CHAPTER ONE

INTRODUCTION

1.0 BACKGROUND OF STUDY

Medical diagnosis, (often simply termed diagnosis) refers both to the process of attempting to determine or identifying a possible disease or disorder to the opinion reached by this process. A diagnosis in the sense of diagnostic procedure can be regarded as an attempt at classifying an individual's health condition into separate and distinct categories that allow medical decisions about treatment and prognosis to be made. Subsequently, a diagnostic opinion is often described in terms of a disease or other conditions.

In the medical diagnostic system procedures, elucidation of the etiology of the disease or conditions of interest, that is, what caused the disease or condition and its origin is not entirely necessary. Such elucidation can be useful to optimize treatment, further specify the prognosis or prevent recurrence of the disease or condition in the future.

Clinical decision support systems (CDSS) are interactive computer programs designed to assist healthcare professionals such as physicians, physical therapists, optometrists, healthcare scientists, dentists, pediatrists, nurse practitioners or physical assistants with decision making skills. The clinician interacts with the software utilizing both the clinician's knowledge and the software to make a better analysis of the patient's data than neither humans nor software could make on their own.

Typically, the system makes suggestions for the clinician to look through and the he picks useful information and removes erroneous suggestions.

To diagnose a disease, a physician is usually based on the clinical history and physical examination of the patient, visual inspection of medical images, as well as the results of laboratory tests. In some cases, confirmation of the diagnosis is particularly difficult because it requires specialization and experience, or even the application of interventional methodologies (e.g., biopsy). Interpretation of medical images (e.g., Computed Tomography, Magnetic Resonance Imaging, Ultrasound, etc.) usually performed by radiologists, is often limited due to the non-systematic search patterns of humans, the presence of structure noise (camouflaging normal anatomical background) in the image, and the presentation of complex disease states requiring the integration of vast amounts of image data and clinical information. Computer-Aided Diagnosis (CAD), defined as a diagnosis made by a physician who uses the output from a computerized analysis of medical data as a “second opinion” in detecting lesions, assessing disease severity, and making diagnostic decisions, is expected to enhance the diagnostic capabilities of physicians and reduce the time required for accurate diagnosis. With CAD, the final diagnosis is made by the physician.

The first CAD systems were developed in the early 1950s and were based on production rules (Shortliffe, 1976) and decision frames (Engelmore & Morgan, 1988). More complex systems were later developed, including blackboard systems (Engelmore & Morgan, 1988) to extract a decision, Bayes models (Spiegelhalter, Myles, Jones, & Abrams, 1999) and artificial neural networks (ANNs) (Haykin, 1999). Recently, a number of CAD systems have been implemented to address a number of diagnostic problems. CAD systems are usually based on biosignals, including the electrocardiogram (ECG), electroencephalogram (EEG), and so on or medical images from a number of modalities, including radiography, computed tomography, magnetic resonance imaging, ultrasound imaging, and so on.

In therapy, the selection of the optimal therapeutic scheme for a specific patient is a complex procedure that requires sound judgement based on clinical expertise, and knowledge of patient values and preferences, in addition to evidence from research. Usually, the procedure for the selection of the therapeutic scheme is enhanced by the use of simple statistical tools applied to empirical data. In general, decision making about therapy is typically based on recent and older information about the patient and the disease, whereas information or prediction about the potential evolution of the specific patient disease or response to therapy is not available. Recent advances in hardware and software allow the development of modern Therapeutic Decision Support (TDS) systems, which make use of advanced simulation techniques and available patient data to optimize and individualize patient treatment, including diet, drug treatment, or radiotherapy treatment.

In addition to this, CDS systems may be used to generate warning messages in unsafe situations, provide information about abnormal values of laboratory tests, present complex research results, and predict morbidity and mortality based on epidemiological data.

1.2 STATEMENT OF THE PROBLEM

Disease diagnosis and treatment constitute the major work of physicians. Some of the time, diagnosis is wrongly done leading to error in drug prescription and further complications in the patient’s health. It has also been noticed that much time is spent in physical examination and interview of patients before treatment commences. The clinical decision support system (CDSS) shall address these problems by effectively providing quality diagnosis in real-time.

1.3 OBJECTIVES OF THE STUDY

- To develop modern interactive diagnostic software that will aid clinicians in diagnostic procedures.

- To offer prescription of medication.
- To enable flexibility in access to information through the World Wide Web or comprehensive knowledge bases.
- To offer information on effective disease prevention.
- To provide for real-time overall effective, efficient and accurate service delivery by clinicians in line with global medical health standards.

1.4 SIGNIFICANCE OF STUDY

Advances in the areas of computer science and artificial intelligence have allowed for development of computer systems that support clinical diagnostic or therapeutic decisions based on individualized patient data. Clinical decision support (CDS) systems aim to codify and strategically manage biomedical knowledge to handle challenges in clinical practice using mathematical modeling tools, medical data processing techniques and artificial intelligence (A.I.) methods.

Its significance is also seen in its ability to:

- Provide diagnostic support and model the possibility of occurrence of various diseases or the efficiency of alternative therapeutic schemes.
- Reduce the potential for harmful drug interactions, prescription errors and adverse drug reactions.
- Enable clinicians report adverse drug reactions to the relevant authorities.
- Promote better patient care by enhancing collaboration between physicians and pharmacists.

1.5 SCOPE OF THE STUDY

Due to the fact that it is difficult to develop an expert system for diagnosing all diseases at a time, financial and time constraints, this research is limited to medical diagnosis and treatment for malaria, typhoid fever and pneumonia.
The therapy covers severe and uncomplicated cases of the treatment of extreme or severe associated cases in patients such as cerebral malaria which causes insanity, blondness, asthma, tuberculosis and so on.
The study will also involve method(s) of diagnosis especially the patient history, physical examination and request for clinical laboratory test but will not go into how these tests are carried out.
Rather, it will only make use of the laboratory and treatment.

1.6 LIMITATIONS OF THE STUDY

In the course of this study, a major constraint experienced was that of time factor and insufficient finance. Others include the inevitability of human error and bias as some information were obtained via interpersonal interactions, interviews and research, making some inconsistent with existing realities or outrightly incorrect.

Great pains were however taken to ensure that these limitations are at their very minimum and less impactful on the outcome of the work.

1.7 DEFINITION OF RELATED TERMS

Here, the researcher shall try as much as possible to explain certain technical terms used during the course of his study.

Prognosis: This is a medical opinion as to the likely outcome of a disease

Etiology: This is the branch of medicine that investigates the causes and origin of diseases.

Diagnostic Criteria: This term designates the specific combination of signs, symptoms, and test results that the clinician uses to attempt to determine the correct diagnosis.

Therapy critiquing and consulting: This function of a clinician implies assessing of the therapy looking for inconsistencies, errors, cross-references for drug interactions and prevents prescribing of allergenic drugs.

Allergen: A substance that causes an allergy.

Epidemiology: The scientific and medical study of the causes and transmission of disease within a population.

CHAPTER TWO

REVIEW OF REATED LITERATURE

2.0 CLINICAL DIAGNOSTIC SUPPORT SYSTEMS

Advances in the areas of computer science and artificial intelligence have allowed for the development of computer systems that support clinical diagnostic or therapeutic decisions based on individualized patient data(Berner and Bell, 1998; Shortliffe, Pennault, Wiederhold, and Fagan, 1990). Medical diagnostic systems according to Wikipedia—the online encyclopedia are interactive computer programs designed to assist healthcare professionals with decision making tasks.

Bankman, 2000, elucidates further by asserting that Clinical Decision Support (CDS) systems aim to codify and strategically manage biomedical knowledge to handle challenges in clinical practice using mathematical modeling tools, medical data processing techniques and Artificial Intelligence (AI) methods. In other words, CDSS are active knowledge systems which use two or more items of patient data to generate case-specific advice (Wyatt and Spiegelhalter, 1991)

This kind of software uses relevant knowledge rules within a knowledge base and relevant patient and clinical data to improve clinical decision making on topics like preventive, acute and chronic care, diagnostics, specific test ordering, prescribing practices. Clinicians, health-care staff or patients can manually enter patient characters into the computer system; alternatively, electronic medical records can be queried for retrieval of patient characteristics. These kinds of decision-support systems allow the clinicians to spot and choose the most appropriate treatment.

However, Delaney, Fitzmaurice et al. 1991; Pearson, Moxey et al. 2009) warns that "regardless of how we choose to define CDS systems, we have to accept that the field of CDSS is rapidly advancing and unregulated. "it has a potential for harm if systems are poorly designed and inadequately evaluated, as well as a huge potential to benefit , especially in health care provider performance,, quality of care and patient outcomes."

CDS system is one of the areas addressed by the clinical information systems (CIS). Clinical information systems provide a clinical data repository that stores clinical data such as the patient's history of illness, diagnosis proferred, treatment as well as interactions with care providers.

There are some principal categories to take into account while striving for excellent decision making as outlined by Shortliffe and Cimono 2006.:

a. Accurate data
b. Applicable knowledge
c. Appropriate problem solving skills.

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 access when additional facts will confuse rather than clarify the patient's case. For example, a usual setting for such a problem is intensive-care units where practitioners must absorb large amounts of data from various monitors, be aware of the clinical status, patient history, accompanying chronic illness, 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 serious adverse effect on patient-care decisions.

Knowledge used in decision making process must be accurate and current. It is a 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 development in the area of the case in question. It is in the clinician's hands to bring proven therapists from research papers to the fore. CDSS analogously needs an extensive well structured and current source of knowledge to appropriately serve the clinician.ood problem solving skills are needed to utilize available data and knowledge.

Above all, good problem solving skills are needed to utilize available data and knowledge deciding clinicians must set appropriate goals for each task, know how to reason about each goal and taste in to account the trade-offs between costs and benefits of therapy and diagnostics. 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, (General Practice Electronic Decision Support 2000).

In order to be able to construct applicable CDS systems, it is imperative to have a broader-based understanding of medical decision making as it occurs in the natural setting. Designing CDSS without understanding the cognitive processes underlying medical reasoning and decision analysis is pliable for ineffectiveness and failure for implementation into clinical workflow (Patel, Kaufman et al. 2002).

2.1 SUCCESS FACTORS OF CDS SYSTEMS

Despite the fact that the computerized CDS systems were continuously in development since the 1970s, their impact on routine clinical practice has not been as strong as expected. The potential benefits of using electronic decision support systems in clinical practice fall into three broad categories (Coiera 2003):

1. Improved patient safety (reduced medication errors and unwanted adverse events, refined ordering of medication and tests);
2. Improved quality of care (increasing clinicians' time allocated directly to patient care, increased application of clinical pathways and guidelines, accelerate and encourage the use of latest clinical findings, improved clinical documentation and patient satisfaction);
3. Improved efficiency of health-care (reducing costs through faster order processing, reductions in test duplication, decreased adverse events, and changed patterns of drug prescribing, favoring cheaper but equally effective generic brands).

Developing CDSSs is a challenging process, which may lead to a failure despite our theoretical knowledge about the topic. Understanding the underlying causes, which lead either to success or either to failure, may help to improve the efficiency of CDSS development and deployment in day-to-day practice. Failures can originate from various developmental and implementation phases: failure to technically complete an appropriate system, failure to get the system accepted by the users and failure to integrate the system in the organizational or user environment (Brender, Ammenwerth et al. 2006).

There is an estimation that 45% of computerized medical information systems fail because of user resistance, even though these systems are technologically coherent. Some reasons for such a high percentage of failure may derive from insufficient computer ability, diminished professional autonomy, lack of awareness of long-term benefits of CDSS-use and lack of desire to change the daily workflow (Zheng, Padman et al. 2005). There is also clear evidence that CDSS services are not always used when available, since too numerous systems' alerts are being overridden or ignored by physicians (Moxey, Robertson et al. 2010).

Despite the problems and failures that might accompany CDSSs, these systems have still been proven to improve drug selection and dosing suggestions, reduce serious medication errors by flagging potential drug reactions, drug allergies and identifying duplication of therapy, they enhance the delivery of preventive care services and improve adherence to recommended care standards.

Recent studies suggest that there are some CDSS features crucial to success of these systems (Kawamoto, Houlihan et al. 2005; Shortliffe and Cimino 2006; Pearson, Moxey et al. 2009; Moxey, Robertson et al. 2010):

- CDSS should provide decision support automatically as part of clinicians' workflow, since systems where clinicians were required to seek out advice manually have not been proven as successful.
- Decision support should be delivered at the time and location of decision-making. If the clinician has to interrupt the normal pattern of patient care to move to a separate workstation or to follow complex, time-consuming startup procedures it is not likely that such system will be good accepted.
- Systems that were provided as an integrated component of charting or ordering systems were significantly more likely to succeed than alone standing systems.

Generally speaking, the decision-support element should be incorporated into a larger computer system that is already part of the users' professional routine, thus making decision support a byproduct of practitioners' ordinary work practices.

- Computerized systems have been reported to be advantageous over paper-based systems.
- Systems should provide recommendation rather than just state a patient assessment. For instance, system recommends that the clinician prescribes diuretics for a patient rather just identifying patient being cardiologically decompensated.
- CDSS should request the clinician to record a reason for not following the systems' advice (the clinician is asked to justify the decision with a reason, e.g. "The patient refused").
- It should promote clinicians' action rather than inaction.
- No need for additional clinical data entry. Due to clinicians' effort required for entering new patient data, they tend to avoid this process, which is essential for new decision support. Systems should rather acquire new data automatically (e.g. data retrieval from EMR).
- The system should be easy to navigate and use, e.g. with quick access and minimal mouse clicks for desired information.
- Timing and frequency of prompts are of great importance. For instance if there are too many messages, this might only lead to ignoring all of them and consequently to missing important information. The timing is as well of great importance - the alerts shouldn't appear at inappropriate times and interrupt the workflow.
- The presentation of data or information on CDSSs shouldn't be too dense or the text to small. Researchers also suggest the use of blinking icons for

important tasks or the arrangement of interactions according to their urgency.

- Decision support results should be provided to both clinicians and patients. Studies have shown beneficial effect of such actions, because they stimulate the clinicians to discuss treatment options with patients, and consequently make the latter feel more involved in their medical treatment.
- Periodic feedback about clinician’s compliance with system decision-making.

What these features have in common is that they all make it easier for clinicians to implement the CDSS into their workflow, thus making it easier to use. An effective CDSS must minimize the effort to receive and act on system recommendations. Clinicians found it also very practical if the CDSS would back up its decision-making with linking it to other knowledge resources across the intranet or Internet. In their opinion the safety and drug interaction alerts were the most helpful feature. Above all the organizational factors, such as computer availability at the point of care and technical perfection of CDSS hardware and software are crucial to implementation (Moxey, Robertson et al. 2010).

Kawamoto 2005 suggests that the effectiveness of CDSS remains mainly unchanged when system recommendations are stated more strongly and when the evidence supporting these prompts is expanded and includes institution-specific data.

2.2 EXAMPLES OF CDSS IN PRACTICE

There have been multiple attempts through history to construct a computer or program, which would assist clinicians with their decisions concerning diagnosis and therapy. Ledley and Lusted published the first article evolving around this idea in 1959. The first really functional CDSS didn't appear until the 1970s.

Some of them are reviewed below:

- Leeds abdominal pain,
- MYCIN,
- HELP and
- Internist-1.

Leeds abdominal pain

F. T. de Dombal and his co-workers at University of Leeds developed Leeds abdominal pain. It used Bayesian reasoning on basis of surgical and pathological diagnoses. These pieces of information were gathered from thousands of patients and put into systems' database. The Leeds abdominal pain system used sensitivity, specificity and disease prevalence data for various signs, symptoms and test results. With help of Bayes' theorem it calculated the probability of seven possible diagnoses resulting in acute abdominal pain: appendicitis, diverticulitis, perforated ulcer, cholecystitis, small-bowel obstruction, pancreatitis, and nonspecific abdominal pain. The system assumed that each patient with abdominal pain had one of these seven conditions, thus selected the most likely diagnose on the basis of recorded observations. Evaluation of the system was done by de Dombal et al. in 1972. It showed that the clinicians' diagnoses were correct in only 65 to 80 percent of the 304 cases, whereas the program's diagnoses were correct in 91.8 percent of cases. Surprisingly, the system has never achieved similar results of diagnostic accuracy in practice outside the Leeds University. The most likely reason for that is the variation of data that clinicians entered into the system for acquiring correct diagnoses (de Dombal, Leaper et al. 1972).

MYCIN

This was a consultation system that emphasized appropriate management of patients who had infections rather than just finding their diagnosis. The developers of this system formed production rules (IF-THEN rules), on basis of current knowledge about infectious diseases. The MYCIN program determined which rules to use and how to chain them together in order to make decisions about a specific case. System developers could update the system's knowledge structure rapidly by removing, altering, or adding rules, without reprogramming or restructuring other parts of the system (Shortliffe 1976).

The HELP System

The HELP system is actually an integrated hospital information system with the ability to generate alerts when data abnormalities in the patient record are noted. It can output data either automatically, in form of printed reports, or it can display specific information, if so requested. Furthermore, the system has an event-driven mechanism for generation of specialized warnings, alerts and reports (Burke, Classen et al. 1991).

Internist-I

This was an experimental CDSS designed by Pople and Myers at the University of Pittsburg in 1974. It was a rule-based expert system capable of making multiple, complex diagnoses in internal medicine based on patient observations. The Internist-I was using a tree-structured database that linked symptoms with diseases. The evaluation of the system revealed that it was not sufficiently reliable for clinical application. Nevertheless, the most valuable product of the system was its medical knowledge base. This was used as a basis for successor systems including CADUCEUS and Quick Medical Reference (QMR), a commercialized diagnostic CDSS for internists (Miller, Pople et al. 1982).

2.3 SELECTED CONTEMPORARY EXAMPLES OF CDSS

ATHENA

The Athena decision support system was deployed in 2002 as a tool to implement guidelines for hypertension. It encourages blood pressure control and issues recommendations about a suitable choice of therapy, concordant with latest guidelines. It also considers co-morbidities of the specific patient in question. ATHENA DSS has an easily changeable knowledge base that specifies criteria for eligibility, risk stratification, set blood pressure margins, it includes relevant co-morbid states and guideline-recommendation, specific for patients with present co-morbidities. The knowledge base also comprises of preferences for certain drugs within antihypertensive drug groups according to the latest evidence.

New pieces of evidence are constantly changing protocols of best hypertension management; ATHENA is thus designed to be accessible to clinicians for knowledge base-customization and to custom local interpretations of guidelines according to the local population structure and other factors.

The system was designed to be independently integrated into a variety of EMR-systems, and is thus interchangeable and adaptable for various health information-systems. The effectiveness, accuracy and success of implementation has been researched and reviewed on many occasions (Goldstein, Coleman et al. 2004; Lai, Goldstein et al. 2004).

ISABEL

Isabel is a web-based diagnosis decision support system that was created in 2001 by physicians. It offers diagnosis decision support at the point of care. The system is eligible for all aged patients, from neonates to geriatrics. Its database covers major specialties like Internal Medicine, Surgery, Gynecology & Obstetrics, Pediatrics, Geriatrics, Oncology, Toxicology and Bioterrorism. Isabel produces an instant list of likely diagnoses for a given set of clinical features (symptoms, signs, results of tests and investigations etc), followed by suggesting the administration of suitable drugs. This is executed by reconciling (i.e. pattern-matching technology) patient data sets with data sets as described in established medical literature. The system allows clinicians to follow their assumptions about differential diagnoses; it hence restricts searches to specific body systems, relatively to diagnoses in question. The system is interfaced with EMR, which allows it to extract existing diagnoses and other patient-specific data.

Furthermore it contains a feature to help Isabel has been extensively validated and been shown to enhance clinician's cognitive skills and thereby improves patient safety and the quality of patient care (Ramnarayan, Tomlinson et al. 2004; OpenClinical 2006).

LISA

LISA is a CDSS that consists of two main components. The first is a centralized Oracle database, holding all patient information about drug schedules, blood and toxicity results, doses prescribed etc. The database is accessible by health professionals from different sectors and locations. The second component represents a web-based decision support module, which is using the PROforma guideline development technology to provide advice about dose adjustments in treatment of acute childhood lymphoblastic leukemia. Clinicians answer their questions with up to date knowledge from textbooks and journals.

CHAPTER THREE
METHODOLOGY AND SYSTEM ANALYSIS

3.0 PREAMBLE

Procedures used in data collection and information gathering are here, outlined and analyzed. Data was carefully collated and objectively evaluated in order to define as well as ultimately provide solutions to the problems for which the research work is based.

During the research work, data collection was carried out in many places. In gathering and collecting necessary data and information needed for system analysis, two major fact-finding techniques were used in this work and they are:

a. Primary source
b. Secondary source

Primary source:

Primary source refers to the sources of collecting original data in which the researcher made use of empirical approach such as personal interview and questionnaires.

This involved series of orally conducted interviews with select clinicians in public and private healthcare practice on the diagnostic procedures they adopt. Also, some patients were interviewed with a view to getting information about their opinion on how medical diagnoses affected them.

Secondary Source:

Perusals through online journals and e-books as well as visits to relevant websites, medical dictionaries and other research materials increased my knowledge and aided my comprehension of diagnostic processes.

3.1 METHODS OF DATA COLLECTION

- **Oral Interview**

This was done between the researcher and the doctors in the hospital used for the studies, and the lab attendance was interviewed. Reliable facts were got based on the questions posed to the staff by the researcher.

- **Study of Manuals**

Manuals and report based used by lab attendance were studied and a lot of information concerning the system in question was obtained.

- **Evaluation of Forms**

Some forms that are necessary and available were assed. These include admission card, lab form, test result, bill card Etc. These forms help in the design of the new system.

3.2 ANALYSIS OF EXISTING SYSTEM

This aims at objectively evaluating the existing system of diagnostics and treatment in the hospital with a view to highlighting its limitations. It also seeks to proffer solutions by offering a knowledgeable expert system which would aid clinicians in diagnostic procedures.

The existing system of medical diagnosis and drug prescription in most hospitals involves manual activities. A proper diagnosis is the first step towards proper medical care. This was the consensus opinion reached by all respondents interviewed. An investigation into how diagnosis is carried out revealed that anytime patients visit the hospital, they are subjected to long waiting hours just to undergo the regular card verification and clearance.

Patients queue accordingly for several hours on a first come first serve (FCFS) basis. A new patient usually registers into the hospital by filling the patient form which signifies that the person is now registered with that hospital. It also, gives the person access to own a hospital folder which is used to record basic information about the diagnoses and drug prescriptions to the patient.

He/she is then referred to a doctor for examination and testing. This examination helps the doctor to determine exactly what a patient may be suffering from. Testing is a great way to find out a medical condition early before it deteriorates.

However it was the widespread practice that in attending to registered patients the attending staff usually retrieved his hospital folder using the patient's form. This form is then sent to the doctor who peruses it, before examining the patient and carrying out the appropriate therapy. The patient is either referred to the laboratory unit for a test (if the need be) or to the pharmacy unit to obtain the prescribed drugs (if the matter is not too complex).

Any treatment proffered to the patient by the doctor must be recorded in the patient's folder to aid future diagnostic references.

This procedure is usually a long and tedious one with attendant bottlenecks.

3.3 BLOCK DIAGRAM OF EXISTING SYSTEM

The diagram below graphically illustrates the process of service delivery to patients in the hospitals visited.

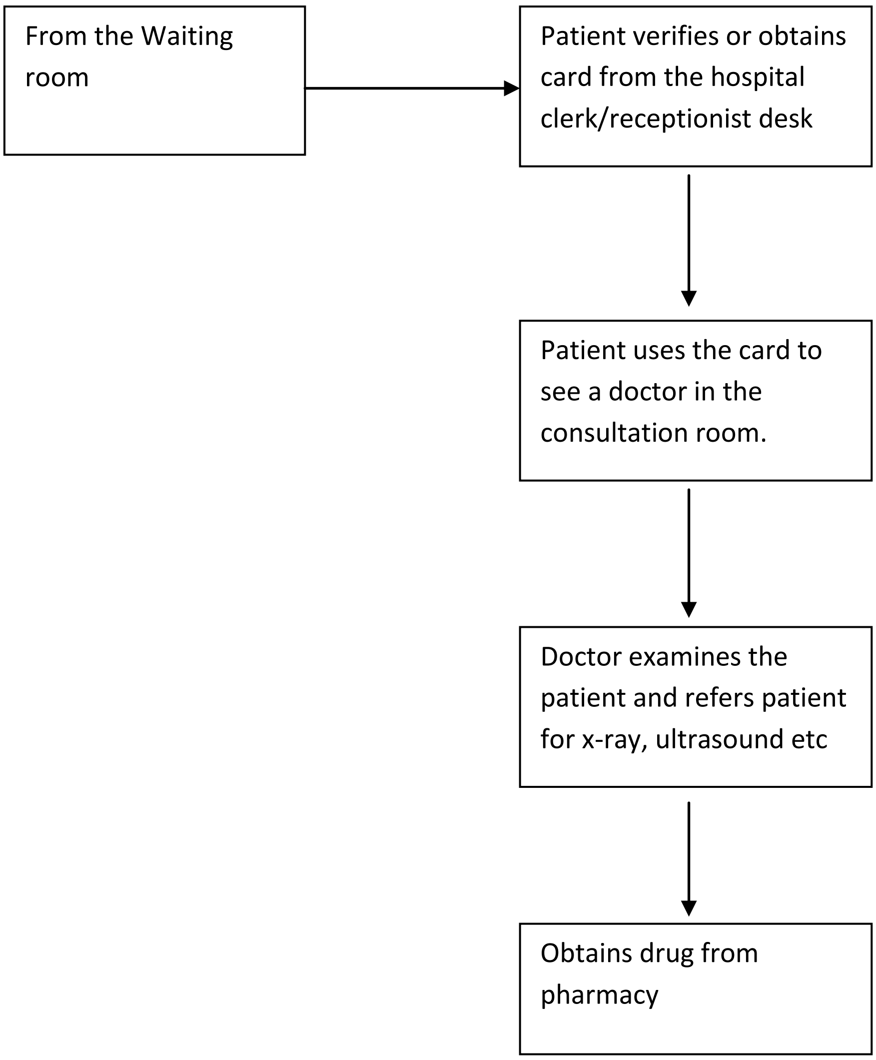


Fig 3.1 Block diagram of existing system service path

3.4 LIMITATIONS OF EXISTING SYSTEM

Some shortcomings were noticed in the existing system after thorough analysis. They include:

a. Manual documentation of patients' records

It was noticed in the course of investigation that the existing system was heavily dependent on manual methods of entering, storing and retrieval of patients' data. This implied patients had to wait for quite long before being referred for diagnosis.

b. Error in diagnosis:

It was discovered that in some cases, wrong diagnosis was given for ailments because they (the ailment) were relatively new and the physician had limited knowledge about it. The situation was even made worse because at the point of medical examination, the physician could not access a wider knowledge base for guidance.

c. Stalling of treatment due to doctor's absence:

Another discovery was that patients had to wait indefinitely in the event of a doctor's prolonged absence and sometimes, end up not accessing treatment. This has led to a further deterioration of their health conditions and in some cases resulted in death of patient.

3.5 INPUT, PROCESS AND OUTPUT ANALYSIS OF PROPOSED SOLUTION

The proposed system is built with the benefit of an object-oriented approach. The system seeks to build a computational model of some problem domain and therefore tends to be exploratory in nature.

The flow of data in the proposed system is in such a way that when a particular disease is highlighted from the disease menu, it will display an interactive submenu that includes the symptoms. The central concepts of the object-oriented paradigm are introduced namely: encapsulation, inheritance and polymorphism.

INPUT ANALYSIS

This deals with the process used to feed data to the system for processing. Here, data could be manually fed in with the help of a keyboard or sourced for electronically by consulting the electronic medical records (EMR) database. The data supplied to the system includes:

a) Patient’s name
b) Home address
c) Sex
d) Age
e) Disease symptoms
f) Date visited

PROCESS ANALYSIS

After the inputs are collected, the system analyzes the data and queries its knowledge base for the actual or related medical condition. Data mining may be conducted to examine the patient’s medical history in conjunction with relevant clinical research. Such analysis can help predict events, which can range from drug interactions to disease symptoms.

OUTPUT ANALYSIS

The CDS system with the aid of its knowledge base, applies rules to patient data using an inference engine and displays the results to the end user(clinician) via his monitor screen. The output here can be

- Clinician diagnosis
- Preventive and control mechanisms
- Drug prescription

3.6 JUSTIFICATION FOR THE NEW SYSTEM

It is expected that with the introduction new system, a lot of positive changes will be noticed. In the design of the web-based diagnostic system, conscientious effort is made to create an effective knowledge based system which would be successfully implemented into the workflow, providing the clinician with the necessary support in their decision making abilities.

The system will also significantly improve health workers' performance and improve patient outcome thus affecting the gross quality of health care delivery.

CHAPTER FOUR

DESIGN, TESTING AND IMPLEMENTATION OF THE NEW SYSTEM

4.0 DESIGN STANDARD

The major objective of this design is to achieve a new system that is more reliable and robust than the existing system in terms of rapid disease prognosis, diagnosis and treatment prescription based on the accurate disease symptoms as provided by the patient in the course of examination and the expert system's inference.

Here the doctor accesses the application on the computer system and keys in the symptoms of the patient's ailment. Once this is done, the software will diagnose the patient based on the symptoms entered. The result of the diagnosis will be displayed on the screen showing the disease the patient is suffering from and the recommended treatment for the disease.

The software design process of the proposed system after a detailed analysis of the current system is carried out using a particular design methodology.

Top down approach has been the best approach in most engineering designers. This involves the disintegration of the project topic itemed as system into subsets called the subsystem.

In the proposed system, the system is divided into different modules and subsystems. These subsystems perform a particular task. At the end of which the whole system is integrated together in line with stated objectives.

The terminals at different locations are connected to the medical knowledge base management system of the expert system. All the files, user forms, diagnostic forms and associated programs will be connected. The design will also provide necessary control both manual and automated to help maintain the integrity of the data base files.

4.1 OUTPUT DESIGN

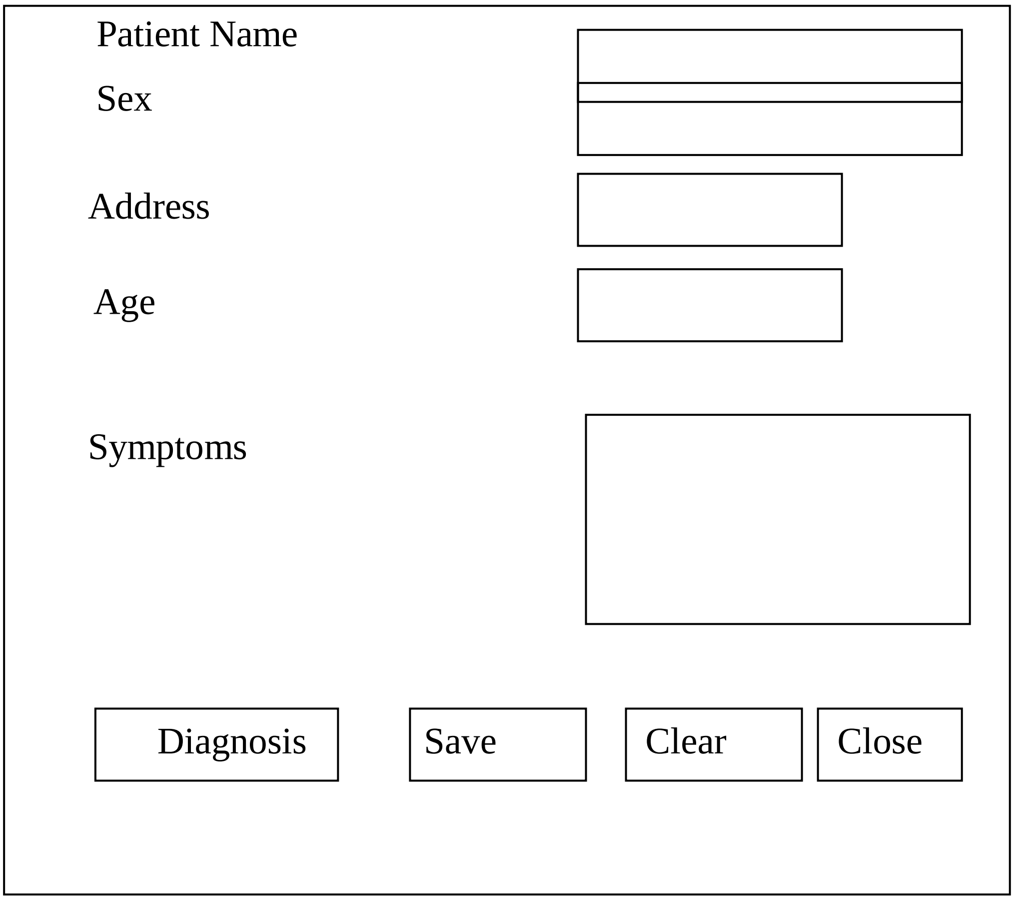
The output form is designed to generate printable reports from the database. The output is place on a database grid and contains information on patient’s records. The output produced can be printed on a hard copy or viewed on the screen. The output generated by the expert system includes:

1. Disease diagnosis report
2. Patients Report
3. Disease treatment report.

4.2 INPUT DESIGN

The input to the new system is the patient's symptoms, which is entered through the keyboard. The input form design takes the format bellow.

Patients Diagnosis Entry Form



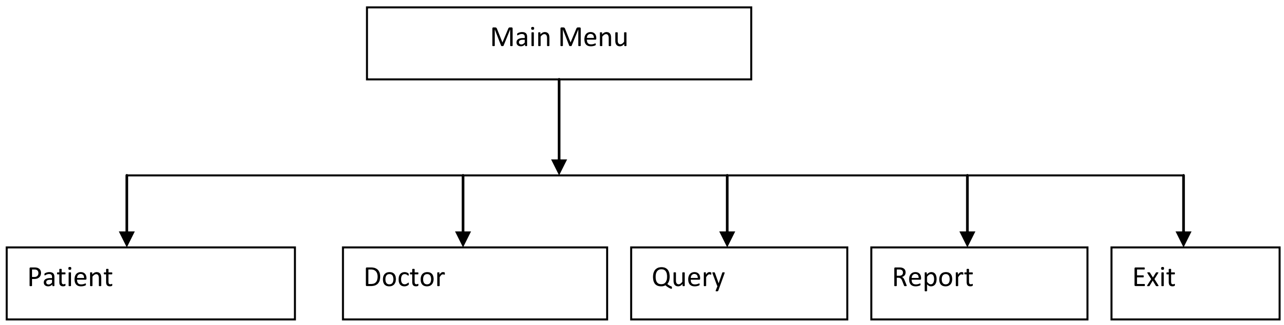
4.3 DATABASE DESIGN

In any good database design, effort should be made to remove completely or at best reduce redundancy. The database design in the software is achieved using Microsoft Access Database. Below is the structure of the database.

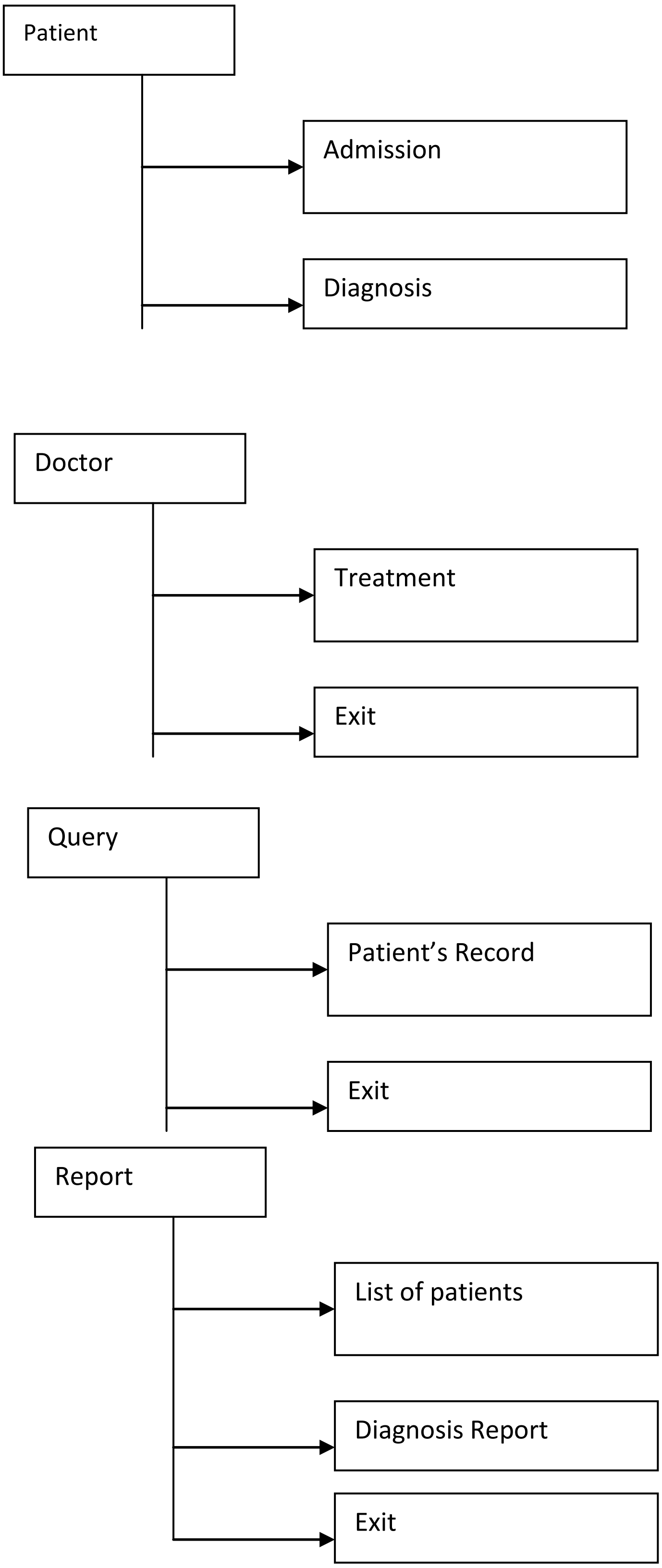
PATIENTS DIEASES SYMPTOMS TABLE

FIELD	FIELD TYPE	FIELD SIZE
No	Text	15
Patients Name	Text	20
Address	Text	30
Age	Integer	2
Sex	Text	1
Symptoms	Text	100
Diagnosis	Text	100
Date	Date/time	8
Treatment	Text	100

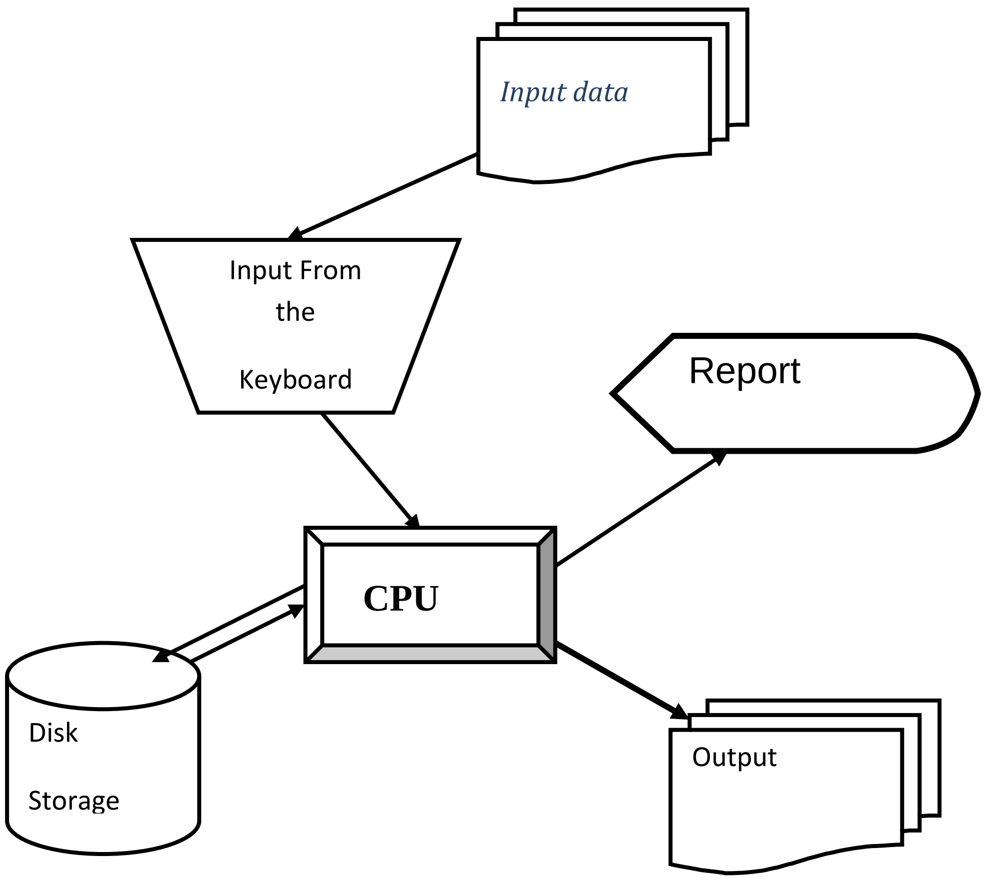
4.4 THE MAIN MENU



4.5 THE SUB MENU



4.6 SYSTEM FLOWCHART



4.7 CHOICE OF PROGRAMMING LANGUAGE

The new system was implemented using Microsoft Visual Basic programming language. This is because the programming language has the advantage of easy development and flexibility. It also has the ability of providing the developer/programmer with possible hints and equally produces a graphical user interface.

Visual Basic is an event driven, graphical user interfaced object oriented programming environment. Structured programming allows the program to be developed in presented module, either by using a top-down or bottom-up method.

The hierarchy of object is in visual basic and it runs the objects, (such as controls) which are placed in frames (another object which group other objects virtually together), and can be placed on the form (windows which open up to display information, or receive input from the user). These forms are linked together by code modules to create a finished visual basic application.

Forms being objects have their own properties and methods attached to them as well, amongst which are caption (which displays text centered at the top of the form, the control box, (which allows one to minimize, maximize, remove, resize, restore or close the form) and the desktop. There exist also two boxes which allow the desktop to change the colour of the form. The toolbox which allows one to design the screen by choosing various options from it such as label text, checkbox and command button is also present.

Considering all these features and much more, the most preferred choice to use was the Visual Basic for window environment, which was quite rewarding.

4.8 SYSTEM REQUIREMENTS

The computer system is made up of units that are put together to work as one in order to achieve a common goal. The requirements for the implementation of the new system are:

- The Hardware
- The Software

Software Requirement

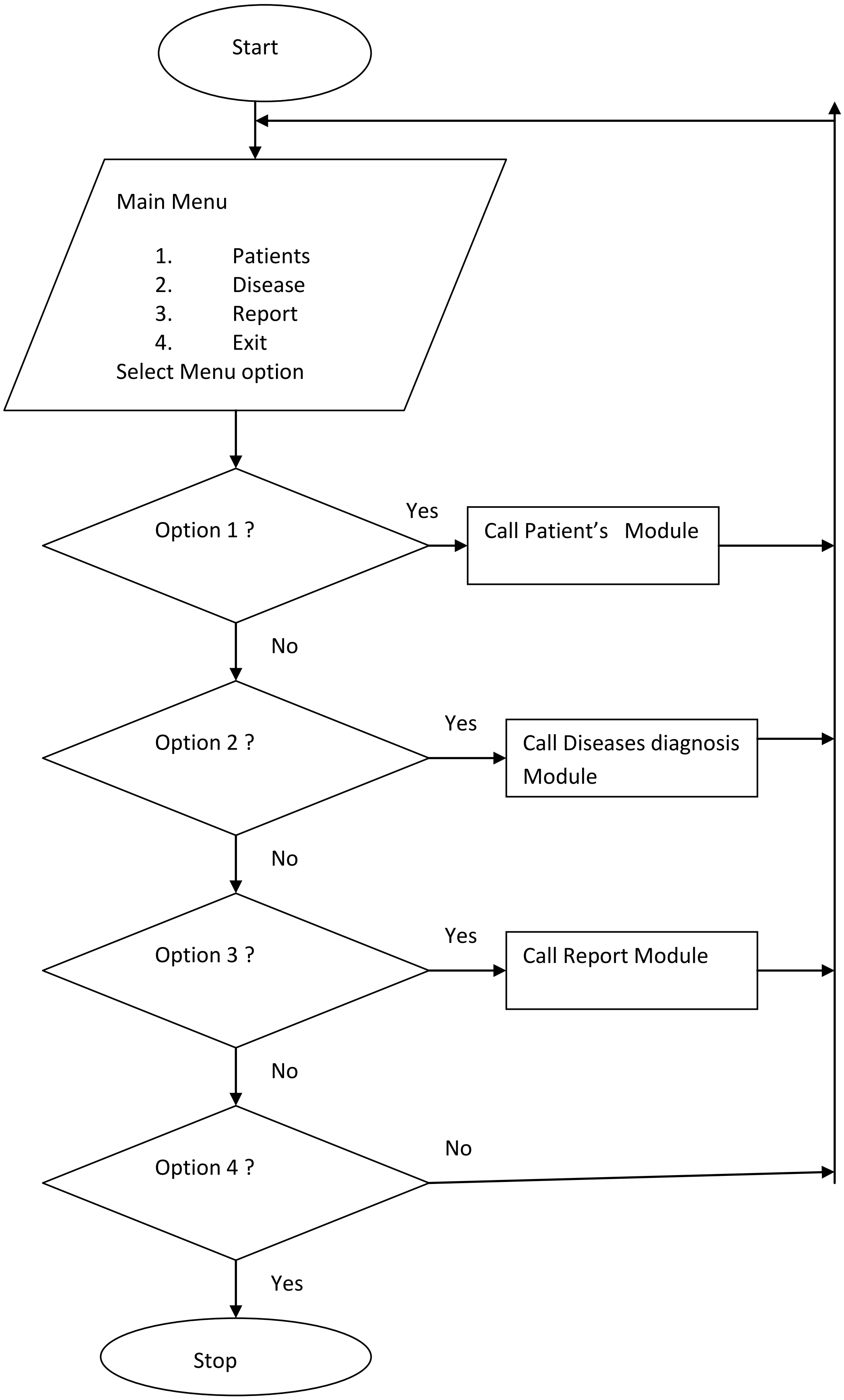
For the effective implementation of the new system, the following software has to be installed on the computer system.

- Windows XP operating system or later
- Microsoft Access Database 2010 or earlier
- Visual basic 6.0

Hardware Requirement

- Pentium VI and Above
- 1GB Ram and above
- 40 GB HD
- Printer

4.9 Program Flowchart



4.10 CHANGE OVER PROCESS

This is the process of changing from the manual system to computerized system. When the entire procedure obtained in an organization is converted to automatic electronic mode. There are many methods of change over which include:

Direct Changeover

In this method the old system is completely replaced by the new system in one move. This may be avoidable where the two systems are substantially different, where the new system is a real time system, or when an extra staff to oversee its parallel running is unobtainable. This method is comparatively cheap but is risky. Program corrections are difficult while the system has to remain operational. The new system should be introduced during stack periods and in large systems. It may be introduced, as an application, allowing several months between each stage to ensure all problems are cleared up before the whole system becomes operational.

Parallel Changeover

In this method , both the manual and computerized system are operated concurrently for sufficiently long period and their outputs compared periodically and possible discrepancies reconciled on the new system until all users are satisfied .The old system is discontinued when discrepancies are seen to have seized arising. It has the advantage of having an old system to fall back on, in case the new system fails. The disadvantage is the cost of running two systems side by side, both of which will achieve similar result.

Phase Changeover

Here, the changeover starts with a department or branch. The effect of the new system in the sample department or branch is observed before some other department or branch which may be more sensitive can adapt to the new system.

Pilot Changeover

In this case, some transactions that are very complex are operated using parallel changeover and in other remaining existing system in application, direct changeover is used. The researcher recommends the “parallel changeover” to avoid drastic problems that may arise due to failure of a newly developed system.

4.11 SOFTWARE TESTING

This defines the test requirement, which the software should meet and it is progressively integrated into complete package. The process of test plan is concerned with providing that a package produces correct and expected result for all possible input data.

For this software testing, we have three basic testing that should be adopted viz:

a. Module Testing
b. Integrated testing and
c. System testing

Module Testing

In this design we have many modules which when triggered up at certain events perform a specific function. So, module testing involves testing of each of the modules in software to verify that they meet their respective objective module testing were carried out to ensure that information properly flows into and out of the program module under test.

The Integration Test

So far, the various modules have been tested and each proved efficiency as an entity. (i.e. module). Though sometimes, the modules can perform their respective functions but when put together, they can function together. So this test therefore checks that when the modules are integrated they can combine to perform their respective functions. Hence, integration testing was done to entire program structure to uncover errors associated with interfacing. These errors were debugged to produce desired results. The essence of integration testing is to ascertain that these modules do not lose their efficiency and reliability. The Integration involved the main form which serves as coordinator and driver for other module.

System Testing

Before bringing and data processing system into use, it is of vital importance that the system is both comprehensive within its intended limits and fully correct. So, each routine must have been written according to specification and tested to complete satisfaction. Also bags must have been removed completely and the program run produced exactly what is required of it.

CHAPTER FIVE

SUMMARY, CONCLUSION AND RECOMMENDATIONS

5.0 SUMMARY

The primary goal of clinical decision support systems development, as for any branch of biomedical research, is to improve the overall health of the population. CDSSs may contribute to this by improving the quality of healthcare services, as well as by controlling the cost-effectiveness of medical examinations and treatment.

The ultimate acceptance of CDS systems will depend not only on the performance of the computerized method alone, but also on how well the human performs the task when the computer output is used as an aid and on the ability to integrate the computerized analysis method into routine clinical practice (Hunt, Haynes, Hanna & Smith, 1998).

Issues, such as a friendly user-interface, a short system response time and low cost, are critical for the daily routine use of CDS systems. Obviously, the development of CDS systems requires close collaboration of two scientific areas: medicine and computer science. This collaboration aims to codify knowledge and define the logical procedures used by the physician to reach a conclusion.

As a result, the engineer must “extract” knowledge from the physician and reproduce it appropriately. This is particularly difficult because the physician’s decisions are the result of a complex procedure combining special knowledge and experience.

5.1 CONCLUSION

The coupling of CDSS technology with evidence-based medicine brings together two potentially powerful methods for improving health care quality. To realize the potentials of this synergy, literature-based and practice-based evidence must be captured into computable knowledge bases, technical and methodological foundations for evidence-adaptive CDSSs must be developed and maintained, and public policies must be established to finance the implementation of electronic medical records and CDSSs and to reward health care quality improvement.

5.2 RECOMMENDATIONS

Based on the remarkable successes recorded by clinical decision support systems in robust health care delivery, this research work is therefore recommended to approved health institutions such as: hospitals, primary health centers, medical laboratories etc to further enhance diagnostic processes by clinicians hereby guaranteeing efficiency in drug or therapy prescription and ultimately ensuring effective treatment.

Quoting Delaney, Fitzmaurice, Riaz & Hobbs, 1999, future trends and challenges in the area of CDS systems include the creation of links to patient electronic medical records and a universally-agreed upon medical vocabulary, so that the entries in the medical records can have well-defined meanings. In addition to this, studies that evaluate the performance of CDS systems in clinical practice, in conjunction with demonstrations of cost-effectiveness, are a critical stage in further developing CDS systems. Users should be responsible for carefully monitoring the introduction of any new system carefully.

REFERENCES

Autonomy, I. (2009). *Autonomy Technology Overview*. U.S.A: Macmillan Press.

Bassingthwaighte, J. B. (2000). “Strategies for the physiome project” *Ann Biomed Eng* 28(8): 1043-1058 Biomed

Berner, E. S. (2007). Clinical decision support systems: Theory and Practice. New York: Springer Inc.

Bohanec, M., Zupan, B., & Rajkovič, V. (2000). Applications of qualitative multi-attribute decision models in health care. *Int J Med Inform,* 58-59, (2000) 191 – 205.

Burke, J. P., & Classen, D. C. (1991). "The HELP system and its application to infection control." *J Hosp Infect* 18 Suppl A: 424-431.

Bury, J., C. Hurt, et al. (2004). "A quantitative and qualitative evaluation of LISA, a decision support system for chemotherapy dosing in childhood acute lymphoblastic leukaemia." *Stud Health Technol Inform* 107(Pt 1): 197-201.

Chan, K.; Felson, D.; Yood, R.; Walker, A. (1994). "The lag time between onset of symptoms and diagnosis of rheumatoid arthritis". *Arthritis and rheumatism* 37 (6): 814–820.doi:10.1002/art.1780370606. PMID 8003053.

Coiera, E. (2003). Guide to health informatics. London : Oxford University Press.

Dombal, F. T., & Leaper, D. J. (1972). "Computer-aided diagnosis of acute abdominal pain." *Br Med J* 2(5804): 9-13.

Doyle, B. J., & Callanan, A. (2009). "Vessel asymmetry as an additional diagnostic tool in the assessment of abdominal aortic aneurysms." *J Vasc Surg* 49(2): 443-454.

Garg, A. X., & Adhikari, N. K. (2005). "Effects of computerized clinical decision support systems on practitioner performance and patient outcomes: a systematic review." *JAMA* 293(10): 1223-1238.

Goldstein, M. K., & Coleman, R. W. (2004). "Translating research into practice: organizational issues in implementing automated decision support for hypertension in three medical centers." *J Am Med Inform Assoc* 11(5): 368-376.

Horstmanshoff, H. F. J. (2004). *Magic and Rationality in Ancient Near Eastern and Graeco-Roman Medicine*. Germany: Brill Publishers,

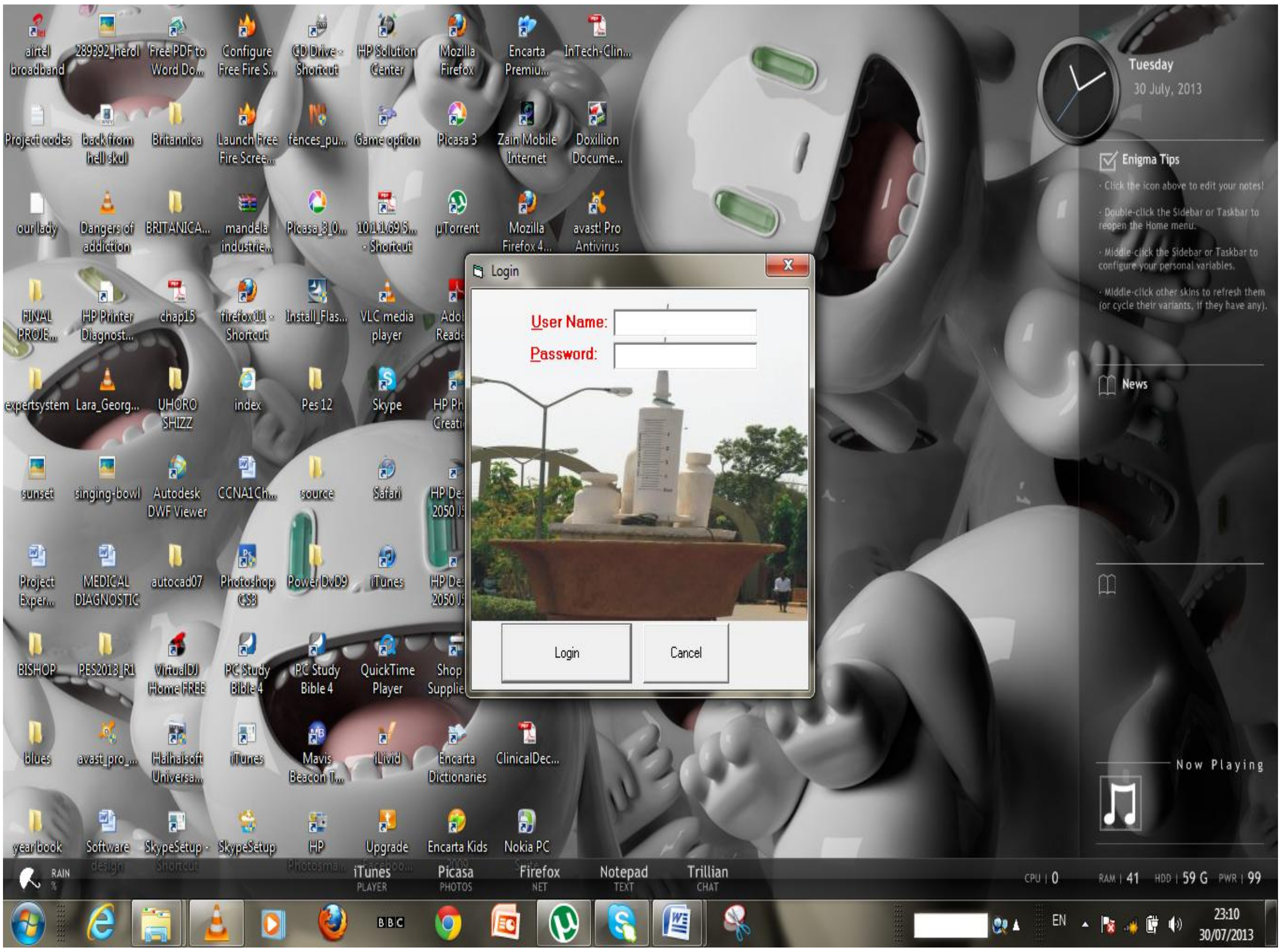
Johnson, P. E., Duran, A. S., Hassebrock, F., Moller, J., Prietula, M., Feltovich, P. J., & Swanson, D. B. (1981). "Expertise and Error in Diagnostic Reasoning". *Cognitive Science* 5 (3): 235–283.doi:10.1207/s15516709cog0503_3.

"Online Etymology Dictionary".

Treasure, W. (2011). *Diagnosis and Risk Management in Primary Care: words that count, numbers that speak*. Oxford: Radcliffe Publishers.

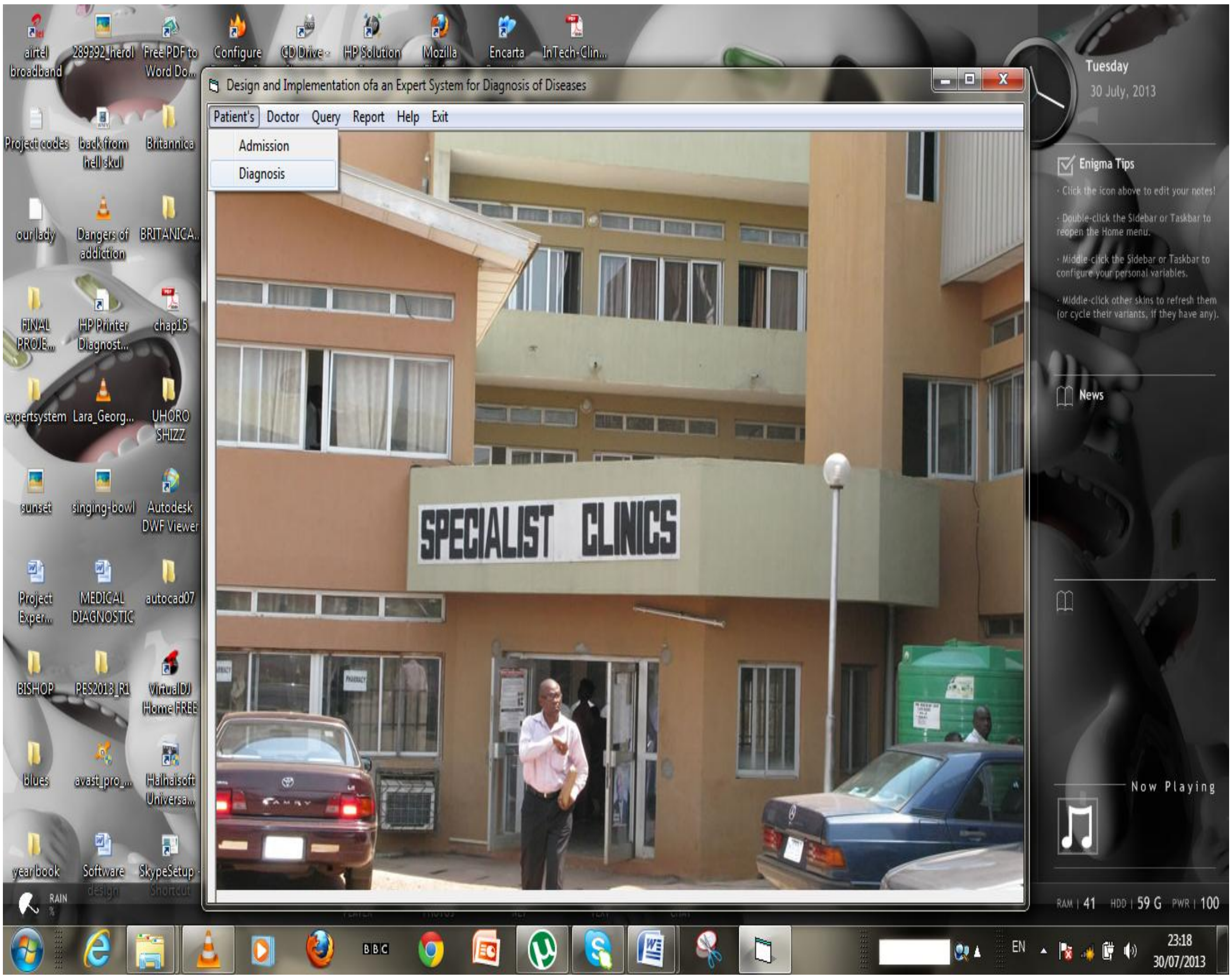
APPENDIX A

Program Login Screen

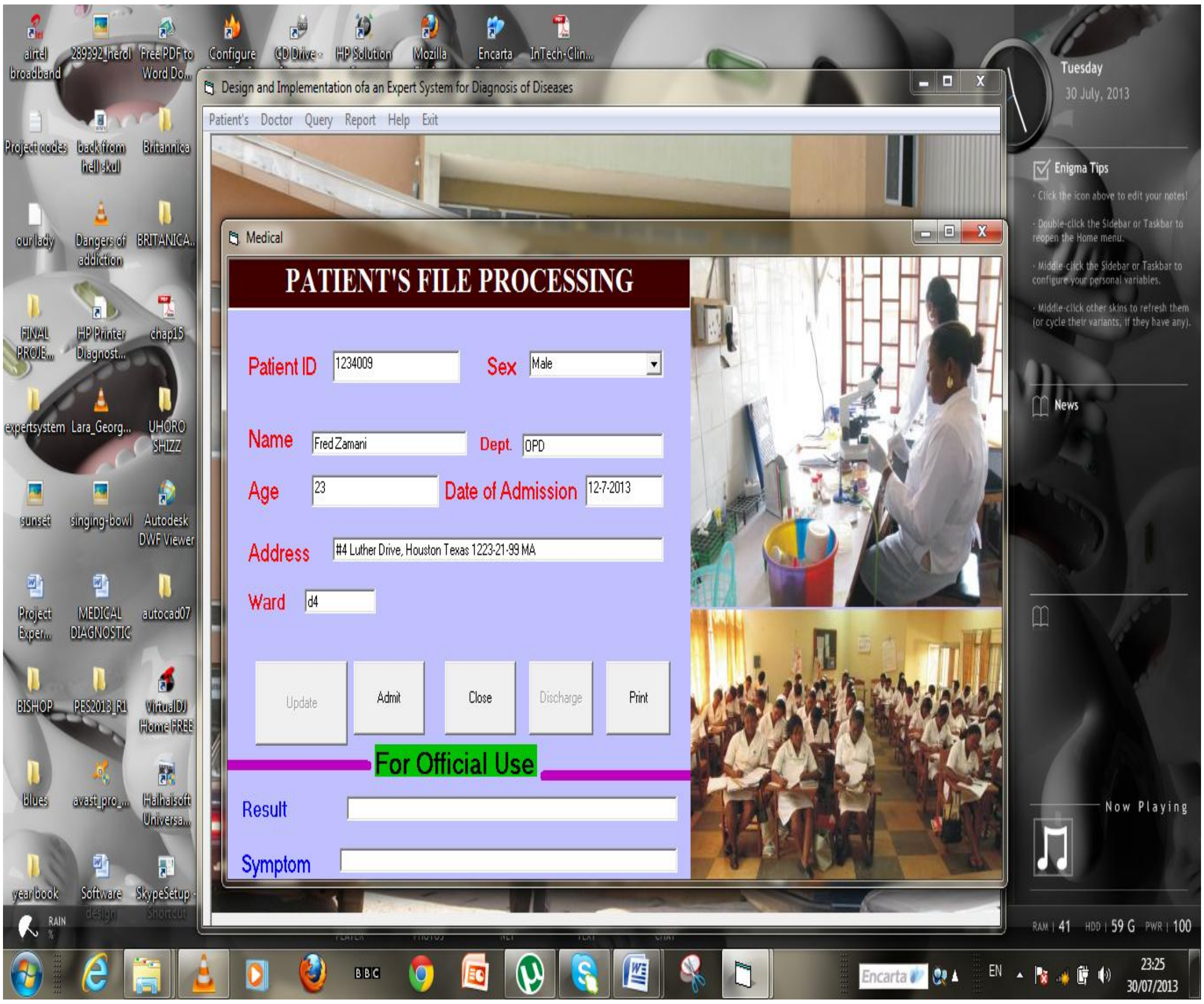


Login Screen

Program Mainmenu

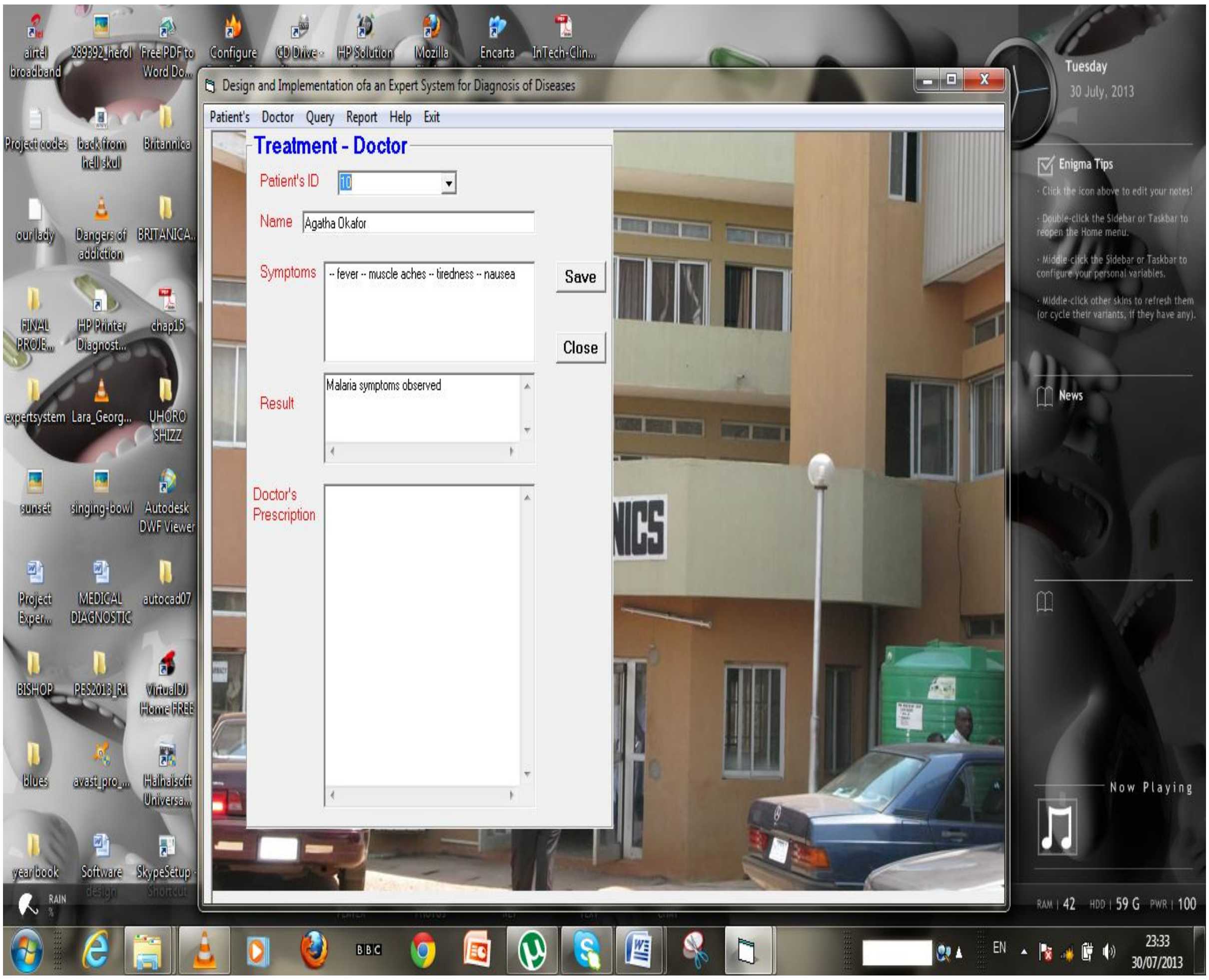


Patient's Registration Form

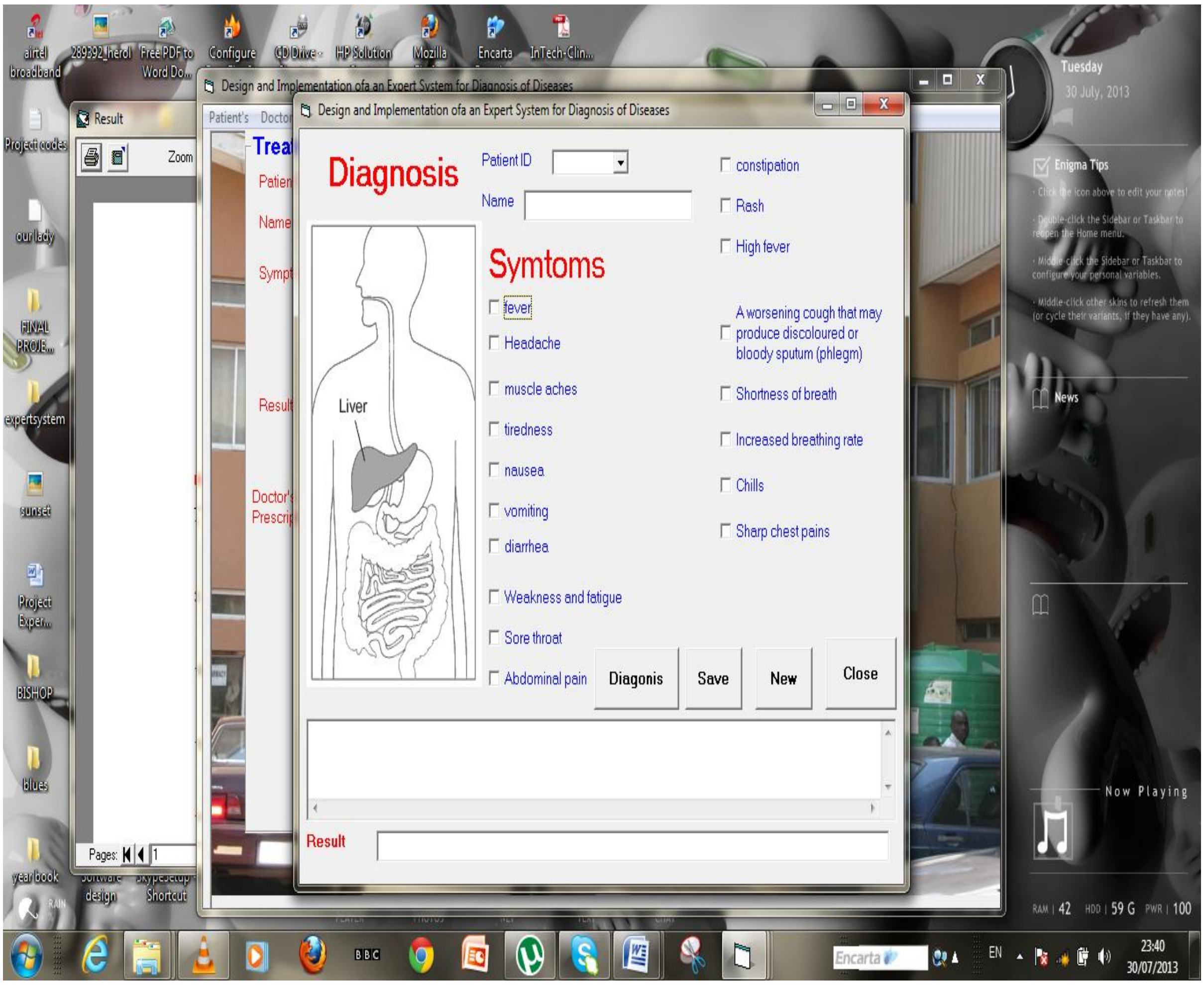


Patient's Registration Form

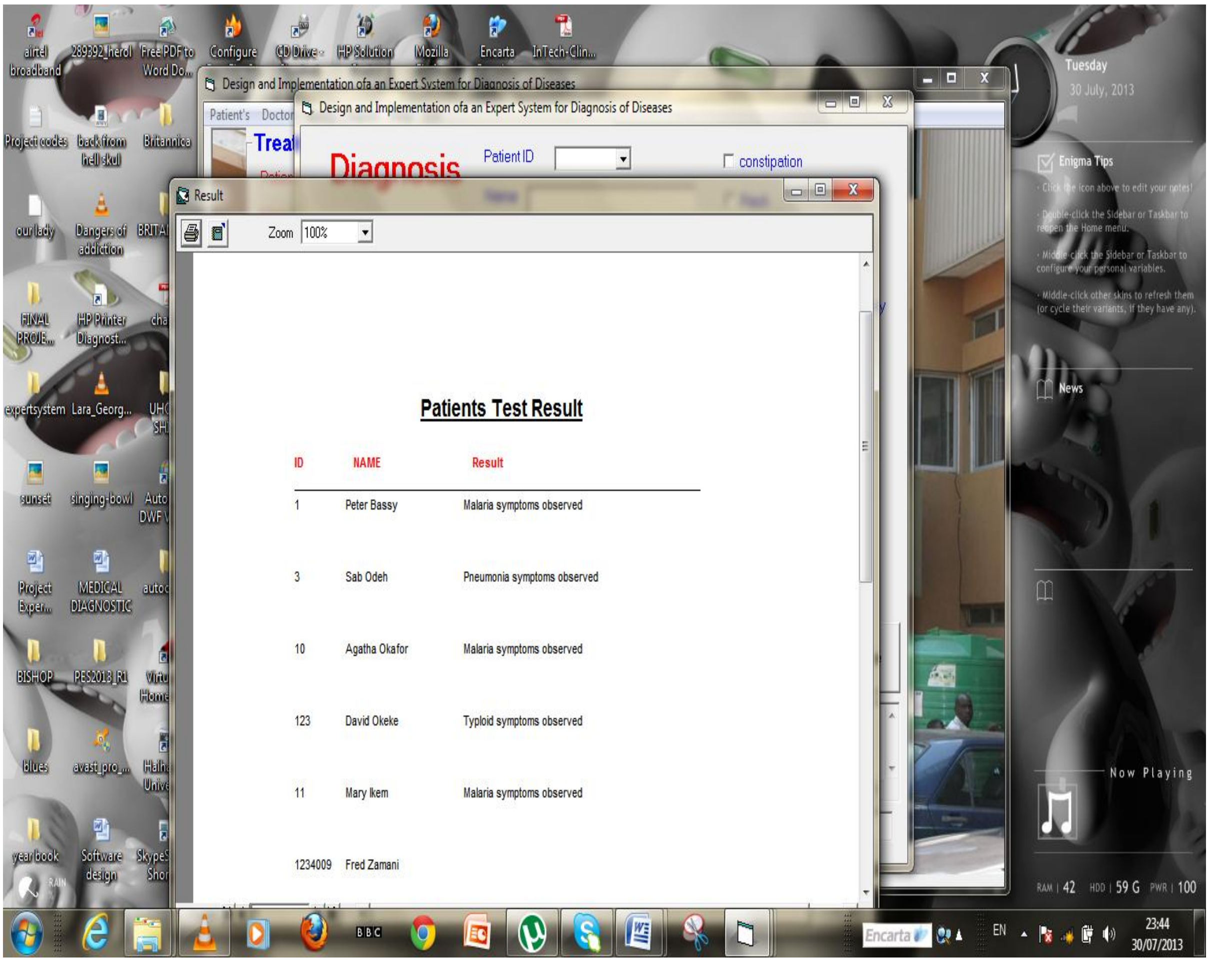
Diagnostic Process



Symptoms box



Generated Patients Test Results



APPENDIX B

PROGRAM SOURCE CODE

```
Private Sub chk_Click(Index As Integer)

If chk(Index).Value = 1 Then

Text2.Text = Text2.Text + " -- " + chk(Index).Caption


n = Index

If n < 7 Then

Text3.Text = Val(Text3.Text) + 1

ElseIf n < 12 Then

Text4.Text = Val(Text4.Text) + 1

Else

Text6.Text = Val(Text6.Text) + 1


End If

End If

End Sub


Private Sub cmdclose_Click()

Form1.Hide

End Sub

Private Sub cmddiagn_Click()

If (Val(Text3.Text) > 1) And (Val(Text4.Text) = 0) And (Val(Text6.Text) = 0)
Then
```

```
MsgBox "Malaria symptoms observed", vbInformation, "Test Result"
Text5.Text = "Malaria symptoms observed"

ElseIf (Val(Text4.Text) >= 1) And (Val(Text6.Text) < 2) Then

MsgBox "Typhoid symptoms observed", vbCritical, "Test Result"
Text5.Text = "Typhoid symptoms observed"

ElseIf (Val(Text6.Text) >= 2) Then
MsgBox "Pneumonia symptoms observed", vbCritical, "Test Result"
Text5.Text = "Pneumonia symptoms observed"

End If

End Sub
Private Sub cmdnew_Click()
Text1.Text = ""
Text2.Text = ""
Text3.Text = ""
Text4.Text = ""
Text5.Text = ""
Text6.Text = ""
For I = 0 To 17
chk(I).Value = 0
```

```
Next I


End Sub
Private Sub cmdsave_Click()
Form2.Data1.Recordset.MoveFirst
Do Until Form2.Data1.Recordset.EOF
If Combo1.Text = Form2.Data1.Recordset.Fields("id") Then
Form2.Data1.Recordset.Edit
Form2.Data1.Recordset.Fields("symptoms") = Text2.Text
Form2.Data1.Recordset.Fields("result") = Text5.Text
Form2.Data1.Recordset.Update
Text1.Text = ""
Text2.Text = ""
Text3.Text = ""
Text5.Text = ""
Text6.Text = ""


Text4.Text = ""
For I = 0 To 17
chk(I).Value = 0
Next I
Exit Do
Else
Form2.Data1.Recordset.MoveNext
```

```
End If
Loop
End Sub

Private Sub Combo1_Click()
Form2.Data1.Recordset.MoveFirst
Do Until Form2.Data1.Recordset.EOF
If Combo1.Text = Form2.Data1.Recordset.Fields("id") Then
Text1.Text = Form2.Data1.Recordset.Fields("name")
Exit Do
Else
Form2.Data1.Recordset.MoveNext
End If
Loop
End Sub

Option Base 1
Private reclenght, reci, recindex, I, t As Integer
Option Explicit

Private Sub Combo2_Change()


End Sub
```

```
Private Sub Command1_Click()

Form2.Data1.Recordset.AddNew
Form2.Data1.Recordset.Fields("id") = Form2.Text1.Text
Form2.Data1.Recordset.Fields("name") = Form2.Text2.Text
Form2.Data1.Recordset.Fields("sex") = Form2.Combo1.Text
Form2.Data1.Recordset.Fields("age") = Form2.Text4.Text
Form2.Data1.Recordset.Fields("address") = Form2.Text6.Text
Form2.Data1.Recordset.Fields("symptoms") = "-"
Form2.Data1.Recordset.Fields("ward") = Form2.Text7.Text
Form2.Data1.Recordset.Fields("date admitted") = Form2.Text5.Text
Form2.Data1.Recordset.Fields("bill") = 0
Form2.Data1.Recordset.Fields("amount paid") = 0
Form2.Data1.Recordset.Fields("balance") = 0
Form2.Data1.Recordset.Fields("date discharged") = "-"
Form2.Data1.Recordset.Fields("treatment") = "-"
Form2.Data1.Recordset.Fields("dept") = Form2.Text12.Text
Form2.Data1.Recordset.Fields("Remark") = "Admitted"
Form2.Data1.Recordset.Fields("view") = "Yes"
Form2.Data1.Recordset.Update
Form2.Text1.Text = ""
Form2.Text12.Text = ""
Form2.Text2.Text = ""
Form2.Text3.Text = ""
```

```
Form2.Text4.Text = ""
Form2.Combo1.Text = ""
Form2.Text5.Text = ""
Form2.Text6.Text = ""
Form2.Text7.Text = ""
Form2.Text8.Text = ""

End Sub

Private Sub Command2_Click()
Command1.Enabled = False
Command4.Enabled = False
Form2.Hide
End Sub

Private Sub Command3_Click()
Form2.Data1.Recordset.MoveFirst
Do Until Form2.Data1.Recordset.EOF
If frmDataEnv.Combo2.Text = Form2.Data1.Recordset.Fields("id") Then
Form2.Data1.Recordset.Edit
Form2.Data1.Recordset.Fields("date discharged") = Date
Form2.Data1.Recordset.Fields("Remark") = "Discharged"
Form2.Data1.Recordset.Update
Form2.Text1.Text = ""
```

```
Form2.Text12.Text = ""
Form2.Text2.Text = ""
Form2.Text3.Text = ""
Form2.Text4.Text = ""
Form2.Combo1.Text = ""
Form2.Text5.Text = ""
Form2.Text6.Text = ""
Form2.Text7.Text = ""
Form2.Text8.Text = ""

Exit Do
Else
Form2.Data1.Recordset.MoveNext
End If
Loop

End Sub

Private Sub Command4_Click()
Form2.Data1.Recordset.MoveFirst
Do Until Form2.Data1.Recordset.EOF
If frmDataEnv.Combo2.Text = Form2.Data1.Recordset.Fields("id") Then
Form2.Data1.Recordset.Edit
Form2.Data1.Recordset.Fields("id") = Form2.Text1.Text
```

```
Form2.Data1.Recordset.Fields("name") = Form2.Text2.Text
Form2.Data1.Recordset.Fields("sex") = Form2.Combo1.Text
Form2.Data1.Recordset.Fields("age") = Form2.Text4.Text
Form2.Data1.Recordset.Fields("address") = Form2.Text6.Text
Form2.Data1.Recordset.Fields("symptoms") = Form2.Text3.Text
Form2.Data1.Recordset.Fields("ward") = Form2.Text7.Text
Form2.Data1.Recordset.Fields("date admitted") = Form2.Text5.Text
Form2.Data1.Recordset.Fields("treatment") = Form2.Text8.Text
Form2.Data1.Recordset.Fields("dept") = Form2.Text12.Text

Form2.Data1.Recordset.Update
Form2.Text1.Text = ""
Form2.Text12.Text = ""

Form2.Text2.Text = ""
Form2.Text3.Text = ""
Form2.Text4.Text = ""
Form2.Combo1.Text = ""
Form2.Text5.Text = ""
Form2.Text6.Text = ""
Form2.Text7.Text = ""
Form2.Text8.Text = ""

Exit Do
```

```
Else
Form2.Data1.Recordset.MoveNext
End If
Loop
End Sub

Private Sub Command5_Click()
Form2.PrintForm
Printer.EndDoc
End Sub

Private Sub pay_Click()
On Error Resume Next
frmDataEnv.Combo3.Clear
Form2.Data1.Recordset.MoveFirst
Do Until Form2.Data1.Recordset.EOF
frmDataEnv.Combo3.AddItem Form2.Data1.Recordset.Fields("id")
Form2.Data1.Recordset.MoveNext
Loop
frmDataEnv.Frame3.Visible = True
frmDataEnv.Text3.Text = ""
frmDataEnv.Text4.Text = ""
frmDataEnv.Text5.Text = ""
```

```
End Sub

Private Sub payment_Click()
On Error Resume Next
frmDataEnv.Combo1.Clear
Form2.Data1.Recordset.MoveFirst
Do Until Form2.Data1.Recordset.EOF
frmDataEnv.Combo1.AddItem Form2.Data1.Recordset.Fields("id")
Form2.Data1.Recordset.MoveNext
Loop
frmDataEnv.Option2.Value = True
frmDataEnv.Frame1.Visible = True
frmDataEnv.Frame1.Caption = "Bill Payment"

frmDataEnv.Text1.Text = ""
frmDataEnv.Text2.Text = ""

End Sub

Private Sub usd_Click()
Dim ht As String
ht = MsgBox("This software is developed by ", vbOKOnly, "Users Guide")

End Sub
```

```
Option Explicit

Public LoginSucceeded As Boolean

Private Sub cmdCancel_Click()

    LoginSucceeded = False
    Me.Hide
    End
End Sub

Private Sub cmdOK_Click()
 '---

      frmLogin.Data1.Recordset.MoveFirst
Do Until frmLogin.Data1.Recordset.EOF
      If txtPassword = frmLogin.Data1.Recordset.Fields("Password") Then
        LoginSucceeded = True
        Me.Hide
         frmSplash.Show

    Exit Do
    End If
```

```
  frmLogin.Data1.Recordset.MoveNext

Loop

    If LoginSucceeded = False Then
        MsgBox "Invalid Password, try again!", , "Login"
        txtPassword.SetFocus
        SendKeys "{Home}+{End}"
     End If

End Sub

Private Sub Timer1_Timer()
frmDataEnv.Show
frmSplash.Hide
frmSplash.Timer1.Enabled = False
End Sub
```