IMNCI system 2. The user selects username and password. 3. Click on sign in button. 4. The system verifies the username and password.[4A, 4B] 5. The system enables the whole features of the system based on the users privilege.
Alternative path	4A. if the username and password mismatched then I. The system notifies the user entered wrong username or password II. The system let the user to try again III. If the user forgets his/her password, then [UC-10]. 4B. If the user hasn't a user account 1. [UC-10]

Table 4.1: Description of login use case.

B. Patient Registration

Use case ID	UC-2
Use case Name	Register Patient
Priority	High
Actor	Nurse or Health Officer
Description	To register patient information when a patient comes to the health center.
Precondition	The user should be authenticated and authorized.
Post condition	Register patient's information.
Main success scenario	1. The user checks whether the patient is registered in the system or not by using patient's name. [2A] 2. When the user confirms that the patient is not registered before then clicks on register button. 3. The system displays patient registration form. 4. The user enters all the required information to the corresponding fields. 5. The system validates the entered patient information.[5A] 6. The system displays the entered data in a table form. 7. The user checks the entered information which is displayed in the table is correct or not. [7A] 8. If the entered data is correct the user confirm to save the patients data. 9. The system saves the patient detail information to the database.
Alternative path	2A. if the patient is registered before, then I. Update patient's status [UC-4] 5A. if the inserted data is incorrect or wrong II. The system displays a message that notifies the users to fill all the required information again. 7A. if the users enters a wrong patients information, I. The user re-writes the inserted wrong information of a patient. II. The system validates and verifies the entered data again. III. The system displays the edited patient's information in table form. IV. The system edits/updates the patient information in the database.

Table 4.2: Description of register patient use case.

C. Diagnosis result registration

Use case ID	UC-3
Use case Name	Register Diagnosis Result
Priority	High
Actor	Nurse or Health Officer
Description	Register observed sign and symptoms, and diagnosis and treatment plan.
Precondition	The user should be authenticated and authorized and Register Patient[UC-2]
Post condition	Patient's diagnosis result is registered to the system.
Main success scenario	1. Search the patient information by using patient's MRN or Name. [1A] 2. If the patient is registered before, the system displays previous patient history. 3. The user clicks on diagnosis result registration button to open diagnosis and treatment result registration form. 4. The system displays diagnosis result registration form. 5. The user first enters the observed sign and symptoms of a patient. Then clicks on next button to register the diagnosis result. 6. The system refreshes the diagnosis and treatment registration form that used to register the diagnosis result. 7. The user enters the diagnosis result to the corresponding fields and clicks on submit button. 8. The system validates and verifies the inserted data.[8A] 9. The patient information is saved to the system.
Alternative path	1A. if the patient is not registered before, then I. Register patient [UC-2] 8A. if the inserted data is incorrect or wrong I. The system displays a message that notifies the users to fill all the required information again

Table 4.3: Description of register diagnosis result use case.

D. Update patient status

Use case ID	UC-4
Use case Name	Update Patient Status
Priority	Medium
Actor	Nurse or Health Officer
Description	Updating patient's information like weight, height, date, and temperature.
Precondition	The user should be authenticated or authorized and Register Patient[UC-2]
Post condition	Patient's data is updated.
Main success scenario	1. The user clicks on OPD register button to open the patient's demographic information and diagnosis history. 2. The system activates the OPD register form. 3. The user enters patient's MRN or Name in the searching field. [3A] 4. If the patient is registered, the system displays patient's information. 5. The user clicks on update button. 6. The system displays a form to update patient's status 7. The user enters all the required data to the fields and click submit button. 8. The system validates the inserted data.[8A] 9. The patient's updated data is saved to the system.
Alternative path	3A. if the patient is not registered before, then I. Register patient [UC-2] 8A. if the inserted data is incorrect or wrong I. The system displays a message that notifies the users to fill all the required information again

Table 4.4: Description of update patient status use case.

E. Search patient history

Use case ID	UC-5
Use case Name	Search Patient's History
Priority	Medium
Actor	Nurse or Health Officer
Description	Describes how the patient previous history is searched.
Precondition	The user should be authenticated or authorized and [UC-3]
Post condition	Patient's history displayed
Main success scenario	1. The user clicks on OPD register button to open the patient's demographic information and diagnosis history. 2. The system activates the OPD register form. 3. The user enters patient's MRN or Name in the searching field. [3A] 4. The system searches the patients detail information from system's database.[4A] 5. If the patient is registered before, the system displays full information about the patient. 6. The user can access the demographic information and all the previous diagnosis history of the patient.
Alternative path	3A. if the user wants to search the patient's information using his/her name, I. The user enters the patient's name. II. The system displays list of patients name at the bottom of searching text-field that are related to the name written in the searching field. III. The user selects the patient name from the suggested name list. IV. The system displays patient's demographic information and previous diagnosis history. 4A. if no result found about the patient history, then I. The system notifies that no patient's history is found in the system.

Table 4.5: Description of Search patient history use case.

F. Generate Report

Use case ID	UC-6
Use case Name	Generate Report
Priority	Low
Actor	Manager
Description	To generate different type of reports from the recorded data.
Precondition	The user should be authenticated or authorized.
Post condition	Display the report.
Main success scenario	1. The user clicks a report button. 2. The system displays a report window that contains different report options. (Case, monthly, quarterly). 3. The user selects a report type and enters report parameters. 4. The system processes and generates the selected report. 5. The system displays the report.[5A]
Alternative path	5A. if the user wants to get the report in different format then, I. Click on export the report button. II. Select the file type of the report.

.

Table 4.6: Description of generate report use case

G. Check IMNCI guideline

Use case ID	UC-7
Use case Name	Check Guideline
Priority	High
Actor	Nurse or Health Officer
Description	To open a guideline for specific problem.
Precondition	The user should be authenticated or authorized.
Post condition	The system displays the guideline.
Main success scenario	1. The user clicks on guideline button. 2. The system displays a guideline window. 3. The user selects the guide that he/ she want to see. 4. The system displays the guideline in detail.
Alternative path	None

Table 4.7: Description of check guideline use case.

H. Diagnosis Assistant

Use case ID	UC-8
Use case Name	Diagnosis Assistant
Priority	Medium
Actor	Nurse or Health Officer
Description	To assist the user when he/she needs a help for diagnosis and treatment plan suggestion to the observed sign and symptoms
Precondition	The user should be authenticated or authorized. [UC-3]
Post condition	The user gets diagnosis and treatment plan suggestion
Main success scenario	1. The user first insert/ select the observed sign and symptoms on diagnosis registration window. 2. The user clicks on diagnosis assistant button. 3. The system dissects and converted the inserted sign and symptoms into attributes. 4. The system compares the attributes with the conditions and rules of the knowledge base. 5. The system displays the suggestion for diagnosis and treatment plan to the observed sign and symptoms.[5A] 6. The user accesses the suggestion and click on exit button to close the diagnosis assistant window when he/she finish.
Alternative path	5A. if there is no much found, then I. The system notifies to the user that no suggestion found that related to the inserted sign and symptoms

Table 4.8: Description of diagnosis assistant use case.

I. Logout

Use case ID	UC-9
Use case Name	Log out
Priority	Medium
Actor	System administrator, nurse, health officer, manager
Description	To logged out from the system.
Precondition	The user must be authenticated and authorized.
Post condition	The user logged out from the system.
Main success scenario	1. If the user wants to log out from the system. 2. The user click on logout button. 3. The system confirms the user is logging out.
Alternative path	None

Table 4.9: Description of logout use case.

J. Manage user accounts

Use case ID	UC-10
Use case Name	Manage User Account
Priority	Low
Actor	System Administrator, Manager
Description	Describes how to create, manage or remove a user account.
Precondition	The user must be authenticated and authorized.
Post condition	User account information is managed.
Main success scenario	1. Open account management window. 2. Search/ check an employee has a user account by using employee ID. [2A] 3. If the employee didn't have an account, then the user click on create new account button. 4. The system displays a window/ form to create new account. 5. The user fills the required filed in the form. And click submit button. 6. The system validates and confirms the inserted data. [6A]. 7. New account information is saved to the system.
Alternative path	2A: If the employee has an account before: I. If the user wants to update the employee's account information. a) click on update/ edit account button b) The system displays a form to update the user account. c) The user fills the required information and click on save button. d) The system validates the inserted data. e) The updated account information is saved. II. If the user wants to delete the employee's account. a) Select and click the employees account. b) The system displays a message to confirm account deletion. c) The user confirms the deletion process by clicking delete button. d) The system deletes the account information from the database. 6A: If the password mismatch with confirm password field I. The system notifies the user entered wrong data. II. The system let the user to enter password and to confirm it.

Table 4.10: Description of manage account use case.

4.1.5.2 Class Diagrams

Class diagrams are used to describe the structure of the system in terms of classes. Class diagrams describe the system in terms of objects, classes, attributes, operations, and their association. Identified classes and relationship among them is shown in Figure 4.2.

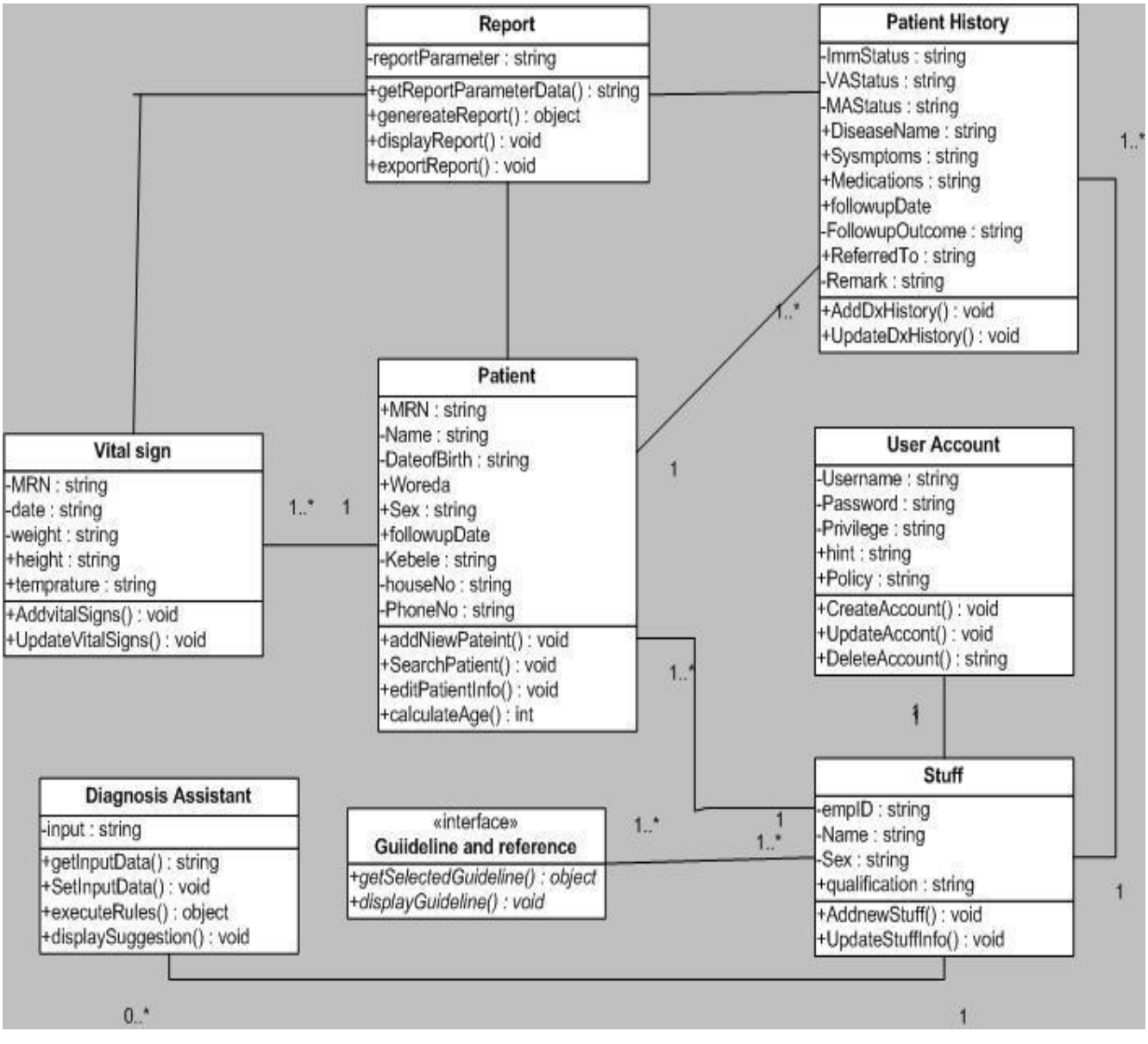


Figure 4.2: Class diagram of e-IMNCI system.

4.1.5.3 Sequence Diagram

UML sequence diagrams is a dynamic modeling technique, that shows an interaction between objects arranged in a time sequence. A sequence diagram aims to represent the detailed object interaction that occurs for a single use case; therefore these diagrams can be seen as an expansion of use cases to the lowest possible level of detail [22]. By looking at what messages are being sent to an object, and by looking at roughly how long it takes to run the invoked method, we quickly get an understanding of where we need to change our design to distribute the load within your system.

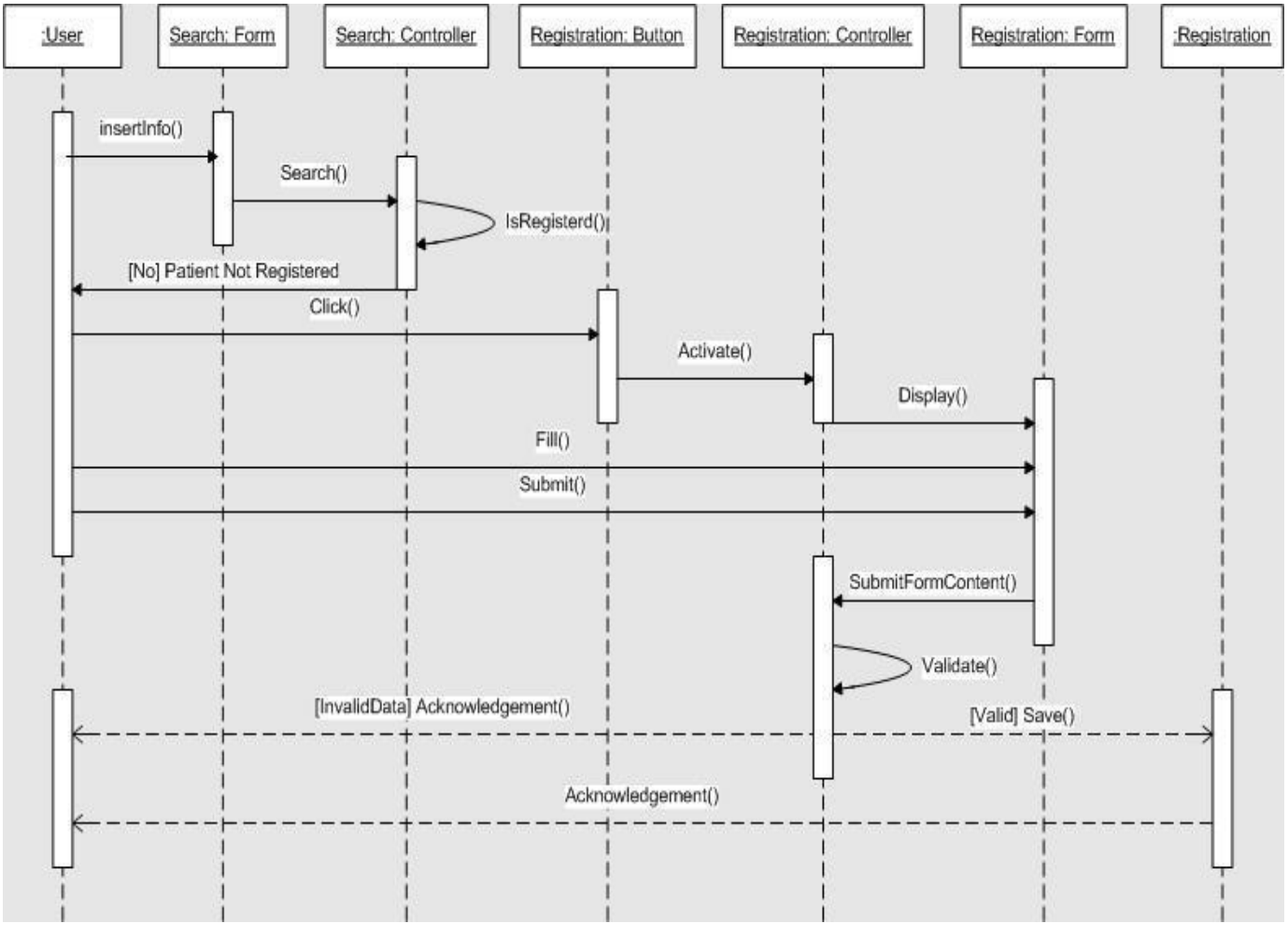


Figure 4.3: Sequence diagram for Register Patient use case.

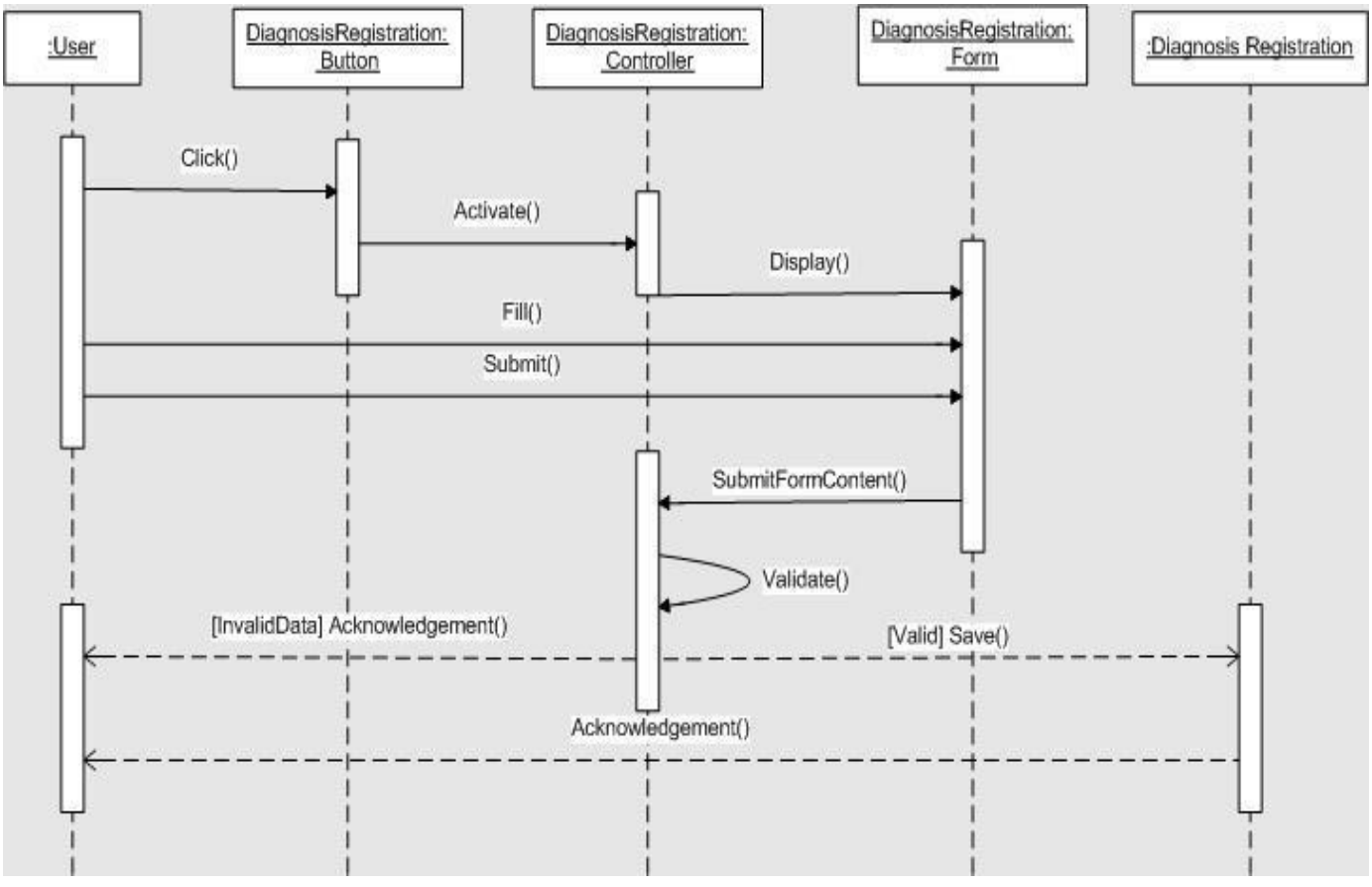


Figure 4.4: Sequence diagram for register diagnosis result use case.

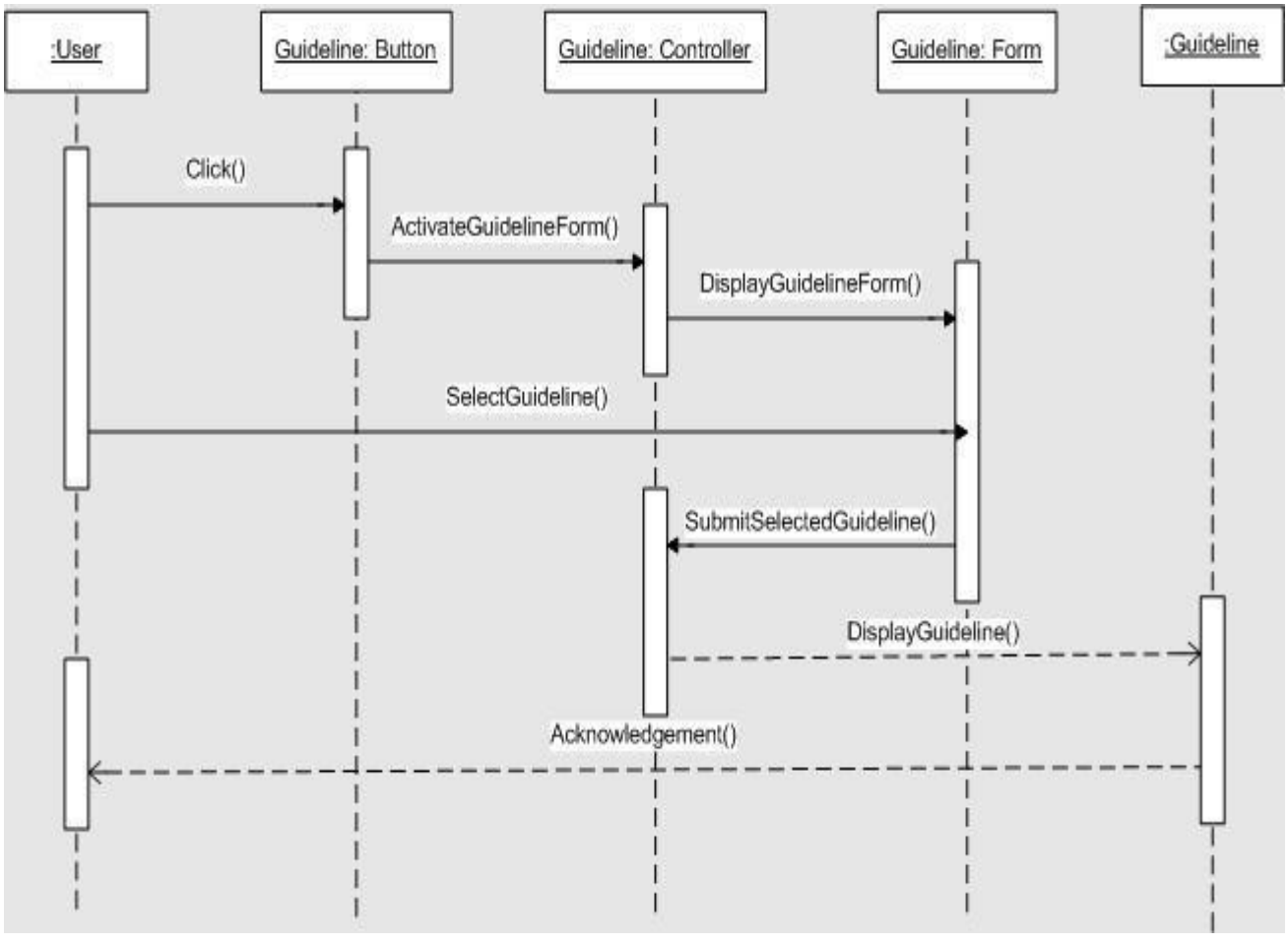


Figure 4.5: Sequence diagram for check guideline use case.

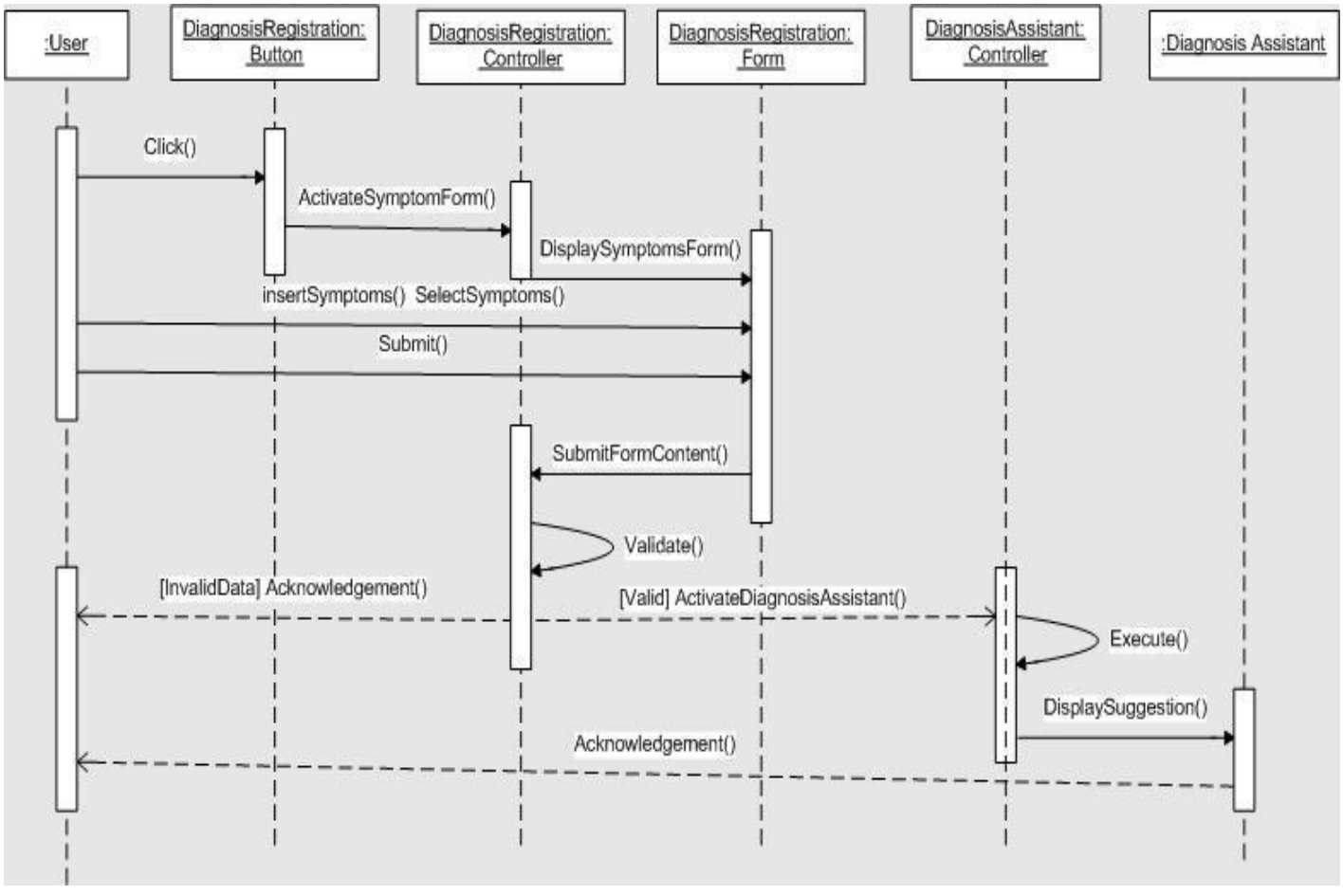


Figure 4.6: Sequence diagram for diagnosis assistant use case.

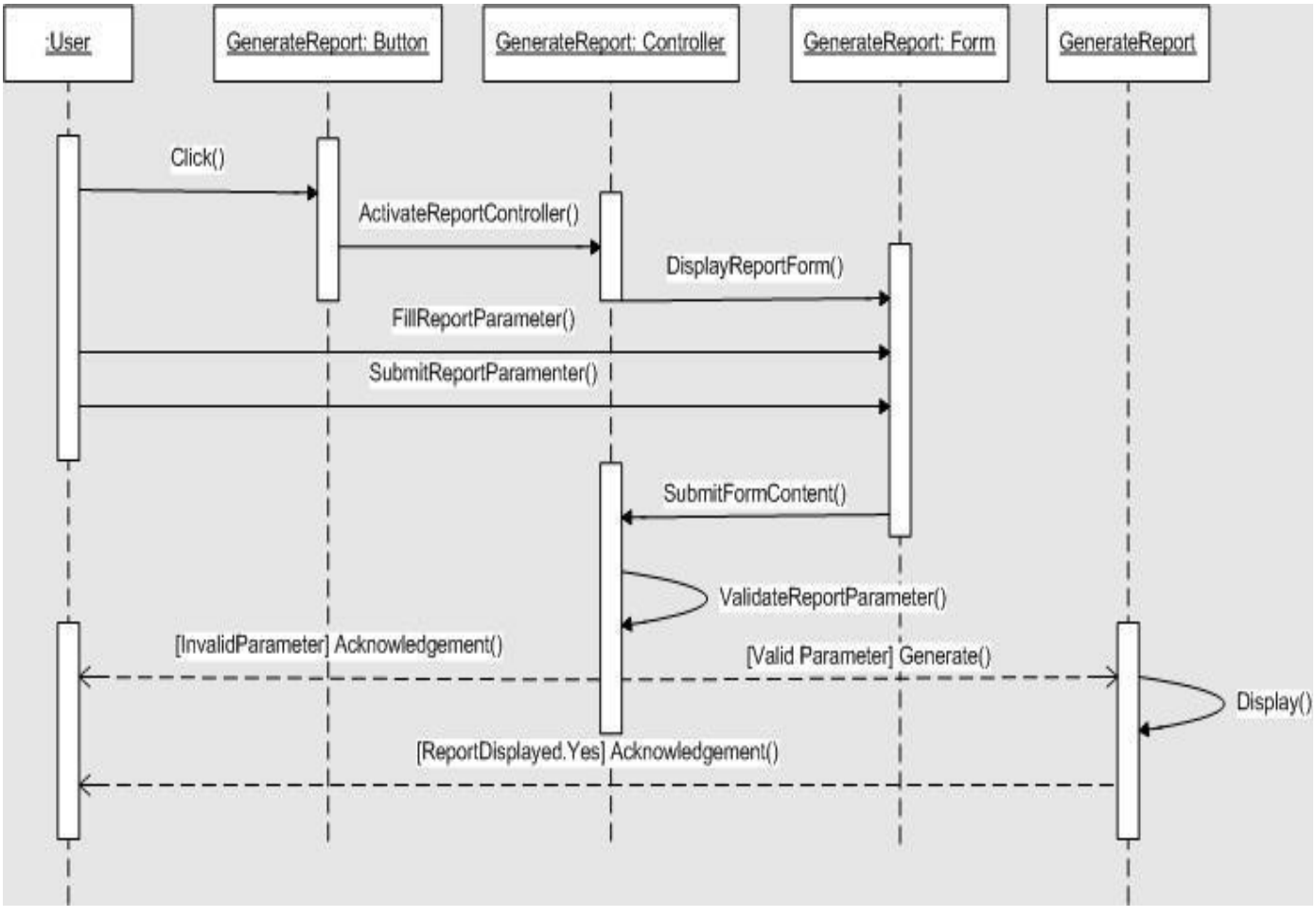


Figure 4.7: Sequence diagram for generate report use case.

4.3 System Design

4.3.1 Architecture of e-IMNCI System

Basically, the idea of this project was to develop electronic version of IMNCI to improve the service delivery of child health in health centers. In the first chapter, the scope of the project was described in detail. The e-IMNCI protocol will follow the same protocol in the current paper system, guiding healthcare workers step by step through the child's assessment, classification, treatment, and communication of instructions to the care-giver. The system includes the features like record management, clinical decision support, report generating and provides a reference & guideline. The overall architecture the system is described below in figure 4.8.

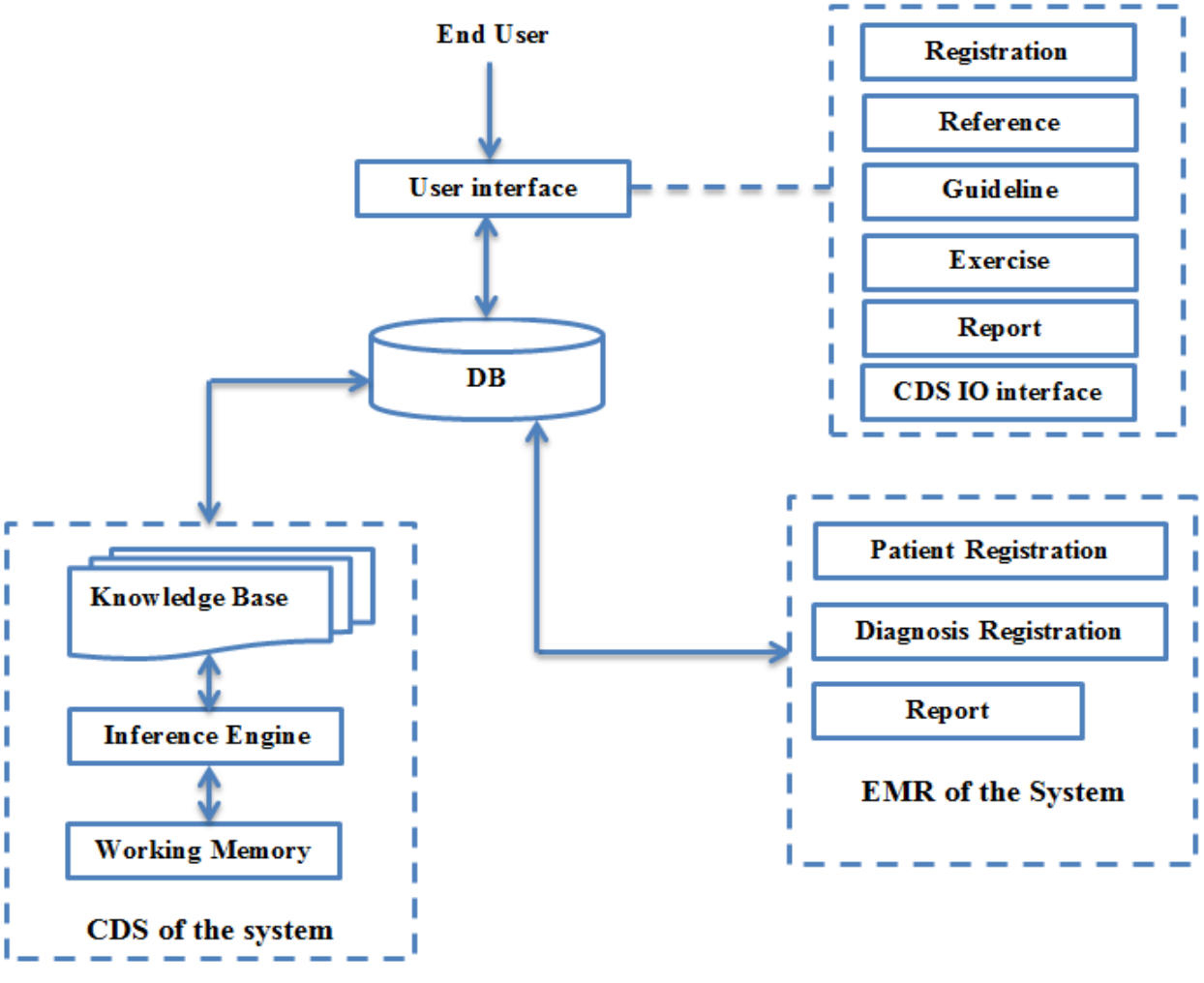


Figure 4.8: Architecture of the e-IMNCI system

4.3.2 Subsystem Decomposition

In order to make system design and development easy, the system is decomposed into smaller parts called subsystems. In this stage, classes with similar functionality are grouped into a single subsystem. During system decomposition, achieving strong coherence within the subsystem and loose coupling between subsystems were taken into consideration. The major subsystems identified are shown in Figure 4.9.

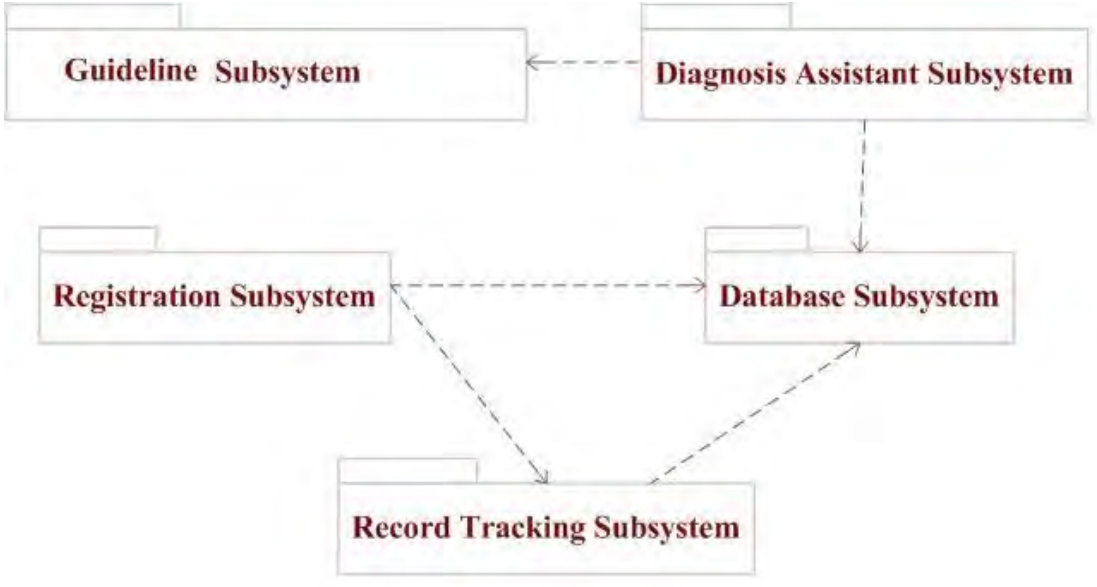


Figure 4.9: Subsystem decomposition of e-IMNCI

Description of each subsystem is given as follows:

- **Database Subsystem:** It provides persistent data storage mechanism for storing data. All other subsystems interact with this subsystem.
- **Registration Subsystem:** it includes the system's components that used to register patient information and diagnosis result.
- **Record Tracking Subsystem**: depending on the recorded data in the registration subsystem, this record tracking subsystem is used to manage storing, updating, searching, and retrieving information efficiently.
- **Diagnosis Assistant Subsystem:** provides a diagnosis and treatment plan suggestions based on the observed sign and symptoms. It also includes treatment plan suggestion about drug's side effect, contraindication, drug interaction and dosage for a specific drug.
- **Guideline Subsystem:** it includes an electronic version of IMNCI guideline.

4.3.3 Hardware/Software Mapping

One of the major and important tasks in system design deals with hardware/software mapping that describes how it is possible to realize the subsystems (i.e. hardware/software) and how the object model is mapped on the chosen hardware and software. In this system, the whole part of the system runs on user's computer which is found in IMNCI department. Issues for such type of components of software architecture can be addressed by asking question like; what is the hardware configuration of the system? Which node is responsible for which functionality? Computers are modeled as nodes in UML deployment diagrams which are used to depict the relationship among run-time components and hardware nodes. Thus we selected a single window machine as the virtual machine on which the system will run. Once the hardware configuration has been defined and the virtual machines selected, subsystems are assigned to nodes. Figure 4.10 shows the deployment diagram of the subsystem.

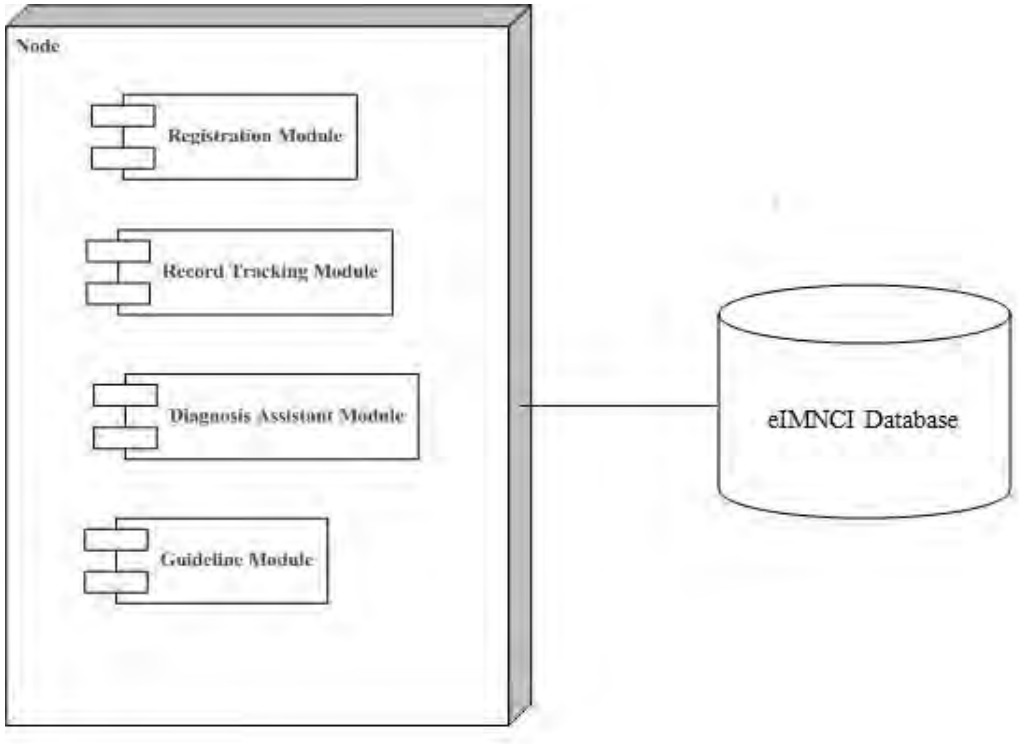


Figure 4.10: Deployment diagram of e-IMNCI system

4.3.4 Persistent Data Management

The requirements and the collection analysis phase produce both data requirements and functional requirements. The data requirements were used as a source of database design. The data requirements were specified in as detailed and complete form. In parallel with specifying the data requirements, the functional requirements of the system were also specified and described in the analysis and design phase of the study. These consist of user-defined operations is applied to the database (retrievals and updates). The functional requirements were used as a source of application software design. The physical data model of a database describes database tables, attributes and relationship between tables. To develop this data model I used ER diagram which designed for to create a common understanding between the developer and end users. Therefore, the ER diagram was mapped to database schema by using different sequential mapping algorithms. After the database schema is created the next step is normalizing the database schema by insert some values to its attribute.

In the picture below (figure 4.11) there are the main phases of database design. Database design is connected with application design.

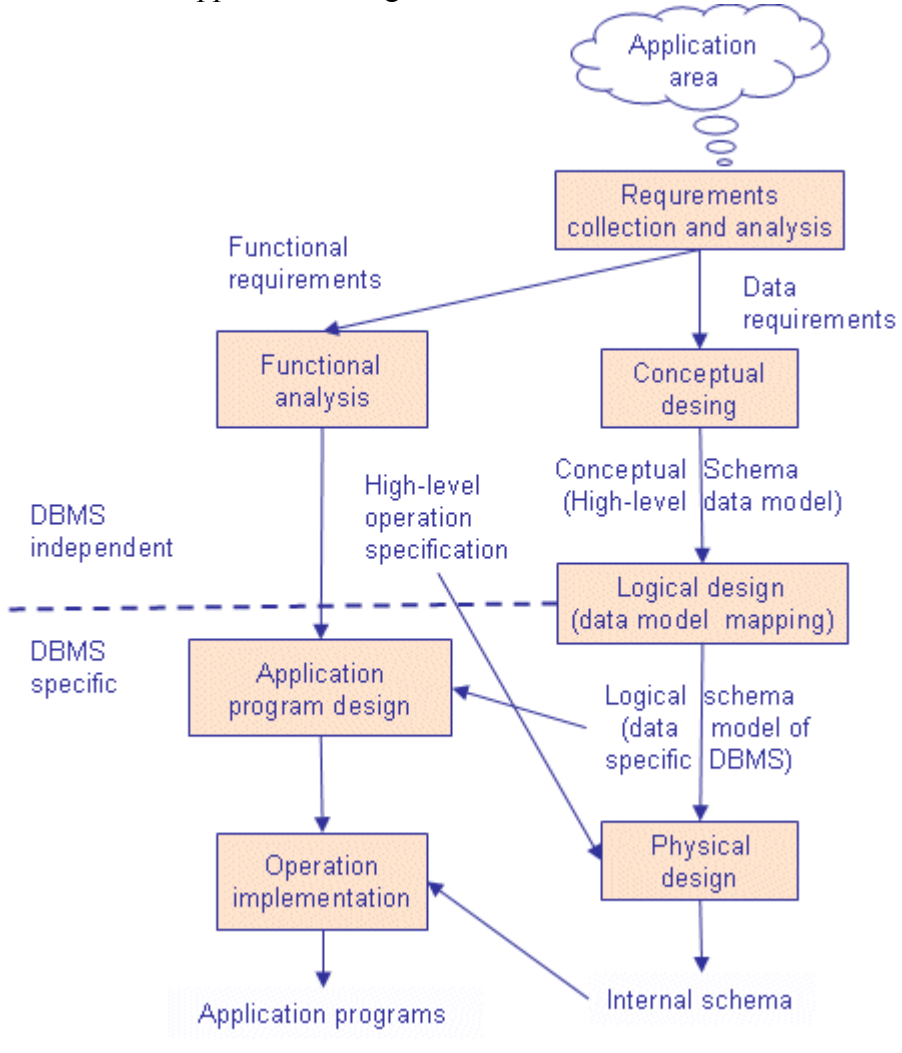


Figure 4.11: Main phases of database design

Conceptual Design

Once all the requirements have been collected and analyzed, the next step was to create a conceptual schema for the database, using a high level conceptual data model. The result of this phase is an Entity-Relationship (ER) diagram or UML class diagram. It is a high-level data model of the specific application area. It describes how different entities (objects, items) are related to each other. It also describes what attributes (features) each entity has. It includes the definitions of all the concepts (entities, attributes & relationship between entities) of the application area. ER diagram was used to demonstrate the conceptual data model of the e-IMNCI system's database.

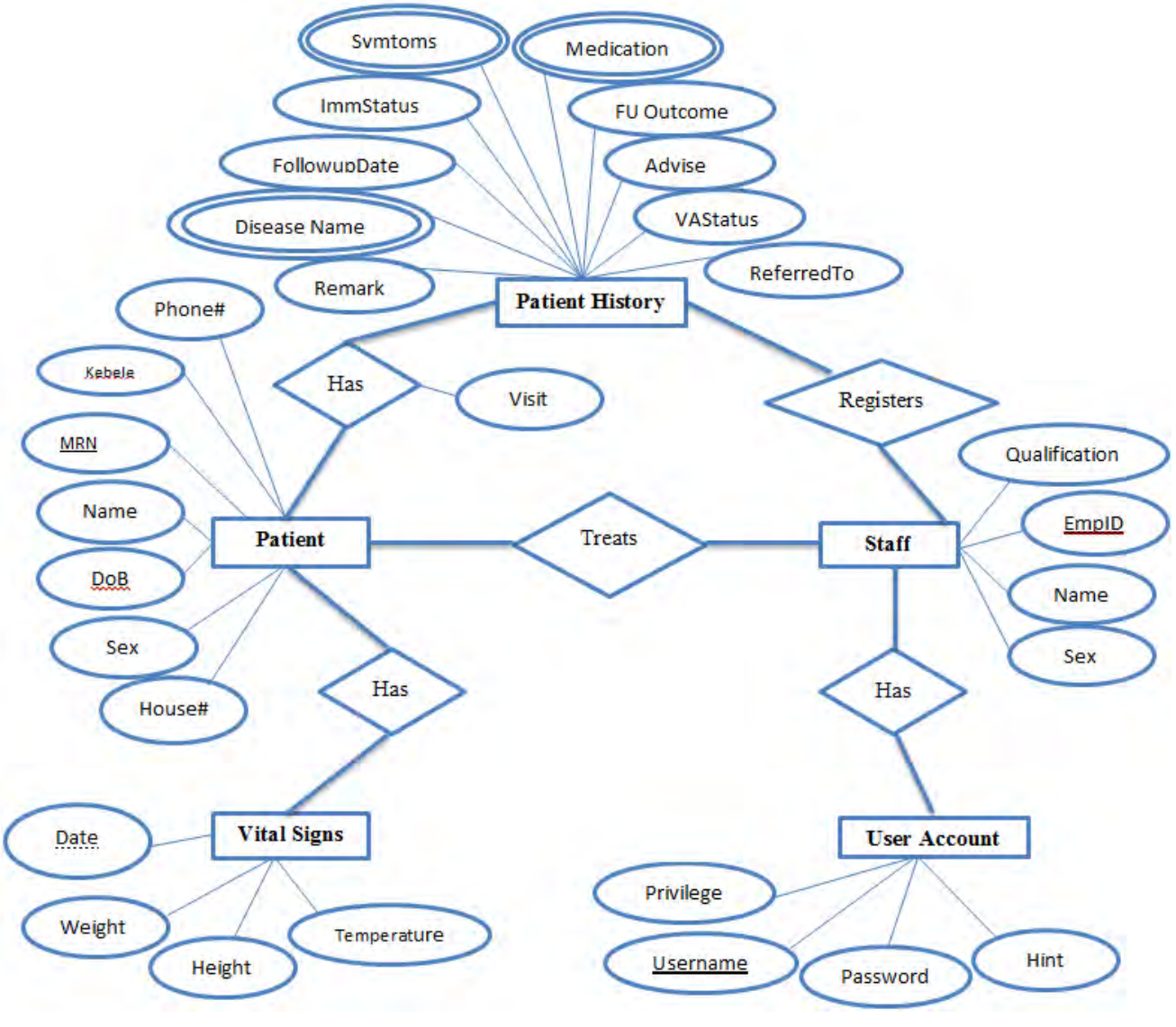


Figure 4.12: Entity relationship(ER) diagram for the prototype of e-IMNCI.

Logical Design

The result of the logical design phase (or data model mapping phase) is a set of relation schemas. The ER diagram or class diagram is the basis for these relation schemas. To create the relation schemas is quite a mechanical operation. There are rules how the ER model or class diagram is transferred to relation schemas. The relation schemas are the basis for table definitions. In this the primary keys and foreign keys were defined which were not defined in previous phase. By applying mapping algorithm to the above ER diagram the following database schema was created.

Patient								
MRN	Name	DoB	Sex	Woreda	Kebele	House#	Phone#	Last Visit

Vital Sign				
MRN	**Date**	Weight	Height	Temperature

Patient History										
MRN	**Date**	**DFlag**	VaStatus	ImStatus	MAStatus	FUDate	Advise	RefTo	Remark	Username

Diagnosis Result			
MRN	**Date**	**Disease**	Visit

Treatment Result		
MRN	**Date**	**Treatment**

Observed Sign & Symptoms		
MRN	**Date**	**Symptoms**

Staff					
EmpID	Name	DoB	Sex	Qualification	Phone#

User Account				
Username	Password	Password Hint	Privilege	EmpID

Figure 4.13: Database schema of e-IMNCI system.

Normalization

Normalization is the last part of the logical design. The goal of normalization is to eliminate redundancy and potential update anomalies. Redundancy means that the same data is saved more than once in a database. Update anomaly is a consequence of redundancy. If a piece of data is saved in more than one place, the same data must be updated in more than one place. Normalization is a technique by which one can modify the relation schema to reduce the redundancy. Each normalization phase adds more relations (tables) into the database. For this study, first normal form, second normal form and third normal form were applied to eliminate redundancy and potential update anomalies in the database schema.

Physical Design

The goal of the last phase of database design, physical design, was to implement the database. At this phase database management system (DBMS) is used. MsSQL 2008 was used to create the physical design of the database. The indexes, the integrity constraints (rules) and the users' access rights were defined here. The physical design of a database describes database tables, attributes and relationship between tables. To develop this database design I used ER diagram in conceptual design, which is designed for to create a common understanding between the developer and end users. After that, ER diagram was mapped to database schema by using different sequential mapping algorithms. Finally, the database schema normalized by using first, second and third normal forms to create the physical design of the e-IMNCI's database. The physical data model of e-IMNCI system is illustrated below in figure 4.14.

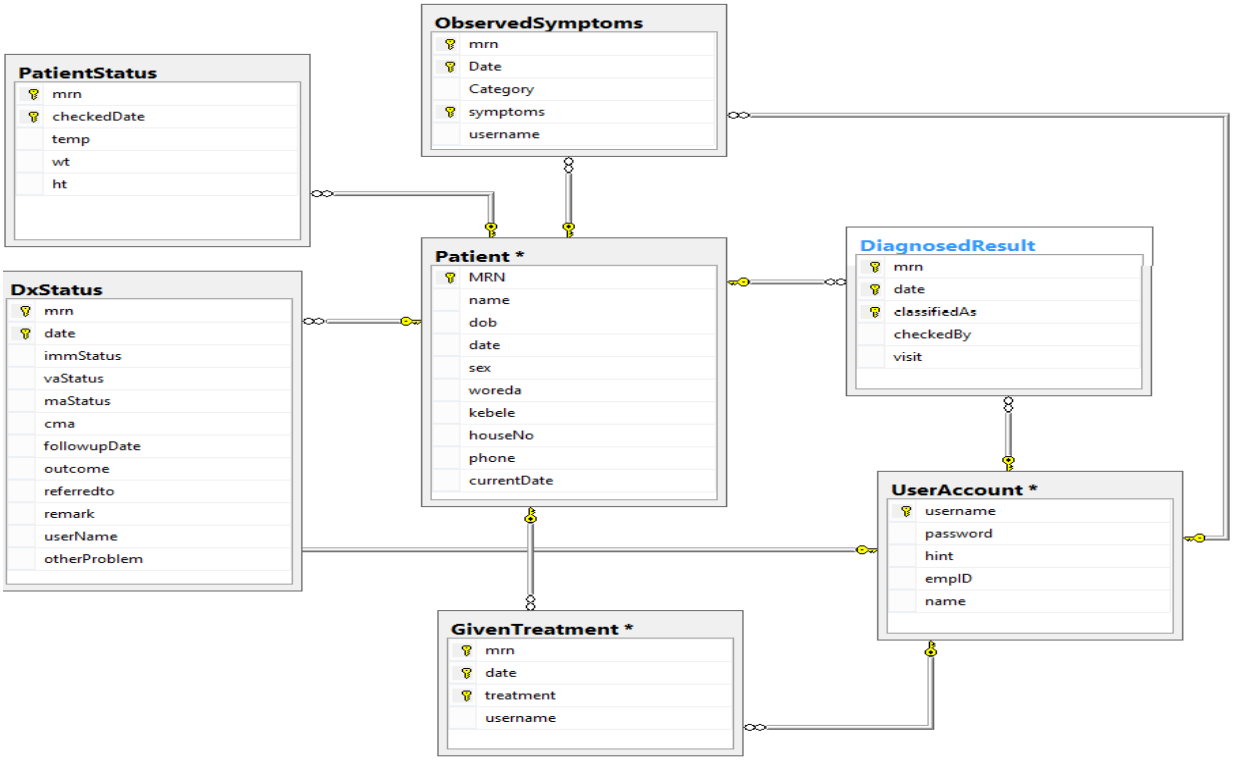


Figure 4.14: Database diagram of e-IMNCI system.

4.4 System Implementation

4.4.1 Guideline

The guideline and reference module of the system is a detail information source that helps health professionals to diagnose and treat under 5 years' child health problems. The guideline is developed under the paper based WHO IMNCI guideline and protocols. The guideline only needs 2 clicks to open a specific guide to see the diagnosis and treatment plan for the observed problem. Hence, the guideline is easier to use and access rather than the paper based guideline which is used before.

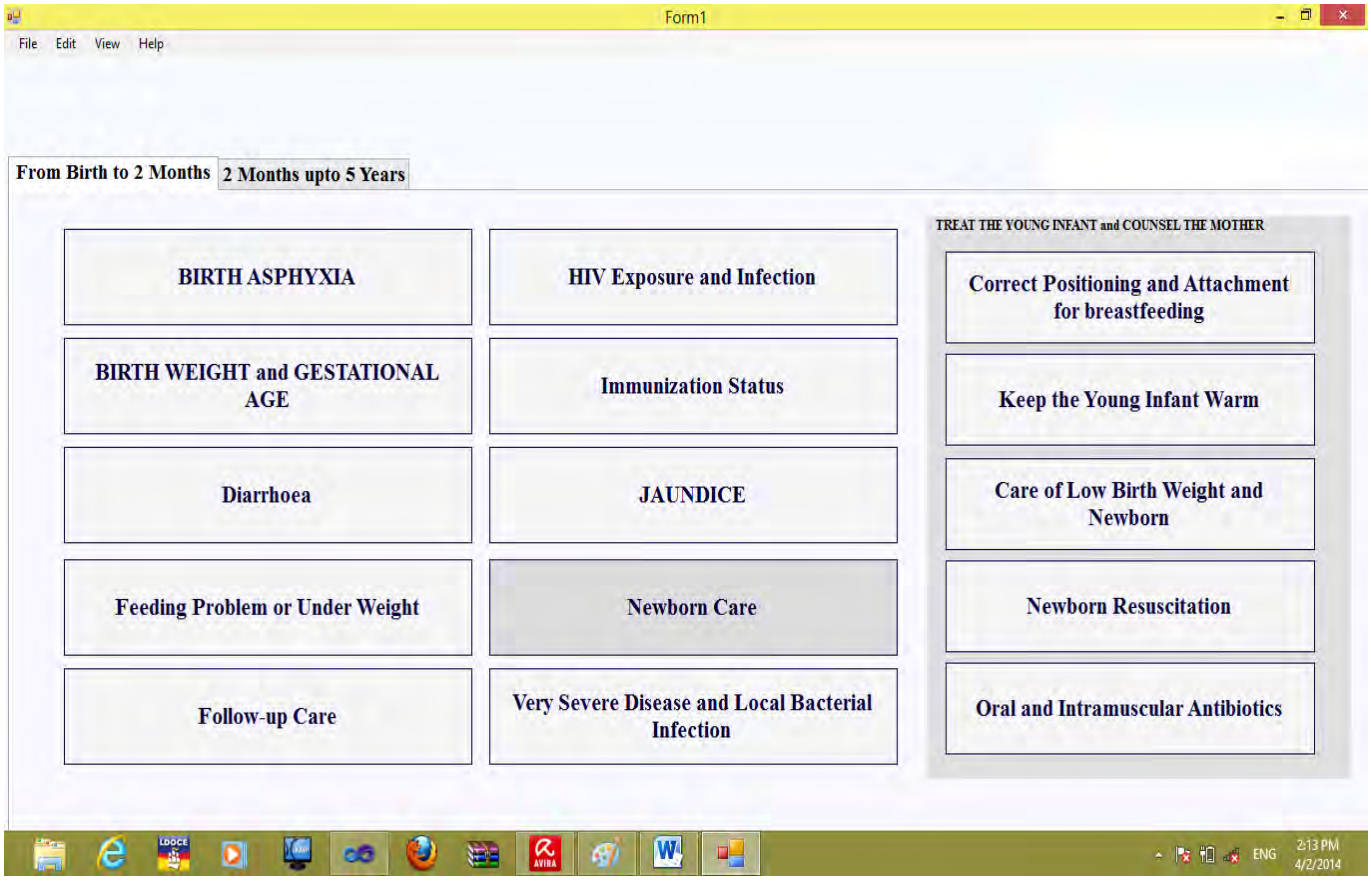


Figure 4.15: A user interface to select the guideline.

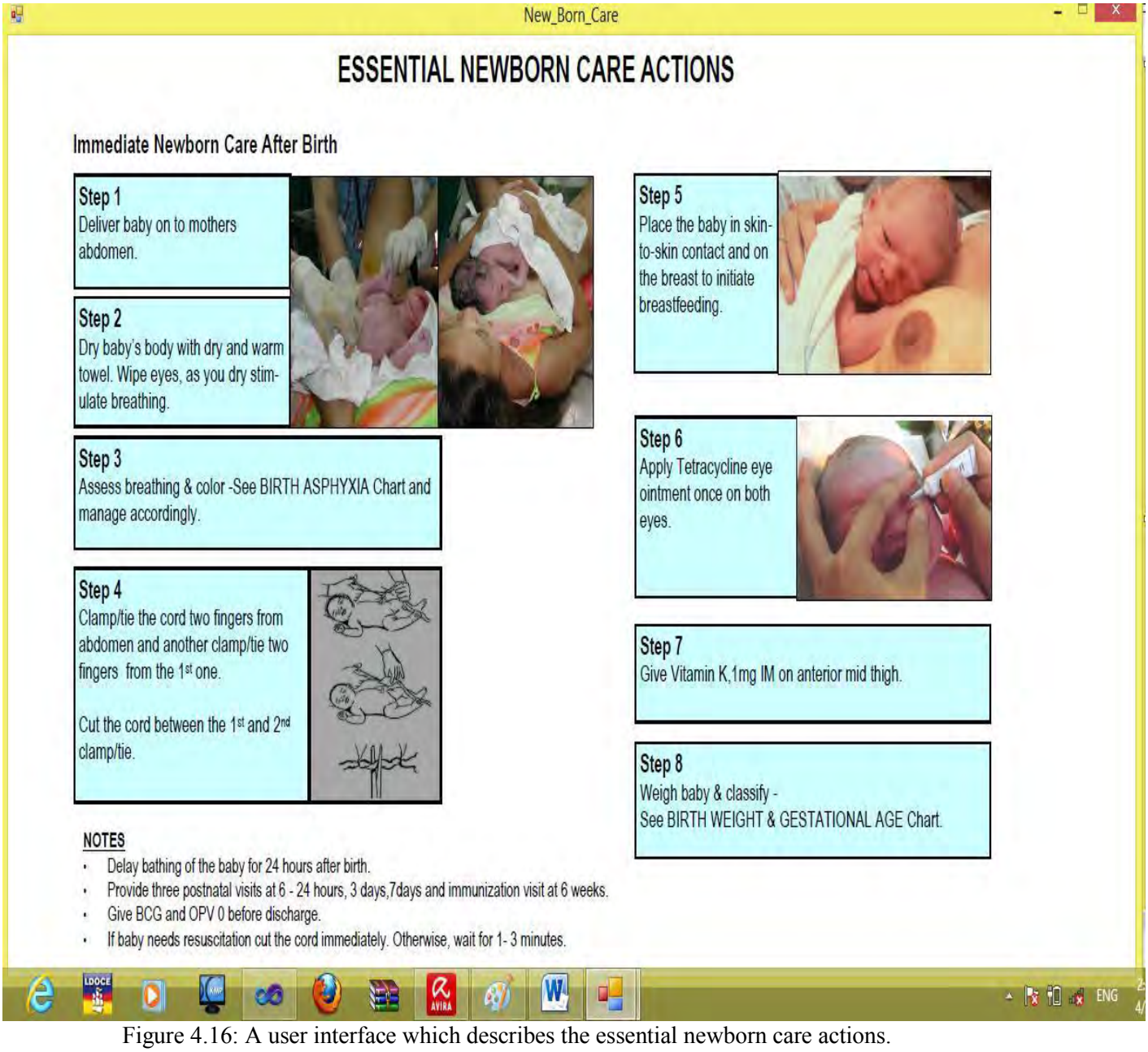


Figure 4.16: A user interface which describes the essential newborn care actions.

4.4.2 Electronic Medical Record of e-IMNCI

Electronic medical records are designed to capture, store, retrieve and present clinical data in proper and easy way. Some of the advantages of EMR is that the records are organized, legible and, therefore, likely to minimize misunderstandings as well as patient care errors.

This e-IMNCI software serves as both an electronic medical record system and a clinical decision support system. Electronic medical record module of the system registers patient's information & diagnosis result and generates different kind of reports. However, the system's EMR doesn't include other department's task like laboratory, pharmacy and card room.

A. Patient Registration

The first component of the EMR module is patient registration. This component used to register a patient's information when the patient comes to the health center for the first time and the system gives a unique id for each patient. Once the patient is registered, only patient's current status (like weight, height, temperature, visit and date) will be registered to the system.

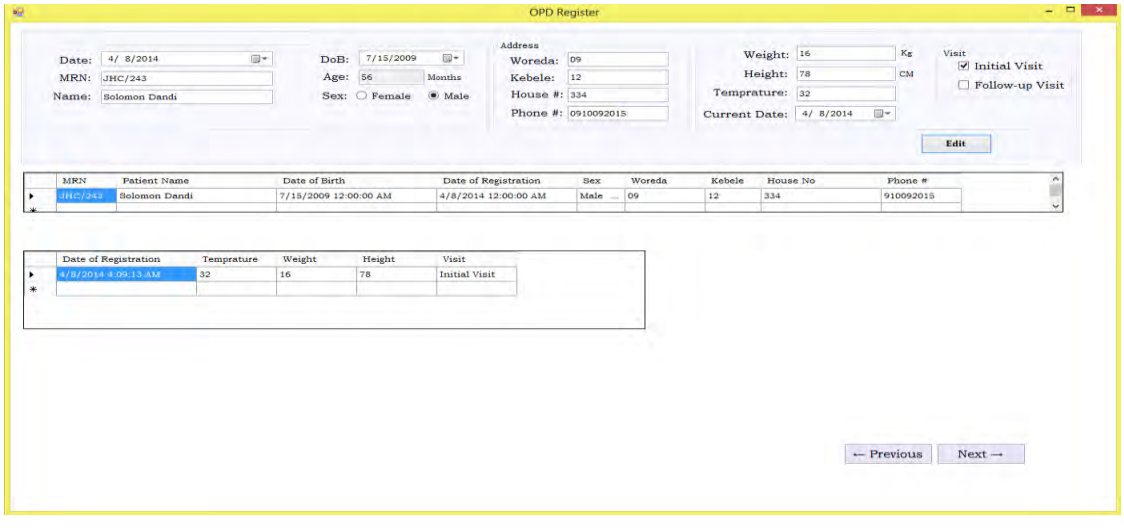


Figure 4.18: A user interface designed to register patient's information.

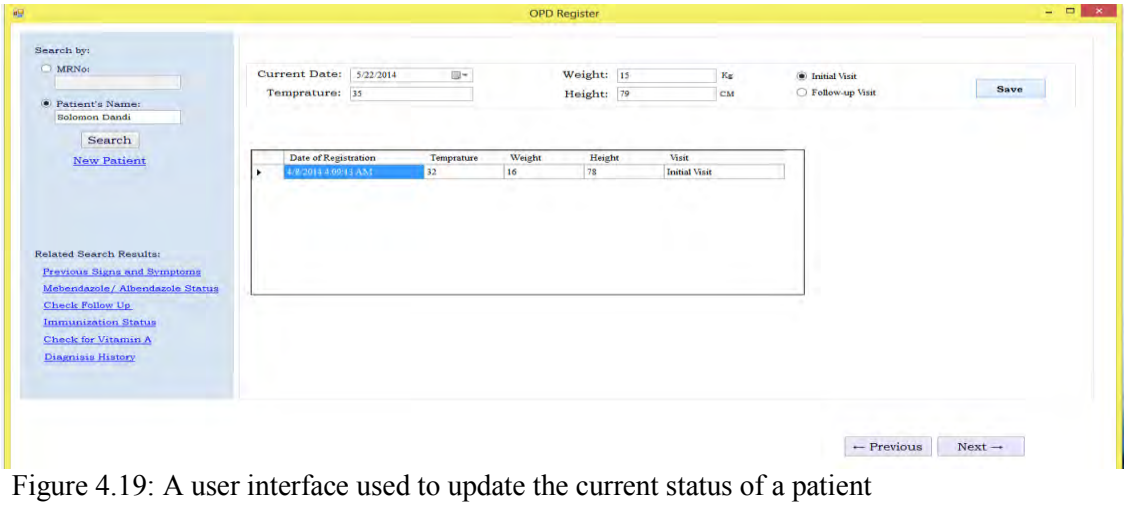


Figure 4.19: A user interface used to update the current status of a patient

B. Diagnosis Registration

Once the patient is registered in to the database, the next step is entering the observed sign and symptoms. The symptoms lists in the system are grouped into different categories depending on the behavior of the symptoms. Figures 20 and 21 show the user interface to register observed sign and symptoms, and diagnosis & treatment result, respectively.

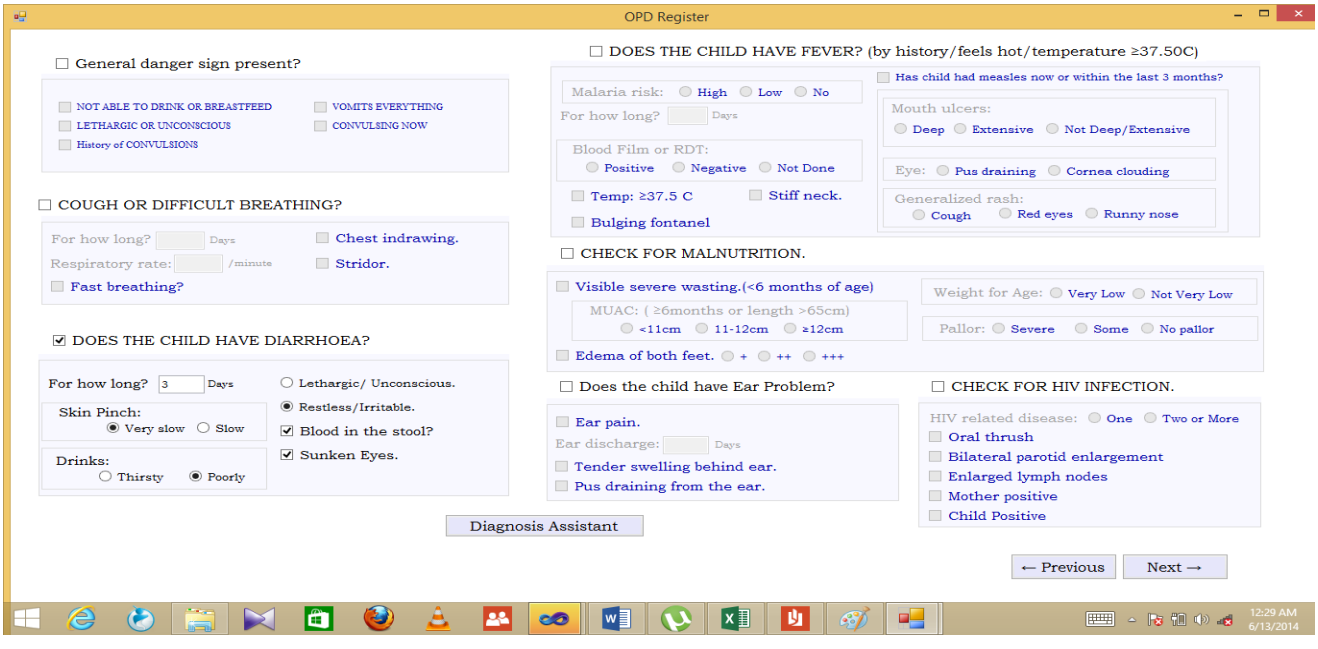


Figure 4.20: A screen to select and enter the observed sign and symptoms of the patient.

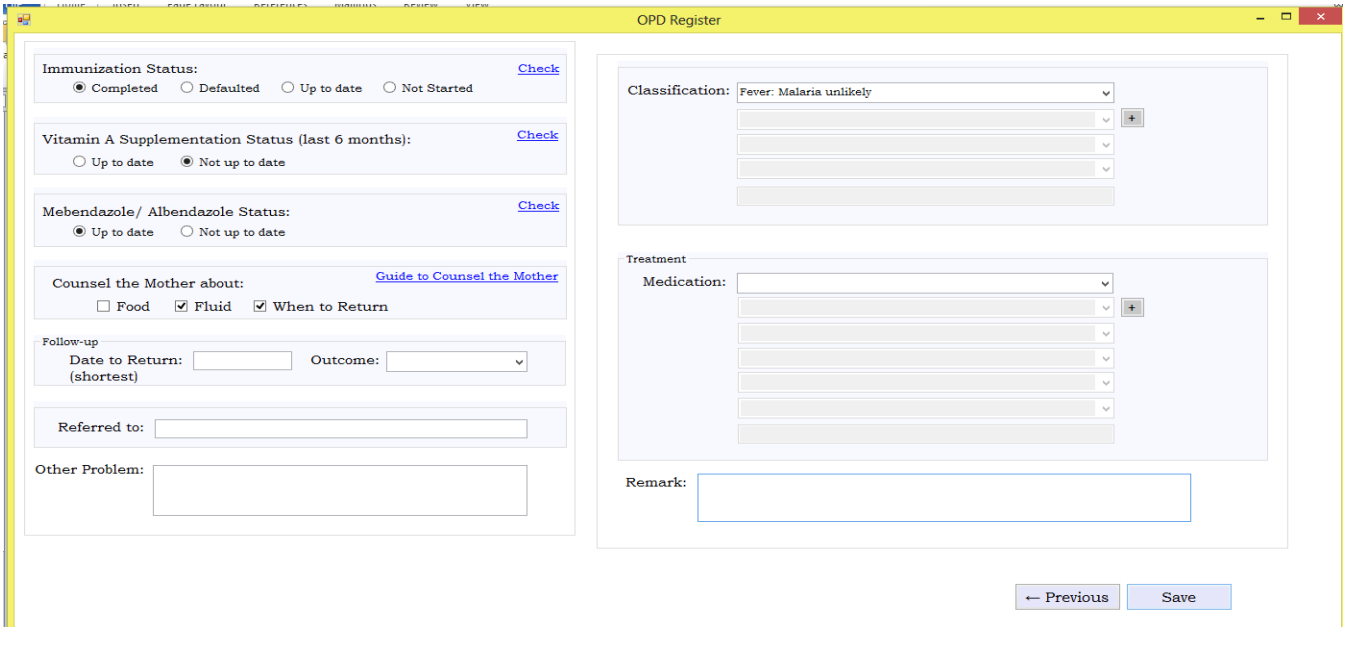


Figure 4.21: A screen designed to register diagnosed results of a patient.

4.2.4 Clinical Decision Support

This module of the system used to identify the possible diagnosis and treatment based on the observed sign and symptoms. It has four main components these are knowledge base, inference engine, temporary working memory and the user interface. The details of those components are described below:

Knowledge Base

In this system a rule-based knowledge representation and reasoning is followed. The rule based reasoning method is a commonly used technique for the development of knowledge-based systems. Knowledge is represented in the form of condition-action pairs: IF this condition (or antecedent-condition or premise) occurs, THEN some action (or conclusion or consequence) will occur.

Knowledge base is a set of rules or the encoded knowledge about diagnosis and treatment for highly prevalent diseases in childhood (Pneumonia, Malaria, Diarrhea and Malnutrition). The data for the knowledge base was collected from IMNCI guideline; this data was processed and converted into the patterns to define rules. The validated knowledge is represented in the form of rules by rule-based representation technique and the rules are codified to the knowledge base of the system using C#.NET programming language. The rules described in table 4.11 were implemented as part of the program but not separately. The following rules in the knowledge base of system are expressed in natural language rules IF … THEN …

Rule #	Rule in details
Rule 1	**IF** Difficult Breathing **AND** Cough **OR** Lower Chest In-drawing **OR** Fast Breathing **OR** Unable to drink or breastfeed **OR** Vomiting Everything **OR** Convulsion History **OR** Convulsion Now **OR** Lethargic/ Unconscious **OR** Stridor in calm child **THEN "**Severe Pneumonia**"**
Rule 2	**IF** Cough **AND** Difficult Breathing **AND** Fast Breathing **THEN** "Pneumonia "
Rule 3	**IF** Cough **AND** Difficult Breathing **THEN** "No Pneumonia: Cough or Cold "
Rule 4	**IF** two or more of the following signs, Lethargic or unconscious **OR** Sunken eyes **OR** Not able to drink or drinking poorly **OR** Skin pinch goes back very slowly **THEN "**Severe dehydration"
Rule 5	**IF** two or more of the following signs, Restless irritable **OR** Sunken eyes **OR** Drinks eagerly, thirsty **OR** Skin pinch goes back slowly **THEN** "Some dehydration"
Rule 6	**IF** there are no enough signs to classify as "some" or "severe" dehydration. **THEN "**No dehydration"
Rule 7	**IF** diarrhea lasts for 14 days or more and dehydration is present. **THEN** "Severe persistent diarrhea"
Rule 8	**IF** diarrhea lasting for 14 days or more and there is no dehydration **THEN** "Persistent diarrhea"
Rule 9	**IF** there is blood in the stool **THEN** "Dysentery"
Rule10	**IF** Unable to drink or breastfeed **OR** Vomiting Everything **OR** Convulsion History **OR** Convulsion Now **OR** Lethargic/ Unconscious **OR** Stiff Neck **AND** High Malaria Risk **THEN "**Very Severe Febrile Disease**"**
Rule11	**IF** Acute Fever **AND** Chills **AND** Sweating and Headache **AND** Coagulation Defects **AND** Shock **AND** Renal and Liver Failure **AND** Joint Pain **AND** Loss of Appetite **AND** Vomiting **THEN "**Malaria**"**
Rule12	**IF** RDT OR Blood Film= Positive **THEN "**Malaria**"**

Rule13	**IF** RDT OR Blood Film= Negative **THEN "**Fever: Malaria Unlikely**"**
Rule14	**IF** Unable to drink or breastfeed **OR** Vomiting Everything **OR** Convulsion History **OR** Convulsion Now **OR** Lethargic/ Unconscious **OR** Stiff Neck **AND** Low Malaria Risk **THEN "**Very Severe Febrile Disease**"**
Rule15	**IF** No runny nose AND No measles AND No other cause of fever **AND** Low Malaria Risk **THEN "**Malaria**"**
Rule16	**IF** Runny nose **OR** other cause of fever **AND** Low Malaria Risk **THEN "**Fever: Malaria Unlikely**"**
Rule17	**IF** Acute Fever **AND** No Malaria Risk **THEN** "Fever: No Malaria "
Rule18	**IF** Unable to drink or breastfeed **OR** Vomiting Everything **OR** Convulsion History **OR** Convulsion Now **OR** Lethargic/ Unconscious **OR** Stiff Neck **AND** Clouding of cornea **OR** Deep or extensive mouth ulcers **THEN "**Severe complicated Measles**"**
Rule19	**IF** Pus draining from the eye **OR** Mouth ulcers (not deep or extensive) **THEN** "Measles with Eye Or Mouth Complications"
Rule20	**IF** Measles now or within the last 3 months **AND** Fever **THEN** "Measles"

Table 4.11: Knowledge base rules expressed in natural language.

Inference Engine and Temporary Working Memory

According to the end-user input and the set of facts from the knowledge base, the inference engine infers facts or makes conclusions from the knowledge base. A rule based inference engine is used to infer the knowledge base. Each rule were defined in the form of

If X, then Y

The set X is called the conditions and the set Y is the consequents. A rule is triggered if all the conditions are satisfied and then the consequents are fired. The working expert system has both a rule base and working memory. The rule base used as a reaction system, each consequent is action.
A dialog is constructed by using the user interface and the system. The user provides information about the observed sign and symptoms and the system attempts to provide a solution (diagnosis and treatment plan suggestion) from the knowledge base. These insights were provided by the inference engine after examining the knowledge base.

Forward chaining inference method was used to infer the rules of the knowledge base. This method involves assigning values to attributes, evaluating conditions and checking to see if all of the conditions in a rule are satisfied. A general algorithm for this method is:

While values for attributes remain to be input
Read values and assign to associate
Evaluate conditions
Fire rules whose conditions are satisfied.

The relationship between attributes (X_I), conditions (C_I), rules (R_I) and actions (A_I) was represented as follows:

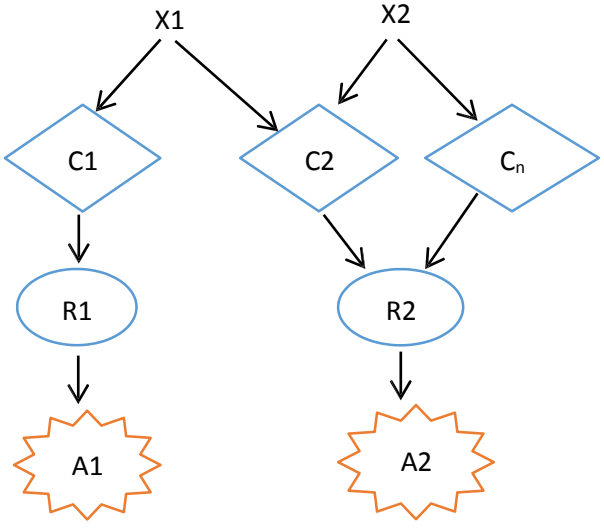


Figure 4.22: Relationship between attributes, conditions, rules and actions in the inference engine.

Sample code for diagnosis of pneumonia

```
//Cough and Difficult Breathing
if (checkBoxCough.Checked == true)
   {
    if (checkBoxDangerConvNow.Checked == true || checkBoxDangerConvulsions.Checked == true ||
        checkBoxDangerDrink.Checked == true || checkBoxDangerVomit.Checked == true ||
        checkBoxDangerUnconcious.Checked == true || checkBoxCoughStridor.Checked == true ||
        checkBoxCoughChest.Checked == true)
        {
         // Severe Pneumonia
        MessageBox.Show("Observed Sign and Symtptoms:\n" +
        "----------------------------------------------------------------------------\n" +
        " Cough or Difficult breathing and Unable to drink or breastfeed, Vomiting everything, Convulsion history, Convulsion now,
         Lethargic/Unconscious, Chest indrawing OR Stridor in calm child.\n" +
        "----------------------------------------------------------------------------\n" +
        "Classified As:\n" +
        "----------------------------------------------------------------------------\n" +
        "SEVERE PNEUMONIA OR VERY SEVERE DISEASE \n" +
        "----------------------------------------------------------------------------\n" +
         "Treatment: \n" +
        "----------------------------------------------------------------------------\n" +
        " * Give first dose of IV/IM Ampicillin or Chloramphenicol. \n" +
        " * Refer URGENTLY to hospital. \n" +
        "----------------------------------------------------------------------------\n");
  }
else if (checkBoxCoughFastBreathing.Checked == true)
        {
         //Pneumonia
         MessageBox.Show("Observed Sign and Symtptoms:\n" +
         "----------------------------------------------------------------------------\n" +
         " Cough and Difficult Breathing AND Fast Breathing \n" +
         "----------------------------------------------------------------------------\n" +
         "Classified As:\n" +
         "----------------------------------------------------------------------------\n" +
         "PNEUMONIA \n" +
         "----------------------------------------------------------------------------\n" +
         "Treatment: \n" +
         "----------------------------------------------------------------------------\n" +
         " * Give Cotrimoxazole for 5 days. \n" +
         " * Soothe the throat and relieve the cough with a safe remedy. \n" +
```

```
        " * Advise mother when to return immediately. \n" +
        " * Follow-up in 2 days. \n" +
          "-------------------------------------------------------------------------------\n");
      }
    else
        {
        MessageBox.Show("Observed Sign and Symtptoms:\n" +
        "-------------------------------------------------------------------------------\n" +
        " No signs of: Very Severe Disease AND Pneumonia \n" +
        "-------------------------------------------------------------------------------\n" +
        "Classified As:\n" +
        "-------------------------------------------------------------------------------\n" +
        "NO PNEUMONIA: COUGH OR COLD \n" +
        "-------------------------------------------------------------------------------\n" +
        "Treatment: \n" +
        "-------------------------------------------------------------------------------\n" +
        " * If coughing for ≥ 14 days, refer for assessment. \n" +
        " * Soothe the throat and relieve the cough with a safe remedy. \n" +
        " * Advise mother when to return immediately. \n" +
        " * Follow-up in 5 days if not improving. \n" +
        "-------------------------------------------------------------------------------\n");
        }
    }
```

Figure 4.23: Working memory and inference engine for cough and difficult breathing problem of the system's CDS.

CDS's User Interface

The user interface is a channel for communication between the CDS and the end-user. Therefore, in order for the CDS to be an interactive tool, it includes a means to show and retrieve information in a simple manner. Examples of information to be shown are the consequences made by the inference engine, the justifications for such consequences, and an explanation for the actions made by the CDS. The reason for the significance of the user interface component is that end-users usually evaluate CDSs based on the quality of the user interface instead of the CDS itself. There are two user interfaces which are used for CDS module of the system. The first interface is used to insert the observed sign and symptoms (figure 4.20) and the other one is used to display the result of the inference engine (figure 4.24). The first interface includes different symptoms which are used as an input for clinical decision support and the second interface is to display the diagnosis and treatment result in detail.

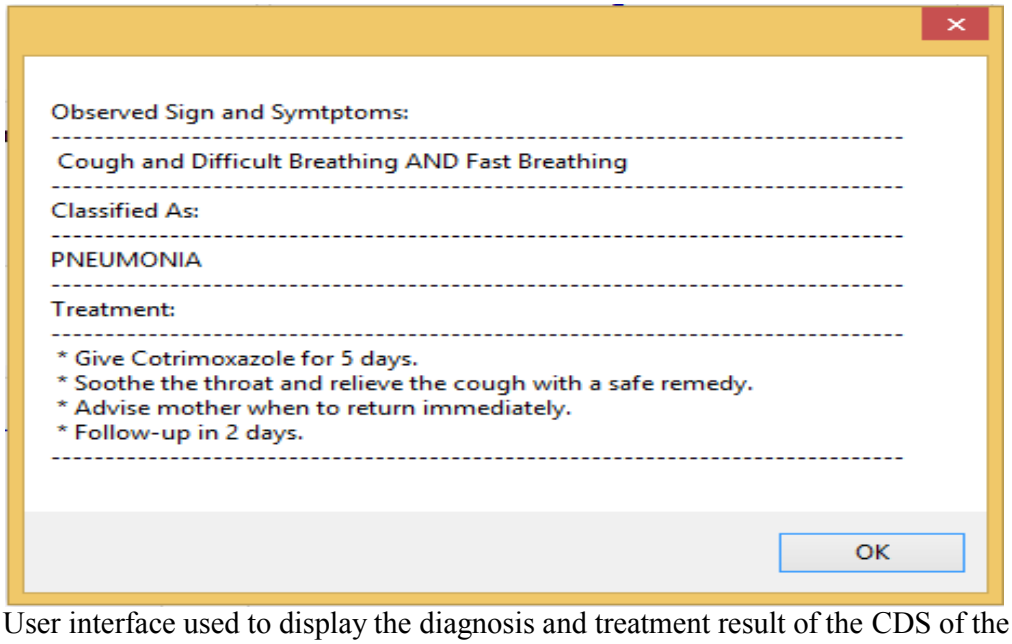


Figure 4.24: User interface used to display the diagnosis and treatment result of the CDS of the system.

4.5 Testing and Evaluation

After the system is developed in C#.net and MsSQL under the need of the user's requirement, the next step was testing and evaluating the system whether the system satisfies the users need and performance of the system. The scope of testing and evaluation that is accomplished and the significance involved to it rely on the size, complexity, and other features of the system. As the aim of testing and evaluation of the system is to assure that the system does what it is required to do. We can test and evaluate a system even we know and understand what to expect. In addition to this evaluation was done without training; however, the participants were oriented about the system's flow and what are the system features. Therefore, in this project testing and evaluation of the system has two aspects. The first one is system performance testing, and the other one is user acceptance testing.

In this study, 6 nurses were selected from Janmeda, Yeka woreda-4 and Shiromeda health centers to evaluate the usability and performance of e-IMNCI system. Typically 5-10 participants are involved in most of system usability testing due to resource limitations. That's why only 6 participants were selected for the testing in due consideration of resource limitation and to get more problems by giving them enough time to consider and check the system. The evaluators of the system were purposively only 2 nurses were selected from each health centers. Among these three selected health centers, 6 health professionals who work on less than 5 years children health were selected purposively. The participants profile includes age, computer skill, typing speed and education qualification. The typing speed and computer skill included because most of the time these two attributes can affect the performance and ease of use of the system. Participant information collected in the before they doing the given tasks (Appendix B). The participants profile summarized in table 4.12.

	Age	Computer Skill	Typing Speed	Education Status
P1	28	Poor	Beginner	BSc
P2	26	Good	Intermediate	BSc
P3	29	Very Good	Advanced	BSc
P4	24	Good	Beginner	BSc
P5	26	Very Good	Advanced	BSc
P6	26	Very Good	Beginner	BSc

Table 4.12: Summary of participant's profile

4.3.1 System Performance Test

System performance testing is the process of determining whether the prototype system is correct, that is whether it meets the level of accuracy as required. It confirms whether the right prototype system has been built. In system performance testing section, a number of patient's histories are selected by the researcher and questionnaire (system performance survey) was conducted in order to test the accuracy of the system. The correct and incorrect outcomes are identified by comparing decisions made by domain experts on the patient's history and with the conclusions of the system.

The performance testing of the system used a registered diagnosis result to test the accuracy of the diagnosis suggestion of the system's CDS module. However, the performance of other feature of the system was tested by allowing the users to do some tasks on the system and they were asked to fill a user satisfaction survey about their thought about the performance of patient record module, guideline, diagnosis assistant and reference of the system. the selected tasks in testing session were Login, Register a new patient's information, Register the observed sign and symptoms, Register the diagnosis result, Search the registered patient's detail information, Open fever care guideline, Open IMNCI knowledge test exercise, Check the diagnosis and treatment assistant, Check the report for number of patients who diagnosed as having Malaria, Check pneumonia's detail by using reference of the system and Logout (the detail of tasks name, description, requirement and instruction is described in appendix B). The participants were asked to do the above tasks. The test result is analyzed and described as follows.

A. Average time on tasks

The average time on task is the average time taken to perform the tasks by participants. . The average time taken to perform the tasks by participants is demonstrated in the figure 4.25.

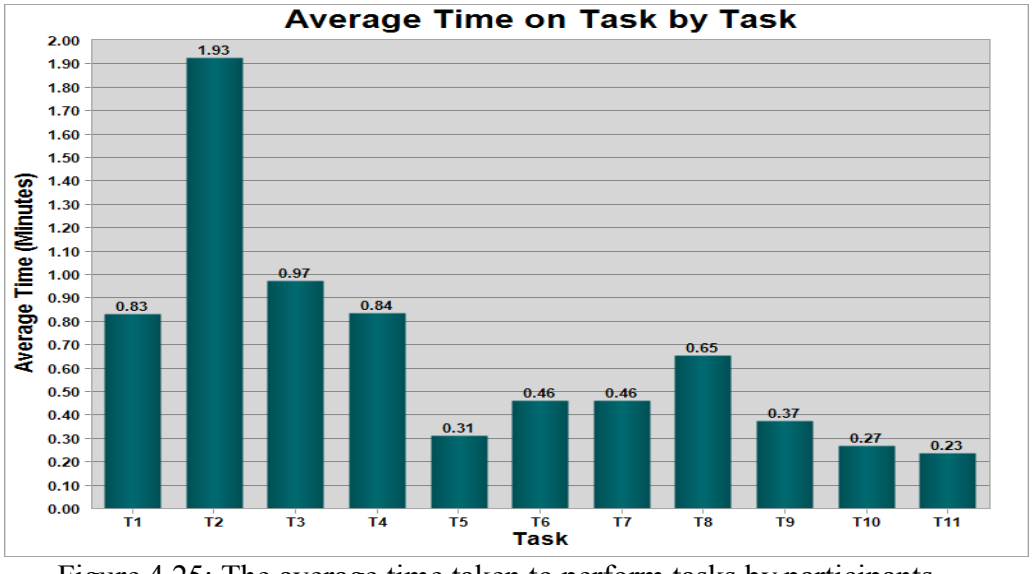


Figure 4.25: The average time taken to perform tasks by participants.

These 11 tasks are described illustrated in appendix B that includes task name, description, instructions, and expected time to complete. The average time needed to complete T2 that is registering new patient information is more than other tasks. The reason for T2 taking more time than other tasks is, it has more fields to fill rather than others.

However, the average time taken for whole tasks is more than the expected time of tasks because of typing speed and computer skill of the user.

B. Average time on task by participants

The participants were asked to perform the tasks which are described in appendix B. During the test session the time to complete the given tasks by participant were recorded. The average time to complete all the given tasks by each participant is illustrated in the figure 4.26.

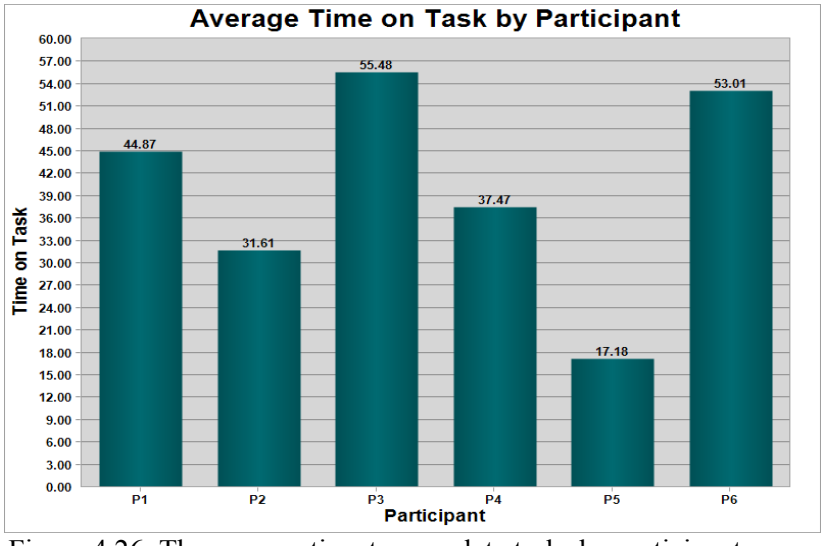


Figure 4.26: The average time to complete tasks by participants.

C. Task Success distribution by participants

Figure 4.27 illustrates success distribution of tasks by participants. The determining criteria's of task success distribution are complete with ease, complete with difficulty and filled to complete the task. The result of task success distribution was obtained from the recorded data about the condition of the user performance to complete the task that has been observed by the testing session observer and the average time taken to perform the tasks.

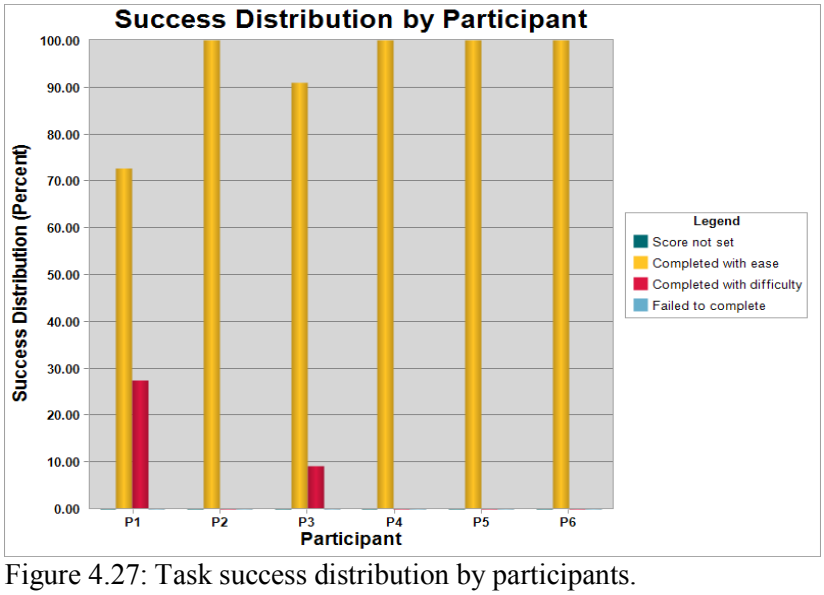


Figure 4.27: Task success distribution by participants.

P2, P4, P5 and P6 were completed the tasks 100% without difficulty. However, P1 and P3 have got a trouble to complete the given tasks. Participant1 has got a difficulty to complete registering new patient's information, registering the observed sign & symptoms and registering the diagnosis result. And Participant3 also faced a difficulty to finish registration of patient information.

D. Average time vs. Expected time on tasks

The summary of task performance showed that the task of working on eIMNCI system took the average time to complete the task versus the expected time. The expected time is the average time of the observers to complete the task (task list and description is shown in appendix B). Figure 4.28 shows, the average time taken and expected time to complete the 11 tasks (Login, Register a new patient's information, Register the observed sign and symptoms, Register the diagnosis result, Search the registered patient's detail information, Open newborn care guideline, Open IMNCI knowledge test exercise, Check the diagnosis and treatment assistant Check the report for number of patients who diagnosed as having Malaria, Check pneumonia's detail by using reference of the system and Logout).

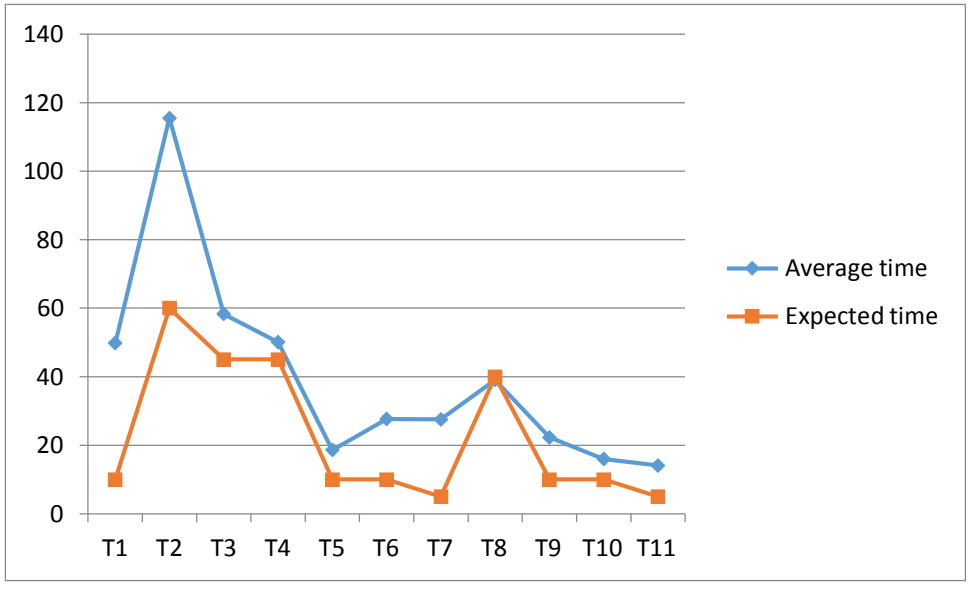


Figure 4.28: Average time vs. expected time taken to perform each task.

The difference between the average time and expected time to complete the task has a significant difference on some tasks that is cause by different factors like typing speed and computer skill of participants.

4.3.2.1 Experiment Result of the system's CDS

A set of 5 diseases along with its symptoms were added to the knowledge of the system. An experiment was carried out by taking 25 patient data from three health centers. For the same 25 patients, nurse's diagnosis reports have been collected. Among the selected 25 patient history, some of the records show the recorded information is beyond the boundary of the IMNCI service. The observed sign and symptoms of each patient were entered to the system then the system displays the corresponding diagnosis and treatment plan suggestions. However, the correct diagnosis results only included in this comparison; the criteria to choose the correct diagnosis result is based on the IMNCI guideline. The system's suggestion was compared to the decision which was decided by nurses. The results are compared and shown in the figure 4.29 below.

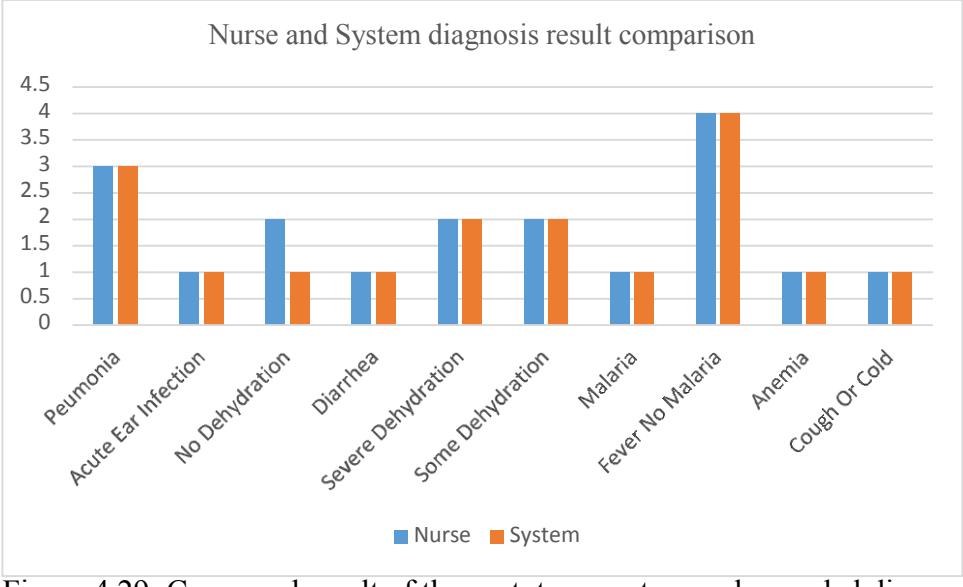


Figure 4.29: Compared result of the prototype system and recorded diagnosis data.

The accuracy of the system's clinical decision supports module scores of 94.44% accuracy. Hence, the accuracy of the system will be improved by adding the additional rules to the knowledge base.

4.3.2 System Usability Test

The goal of the system usability test was to determine the usability of e-IMNCI system. Usability often refers as the question of how well users can use system functionality. Usability is not one-dimensional property of user interface. It's associated with five attributes: learnability, efficiency, memorability, errors and satisfaction.

Under system usability testing section, the process of ensuring that whether the system satisfies the requirements of its end-users is performed. This permits end-users to test the system by actually using it and evaluating the benefits received from its use. Morae TechSmith software was used to conduct the usability test of the system by using its user satisfaction survey question.

Cognitive walk through evaluation method was applied to test the usability of the system. Through this method the evaluator first determines the exact sequence of correct task performance, and then estimates, on a screen by screen basis, the likely success or failure of the user in performing such a sequence. Concentrates more on the difficulties users may experience in learning to operate an application to perform a given task. The importance of this evaluation form is to evaluate to what extent the system is usable with regard to the end-users in the domain area.

The result of usability testing is demonstrated as follows. The values for all attributes in table 4.13 are fixed as: Strongly agree = 5, Agree = 4, undecided = 3, Disagree = 2 and Disagree = 1.

Usability Testing							
No	**Criteria of evaluation**	**Strongly Disagree**	**Disagree**	**Undecided**	**Agree**	**Strongly Agree**	**Average**
1	I think that I would like to use this system frequently.				16.67%	83.5%	96.6%
2	I found the system not complex.	33.34%			33.34%	33.34%	66.6%
3	I thought that the system was easy to use					100%	100%
4	I think that I wouldn't need the support of a technical person to be able to use this system.	33.34%		16.67%	50.01%		62.4%
5	I found the various functions in this system were well integrated				33.34%	66.68%	93.4%
6	I thought the system doesn't have inconsistency.			16.67%	16.67%	66.68%	90%
7	I would imagine that most people would learn to use this system very quickly.				33.34%	66.68%	93.4%
8	I found the system not cumbersome to use		16.67%	16.67%	66.68%		70%
9	I felt very confident using the system			16.67%	16.67%	66.68%	90%
10	I needed to learn a lot of things before I could get going with this system.		33.34%	16.67%	16.67%	33.34%	70%
		Average Result					**83.4%**

Table 4.13: User satisfaction survey result evaluated by participants.

The evaluators scored 83.5% strongly agree and 16.67% agree to use the system frequently. The 2nd evaluation criteria the system is not complex scored 33.34% as strongly agree, 33.34% as agree and 33.34% as strongly disagree. The evaluators were 100% strongly agree on the criteria of system's ease to use. For the criteria of technical person not need to be able to use the system scored as 50.01% agree, 16.67% as undecided and 33.34% a strongly disagree. For the 5th criteria that is whether the various functions in this system were well integrated or not scored as 66.68% as strongly agree and 33.34% agree. The evaluators were scored 66.68% as strongly agree, 16.67% as agree and 16.67% as undecided for the criterion of the system doesn't have inconsistency. For the criteria their imagination about most

people would learn to use this system very quickly scored as 66.68% as strongly agree and 33.34% agree. The evaluators were scored 66.68% as agree, 16.67% as undecided and 16.67% as disagree about the system is not awkward to use. The evaluators felt confident to use the system as 66.68% as strongly agree, 16.67% agree, 16.67% as undecided. For the last criteria the evaluators scored 33.34% as strongly agree, 16.67% as agree, 16.67% as undecided and 33.34% as disagree.

Finally, the average usability of the prototype system according to the evaluation results filled by the participants (domain experts) is 4.17 out of 5 or 83.4%.

CHAPTER FIVE

CONCLUSIONS AND RECOMMENDATIONS

The earlier chapters have brought to light some significance issues in the development of the electronic version of IMNCI system for the improvement of clinical decision support in health centers. In this part, conclusion of the study works and gives recommendation for future investigation in the health care services.

5.1 Conclusions

IMNCI is a strategy that integrates all available measures for health promotion, prevention and integrated management of childhood diseases through their early detection and effective treatment, and promotion of healthy habits within the family and community. The importance of having an Integrated Management of Newborn and Childhood Illness strategy is that it enables a consistent and standardized approach that addresses the major causes of under-five morbidity and mortality which are responsible for more than 90% of mortality in this age group in Ethiopia. The clinical guidelines include methods for assessing signs that indicate severe disease; assessing a child's nutrition, immunization and feeding; teaching parents how to care for a child at home; counseling parents to solve feeding problems; and advising parents about when to return to a health facility.

This work represents my first steps towards the goal of creating an electronic version of IMNCI that improves the clinical decision making process that will be done by the health professionals in health centers. In developing the prototype system, information is acquired by using interviews with domain experts, observation of the working process and from relevant documents by using documents analysis method to find the solution of the problem.

Unified modeling language method was used to analyze the collected requirements and to create the design of the system. Based on design of the system, development of the system was proceeding by using Ms Visual studio 2010 with C# programming language and MsSQL database management system. Testing and evaluation of the prototype system covers the performance and usability of e-IMNCI. The usability test of the system was used to describe to what extent the system is usable, then the users were asked to evaluate the usability of the system by using user satisfaction survey. The result of the test shows 83.4% willingness to use the system easily. In addition to the usability test, performance test measures were considered during the usability testing. The performance test measures time spent to perform a task, count of all incorrect selections and accuracy of system's diagnosis assistant feature.

From three health centers 25 patient recorded data were taken to check the accuracy of the diagnosis assistant of the system. The result shows 94.4% accuracy of diagnosis and treatment plan suggestion compared to the diagnosis and treatment result done by nurses.

Generally, the prototype system serve as a guideline, patient information management system and diagnosis assistant. The initial feedback from health works has been extremely positive. Hence the prototype system achieves a good performance and meets the objectives of the project.

5.2 Recommendations

From the presentations of the findings of the study, the following recommendations for future study are forwarded:

- To make the diagnosis assistant's decision more accurate than this system, the developer or researchers should improve the rules of the knowledge base by adding other symptoms of a disease which are not included in the IMNCI guideline.
- However, almost all health centers' IMNCI department didn't have a computer in the office; and also currently in Ethiopia, infant and child health care is given by health extension workers through house to house service. Hence, the system should be run on different platforms and devices like PDA, tablet, or mobile application to make accessible in remote areas and personal uses.
- Developing active mode of information transmission so as to make the diagnosis assistant more interesting.
- The system should be dynamically updates rules of the knowledge base.
- Developing the Amharic version of the system that makes the system preferable and accessible by rural health extension workers.

References

1. Quality Assurance Project. Evaluation of an IMCI Computer-based Training Course in Kenya. Operations Research Results. Published for the U.S. Agency for International Development (USAID) by the Quality Assurance Project (QAP), Bethesda, MD. 2006.
2. World Health Organization. Integrated Management Of Childhood Illness, 1st ed., Department of Child and Adolescent Health and Development (CAH), Geneva, pp. 3–6, and pp. 12-20. 2005.
3. Integrated Management of Newborn and Childhood Illness: Blended Learning Module for the Health Extension Program part1& part 2, 1st ed., Ethiopian Federal Ministry of Health, Addis Ababa, 2012
4. Serena Software, Inc., an Introduction to Agile Software Development June 2007, California, United States. 2007.
5. Supporting life Project 2013. http://www.supportinglife.eu/imci-e-imci/about-e-imci
6. Brian DeRenzi et al, "e-IMCI: Improving Pediatric Health Care in Low-Income Countries." University of Washington, Ifakara Health Research & Development Centre, Tanzania, 2008.
7. Berner ES. Clinical decision support systems: State of the Art. AHRQ Publication No. 09-0069-EF. Rockville, Maryland: Agency for Healthcare Research and Quality. June 2009.
8. Mitchell et al.: Using electronic technology to improve clinical care – results from a before-after cluster trial to evaluate assessment and classification of sick children according to Integrated Management of Childhood Illness (IMCI) protocol in Tanzania. BMC Medical Informatics and Decision Making 2013 13:95.
9. ICATT Integrated management of childhood illness Computerized Adaption Training Tool, WHO and Novartis Foundation for Sustainable Development, 2007: http://www.icatt-training.org
10. Isabel - OpenClinical AI Systems in clinical practice, http://www.openclinical.orga/isp_isabel.htm
11. Quick Medical Reference (QMR) - OpenClinical AI Systems in clinical practice, http://www.openclinical.org/aisp_qmr.htm
12. TheraSim - OpenClinical AI Systems in clinical practice, http://www.openclinical.org/aisp_therasimCS-HIV.htm
13. Iliad - OpenClinical AI Systems in clinical practice, http://www.openclinical.org/aisp_iliad.htm
14. HELP - OpenClinical AI Systems in clinical practice, http://www.openclinical.org/aisp_help.htm
15. CHICA - OpenClinical projects, http://www.openclinical.org/prj_chica.htm
16. DXplain - OpenClinical AI Systems in clinical practice, http://www.openclinical.org/aisp_dxplain.htm
17. Dejan Dinevski et al (2011). Clinical Decision Support Systems, Telemedicine Techniques and Applications, Prof. Georgi Graschew (Ed.), ISBN: 978-953-307-354-5.

18. Chiang S. Jao and Daniel B. Hier (2010). Clinical Decision Support Systems: An Effective Pathway to Reduce Medical Errors and Improve Patient Safety, Decision Support Systems, Chiang S. Jao (Ed.), ISBN: 978-953- 7619-64-0.
19. Wilfred Bonney. Impacts and Risks of Adopting Clinical Decision Support Systems, Efficient Decision Support Systems - Practice and Challenges in Biomedical Related Domain, Prof. Chiang Jao (Ed.), ISBN: 978- 953-307-258-6, 2011.
20. Anne-Marie J.W et al. Success Factors and Barriers for Implementation of Advanced Clinical Decision Support Systems, Efficient Decision Support Systems - Practice and Challenges in Biomedical Related Domain, Prof. Chiang Jao (Ed.), ISBN: 978-953-307-258-6, (2011).
21. Rainu Kaushal et al. Effects of Computerized Physician Order Entry and Clinical Decision Support Systems on Medication Safety: A Systematic Review. Arch Intern Med. 2003;163:1409-1416 2003.
22. Simon, B., McRobb, S., Farmer, R.,Object Oriented Systems Analysis and Design using UML, McGraw-Hill Web References, 1999.
23. Standard treatment guideline for health centers, 2nd ed., Drug authority and control agency (DACA), Addis Ababa, 2010.
24. Mitari Patel et al, "Rule based expert system for viral infectious disease", International Journal of Advanced Research in Computer Science and Software Engineering, 2013, vol 3, pp. 591-595.
25. Grady Booch, James Rumbaugh , Ivar Jacobson. "The Unified Modeling Language User Guide", 1st ed, ISBN: 0-20201-87168-4., 1998.
26. S. Sumathi, S. Esakkarajan. "Fundamentals of Relational Database Management Systems", German, ISSN electronic edition:1860-9503, 2007.

APPENDIX

Appendix A: Requirement Collection Checklist.

Interview Checklists:

1. Is any available software/ system you are using in your office?

_____
_____
_____
_____

2. What is your need which will be included in the system?

_____
_____
_____
_____

3. What organizational standards do you expect from the system?

_____
_____
_____
_____

4. What are the domain information in your department?

_____
_____
_____
_____

5. How do you use the paper based system?

_____
_____
_____
_____

6. Is the guideline clear and simple to use? If there is a problem with it please mention some of it.

_____
_____
_____
_____

Observation Checklists:

1. Do you have a computer in your office?
   ☐ Yes ☐ No

   If yes, for what purpose you used it now?

_____
_____
_____
_____

2. How do you register patient's information?

_____
_____
_____
_____

3. How could you analyze the patient's information?

_____
_____
_____
_____

Document Review and Analysis:

1. What kind of forms you used for registering patient information, diagnosis result and report?

_____
_____
_____
_____

2. How do you deal with forms?

_____
_____
_____
_____

3. How could you generate a report from the registered patient diagnosis history?

_____
_____
_____
_____

Appendix B: Usability testing tasks and survey questions

Survey 1: Participants Profile

1. Age?

2. Computer Skill?

   Poor Good Very Good

3. Typing Speed?

   Beginner Intermidate Advanced

4. Education Qualification?

   Diploma BSc MSc

Close

Survey 2: User Satisfaction

1. I think that I would like to use this system frequently.

	1	2	3	4	5	
Strongly Disagree	○	○	○	○	○	Strongly Agree

2. I found the system unnecessarily complex.

	1	2	3	4	5	
Strongly Disagree	○	○	○	○	○	Strongly Agree

3. I thought that the system was easy to use

	1	2	3	4	5	
Strongly Disagree	○	○	○	○	○	Strongly Agree

4. I think that I would need the support of a technical person to be able to use this system.

	1	2	3	4	5	
Strongly Disagree	○	○	○	○	○	Strongly Agree

5. I found the various functions in this system were well integrated

	1	2	3	4	5	
Strongly Disagree	○	○	○	○	○	Strongly Agree

6. I thought there was too much inconsistency in this system

	1	2	3	4	5	
Strongly Disagree	○	○	○	○	○	Strongly Agree

7. I would imagine that most people would learn to use this system very quickly.

	1	2	3	4	5	
Strongly Disagree	○	○	○	○	○	Strongly Agree

8. I found the system very cumbersome to use

	1	2	3	4	5	
Strongly Disagree	○	○	○	○	○	Strongly Agree

9. I felt very confident using the system

	1	2	3	4	5	
Strongly Disagree	○	○	○	○	○	Strongly Agree

10. I needed to learn a lot of things before I could get going with this system.

	1	2	3	4	5	
Strongly Disagree	○	○	○	○	○	Strongly Agree

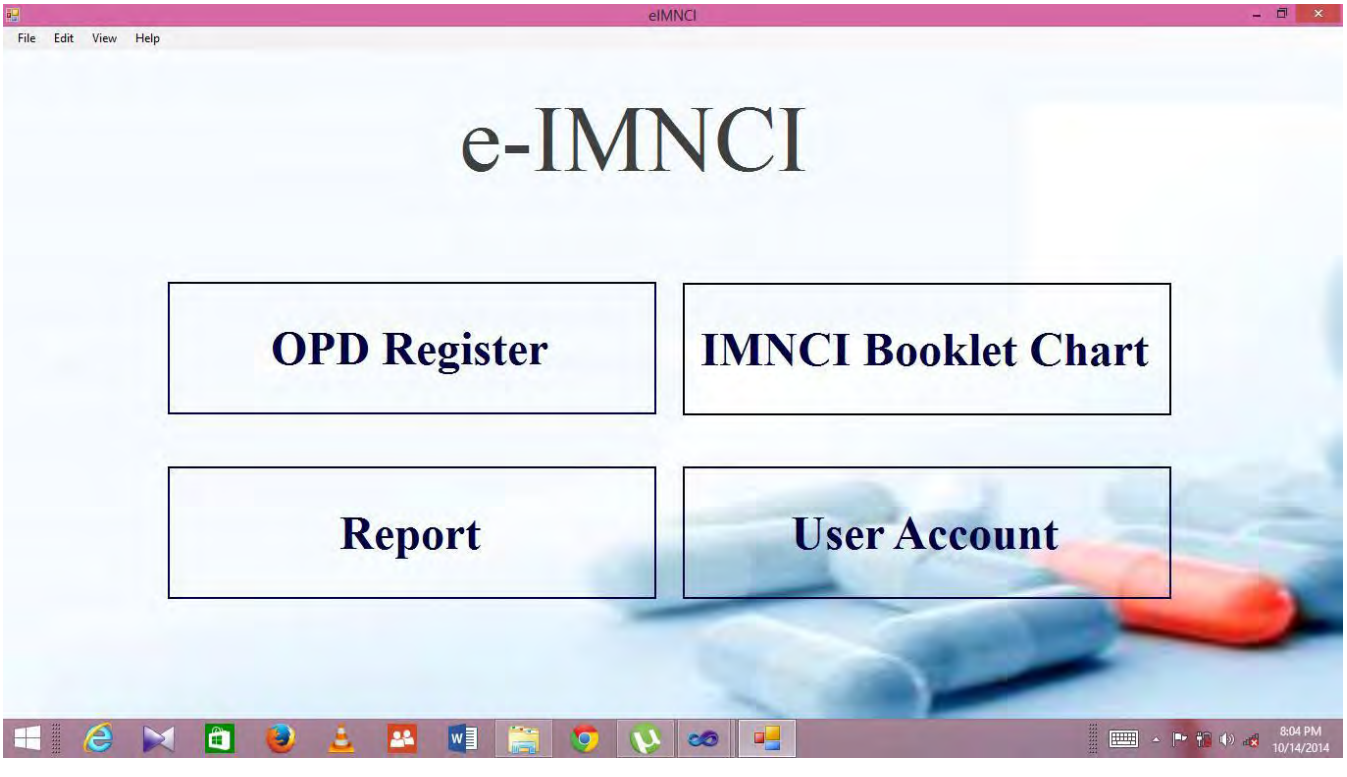
Close

Task No	Task Name	Task Description	Task Requirement	Task Instruction
1	Login	Login to the system so as to use the features of the software. Task performance ➢ Always Measurement attributes: ➢ Wrong clicks ➢ Time to complete the task	This test requires: • Open the e-IMNCI system • It should be completed within 10 seconds.	Login to the system by using your username and password to use the software.
2	Register a new patient's information	Registering a new patient's information to the system. Task performance ➢ Always Measurement attributes: ➢ Wrong clicks ➢ Time to complete the task	This test requires: • Logged into the system • It should be completed within 1 min.	Register new patient who is come to the department for the first time.
3	Register the observed sign and symptoms	Register the observed sign and symptoms. Task performance ➢ Always Measurement attributes: ➢ Wrong clicks ➢ Time to complete the task	This test requires: • Logged into the system • The patient should be registered in the system. • It should be completed within 45 seconds.	Open sign and symptoms registration page and Register the observed sign and symptoms.
4	Register the diagnosis result	Register the diagnosis result after the observed sign and symptoms were registered. Task performance ➢ Always Measurement attributes: ➢ Wrong clicks ➢ Time to complete the task	This test requires: • Logged into the system • The patient should be registered in the system. • It should be completed within 45 seconds.	After you register patient information and observed sign and symptoms.
5	Search the registered patient's detail information	Search patient's detail information who is registered before. Task performance ➢ Sometimes Measurement attributes: ➢ Wrong clicks	This test requires: • Logged into the system • The patient should be registered in the system. • It should be completed within 10 seconds.	Search the registered patient's detail information

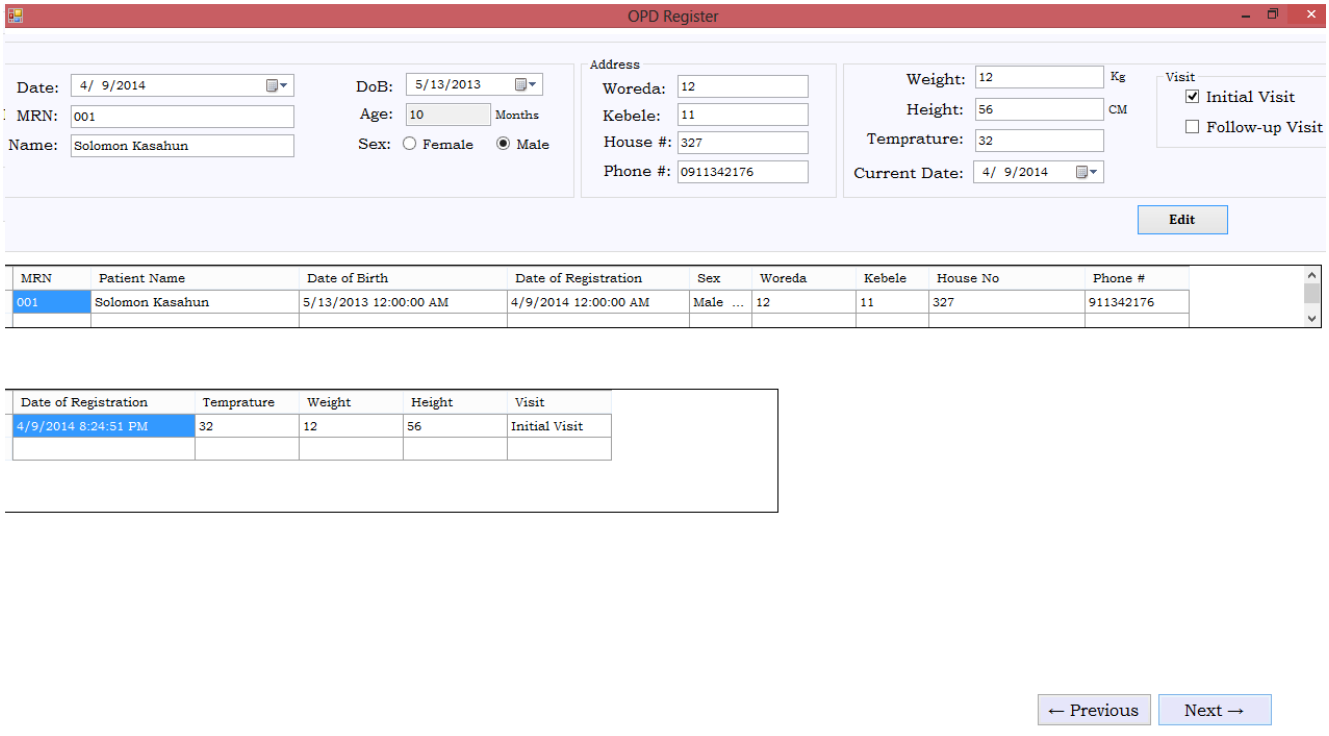
		➢ Time to complete the task		
6	Open fever care guideline.	Open fever guide from the guidelines. Task performance ➢ Seldom Measurement attributes: ➢ Wrong clicks ➢ Time to complete the task	This test requires: • Logged into the system • It should be completed within 10 seconds.	Open fever guide from the guidelines booklet chart.
7	Open IMNCI knowledge test exercise	Open an exercise to do some questions about IMNCI care. Task performance ➢ Sometimes Measurement attributes: ➢ Wrong clicks ➢ Time to complete the task	This test requires: • Logged into the system • It should be completed within 5 seconds.	Click on IMNCI booklet chart button and then select a question from given knowledge test.
8	Check the diagnosis and treatment assistant	Check the diagnosis and treatment assistant Task performance. ➢ Sometimes Measurement attributes: ➢ Wrong clicks ➢ Time to complete the task	This test requires: • Logged into the system • It should be completed within 40 seconds.	Insert some symptoms that you need to see assistance for diagnosis and treatment.
9	Check the report for number of patients who diagnosed as having Malaria	Check the report for number of patients who diagnosed as having Malaria Task performance.: ➢ Seldom Measurement attributes: ➢ Wrong clicks ➢ Time to complete the task	This test requires: • Logged into the system • It should be completed within 10 seconds.	Open report page and Check the report for number of patients who diagnosed as having Malaria
10	Check pneumonia's detail by using reference of the system	Check pneumonia's detail by using reference of the system Task performance.: Seldom Measurement attributes: Wrong clicks Time to complete the task	This test requires: • Logged into the system • It should be completed within 10 seconds.	Check pneumonia's detail by using reference of the system by clicking reference button from home page of the system.
11	Logout	Logout from the system to leave the page. Task performance: Always Measurement attributes: Wrong clicks ➢ Time to complete the task	This test requires: • Logged into the system • It should be completed within 5 seconds.	Logout from the system to leave the page.

Appendix C: User interfaces of the system

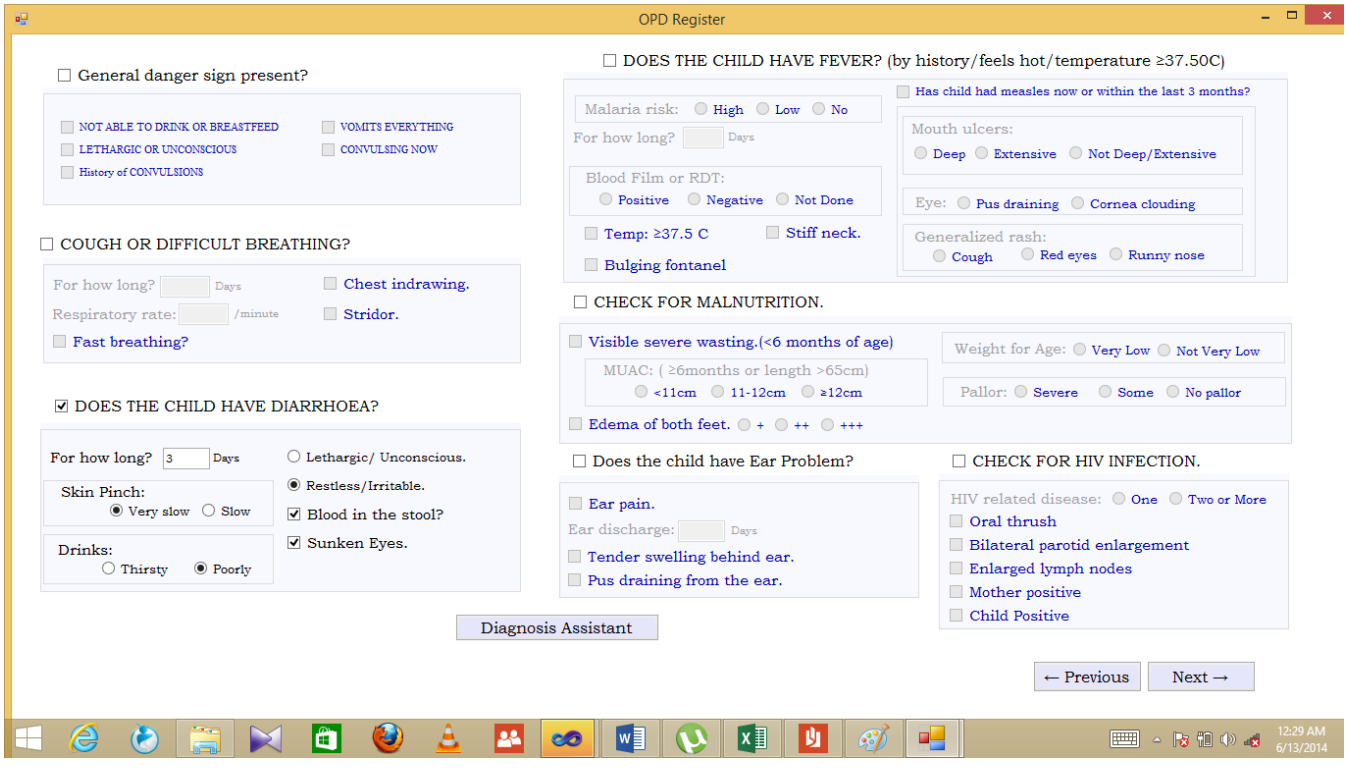
Login user interface.



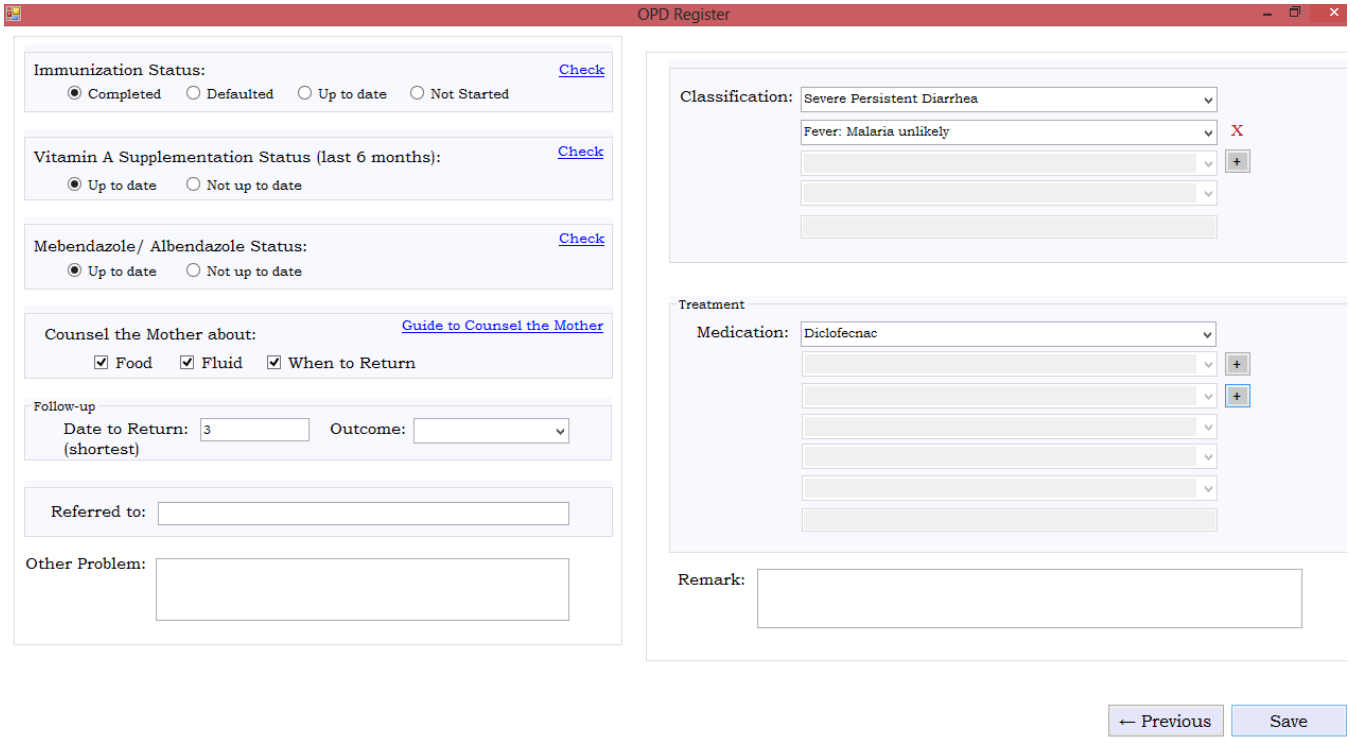
Front Page



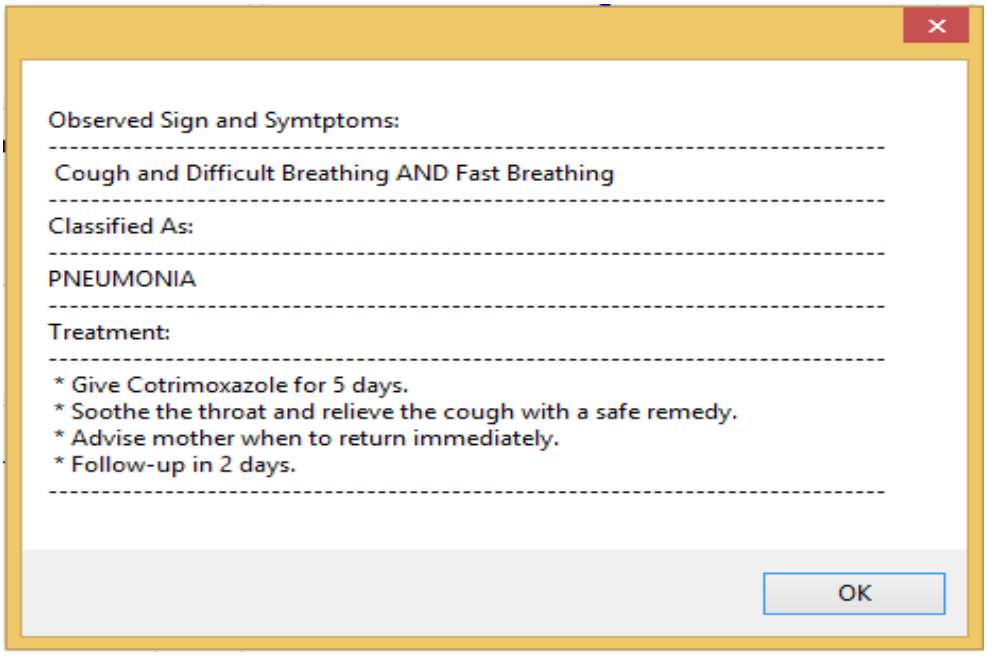
Patient Registration user interface.



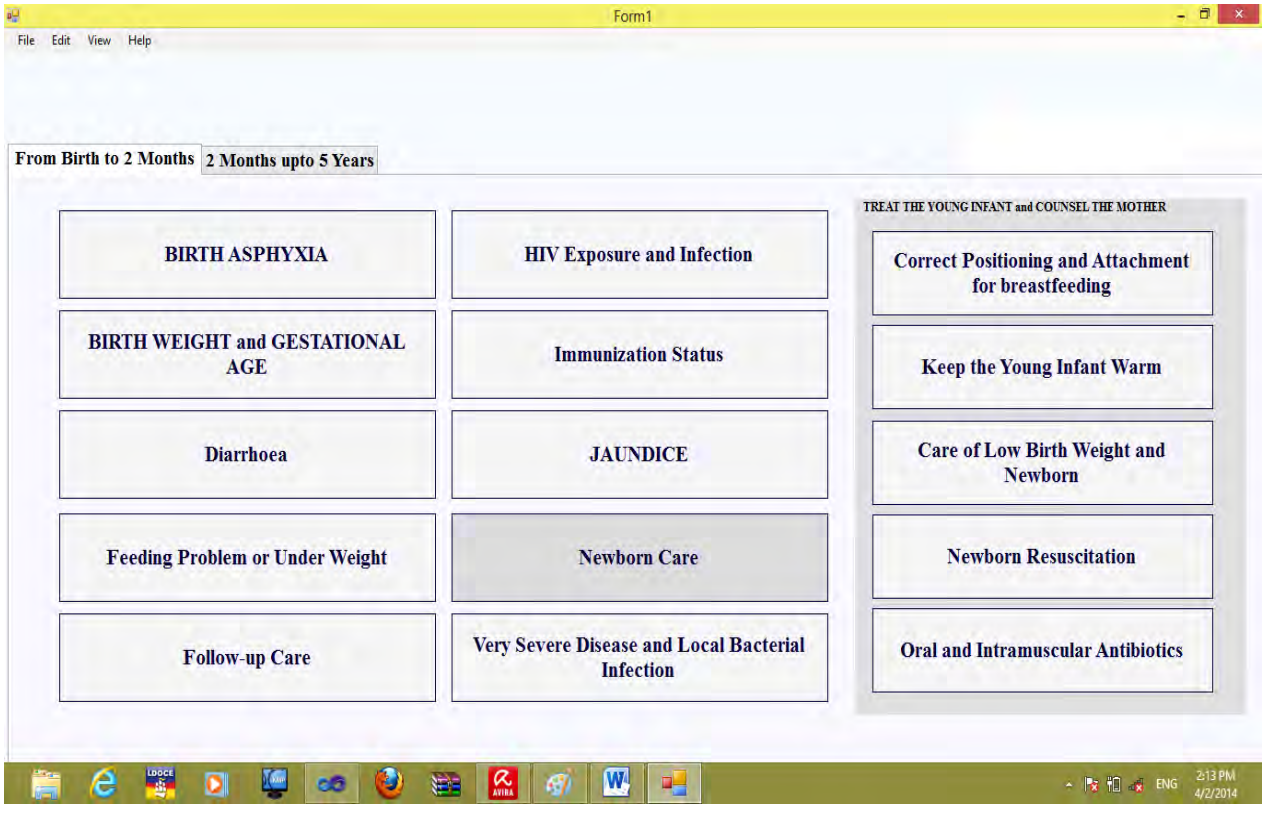
A user interface to register Observed sign and symptoms.



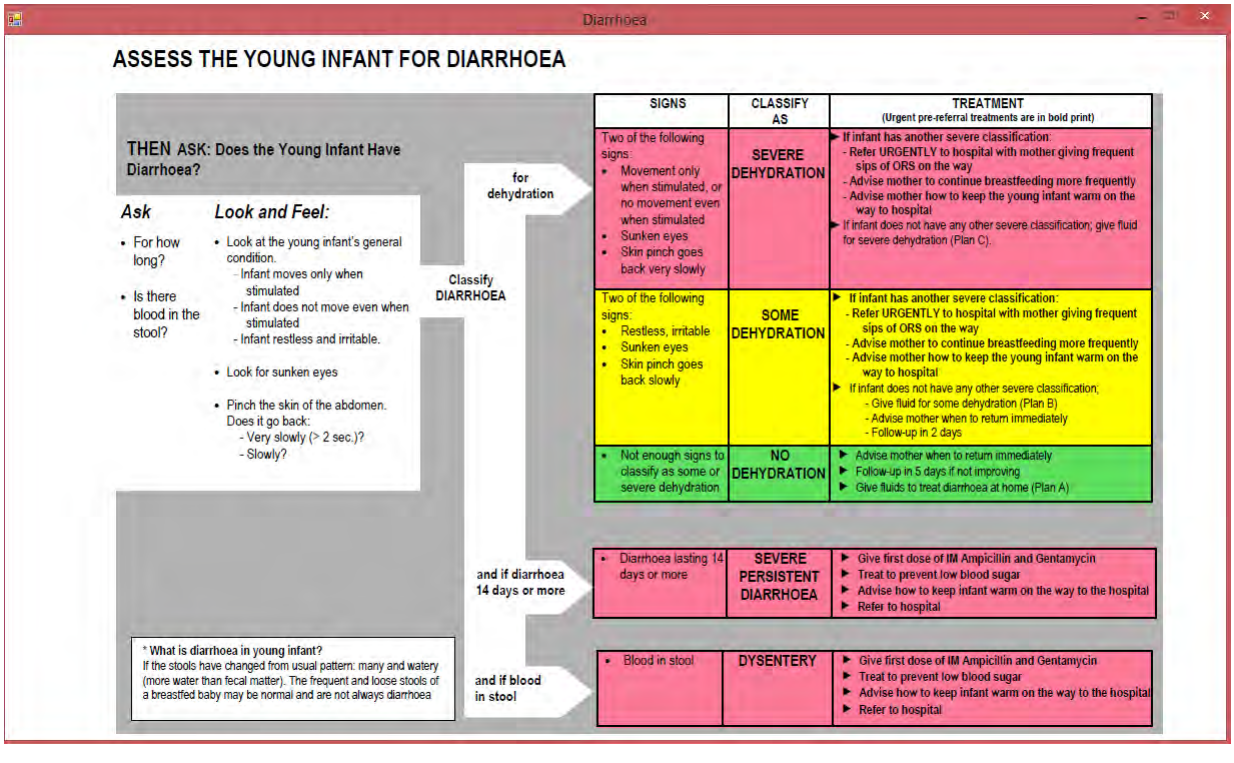
A user interface to register diagnosis result.



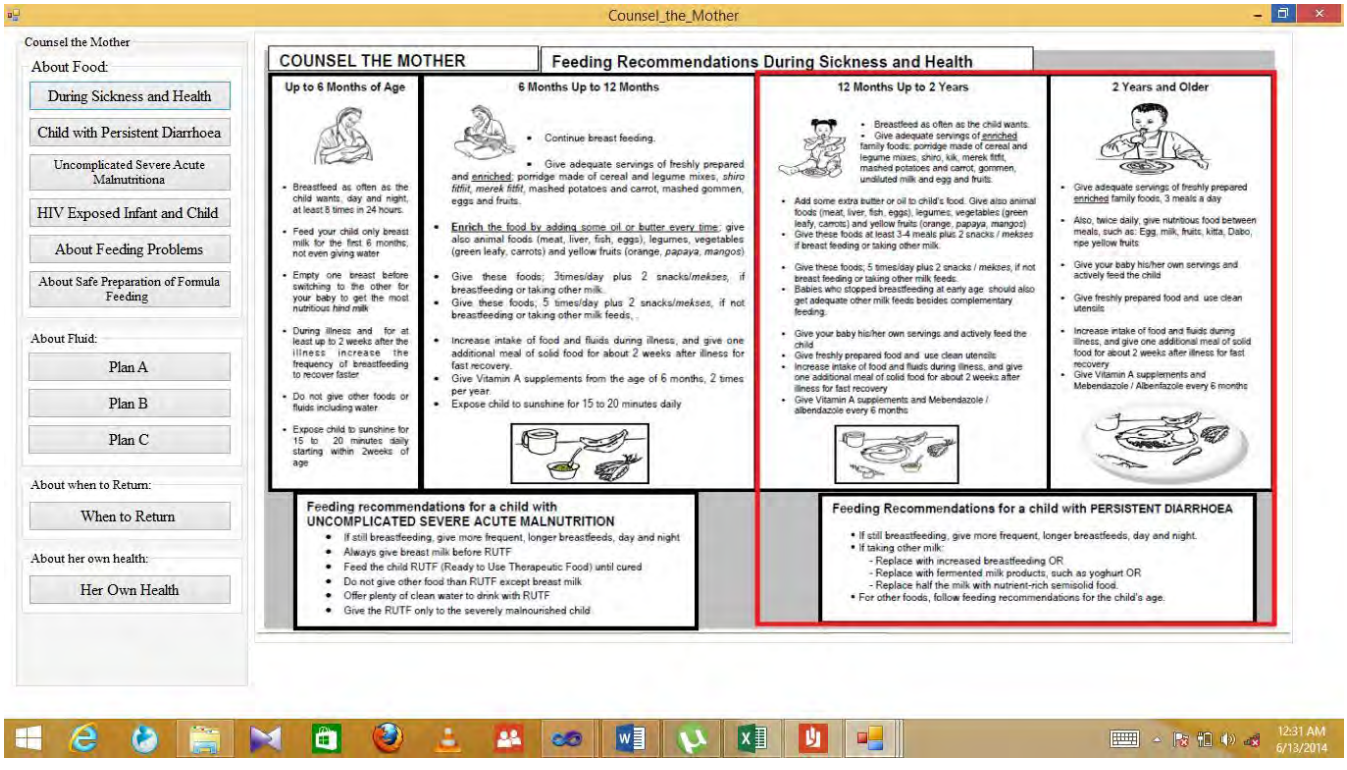
A user interface to display system's diagnosis and treatment plan suggestion.



Guideline user interface.



A user interface to display the result of fever guide.



A user interface to advice the m

DECLARATION

I declare that this project is my original work and has not been presented for degree in any other university, and that all sources of materials used for the project have been acknowledged.

Tuha Mustefa

This project has been submitted for examination with our approval as university advisors.

Dr. Martha Yifru	Dr. Ayele Belachew

Place and Date of submission: Addis Ababa, June 2014