USING SNOMED CT TO DEVELOP AN ELECTRONIC HEALTH RECORD SYSTEM FOR SURGERY DEPARTMENT: A CASE STUDY OF THE UNIVERSITY TEACHING HOSPITAL ZAMBIA

BY

DANNY LEZA

A Dissertation Submitted to the University of Zambia in Partial Fulfillment of the Requirements for the Award of the Degree in Masters of Computer Science

THE UNIVERSITY OF ZAMBIA

LUSAKA

2019

COPYRIGHT

DECLARATION

I, **Danny Leza**, do hereby declare that this work has not previously been submitted in candidature for any degree. The dissertation is the result of my own work and investigations except where otherwise stated. Information from other sources has been duly acknowledged. A complete list of references is appended.

Signature…….. **Date**…….

Computer Number: 2015078876

Supervisor: **Dr. Jackson Phiri**,

Signature :……. Date: ……..

CERTIFICATE OF APPROVAL

This dissertation by **Danny Leza** is approved as fulfilling part of the requirements for the award of the degree of Master of Computer Science of the University of Zambia.

Examiner 1: ……..…… Signature:…… Date:…….

Examiner 2: ……..…… Signature:…… Date:…….

Examiner 3: ……..…… Signature:…… Date:…….

ABSTRACT

The University Teaching Hospital is an integral national referral Hospital made up of eight departments. Systems interoperability is key for successful flow of patient information from one department to another and from section to section within a department. In this study, the Systemized Nomenclature of Medicine-Clinical Terms (SNOMED CT) is proposed to develop an Electronic Health Record System for the Surgery Department at UTH. SNOMED CT is widely recognized as the leading global clinical terminology standard which is used to develop interoperable Electronic Health Records (E.H.R). Lack of a SNOMED CT E.H.R System in surgery departments causes inefficient scheduling of surgical procedures, insufficient and inaccurate pertinent patient historical information, misconceptions and error arising from ambiguities in terminology usage. The result is unhealthy clinician working environment leading to high death rates among patients. The Study aim at examining the patient record system being used at UTH and its drawbacks. Further the study sort to model and develop a prototype as a solution to the challenges. Baseline survey in which interviews, questionnaires and record inspection were used to give a full picture of the situation on the ground. Convenient sampling was used with a sample size of 40. The data from the questionnaires were analyzed using a statistical package called SPSS. The results from the Quantitative data analysis, Interviews and Record inspection were used to come with the requirements for the design of the SNOMED CT E.H.R Model. The SNOMED CT Model was designed using Traditional Systems Development Life Cycle with prototyping. Further Universal Modelling Language and User Case Diagrams were used to model the Objects and User Interactions. The SNOMED CT prototype was developed using Java and MySQL for the Database. Baseline Survey Reviewed that out of 40 respondents 72.5% had computers in their section 27.5% did not have, 60% were using Paper Records and Microsoft Excel, 37.5% were using Paper Based Records, 2.5% were using electronic record system. Further, more than 50% of the medical practitioner ranging from nurses to surgeon were dissatisfied with the manual paper based system. In addition, Records Inspected showed that paper based system had redundancy. Records of patients are destroyed every after ten to fifteen years in order to create space for new ones. The Model as a solution was designed incorporating Snomed CT as a Sematic Standard to aid interoperability. A prototype was developed in which a link to Snomed CT International Browser was embedded and an offline version of Snomed was added using a system called Clinic Clue Xplore. Prototype is able to capture patient details, scheduled patients for surgery, capture pre-surgery details and enter post-surgery details. All diagnosis and prescription have a Snomed CT concept ID and its description. It is recommended that further development be done to interconnect the prototype to other systems in the department.

Key Words: Electronic Health Records, Snomed CT, Interoperability, Paper Based Records, Standardization.

ACKNOWLEDGEMENT

First, I thank God almighty the giver of life and wisdom for enabling me to have the opportunity, strength, courage and understanding to undertake this research. Secondly, but very importantly, many thanks to my supervisor Dr Jackson Phiri for his patience, encouragement, guidance throughout my studies. The motivation he gave was always timely and exact. Lastly, I would like to thank the University of Zambia and University Teaching Hospital Department of Surgery for the support and cooperation given to me during the course of this research.

DEDICATION

This document is dedicated to my wife, daughters, father, mother and brothers and sisters for their support and encouragement.

TABLE OF CONTENT

COPYRIGHT ii
DECLARATION iii
CERTIFICATE OF APPROVAL iv
ABSTRACT v
ACKNOWLEDGEMENT vi
DEDICATION vii
1. CHAPTER 1:INTRODUCTION 1
1.1 Introduction 1
1.2 Introduction to the Research 1
1.3 Background 3
1.4 Statement of the Problem 3
1.5 Aim 4
1.6 Objectives 4
1.7 Research Questions 4
1.8 Significance of the Study 5
1.9 Research Contribution 5
1.10 Organization of the Dissertation 6
1.11 Chapter Summary 6
2. CHAPTER 2: LITERATURE REVIEW 7
2.1 Introduction 7
2.2 Human Health 7
2.2.1 Information Technology and Health 7
2.3 Record Management Systems 8
2.3.1 Disadvantages of Paper Based Records 9
2.3.2 Patient Record 10
2.4 Definition of E-health 11
2.4.1 Essential Elements of E-health 11
2.5 Electronic Health Record 12
2.5.1 Definitions of an Electronic Health Record System 12
2.5.2 Advantages of Electronic Health Records 15
2.6 Implementation of Records Systems in Hospitals 16
2.6.1 Examples of Electronic Health Records in Developing Countries 17
2.6.2 Potential Benefits of Electronic Health 18
2.7 Systems in at UTH 19

2.8 Interoperability 20
2.8.1 Healthcare Interoperability 20
2.8.2 Standards 21
2.8.3 Logical Observation Identifiers, Names, and Codes (LOINC) 27
2.8.4 Health Level 7 (HL7) 28
2.8.5 Example Application of Snomed CT for Sematic Interoperability 28
2.9 Design and Development of an E.H.R 30
2.9.1 Data model 30
2.9.2 Architecture 32
2.10 Example Designs 39
2.11 System Development 43
2.11.1 Software development stages 43
2.12 Review of Software Development Tools 56
2.13 Chapter Summary 58
3. **CHAPTER 3: METHODOLOGY** 59
3.1 Introduction 59
3.2 Baseline study 59
3.3 Study Setup 59
3.4 Summary of the baseline survey 60
3.5 Ethical Consideration 60
3.6 Limitations of the Baseline Study 60
3.7 System Automation 61
3.8 Current Business Process OR Patient Flow 61
3.9 System Architecture 62
3.10 Functional Specification 64
3.10.1 Requirement determination 64
3.11 Nonfunctional Requirements 68
3.11.1 System Modelling and Design 69
3.11.2 System Implementation 87
3.11.3 Limitations of the Prototype Developed 89
3.12 Chapter Summary 89
4. CHAPTER 4: RESULTS 90
4.1 Introduction 90
4.2 Baseline Study Results. 90
4.2.1 Survey Sampling and Data collection 90

4.3 Survey Results and Discussion 90
Demographic Findings 91
4.4 Findings from Record Inspection 107
4.5 Development of a Prototype 109
5. CHAPTER 5: DISCUSSION AND CONCLUSION 137
5.1 Introduction 137
5.2 Discussion 137
5.2.1 Baseline study 137
5.2.2 Record Inspection 138
5.2.3 Business Process Mapping 138
REFERENCES 141
Appendices 151
Sample Code 151
Appendix B: Questionnaire 195

LIST OF FIGURES

Figure 2.1: Records Management System 8
Figure 2.2:Paper based records 9
Figure 2.3: Components of EHR 13
Figure 2.4: General data features of an Electronic Health Record 15
Figure 2.5: levels of Interoperability 20
Figure 2.6: Snomed CT structure of terms 25
Figure 2.7: Snomed CT concept, description, description type 26
Figure 2.8: Relationships in Snomed 27
Figure 2.9: Interface Design 35
Figure 2.10: Example system ERD 41
Figure 2.11: Example system login form 41
Figure 2.12: Example system landing page 42
Figure 2.13: Example System admission and discharge form 43
Figure 2.14: Waterfall Methodology diagram 46
Figure 2.15: Prototyping methodology diagram 47
Figure 2.16: Iterative and incremental methodology 48
Figure 2.17: Spiral methodology 49
Figure 2.18: Rapid application 50
Figure 2.19: different languages and their popularity on GitHub 57
Figure 3.1: summary baseline survey method 60
Figure 3.2: Patient flow chart at UTH surgery 61
Figure 3.3: General system architecture 63
Figure 3.4: conceptual working of the system in view of system interoperability and semantic interoperability 63
Figure 3.5: System Level use case diagram 70
Figure 3.6: Administrator use case diagram 71
Figure 3.7: Surgeon Use Case Diagram 72
Figure 3.8: Anesthetist use case 73
Figure 3.9: Doctor GP use Case Diagram 74
Figure 3.10: Receptionist Use Case Diagram 75
Figure 3.11: Nurse Use Case Diagram 76
Figure 3.12: symbols of communication diagram 77
Figure 3.13: Sequence Diagram symbols 78
Figure 3.14: Symbols and description of messages 78
Figure 3.15: Login Communication Diagram 79
Figure 3.16: Login Sequence diagram 79
Figure 3.17: Receptionist View 80
Figure 3.18: Queue Patient 80
Figure 3.19: Doctor attend to patient 81
Figure 3.20: Nurse Sequence Diagram 81
Figure 3.21: Anesthetist sequence diagram 82
Figure 3.22: Snomed CT Electronic Health Record System Class Diagram 83
Figure 3.23: ER Diagram 85
Figure 4.1: Gender of Respondent 91
Figure 4.2: Age of respondent 92
Figure 4.3: Highest level of education 93
Figure 4.4: Job Title 95

Figure 4.5: Length of service 96
Figure 4.6: Existence of Computers 97
Figure 4.7: Level of electronic medical record implementation 98
Figure 4.8: Schedule of patients to be operated on 100
Figure 4.9: Information regarding a particular patient before surgery is readily available during operation procedures 101
Figure 4.10: Availability of information before Surgery 102
Figure 4.11: Privacy and Security rating 103
Figure 4.12: Rating of report generation with the current system 104
Figure 4.13: Level of satisfaction with the current system 106
Figure 4.14: Pictures of Manual Records in Use 107
Figure 4.15: Sample Ms Excel Information System 108
Figure 0.16: JAVA database connection code 109
Figure 4.17: Sample Database Connection 110
Figure 4.18: Login form 111
Figure 4.19: Landing Page for System Administrator 111
Figure 4.20: Administrator Menus 112
Figure 4.21: Hospital Creation 113
Figure 4.22: Create Patient Button 114
Figure 4.23: Patient Bio Data 115
Figure 4.24: Patient's other details 116
Figure 4.25: Capturing the patient image using webcam 117
Figure 4.26: View patient details doctor identification number 117
Figure 4.27: View Patient details filtered by Gender 118
Figure 4.28: View Patient Details by Point of Care 118
Figure 4.29: Generating Reports from patient list 119
Figure 4.30: Queued Patients 120
Figure 4.31: Patient biodata 121
Figure 4.32: Selecting patient 122
Figure 4.33: Searching for a patient 123
Figure 4.34: Doctor Home page 124
Figure 4.35: Doctor attending to patient 125
Figure 4.36: Summary patient details as seen by the doctor 126
Figure 4.37: Attendance Details 127
Figure 4.38: Snomed CT Incorporation in diagnose 128
Figure 4.39: Offline Snomed CT Concept ID Access 129
Figure 4.40: Diagnosis with SNOMED CT Concept ID and Description 130
Figure 4.41: Challenges table form 131
Figure 4.42: Admitting a patient 132
Figure 4.43: Patient admission 132
Figure 4.44: Admission Details 133
Figure 4.45: Attendance list 133
Figure 4.46: Other attendance tab form 134
Figure 4.47: Patient queued up for surgery 135
Figure 4.48: patient attendance record 135
Figure 4.49: Patient attendance record 136

LIST OF TABLES

Table 1: Application of Snomed CT 30
Table 2: methodology, characteristics, strengths and weakness 50
Table 3: Low-Level functional requirements 64
Table 4: low-level functional requirement 66
Table 5: nonfunctional requirements 69
Table 6: Data Dictionary 86
Table 7: Snomed CT standardized Electronic Health Record System hardware components 88
Table 8: Gender of Respondent 91
Table 9: Age Group 92
Table 10: Highest level of education 93
Table 11: Job Title 94
Table 12: length of service 95
Table 13: Existence of computer in the department or section 96
Table 14: level of electronic records implementation 98
Table 15: scheduling of patients for operation 99
Table 16: Information regarding a particular patient before surgery is readily available during operation procedures 100
Table 17: Operating room and patient scheduling 101
Table 18: Rating in terms of privacy and security of patient's medical data 102
Table 19: Rating of report generation with the current system 104
Table 20: Patient scheduling, report generation, level of satisfaction 105

ACRONYMS

ADT	Admission, Transfer And Discharge
AMPATH	Academic Model for Prevention and Treatment of HIV/AIDS
ASTM	American Society for Testing and Materials
CEN TC	Comite Europeen de Normalization – Technical Committee
CPOE	Computerized Provide Order Entry
CPR	Computer-Based Patient Record
CPT	Current Procedure Terminology
DBMS	Database Management System
EHR	Electronic Health Record
EMR	Electronic Medical record
ERD	Entity Relation Diagram
FSN	Fully Specified Name
FTP	File Transfer Protocol
HL7	Health Level Seven
HMIS	Health Management Information System
HTTP	Hyper Text Transport Protocol
ICD	International Classification of Diseases
ICD-9-CM	Clinical Modification
ICT	Information Communication Technology
LAN	Local Area Network
LOINC	Logical Observation Identifiers, Names and Codes
MAR	Medication Administration Record
MS	Microsoft
RIM	Reference Information Model
RIS	Radiology Information System
SNOMED CT	Systematized Nomenclature of Medicine—Clinical Terms
UPS	Interruptible Power Supply
UTH	University Teaching Hospital
WAN	Wide Area Network
WHO	World Health Organization

CHAPTER 1

1.1 Introduction

This chapter presents the background to this study on the Electronic Health Record Management System at the University Teaching Hospital in Lusaka Zambia. We begin by looking at a brief introduction to the research, background, scope, statement of the problem, aim, objectives, significance of the study and research contribution. The chapter concludes by looking at the organization of the thesis and the summary of the chapters.

1.2 Introduction to the Research

As time goes by, medical care is getting more multifaceted and as new technologies are discovered, there is a need for the medical team to come up with better structures of maintaining the patients' information. Proper and accurate documentation comes in hand in hand with better medical care and implementation policies. The electronic medical record (EMR) is one of the medical tools that seek to improve medical care by providing hospitals with the kind of platform that allows for new services and new functionality [1] . The patient information can then be updated as the patient undergoes new treatment and newer health information is discovered.

According to NIH NCRR [2], the Electronic Health Record (EHR) is a compiled report of all of a patient's health information that includes the patient's demographics, progress notes, problems, medications, vital signs, past medical history, immunizations, laboratory data, and radiology reports. The EHR computerizes this information in an organized manner in which the patient has acquired health care. This is a very important tool in the provision of evidence-based care of a patient and it incorporates different health care departments to ensure an effective and comprehensive health record.

The patients' information also requires being secure and available in the computerized file for future references. All the medical personnel must be able to assess and understand the information in the patient's file to ensure that the patient undergoes proper treatment and to lessen the workload of having to ask the patient for his basic health information each time he visits the hospital [3].

A computerized medical record brings with it many advantages. It presents data in a very organized manner so that each hospital department finds the required information without difficulties. This changes the way health care is practiced in that it is very unlikely to overlook important findings.

The success implementation of Electronic Health Records provides another challenge when it comes to common definition of medical terms in which one term can mean different things to different people. The solution is applying a standard in the storage and retrieval of medical records. SNOMED CT is an international standard that can be implement together with an E.H.R.

The primary purpose of SNOMED CT is to code the meanings that are used in health care delivery and support the clinical recording of health information. SNOMED CT contributes to the improvement of patient care by underpinning the development of Electronic Health Records that record clinical information in ways that enable meaning-based retrieval. This provides effective access to information required for clinical decision support and consistent.

In this study, we are proposing the Systemized Nomenclature of Medicine-Clinical Terms (SNOMED CT) to develop an Electronic Health Record System for the Surgery Department at UTH. SNOMED CT is widely recognized as the leading global clinical terminology standard which is used to develop interoperable Electronic Health Records (E.H.R). Clinical terminologies are highlighted in the scientific literature as a key factor for improving communication of clinical data and increase availability of relevant information for the various stakeholders within the health sector.

Clinical terminologies have potential to support development and configuration of Electronic Health Records or Clinical Information Systems (CISs) that enable semantic interoperability and support efficient and effective data entry and retrieval [4]. Various definitions of Semantic interoperability in health exist.

“semantic interoperability means ensuring that the precise meaning of exchanged information is understandable by any other system or application not initially developed for this purpose .” [4]

Hence a prerequisite in achieving Semantic Interoperability is standardized concept terminology Standard (like SNOMED CT) because they are the mean to representing clinical and medical terms meaning unambiguously. E.H.R are configured to transform the clinicians’ documentation needs and requirements for functionality into templates which best support the clinical practice [5]–[7].

1.3 Background

The University Teaching Hospital is the highest referral hospital in Zambia and the oldest medical training school in Zambia. It is divided into two categories of departments. It has the administrative departments and clinical care departments. The surgery department is the biggest of the of the eight clinical departments [8].

The Department of Surgery has been an integral part of The University Teaching Hospital since the time it was established, by then called Lusaka Central Hospital. It is currently the largest department and with the most diverse specialized units such as; General surgery, Othorpaedic and Trauma, Ophthalmology, Urology, Paediatric Surgery, Otorhinolaryngology (ENT), Cardiac, Laparoscopy, Neurosurgery, Maxillofacial [8]

The department has casualty unit, which are entry point for most surgical patients. It also functions as emergency and disaster management unit. They are later channeled to either surgical admission wards for those who need urgent attention or the various specialized clinics appropriately.

The department also boasts of having the largest theatre, which caters for emergency operations every day of the week and elective operations only during weekdays by the individual specialized unit.

Although not enough, the large proportion of surgical patients is admitted to G-block. Others with bone, ENT and maxillofacial problems are kept in C-Block.

The department also has number of outpatient clinics, which handle a patient care after discharge from operation theatre [8].

As is the case with many African countries faced with many challenges among them poverty, also lack of robust healthcare infrastructure in the form of information and Communication technology (ICT) to ensure continuity of patient health care which many researchers considered a lifesaving resource [9].

1.4 Statement of the Problem

In a delicate field like medicine were patient care is critical, it is important to reduce misconceptions and error which may arise from ambiguities in terminology usage. Delay arise in patient care especially in surgery operations which require clear patient information (patient medical history) before proceeding with surgical procedures [10].

Patient Records which are held in both manual and sometimes in systems like Health Management Information System (HMIS) [11] and Smartcare [12] for patient on Antiretroviral treatment are not standardized to aid interoperability. This challenge delays patient care because clinicians have to carry out interdepartmental communication to make clarifications in the cases where a patient's record has missing information, which is vital for prescription and surgical procedure. Patients are subjected to long waits just so that clarifications can be made before treatment is administered. This scenario creates a difficult working environment for clinicians because which lead to frustration.

1.5 Aim

To assess the system being used to manage patient's records at UTH Surgery Department and apply SNOMED CT as a Standard in developing an E.H.R system meant to mitigate the drawbacks of the system in use.

1.6 Objectives

The following objectives will guide the research.

1. To investigate and assess the patient record systems in use at the University Teaching Hospital.
2. Based on the findings of (1) to develop a model using SNOMED CT as a standard for UTH Electronic Health Record (E.H.R) System.
3. Develop an E.H.R prototype based on the proposed model in (2).

1.7 Research Questions

In order to achieve the above stated objectives, the following research questions will have to be answered:

1. What are the major challenges faced by the stake holders when dealing with the current health system?
2. How can we develop a model for the UTH E.H.R System that is based on SNOMED CT standardization?

3. Is it possible to develop an E.H.R system prototype for the UTH based on the Model in (2)?

1.8 Significance of the Study

The significance of the study is realized in the benefits it will give by providing a standardized means to manage patient records among them being:

- The efficient scheduling of surgical procedures in the operating room suites to maximize the use of each suite, while accommodating the surgeons' requests for a specific date, time, routine supplies, and any special instruments and/or supplies required for the procedure.
- Tracking surgeons' preferences and average times for procedures will be made easier and accurate.
- It will provide easily accessible and accurate historical information for average times, this will reduce delays in the schedule and help to improve on patient care and reduce costs.

Report generation, easy access to information by different specialists in the Department of Surgery of UTH will be one of the significant contribution that the study will provide. The proposed E.H.R System will also function as a source of critical data needed to carry out research. It study will contribute information to the academic body of knowledge through the papers to be published.

1.9 Research Contribution

The research has highlighted the main challenges faced by the clinician at the largest Hospital in the country. The Surgery department at the UTH is the largest, which acts as the national referral center for cases, which cannot be handled by other hospitals. The research reviewed the system used for recording keeping and its lack of inter-operable standards, which befit an E.H.R.

An Electronic Health Record System was designed and developed incorporating Snomed CT as a sematic standard for interoperability.

The system will be installed and piloted at the surgery department of the University teaching hospital.

1.10 Organization of the Dissertation

The study includes the following chapters: Chapter 2 discusses the background of the study and problem statement. Chapter 2 discusses the literature review and related works. Chapter 3 highlights the determination of the research design and methodology of both the baseline study and system automation. Chapter 4 is the presentation of the findings of the baseline study and the system implementation and testing. Finally, Chapter 5 discusses the findings, draws conclusion and recommendations are given.

Appendices are given at the end of chapter five, which includes sample code and the questionnaire.

1.11 Chapter Summary

This chapter presented the introduction to the research, background, statement of the problem, aim of the study, the objectives of the study, research questions, motivation, significance and scope of the work.

The next chapter will focus on evaluating the significance of the study in relation to the related literature from studies that were conducted responding similar challenges arising from the need computerize hospital patient records.

CHAPTER 2: LITERATURE REVIEW

2.1 Introduction

This chapter is a presentation of literature reviewed from different sources records management to electronic health record system. The Chapter is organised by looking at Human health, Information technology and its contribution to health systems, record keeping, Electronic health systems, interoperability and standard and implementation of electronic health systems.

2.2 Human Health

The World Health Organization (WHO) defined health in its broader sense in its 1948 constitution as "a state of complete physical, mental, and social well-being and not merely the absence of disease or infirmity." [13] This term has been defined elsewhere as “Health is the level of functional and metabolic efficiency of a living organism. In humans it is the ability of individuals or communities to adapt and self-manage when facing physical, mental, psychological and social changes with environment.” [14]. Healthy population is lives longer and contributes to the development of the nation. It therefore goes to say healthy is wealthy [15].

It is for this reason that most countries have prioritized investment in health care facilities. This is evidenced by the budget allocation towards the area under discussion [16].

2.2.1 Information Technology and Health

Information Communication Technologies have been seen to drive business growth globally [17], hence the growth of global spending in ICT which is forecasted at $4.8 Trillion by the end of 2018 [18]. If the business community have benefited from the growth in the ICT sector, then the healthy sector can make positive strides once ICTs have been put to use [19].

Studies have shown that the use of ICT in the health sector is capable of increasing efficiency, reducing errors, supporting more team-based care, improving integration of best practice into routine care, enabling consumers to engage more actively in their care, and producing more efficient services through changes in professional roles and responsibilities [20]. The ICT infrastructure required to revolutionaries the business processes in a particular health sector are cheaper to acquire if the cost is compared to the benefits which would accrue [21].

Information and Communication Technology (ICT) is revolutionizing our life, our ways to interact with each other, and day-to-day life and work. Its application in health is described broadly as eHealth, which includes telemedicine, electronic medical records, electronic health records and health information systems with decision support, mobile health and eLearning tools. EHealth has shown potential in facilitating a better health care delivery system, leading to better health and universal health coverage. It creates access, enhances quality, improves primary health care interventions and can act as a solution for situations where human resources for health are scarce [22].

2.3 Record Management Systems

A Record Management System manages Physical and Electronic documents and records from their creation, use, storage, and maintenance to eventual destruction or permanent preservation while retaining integrity, authenticity and accessibility of corporate records. The application manages the complete life cycle of both Physical and Electronic records as per Records Management policies laid down by the organization and those required by law [23]. Figure 2.1 shows the life cycle of a record from creation to archiving or destruction.

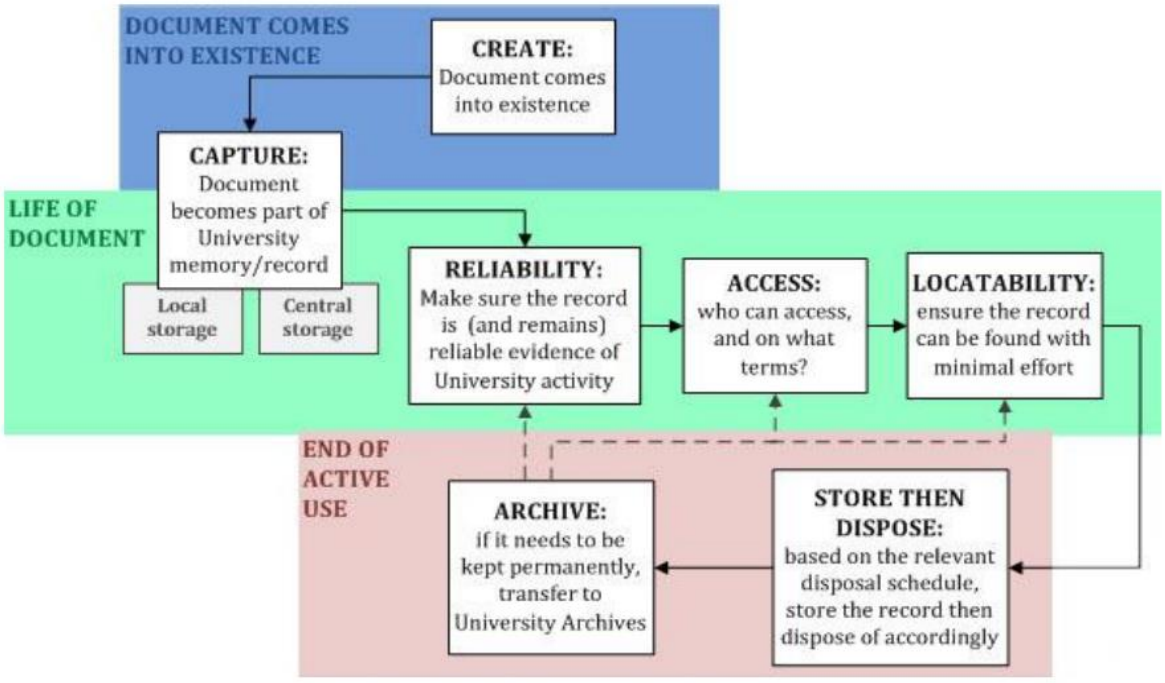

Figure 2.1: Records Management System (source: https://www.adelaide.edu.au/records/images/rmo_scheme.jpg)

A record can exist in two forms. It can be in electronic format or manual form. Many institutions have kept record in manual form and this make them suffer from the many drawbacks of the manual systems.

Figure 2.2:Paper based records(source: https://www.fotosearch.com/BLD169/blm019420/)

2.3.1 Disadvantages of Paper Based Records

They are very difficult to search compared to electronic records. The system of using tabs and indexes is subject to human in accuracies.

1. Paper based records only have one way of phrasing a condition, treatment, or issue. Abbreviation, alternate emphasis, or obvious generalization of the terms, are not recorded. This because it is very tedious to write all metadata or synonyms about a record.
2. The records have to be managed by a person and the physical space they occupy. If the records are too large, additional expense must be budgeted to pay for off-site storage. The person managing records bring about additional personnel expenses. There is only one copy of a paper record. This means only person can use it per time and in an event of loss, then all the valuable information it contains is lost. If several people separate the record for use concurrently, it must be merged again, or it will not be available the next time it is needed.

3. A competent filing person must be employed to put the Records the proper place so they can easily be retrievable. Alternate indices such as a card catalog, must be created and maintained. This requires training expenses, recruitment and retainment expenses, as well as salary and benefits for that person.
4. Accurate summary information must be accumulated through an exhaustive review of all the Records.
5. Any backup copy of the Records must be made by hand or by automatic copier (with the associated expenses of a copy machine), and physically transported to a safe archival location.
6. Paper Records are susceptible to unplanned destruction, whether by rodents, flooding, fire, etc. Planned destruction must be done securely and involves environmentally unfriendly shredding or incineration. Paper Records are not compact enough to be physically moved by a work force.

In view of the disadvantages of manual records, it is very important that records are automated. Having looked at a record in general, attention is now given to a patient's record.

2.3.2 Patient Record

A patient record is the repository of information about a single patient [24]. Health care professionals generate this information as a direct result of interaction with a patient or with individuals who have personal knowledge of the patient (or with both). Traditionally, patient records have been paper and have been used to store patient care data. As earlier outlined, this makes a patient's record susceptible to the pitfalls of manual records. Hence, the need to have a computer based patient record.

A computer-based patient record (CPR) is an electronic patient record that resides in a system specifically designed to support users by providing accessibility to complete and accurate data, alerts, reminders, clinical decision support systems, links to medical knowledge, and other aids [25].

Health care professionals while providing patient care services to review patient data or document their own observations, actions, or instructions use a primary patient record.

A secondary patient record is derived from the primary record and contains selected data elements to aid nonclinical users (i.e., persons not involved in direct patient care) in supporting, evaluating, or advancing patient care. Patient care support refers to administration, regulation, and payment functions [26]. Patient care evaluation refers to quality assurance, utilization review, and medical or legal audits. Patient care advancement refers to research. These records are often combined to form what is terms a secondary database (e.g., an insurance claims database).

A patient record system is the set of components that form the mechanism by which patient records are created, used, stored, and retrieved. A patient record system is usually located within a health care provider setting. It includes people, data, rules and procedures, processing and storage devices (e.g., paper and pen, hardware and software), and communication and support facilities [27].

2.4 Definition of E-health

E-health is an umbrella term used to refer to the implementation and use of information technologies within a health care system. It is often defined as information and communication technology (ICT) within an eco-system comprising patients and other stakeholders that deliver health services [10].

2.4.1 Essential Elements of E-health

According to [28], e-health has 10 essential elements or "10 essential Es". These are: efficiency, enhancing quality of care, evidence-based health, empowerment of consumers, encouragement of a new patient–health professional relationship, education of physicians, enabling health care

Information exchange and communication, extent of the scope of health care, ethics and equity.

Efficiency: The avoidance of unnecessary diagnostics and therapy with the more active involvement of the patient [28].

Enhancement of the quality of health care provision means the patients have the power to choose between different health care providers and specialists who should be evaluated according to their quality and performance [28].

Evidence-based health means that proven science and evaluation must form the basis of eHealth interventions [28]. It is often difficult to prove the effectiveness of the role of e-health in long-term

health conditions. Empowerment refers to patient-centered health care, patient choice, collaboration, quality, participation and knowledge with better access to patient records and health care information on the Internet. These elements mean that the patient has more responsibility for their own health and is expected to make healthier decisions [28].

Encouragement of the patient–health professional relationship develops shared decision making and evens the power balance between the two roles [28]. Education of health professionals and patients is essential. Physicians can access online medical education and patients (or consumers) can obtain individualized health education alongside prevention information [28]. Exchange and communication: Standardized interoperability enables better communication between different health care establishments [28]. Extending the scope of health care: This means patients can use the Internet to access and obtain health services on a global level including advice and information regarding pharmaceutical products [28].

2.5 Electronic Health Record

The electronic health record (EHR) has been considered a critical and important application of E-health systems [29][30].

2.5.1 Definitions of an Electronic Health Record System

Electronic health record (EHR) systems record health-related information on an individual so that it can be consulted by clinicians or staff for patient care. One formal definition of an EHR is "an electronic version of a patient's medical history, that is maintained by the provider over time, and may include all the key administrative and clinical data relevant to that person's care under a particular provider, including demographics, progress notes, problems, medications, vital signs, past medical history, immunizations, laboratory data, and radiology reports" [27].

2.5.1.1 Components of an electronic Health Record System

The basic key components of an EHR include Patient Management, Clinical component, Laboratory Component, radiology Component and Billing. Figure 2.3 show components which constitutes an Electronic Health Record System.

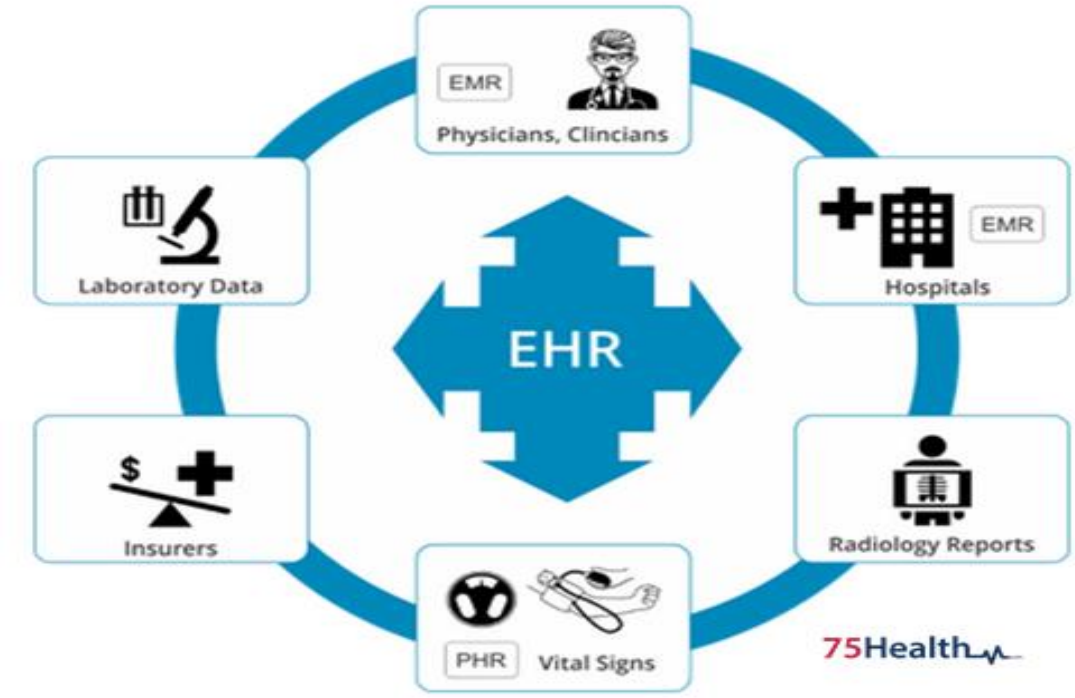

Figure 2.3: Components of EHR (source:http://blog.75health.com/what-components-constitute-an-electronic-health-record/)

1. **Patient Management Component**:

This component is required for patient registration, admission, transfer and discharge (ADT) functionality. Patient registration includes key patient information such as demographics, insurance information, contact information etc. When a patient is registered in an EHR for the first time, a unique ID (often called "Medical Record Number") is generated. Whenever a patient has an encounter with the organization, another unique "encounter" number is generated.

2. **Clinical Component:**

This component can consist of multiple sub-components like Computerized Provide Order Entry (CPOE), electronic documentation, nursing component among others.

CPOE allows providers to enter orders that are needed for patient management directly in the computer. This component can make use of clinical decision support tools such as drug-drug, drug-allergy, and drug-diagnoses interactions. This module also allows providers to enter multiple orders from order sets.

Electronic documentation by providers allows them to document notes such as History & Physical, consults, discharge summaries, operative notes etc. Multiple tools may be used to enable electronic documentation such as templates, speech recognition and transcription services.

The pharmacy system allows for maintaining a drug formulary, filling prescriptions and cross-checking any orders that are placed by providers in the EHR.

Nursing component allows for collection of key patient information such as vital signs, input and output among others. This component also allows for medication administration record (MAR), barcode medication charting and nursing documentation.

3. **Laboratory component:**

Laboratory components are divided into two subcomponents;

1) Capturing results from lab machines, and
2) Integration with orders, billing and lab machines. The lab component may either be integrated with the EHR or exist as a standalone product.

4. Radiology Information System:

Radiology information system (RIS) and Picture Archiving & Communications System (PACS) are used to manage patient workflow, ordering process, results and the images themselves.

5. Billing System

The billing system (hospital and professional billing) is used to capture all charges generated in the process of taking care of patients. These charges generate claims, which is submitted to insurance companies, tracked and completed.

Overlaid on top of the core application layers is generally a data layer, which is fed data by the EHR. This data layer allows healthcare professionals to monitor Key Performance Indicators, view dashboards with relevant business data, and run analytical reports to monitor and improve the performance of the health care organization. One of the successes of using the data captured by EHR's is the ability to track organizational expenses, inventory, and revenue cycle performance. These basic tasks were very complicated in the pre-EHR era.

One of the biggest challenge today around managing electronic health record data is in obtaining clean, discreet data that can be used for analytics. Natural language processing and other tools are being developed to solve some of these problems.

We are in the early years of using this technology in healthcare. Organizations are trying to rebuild paper processes in the EHR, the government regulations are suppressing innovation, and this is leaving healthcare providers frustrated and disenfranchised. It will be a long time before healthcare is able to reengineer it's processes and adopt this technology to its fullest.

All the above outline components have a special storage location. Figure 2.4 is component based EHR structure.

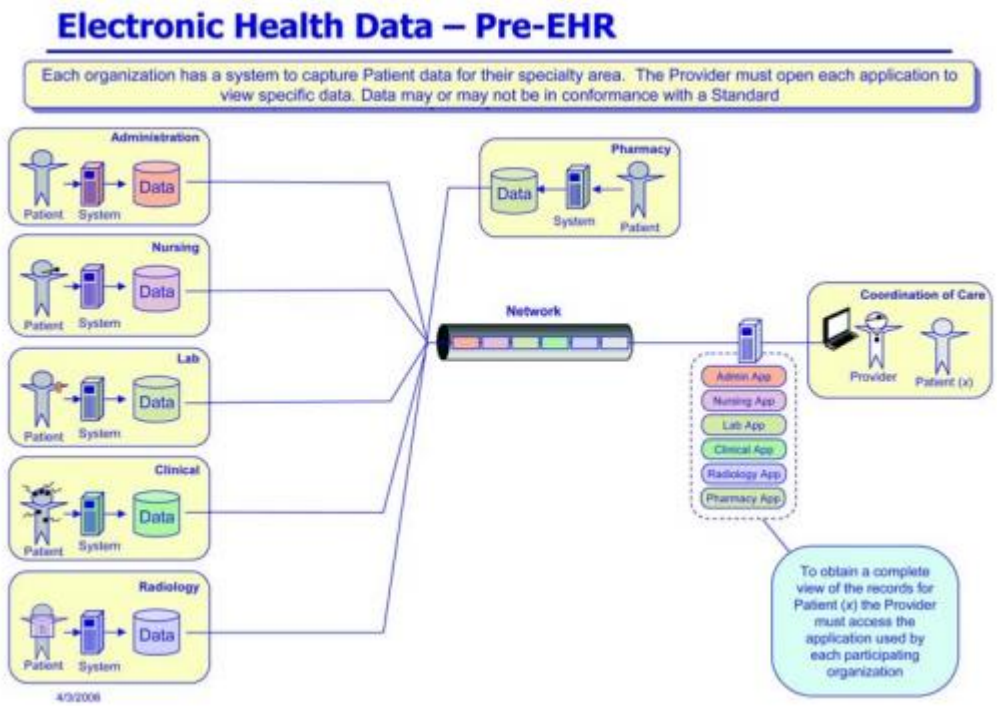

Figure 2.4: General data features of an Electronic Health Record (https://s3.amazonaws.com/rdcms-himss/files/production/public/HIMSSorg/Content/files/Code%20180%20MITRE%20Key%20Components%20of%20an%20EHR.pdf)

2.5.2 Advantages of Electronic Health Records

EHRs help providers better manage care for patients and provide better health care by [31]:

- Providing accurate, up-to-date, and complete information about patients at the point of care.

- Enabling quick access to patient records for more coordinated, efficient care.
- Securely sharing electronic information with patients and other clinicians.
- Helping providers, more effectively diagnose patients, reduce medical errors, and provide safer care.
- Improving patient and provider interaction and communication, as well as health care convenience
- Enabling safer, more reliable prescribing
- Helping promote legible, complete documentation and accurate, streamlined coding and billing
- Enhancing privacy and security of patient data
- Helping providers improve productivity and work-life balance
- Enabling providers to improve efficiency and meet their business goals
- Reducing costs through decreased paperwork, improved safety, reduced duplication of testing, and improved health.

2.6 Implementation of Records Systems in Hospitals

Many countries are strategically pushing the agenda of implementing EHR through incentives. This is because in most of the developed countries particularly in the USA, a lack of adoption of Electronic Health Record is interpreted as a deviation from standard of care [32] .

World over there has been an increase in adoption and implementation of Electronic Health Record System.

According to the[32] , the top 10 countries with the highest EHR adoption rates are:

Norway (98 percent), Netherlands (98 percent), United Kingdom (97 percent), New Zealand (97 percent), Australia (92 percent), Germany (82 percent), United States (69 percent), France (67 percent), Canada (56 percent), Switzerland (41 percent).

Implementation of Electronic Health Record in developing countries has taken shape despite the many challenges faced. In developing countries, healthcare information systems have been driven mainly by the need to report aggregate statistics for government or funding agencies [33]. Such data collection can be performed with simple paper forms at the clinic level, with all electronic

data entry done centrally, but that approach tends to be difficult and time-consuming and may provide little or no feedback to the staff collecting data.

2.6.1 Examples of Electronic Health Records in Developing Countries

Electronic Health Record systems (EHRs) are increasingly being implemented in resource poor countries, with systems like OpenMRS currently being used in over 15 different African countries [34].

In Kenya [35], an Electronic Health Record called Mosoriot Medical Record System(MMRS) was adopted. This System which was implemented in 2001, and later adapted to support a project called Academic Model for Prevention and Treatment of HIV/AIDS (AMPATH) and the system was renamed as AMR. The design is such that it has two networked computers running Microsoft Access, powered by a UPS with solar battery backup. The System was later expanded to seven networked computers linked to a single MS Access database. The database has 60000 patient's records and over 150 000 visits records in four years.

Data entry: In the MMRS, patients are registered in the system upon arrival, travel through the clinic with a paper visit form, and present the visit form as they depart. Clerks perform the registration and transcribe visit data. AMRS data are collected on paper forms at each visit, delivered to a central location for data entry, and then returned to the patient's paper chart. Functions: MMRS provides both patient registration and visit data collection functions. Data are collected on all patients seen in the medical clinic, including their laboratory results and medications. AMRS supports comprehensive HIV care as well as mother-to-child transmission prevention, while serving as a rich database for quality improvement and answering research questions. Pharmacy management: Based on drug regimens analysis available.

In Haiti [35], A System called HIV-EMR was developed to cater for seven public health clinics. The system is Open source web system backed by an Oracle database with an additional offline client for data entry and review. Functionality includes: History, physical examination, social circumstances and treatment recorded. Decision support tools provide allergy and drug interaction warnings, and generate warning emails about low CD4 counts. An offline component of the EMR was developed to overcome unreliable internet communications in some sites.

In Uganda [35], system called Careware was developed by a team based at the US Department of health and Human Services. It is designed as a standalone database built with MS Access. Provides comprehensive tools for tracking HIV patients and their treatment, including clinical assessment, medications and billing data. It is widely used in health centers and hospitals in the US, and has recently been internationalized and deployed in Uganda in October 2003. Pharmacy management: Drug inventory support in international version.

In Malawi [35], an EMR, was implement at Kamuzu Central Hospital located in Lilongwe. Malawi has made extensive use of a touchscreen patient management information system for a wide range of clinical problems in the 216-bed paediatric department since 2001. The system is designed to runs over a local area network built on Linux/ MySQL with Visual BasicTM for the client programs. Functionality, Data are collected on patient demographics, medication, laboratory tests and X-rays. A potential limitation of the touch screen approach is that it is difficult to enter free text, though an 'on-screen' keyboard is available and has been used by local staff to enter all the patients' names. Pharmacy management: Recording of regimens only. Significance: The extensive use of this system directly by healthcare workers in a poor country with limited IT skills is a convincing demonstration of the potential of EMRs with user-friendly data entry mechanisms.

2.6.2 Potential Benefits of Electronic Health

EHR bring about great potential benefits and a high likelihood of outcome. Among them are:

- The reduction of costs achieved through the reduction in duplication of services and the reduction in the number of personnel through computerization of manual services and automation of coding [36] .
- EHRs improve quality of care due to diminished medical errors by providing healthcare workers with decision support systems. They also promote evidence-based medicine by providing access to unprecedented amounts of clinical data for research that can increase the level of knowledge of effective medical practices [37][38].
- EHRs improve the efficiency and effectiveness with which patient care services are delivered by clinicians. They allow for simultaneous remote access to patient data, legibility of records, safer data storage, patient data confidentiality, flexible data layouts, and continuous data processing [14][39].

- EHRs are more reliable due to the presence of a good backup system for disaster recovery.

  Patient satisfaction is enhanced through the smooth handling of referrals, reduction of the need for multiple tests, ease in accessing results and detection of serious health threats that may be life threatening [40].

In view of the aforementioned, benefits of ICTs and particularly EHRs, it is imperative to look at the use of manual records in health care management focusing on the University Teaching Hospital Lusaka.

2.7 Systems in at UTH

The University Teaching Hospital (UTH) is an integral national referral hospital made up of eight departments. The Department of Surgery has been an integral part of The University Teaching Hospital since the time it was established, by then called Lusaka Central Hospital. It is currently the largest department and with the most diverse specialized units such as; General surgery, Orthopedic and Trauma, Ophthalmology, Urology, Pediatric Surgery, Otorhinolaryngology (ENT), Cardiac, Laparoscopy, Neurosurgery, Maxillofacial, Infrastructure and patient care;

The department has casualty unit, which are entry point for most surgical patients. It also functions as emergency and disaster management unit. They are later channeled to either surgical admission wards for those who need urgent attention or the various specialized clinics appropriately [8].

Currently, there are two major systems, which are used for patients' records, and for statistical information and these are Smartcare and DHMIS.

Smartcare is an Electronic Health Record (EHR) System used for management of client health records, generation of reports and in auxiliary services such as pharmacy, labs, logistics and user and provider management. Smartcare is generally considered as being made up of 3 subsystems which are: the trained and certified users, the software system and computers and other physical infrastructure that supports use of the system [41][42].

The rationale for DHMIS has been that the availability of operational, effective and efficient health management information systems is an essential component of the required district management capacity. The logic is that effective and efficient HMIS will provide district health managers with the information required to make effective strategic decisions that support district performance and

sustainability in these decentralized health systems[43]. This system is mainly focused on the management information part. It has been in use at the University Teaching Hospital as well as in many of the Government Hospital through Zambia. The system is an Open source and many countries of the World are using it [44]. Its main use is for statistical data for manager to help them make decision. These two systems cannot adequately solve the problems associated with the surgery department at the University Teaching Hospital in Lusaka.

2.8 Interoperability

2.8.1 Healthcare Interoperability

In healthcare, interoperability is the ability of different information technology systems and software applications to communicate, to exchange data accurately, effectively, and consistently, and to use the information that has been exchanged [45]. Figure 2.5 shows the four levels of systematic interoperability.

Systematic Interoperability Taxonomy

Level	Description	Examples
1	Non-electronic data	No PC/information technology
2	Machine-transportable data	Fax/Email
3	Machine-organizable data	Structured messages, non-standard content/data
4	Machine-interpretable data	Structured messages, standardized content/data

Figure 2.5: levels of Interoperability (https://www.slideshare.net/poikonen/himss-10-myths-of-pharmacy-interoperability)

Level 1: Non-electronic data. Examples include paper, mail, and phone call [46].

Level 2: Machine transportable data. Examples include fax, email, and unindexed documents. Level 3: Machine organizable data (structured messages, unstructured content). Examples include HL7 messages and indexed (labeled) documents, images, and objects [47].

Level 4: Machine interpretable data (structured messages, standardized content). Examples include the automated transfer from an external lab of coded results into a provider's EHR.

Data can be transmitted (or accessed without transmission) by HIT systems without need for further semantic interpretation or translation [48].

Level 4: Semantic Interoperability is defined as the ability to automatically interpret the information exchanged meaningfully and accurately in order to produce useful results as defined by the end users of both systems [49]. To achieve semantic interoperability, both sides must defer to a common information exchange reference model. The content of the information exchange requests is unambiguously defined: what is sent is the same as what is understood.

2.8.2 Standards

A "standard" is "established by consensus and approved by a recognized body that provides rules, guidelines, or characteristics for activities." [46] [50].

The use of standard clinical vocabularies and structured data organization (ontologies) greatly enhances the ability of clinical systems to interoperate in a meaningful way and for EHR data to be used in clinical trials.

To create interoperable EHRs, standards are needed for:

Clinical vocabularies Healthcare message exchanges, in which one system exchanges messages with another EHR . In addition, EHR systems must follow appropriate privacy and security standards, especially as they relate to HIPAA regulations. Three main organizations create standards related to EHRs:[51] Health Level Seven (HL7), Comite Europeen de Normalization – Technical Committee (CEN TC) 215, and the American Society for Testing and Materials (ASTM) E31 [34]. HL7, which operates in the United States, develops the most widely used healthcare-related electronic data exchange standards in North America. CEN TC 215, which operates in 19 European member states, is the preeminent healthcare IT standards developing organization in Europe. Both

HL7 and CEN collaborate with the ASTM, which operates in the United States and is mainly used by commercial laboratory vendors. Clinical Vocabularies play a strategic role in providing access to computerized health information because clinicians use a variety of terms for the same concept. For example, either “leukopenia” or “low white count” might be written in a patient record—usually these are synonyms. Without a structured vocabulary, an automated system will not recognize these terms as being equivalent.

Standard vocabularies are a means of encoding data for exchange, comparison, or aggregation among systems. Specifically, they are used to Search knowledge resources (e.g., key word searches, tagging) Identify the correct guidelines, critical paths, and reminders to be used in prompting high quality patient care Support practice analysis, quality improvement, and outcomes research Provide data for clinical epidemiological analyses Vocabularies are absolutely essential for data interchange and analyses within and across institutional domains. They are required for all secondary uses of clinical data and for functions such as generating flow sheets. When a clinician evaluates a patient, the documentation usually captures free text or unstructured information, such as history and physical findings. As the clinician evaluation process continues, the unstructured data is transformed (often by a clinical coding specialist) into more structured data that is often linked to payment processing and reimbursement. These claims-related structured data sets (which are different from clinical vocabularies) include Current Procedure Terminology (CPT) codes, International Classification of Diseases (ICD), and Diagnosis Related Groups (DRG). These data sets are primarily used for structured billing and are not designed to capture clinical details that would be most useful for research purposes. Implementing standardized clinical vocabularies and disease ontologies into clinical data capture systems can alleviate terminology inconsistencies when data is captured at the point of care. Logical Observation Identifiers, Names and Codes (LOINC) for ordering lab tests and Systematized Nomenclature of Medicine—Clinical Terms (SNOMED-CT) for recording test results, along with many other existing vocabularies, provide well-defined meanings for specific terms that can be standardized across applications. These vocabularies lend themselves to much more detailed and relevant clinical analyses, especially when measuring outcomes for clinical research support, but only when they are implemented in a uniform way.

2.8.2.1 International Classification of Disease

The ninth revision of ICD is the most commonly used version (the tenth edition is slowly being adopted). It is published by the World Health Organization (WHO). The ICD-9-CM (Clinical Modification) was developed by the National Center for Health Statistics for use in the United States [52]. The ICD is primarily used to code data for billing purposes to identify the disease or problem for which the patient was treated [53].

2.8.2.2 Systematized Nomenclature of Medicine (SNOMED)

SNOMED CT (Systematized Nomenclature of Medicine -- Clinical Terms) is a standardized, multilingual vocabulary of clinical terminology that is used by physicians and other health care providers for the electronic exchange of clinical health information [54].

- It is the most comprehensive, multilingual clinical healthcare terminology in the world
- It is a resource with comprehensive, scientifically validated clinical content.
- It enables consistent representation of clinical content in electronic health records
- It is mapped to other international standards
- It is in use in more than fifty countries

The number of concepts in SNOMED CT continues to grow, and the July 2018 release contained 340,659 active concepts. When implemented in software applications, SNOMED CT can be used to represent clinically relevant information consistently, reliably and comprehensively as an integral part of producing electronic health information. When implemented in software applications, SNOMED CT can be used to represent clinically relevant information consistently, reliably and comprehensively as an integral part of producing electronic health information.
SNOMED CT supports the development of comprehensive high-quality clinical content in health records. It provides a standardized way to represent clinical phrases captured by the clinician and enables automatic interpretation of these. SNOMED CT is a clinically validated, semantically rich, controlled vocabulary. It supports evolutionary growth in expressivity to meet emerging requirements. SNOMED CT based clinical information benefits individual patients and clinicians as well as populations while supporting evidence-based care.

Benefits of SNOMED CT

The use of an electronic health record improves communication and increases the availability of relevant information. If clinical information is stored in ways that allow meaning-based retrieval, the benefits are greatly increased. The added benefits range from increased opportunities for real time decision support to more accurate retrospective reporting for research and management.
SNOMED CT enabled clinical health records benefit populations by:

Facilitating early identification of emerging health issues, monitoring of population health and responses to changing clinical practices. Enabling accurate and targeted access to relevant information, reducing costly duplications and errors. Enabling the delivery of relevant data to support clinical research and contribute evidence for future improvements in treatment. Enhancing audits of care delivery with options for detailed analysis of clinical records to investigate outliers and exceptions.

SNOMED CT enabled clinical health records support evidence-based care by:

Enabling links between clinical records and enhanced clinical guidelines and protocols.

Enhancing the quality of care experienced by individuals. Reducing costs of inappropriate and duplicative testing and treatment, limiting the frequency and impact of adverse healthcare events.

Raising the cost-effectiveness and quality of care delivered to populations [55].

2.8.2.3 How SNOMED CT works

The SNOMED CT logical model defines the way in which each type of SNOMED CT component and derivative is related and represented.
The core component types in SNOMED CT are concepts, descriptions and relationships.
Our model specifies how the components can be managed in an implementation setting to meet a variety of primary and secondary uses [56].

Every concept represents a unique clinical meaning, which is referenced using a unique, numeric and machine-readable SNOMED CT identifier. The identifier provides an unambiguous unique reference to each concept and does not have any ascribed human interpretable meaning.

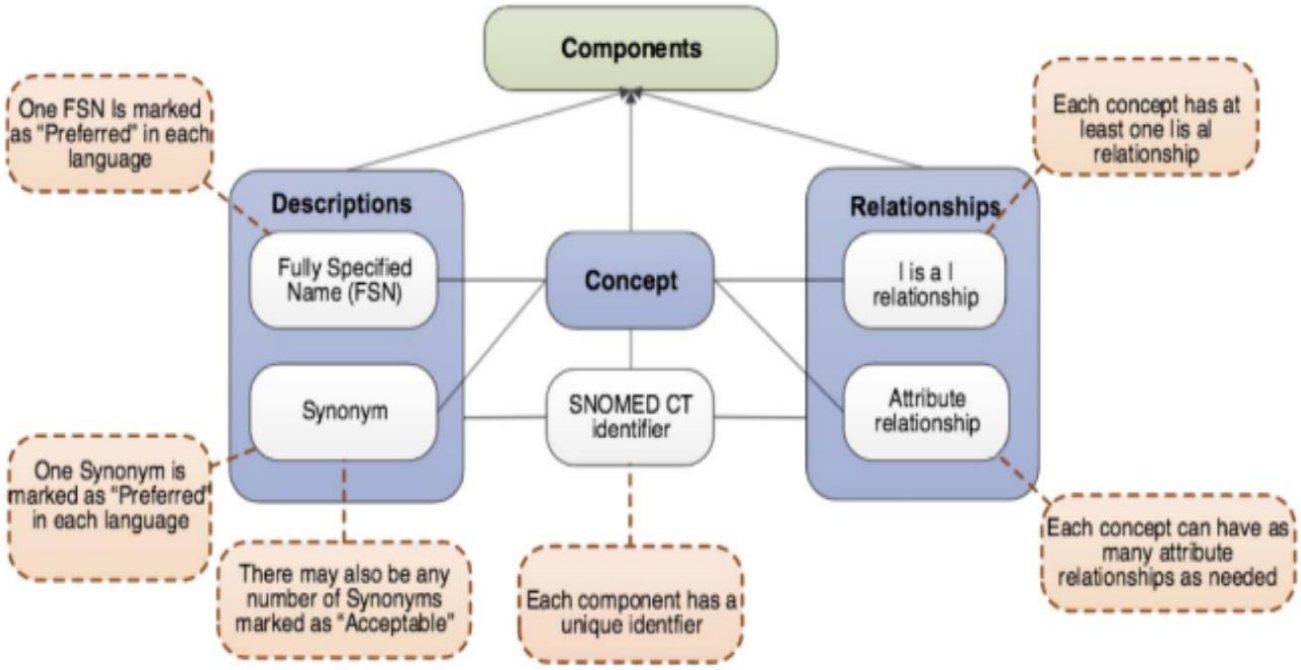

Figure 2.6: Snomed CT structure of terms (https://confluence.ihtsdotools.org/display/DOCSTART/5.SNOMEDCT Logical)

Two types of description are used to represent every concept – Fully Specified Name (FSN) and Synonym. This is shown in figure 2.6. The FSN represents a unique, unambiguous description of a concept's meaning. This is particularly useful when different concepts are referred to by the same commonly used word or phrase. Each concept can have only one FSN in each language or dialect [57].

A synonym represents a term that can be used to display or select a concept. A concept may have several synonyms. This allows users of SNOMED CT to use the terms they prefer to refer to a specific clinical meaning.

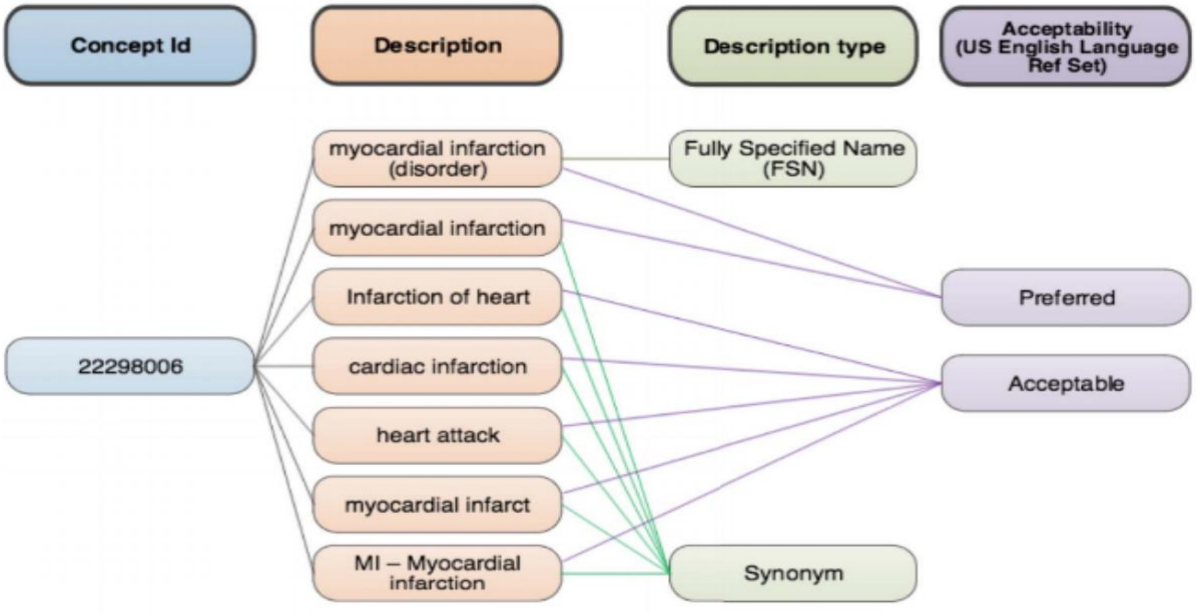

Figure 2.7: Snomed CT concept, description, description type (https://confluence.ihtsdotools.org/display/DOCSTART/5 SNOMEDCT Logical Model)

Figure 2.7 shows SNOMED CT concepts, description and description type. A relationship represents an association between two concepts. Relationships are used to logically define the meaning of a concept in a way that can be processed by a computer. A third concept, called a relationship type (or attribute), is used to represent the meaning of the association between the source and destination concepts. There are different types of relationships available within SNOMED CT [58]. Figure 2.8 shows relationships in SNOMED CT.

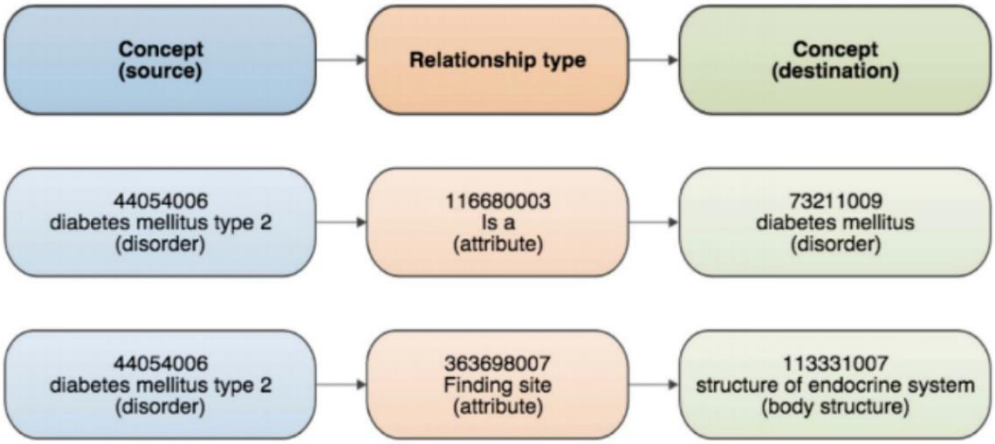

Figure 2.8: Relationships in Snomed CT(https://confluence.ihtsdotools.org/display/DOC-START/5.SNOMED CT Logical Model)

2.8.2.4 Use of SNOMED CT

SNOMED CT itself is only a part of the solution to addressing the requirements for effective electronic health records. A terminology on its own 'does nothing'. To benefit from use of a clinical terminology, it must be implemented and used as part of an application.
The design of the software application in which it is used, and the objectives and motivation of its users, are key factors in determining success.
SNOMED CT is critical for clinical documentation, as it supports the representation of detailed clinical information in a way that can be processed automatically.
Realization of the capability of SNOMED CT to support clinical information and meaning based retrieval requires careful consideration of the actual setting, in terms of scope of use, record structure, data entry, data retrieval and communication.

2.8.3 Logical Observation Identifiers, Names, and Codes (LOINC)

LOINC codes are used to identify individual laboratory results (e.g. hemoglobin values), clinical observations (e.g., discharge diagnosis), and diagnostic study observations (e.g., chest x-ray impression). LOINC is most widely used in laboratory systems [56].

2.8.4 Health Level 7 (HL7)

HL7, (Health Level Seven), is a standard for exchanging information between medical applications. This standard defines a format for the transmission of health-related information.

Information sent using the HL7 standard is sent as a collection of one or more messages, each of which transmits one record or item of health-related information. Examples of HL7 messages include patient records, laboratory records and billing information.

Although HL7 and their messages are widely used, many systems don't know how to speak the language and require a translator. HL7 interface engines work alongside existing applications as an interpreter, speaking the language of HL7 [59][60].

There are two major versions of HL7 in use today. One is HL7 v. 2x, which is commonly used by the existing COTS applications and the other is HL7 v. 3, the Reference Information Model (RIM) which provides a much more robust ability to represent complex relationships. While the RIM is not yet implemented by many COTS EHRs, it can potentially be used for representation of translational research data in a form that can be exchanged with EHRs in the future.

2.8.5 Example Application of Snomed CT for Sematic Interoperability

It would seem obvious that the rationale for using appropriate terminology in health information systems applies to Africa as it does for the US and Europe [33]. This would be true, but there are particular conditions in which terminology is even more important for implementation in Africa. Although Africa has large language groups such as English-, French- and Portuguese-speaking countries, there are numerous linguistic differences at the country and local levels. Significant differences exist in the names of specific disease entities, medications, and laboratory tests. Although Ministry of Health reporting for individual countries is frequently in one of the European languages mentioned above, local clinicians and providers who are recording the data may not speak these fluently. Conversely, clinicians trained in European languages may not be fluent in the languages used by their patients. It would seem obvious that the rationale for using appropriate terminology in health information systems applies to Africa as it does for the US and Europe.

This would be true, but there are particular conditions in which terminology is even more important for implementation in Africa. Although Ministry of Health reporting for individual countries is frequently in one of the European languages mentioned above, local clinicians and providers who

are recording the data may not speak these fluently. Conversely, clinicians trained in European languages may not be fluent in the languages used by their patients.

According to [61], It would seem obvious that the rationale for using appropriate terminology in health information systems applies to Africa as it does for the US and Europe. This would be true, but there are particular conditions in which terminology is even more important for implementation in Africa. Although Africa has large English-, French- and Portuguese-speaking groups, there are numerous linguistic differences at the country and local levels. Significant differences exist in the names of specific disease entities, medications, and laboratory tests. The gap between recording the information from the patient and what eventually arrives in the end as information, either locally or at ministerial level, is significant. Capturing knowledge in a manner in which it contains the most information is important, not only for the reasons propositioned by [62], but also because much indigenous knowledge is being lost. Experience of health care providers in Africa is fragmented and knowledge is locked away in unanalyzed paper records and reports which take years to make their way to government where they are frequently ignored [54]. Information technology should support the collecting and preservation of this knowledge, rather than suppress it [3]. If terminology can help capture clinical experience accurately and quickly, it can then be put to use at the local and national level. Capturing the data using a method which incorporates reference terminology would make it possible for countries to share their knowledge of the status of health, disease and the effectiveness of interventions. In particular, the experience of neighboring countries would be more relevant to Africa providers and policy makers than data available from developed countries. The need for information technology in the developing world is great. Effective implementation and application of health information technology will likely be required if any significant progress is to be made against the devastation of HIV/AIDS, tuberculosis and malaria [63]. There are many obstacles to successful design and implementation of EHR systems in Africa. One challenge is that systems designed for US and European health facilities may not be applicable in rural and urban developing countries. Terminology has been identified as an important barrier, and a properly implemented interface terminology has the potential to solve this problem. Taking evidence-based experience from other African situations and applying them within the information system holds promise for improving quality and helping to train providers in resource-poor settings. However, the inclusion of medical terms in local languages, important for an interface terminology, is difficult and requires access to experts and reference materials [64]. To test whether it is possible, the Millennium Villages™ Project (MVP) built a network of health information

systems called the Millennium Global Village Network (MGV-Net). MVP is the product of five years of intensive preparation by hundreds of scientists and development experts and works in eleven countries throughout Africa to help people lift themselves out of extreme poverty. MGV-Net uses a common data dictionary based on reference terminologies (SNOMED CT, LOINC and RxNorm), but employs a centralized Terminology Service Bureau (TSB) to manage an interface terminology distributed throughout the network. The TSB is a critical component of the MGV-Net and maintains all terminology additions and mappings. To begin the process, a workbook identifying the most common diseases, diagnostic tests and medications was sent to all MVP villages for translation. Clinicians in each of the MVP countries work together online as part of the TSB to update the database. Table 1 shows application of Snomed CT in which the terminologies have been translated to local languages.

Table 1: Application of Snomed CT

SNOMED CT	English	French	Kinyarwanda	Swahili
271737000	Anemia	Anémie	Kubura amaraso	Upungufu wa damu
195967001	Asthma	Asthme	Asima	
82272006	Common cold			Homa ya mafua
9826008	Conjunctivitis	Conjonctivite	Indwara z'amaso	
80967001	Dental caries	Carie dentaire		
62315008	Diarrhea			Kuhara
7520000	Fever of unknown origin	Fièvre	Umuriro	
25374005	Gastroenteritis	Gastroentérite	Kuruka no kuhitwa	
399221001	Genital bleeding (vaginal)	Hémorragie génitale	Kuva	
25064002	Headache			Kuumwa na

2.9 Design and Development of an E.H.R

2.9.1 Data model

According to [35], the design of the database tables and their relationships, the data model, is the core of any EMR/EHR system. The strength of the data model will dictate the scalability and flexibility of a system. The design of the database schema is usually driven by the functional requirements of the EMR/EHR system; if the system is primarily for reporting and health statistics, there is a tendency to represent all data items as columns, similar to a spreadsheet. This approach is suitable for simple single functional function systems, such as for clinical trials, but tends to be

inflexible, especially for chronic care. For more multi-functional systems, a data model is required that:

- Can support a variety of functions within clinical care, programme monitoring and reporting, supplies and logistics, and research. These more complex systems need to be able to handle different types of data and accommodate new data such as drugs, clinical conditions and outcomes without modifying the data model.
- Allows for temporal data; data for clinical diagnoses, laboratory tests, treatments and outcomes are often temporal in nature, particularly for chronic disease management.
- Allows for data to be exported in standard formats for analytic and statistical packages, third party software, among others allows for, or can gracefully expand to, support different spoken languages and variations in medical terminology. One approach in designing a data model for more complex EMR/EHR systems is the use of a concept dictionary. For example, hard-coding types of clinical conditions and outcomes into the database schema often results in making frequent changes to the data model as the system is expanded to allow for more types of clinical conditions. This expansion is not always possible or easily made and the data model can end up restricting certain extensions to the system. Instead codes from a central concept dictionary can be created that map to the clinical conditions.
- Allow the database schema to store these concepts as data. The meaning of these codes will not be a fixed part of the database schema. When the system expands to allow for new clinical conditions, all that is required is to insert new concept codes without changes to the database schema. These codes can include fields in a second language, allowing straightforward translation in bilingual environments. This approach can also deal with variations and nuances in terminology between projects and sites. It is possible to map a single code to many natural language phrases. Each coded concept must be well described and non-ambiguous. Where possible, the concept codes should be mapped to standard coding systems such as ICD10, SNOMED, LOINC or HL7, though none provide complete coverage [35]. The data model can even be extended to assist in the automatic generation of user interfaces and to support validation rules for correct data entry. To move beyond successful prototypes to widespread use, it is essential that EMR/EHR systems are developed with open standards and sharable components. A common data model can efficiently

link the wide range of technology platforms discussed earlier and ease collaboration between projects.

2.9.2 Architecture

Network architectures Stand-alone Systems

In these systems, a database and user interface is deployed on a single machine. The EMR/EHR system has no explicit functionality to communicate with other machines over a network [65].These systems range from a simple spreadsheet for storing patient data to a simple database. Stand-alone systems are the easiest type of EMR/EHR to design and implement and are suitable for isolated applications such as a small EMR/EHR, patient registry or a clinical trial database.

Local area network (LAN) systems

A LAN EMR system is deployed at a single site and machines have a relatively fast connection to each other (= 10mb/s).

Typically, these systems revolve around a database (Oracle, MySQL, MS SQL Server) deployed on a central server [66]. Users have local client application interfaces in which they enter, query and modify data directly on the central database [67].

Wide area network (WAN) systems

These consist of a networked system that operates across multiple geographical sites. Sites could be spread across a single city, state, country or could even span multiple countries [68] [69]. There are many approaches to WAN EMR systems.

EMR User Interface Issues

The user interface of a good EMR/EHR should provide a portal through which all aspects of patients' health record can be seamlessly and accurately recorded and retrieved [70]. All this information must then be made easily accessible to the physician and presented in a meaningful way by the EMR/HER [71]. The literature suggests two aspects that are particularly important with respect to the usefulness of an EMR to physicians: the recording of information into the EMR and

the navigation of the interface itself. The goal of this paper is to present an EMR design that makes it easier for the physician to accomplish these tasks [72]. The challenges associated with these issues are discussed as follows:

Recording Patient Information

Paper records offer physicians enormous flexibility when documenting and annotating patient information [73]. Unfortunately, the interface for many electronic health record systems emphasize capturing patient data in a highly structured or restricted coded form. In recent studies, it has been shown that some physicians regularly eschew coded data entry, opting instead to record patient information in free-form text [74]. Providing physicians, the ability to record free-form text offers them the flexibility to enter information in an order and manner which is most appropriate to their personal workflow and style. While coded and/or structured data can be useful for clinical decision support and administrative purposes such as report generation, forcing a physician to pigeon-hole clinical observations may result in information critical to patient care being omitted from the patient's health record or recorded in the wrong "box." Information which is outside the domain covered by the restricted coded data of the required by the software may subsequently be forgotten or not communicated to relevant parties. During an encounter, patients often communicate their complaints and symptoms to the physician in the form of a story. For many physicians, accurately capturing these stories (also known as patient narratives), is essential to patient care [75]–[77]. An EMR designer must appreciate the physician's need to capture all aspects of the patient's narrative during the encounter and allow the physician to construct the narrative in a natural manner. Allowing the physician to enter information using free-text offers more flexibility over more restrictive structured or coded data entry. As a result, the method of data entry becomes

more fluid and accommodating for the physician. Ultimately, an EMR must never force the physician to forget that he or she is recording information from a patient and not just recording information to a database. By making the entry of patient information more natural to the physician, the disruption to communication between the physician and patient is kept at a minimum.

Workflow and Interface Navigation

The importance of document management and the clerical tasks implicitly performed by physicians must be acknowledged in the design of an EMR interface. In particular, patient information must be immediately accessible and not hidden within a labyrinth of windows or dialog boxes [70] [78]. For example, navigating between physician orders and their associated results should be seamless and natural. Physicians often complain about the "loss of overview" as they navigate an EMR interface [79][80]. This loss of overview can be caused or exacerbated by the plethora of pop-up windows or dialog boxes that are used by some EMRs to present or request information or to alert physicians about exceptional circumstances such as potentially serious drug-drug interactions [23]. Sometimes, these dialog boxes are implemented as "modal windows" which prevent the physician from continuing until the event has been addressed. Such modal dialog boxes are distractive and have been shown to introduce inefficiencies in a physician's workflow [70]. The navigation of an EMR interface should be very accommodating to a physician's unique style and needs. Retrieval of relevant information from a variety of sources should be efficient and tasks such as orders and requisitions should be performed with a minimal number of steps. The numerous health events that occur during the care of a patient must be presented in a way that is contextually meaningful but without overwhelming the physician or cluttering the user interface. Details regarding these events and their interactions with one another must be made readily available by the interface. Some of the inefficiencies associated with EMR navigation may be attributed to

fundamental aspects of the underlying EMR architecture bubbling up to the interface itself. When EMRs allow database abstractions and other low-level implementation details to bleed into the user interface, a physician may become infected by technical considerations that are unrelated to the immediate care of his or her patients. These artifacts can cause a physician to lose overview during the assessment and planning phases of a patient's treatment; patient care may degrade as a result. By reducing or eliminating conventional user interface elements such as modal pop-up windows, pull-down menus, buttons, checkboxes, lists, etc., a more streamlined interface that offers free-text entry and easier navigation can be developed. The foundation for this interface is proposed in the next section.

Interface Design

To address the issues described in the previous section, the proposed interface design presents to physicians a pane-based view of the patient record, as shown in Figure 2.9.

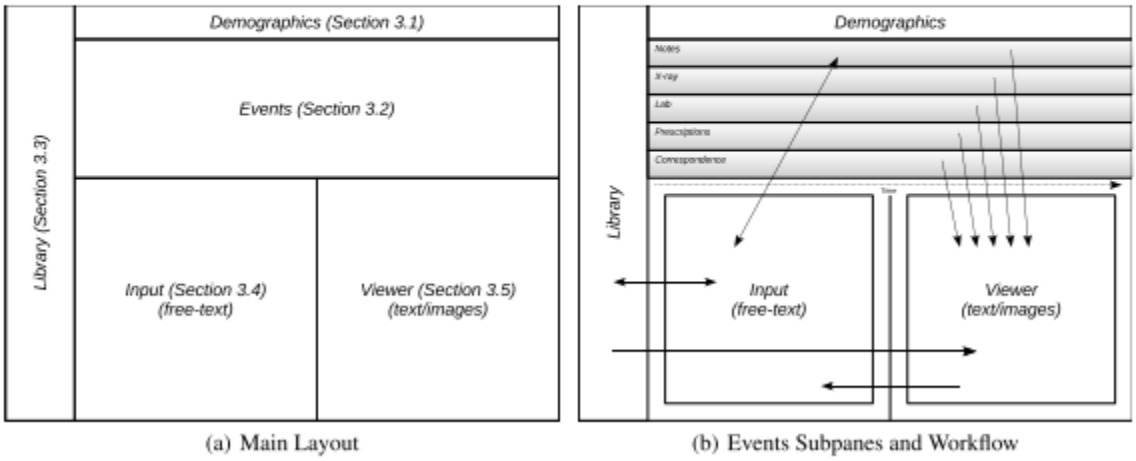

Figure 2.9: Interface Design

This interface is designed to offer physicians a broad overview of the patient record while allowing them to quickly focus on smaller details as the need arises. As the physician records and retrieves clinical information to and from the EMR during the patient encounter, the interface does not change considerably from one view to the next. No disruptive or obstructive pop-up windows are ever displayed by the interface. In particular, the physician is not required to navigate an abundance of windows and dialog boxes in order to record or retrieve information to and from the EMR, regardless of whether the information being retrieved is local or remote. In addition to offering a more natural workflow, this interface may help reduce the loss of overview for the physician. The components of the interface are described in further detail in the following: (1) Demographics Pane; the Demographics pane at the top of the window simply contains generic information about the patient, such as name, age, date of birth and address. Unlike other panes, the demographic pane cannot be resized or closed — it is always present so that that the physician never loses focus of the patient currently being treated. The demographics pane can also be used to navigate to other patients. As data is entered in the various fields of this pane, the contents of the Viewer pane below it will contain a list of patients that satisfy the supplied criteria. Patients can then be selected from this list and their associated details will be presented in the Events pane.

(2) Events Pane the Events pane provides a segregated, time-ordered view of a single patient's health. This puts the entire patient's history in temporal context and offers the physician a "big picture" view of the patient's general history. This user interface element is the primary vehicle used for navigating the patient chart and for maintaining a physician's overview of the patient record. As shown in Figure 9(b), this pane is divided into subpanes, each of which correspond to various domains of health information. There are five broad domains containing the Notes recorded by the physician, Lab reports, X-ray reports, Prescriptions and Correspondences related to the patient. Each of these subpanes can be expanded, collapsed, zoomed and scrolled to reveal

more or less information as required by the physician. Events are represented by small squares (colloquially referred to as rhondots) within the subpane. The horizontal position of each event represents the date the event took place, whereas the vertical placement indicates the time of day the event occurred. Events that are related to one another can be connected by arrows, thereby making it clear which events were triggered by an initiating event. This would enable a threaded event viewer across several provider domains, making it relatively easy to find the originating order for a given lab result, for example. Such a feature would rely upon the underlying communication infrastructure using consistence reference identifiers to relate the events together. Colours and other visual attributes can be used to indicate the urgency of the events. For example, green could be used to represent normal events, while blue and red could be used to represent abnormal and critical/important events, respectively. Unread events can be represented by a square; events that have been viewed by the physician can be changed to circles. More details on each event can be viewed by dragging its square to the Viewer pane. An event can be annotated by dragging it to the Input pane, adding the annotation, and dragging it back to the event pane; a new note linked with the original event will then be created in the Notes subpane. Context menus (activated by right clicking the mouse in the subpane) can be used to select and plot various quantifiable data items extracted from the events contained in that subpane. An Events subpane can be expanded, if necessary, to make the plots easier to read.

(3) Library Pane; Library pane is a vertical container along the left hand side of the display which acts as a repository for a generic collection of clinical practice guidelines, templates, formularies, patient educational pamphlets, diagnostic checklists, requisitions, etc. Each element in the library has a title and a body of associated text. The titles are arranged in alphabetical order in the pane. To prevent titles from overlapping, only frequently referenced titles are initially displayed and

accessible inside the pane. Dragging these to the Input or Viewer panes will cause the title's corresponding body of text to be placed in the appropriate pane. As with the subpanes in the Events pane, the library pane can be zoomed and scrolled as needed. In addition to dragging information out of the Library pane, the contents of this pane can be queried directly as information is entered in the Input pane. The physician can add new resources to the li brary simply by writing text in the Input pane and then dragging it to the Library pane. The first line of the text can be used as the title. A body of medically related information can be programmatically imported into the library, if necessary, as part of the installation procedure of the EMR in the physician's office. The Library pane may also contains documentation related to the usage of the EMR itself, which can be requested by making queries in the Input pane.

(4) **Input Pane**; All information that the physician wishes to record in the EMR is done via the Input pane. All progress and SOAP notes as well as requisitions, prescriptions and correspondences are created here in a relatively freetext manner. Once the note is finished, it is dragged to the Events pane where it will automatically be stored in the Notes subpane. Generic information not particular to any patient can also be created here and dragged to the Library, as described above. The objective of the Input pane is to provide the physician more flexibility when recording patient information than is offered by traditional EMRs that may require a large amount of structured or coded data entry. This pane is intended to serve the same purpose as the blank piece of paper in a traditional chart — the physician has the ability to record unstructured text to capture the patient narrative in a way most natural to his or her style.

As the physician types the note, dynamic analysis of the text and data extraction takes place. This means that as characters are typed, words and phrases can automatically change colour to inform the physician that something of interest has been understood by the EMR. This provides the physician with instant feedback that the EMR has successfully extracted information from the note.

For example, quantitative data such as heart rate and blood pressure could be identified, provided a consistent syntax is used. Notes are traditionally subdivided into coarse sections. For example a SOAP note has subjective, objective, assessment and plan sections. As various subsection heading text is entered by the physician, the editor would enter different contextual modes of operation which would affect the dynamic analysis of the following text. Ideally, all prescriptions and requisitions for lab work or diagnostic could be made inside the Input pane. This makes prescriptions and requisitions possible without having to navigate through numerous pop-up screens and meticulously enter coded data along the way (possibly losing patient overview in the process). Any incomplete or erroneous requisitions and dangerous prescription interactions could be detected dynamically and reported in an unobtrusive manner to the physician by turning the background colour of the text red with an annotation indicating the problem, for example. The success of this Input pane interface depends, in large part, on how well the dynamic text processing algorithm can glean the intentions of the physician as they construct their patient notes and requisitions [70]. By having physicians adopt modest syntactic conventions as part of their note taking, we believe that it will be possible to construct a text processor that will be able to provide the physician with useful real-time feedback during the note taking process.

(5) **Viewer Pane**; Viewer pane is used to display a wide variety of data related to the patient chart. Events from the Events pane can be dragged and dropped onto this pane and the pane will render the contents of the event in a human readable format. The contents could be either text, images or a combination of both. Annotations of the event can be made by dragging the event to the Input pane, making the annotation, then dragging it back to the Events pane.

2.10 Example Designs

This section looks at a system that was development in SAUDI ARABIA.

Review of the Health Record in KSA & Health Care Planning in Saudi Arabia. Health care planning system in Saudi Arabia has different agencies that play important roles in providing health care to residents [81]. These agencies are The Ministry of Health, The National Guard, The Ministry of Defense and Aviation, and The Ministry of Interior. In addition to these agencies there are specialist hospitals in Saudi Arabia that provide health care to specialist health cases. These agencies provide health care services on the basis of exclusive free health care to all citizens. In addition, the private sector in Saudi Arabia plays an increasingly significant role in the Kingdom and coordinates with the referral network and the regulatory requirements of health sector as a whole [81]. The Ministry of health in Saudi Arabia has seen that the primary objective of both the public and private health sector is to improve the health conditions of all citizens through the provision of comprehensive preventive and curative health services throughout the Kingdom, with particular emphasis.

The system was designed and represented by the following XML, and Screenshot.

A. B-XML Instance Representation of the whole EHR system code

   <Area> Health Area </Area> [1] <Hospital> Hospital </Hospital> [1] <Primary Information> Primary Information </Primary Information> [1] <Patient-Record> Patient-Record </Patient-Record> [1] <Outpatient Form> Outpatient Form </Outpatient Form> [1] <Doctor-OrderSheet-Form > Doctor-Order Sheet-Form </Doctor-OrderSheet-Form > <Progress Notes Form> Pro-gres Notes Form </Progress Notes Form> [1] <Admission and Discharge Form> Admission and Discharege Form </Admission and Discharge Form>

B. C- Creation of Rational Data base system and Tables A Relations between tables was established and generated, forms was built of the patient data entry, consists of:

   Hospital manpower basic information,Patients basic information, Outpatient transaction record, The clinical system, In case of accident, X_ray and 7-Microbilogy requesting form.

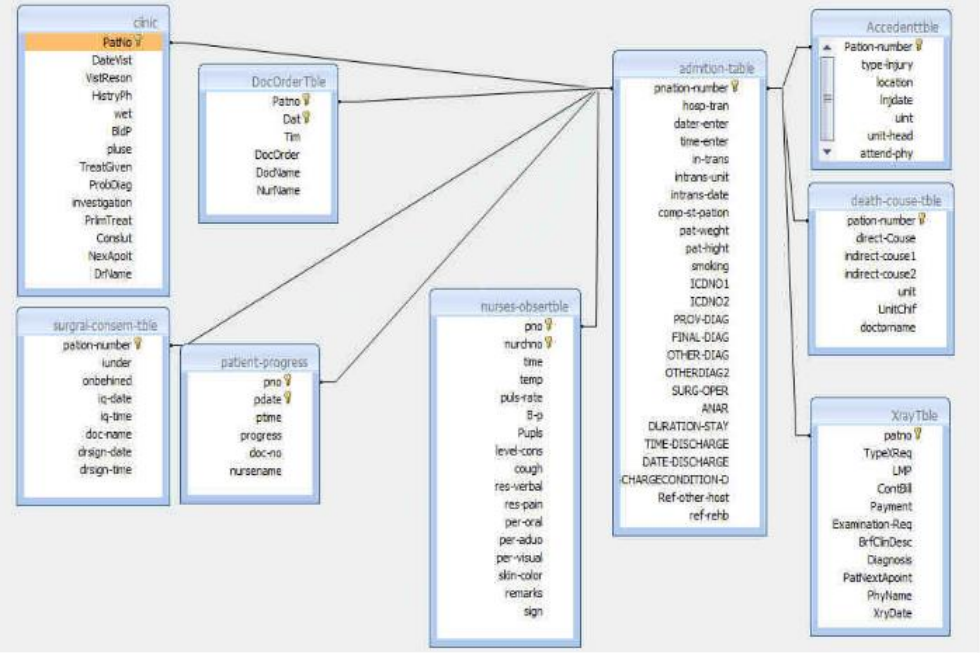

Figure 2.10: Example system ERD

Figure 2.10 is the ERD diagram for the system, which was made up of 9 tables for storing system data.

The design of electronic forms based on the existing manual forms.

Figure 2.11: Example system login form

Figure 2.11 is the login form, which is first displayed when the user wants to access the system. The login form was designed with the ability to display a registration form to help capture details for user who have not been registered in the system.

Figure 2.12: Example system landing page

Figure 2.12 shows the landing page displayed when a user has successfully login.

Figure 2.13: Example System admission and discharge form

Figure 2.13 shows the form used to capture admission and discharge patient details.

2.11 System Development

2.11.1 Software development stages

Building a software product is a process consisting of several distinct stages. Each stage has its own deliverables and is bound by a specific time frame. Depending on the project, certain stages gain additional weight in the overall effort to implement the software product [82].

Research

Research is the stage where the project owner, the project manager and the project team gather and exchange information. The project owner is responsible for formulating requirements and passing them on to the project manager. In order to properly formulate requirements, the project owner has to first define a set of goals. Then he has to envision the way a software product will help him achieve those goals. In the research stage the project owner will try to find people or companies with similar goals and document the way those people or companies acted upon fulfilling their goals. The project manager is responsible for receiving the requirements from the project owner, evaluating them and passing them to the project team as technical specification. The project manager has to be able to evaluate the requirements from both a business perspective and a technical perspective. The project manager has to research market characteristics and user behavior patterns. The project team is responsible for evaluating the requirements from a technical perspective. The project team will have to research the frameworks, API's, libraries, versioning tools and hosting infrastructure

that will be required in order to build the software product. Planning is the stage where all the elements are set in order to develop the software product. Planning starts with defining the overall flow of the application. Next step is to breakdown the flow into smaller, easier to manage subassemblies. For each subassembly a comprehensive set of functionalities has to be defined. Based on the required functionality a database structure is designed. Taking into account the overall flow of the application, the subassemblies, functionalities and database structure, the project manager together with the project team have to choose the technology that will be employed to develop the application. Also the project manager should decide on the best suited management methodology and the proper work protocol for the project at hand. Design is the stage where the layout of the application is created. Web applications and mobile applications tend to grant more impotence to layout than desktop applications. Depending on the nature of the application designs can range from rough and functionality driven to complex and artistic. An accounting application will only require basic graphic design but an online museum will require high end design work. In an accounting application design has to emphasize and enhance functionality whereas in an online museum.
functionality has to be tailored in order to fit the design. The graphic design can overlap with the planning and with the programming stage. The graphic design stage is important because it will display to the project owner a preview of the application before it is actually built. At this stage usually the project owner comes up with new requirements that have to be summited to research and planning. Development is the stage where code is written and the software application is actually built. The development stage starts with setting up the development environment and the testing environment. The development environment and the test environment should be synchronized using always the same protocol. Code is written on the development environment and uploaded on the test environment using the synchronization protocol. Another important aspect of the development stage is progress monitoring. The project manager has to determine actual progress and evaluate it against the initial planning. The project manager should constantly update the project owner on the overall progress. When writing code the software developers should also perform debugging operation in order to upload clean and bug free updates on the testing environment. Software developers should also comment their code so that they can easily decipher later or make it easy to understand for other developers. Testing is the stage where programming and design errors are identified and fixed. Programming errors are scenarios were the application crashes or behave in a way it was not supposed to according to the designed architecture. Programing errors also consist in security or usability issues. If the application is vulnerable to attacks and can therefore allow attackers access to private

data, then that is regarded as a programming error. If users have problem with slow response time from the application than that also is a programming issue. Design errors are actually inconsistency between what the project owner requested and what the project team ended up implementing. Design errors occur in the planning stage, have a significant impact on the project and are usually harder to fix. Identifying design errors is considerably more efficient when the project owner is involved as he is the one that formulated the application requirements. Setup is the stage where the application is installed on the live environment. The setup stage precedes the actual exploitation of the software product. The setup entails configuring the live environment in terms of security, hardware and software resources. Back-up procedures are defined and tested. The actual setup of the software product includes copying the source code, importing the database, installing third party applications if required, installing cron-jobs if required and configuring API's if required. Once the application is installed it will go through another full testing cycle. When testing is completed content is added to the application. Maintenance is the stage that covers software development subsequent to the application setup and also the stage responsible for ensuring that the application is running within the planned parameters. Ensuring that the application is running properly is done by monitoring the firewall, mail, HTTP, FTP, MySQL and SSH error logs. Also monitoring traffic data will provide valuable input on potential issues that may affect the application's performance. An important part of the maintenance stage consists of systematically testing functionalities for errors that were not identified in the testing stage or for issues that are not displayed in the error logs. The maintenance stage also provides the opportunity to add new features of functionality to the software application. Adding new code or changing the old code will have to be submitted to
research, planning, programing, testing and setup. The above mentioned stages are generally agreed by the software development community as being the cornerstones of every software development project. Depending of software development methodology, they may be found under different naming conventions, they may be overlapping changing order or missing altogether. Current software development methodologies A software development methodology is a set of rules and guidelines that are used in the process of researching, planning, designing, developing, testing, setup and maintaining a software product. The methodology also includes core values that are upheld by the project team and tools used in the planning, development and implementation process. The analysis includes specifying the scale of the project the methodology is suited for, the stage project owner feedback is delivered and a graphic representation of the methodology. For coherence reasons stages defined in the second section of the article are also used in depicting the graphic representations of

the methodologies. Waterfall is the first methodology generally acknowledged as being dedicated to software development. Its principals are for the first time described by Winston W. Royce even though the actual term waterfall is not used in the article [82]. It emphasizes meticulous planning and it outputs comprehensive documentation.

The Waterfall methodology is linear sequential process where every stage starts only after the previous has been completed. Each stage has its own deliverables. The Waterfall methodology is predictable and values rigorous software planning and architecture. Figure 2.14 shows the stages involved in the Waterfall methodology.

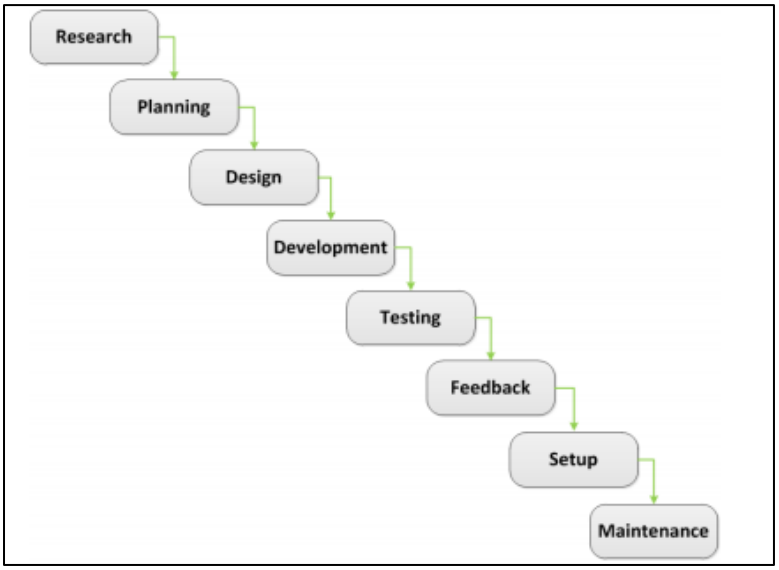

Figure 2.14: Waterfall Methodology diagram

The project owner's feedback is received after the software application is completely developed and tested. The Waterfall methodology is suitable for small scale software development projects where requirements are clear and detailed planning can be easily drafted for the entire project[82].

Prototyping

Prototyping is a methodology that evolved out of the need to better define specifications and it entails building a demo version of the software product that includes the critical functionality. Initial specifications are defined only to provide sufficient information to build a prototype. The prototype is used to refine specifications as it acts as baseline for communication between project team and project owner. The prototype is not meant to be further developed into the actual software product.

Prototypes should be built fast and most of the times they disregard programming best practices [83]. Figure 2.15 is a graphical summery of the prototyping methodology.

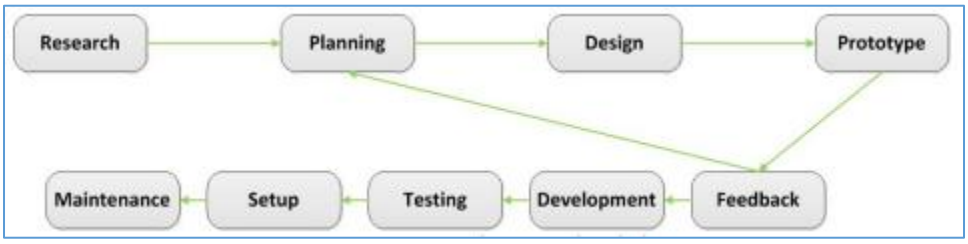

Figure 2.15: Prototyping methodology diagram

The project owner's feedback is received after the prototype is completed. The Prototyping methodology is suitable for large scale projects where is almost impossible to properly define exhaustive requirements before any actual codding is performed. Prototyping methodology is also suitable for unique or innovative projects where no previous examples exist[84].

Iterative and incremental

Iterative and incremental is a methodology that relies on building the software application one step at the time in the form of an expanding model [85]. Based on initial specification a basic model of the application is built. Unlike the prototype, the model is not going to be discarded, but is instead meant to be extended. After the model is tested and feedback is received from the project owner specifications are adjusted and the model is extended. The process is repeated until the model becomes a fully functional application that meets all the project owner's requirements. Figure 2.16 is a graphical illustration of the iterative and incremental methodology.

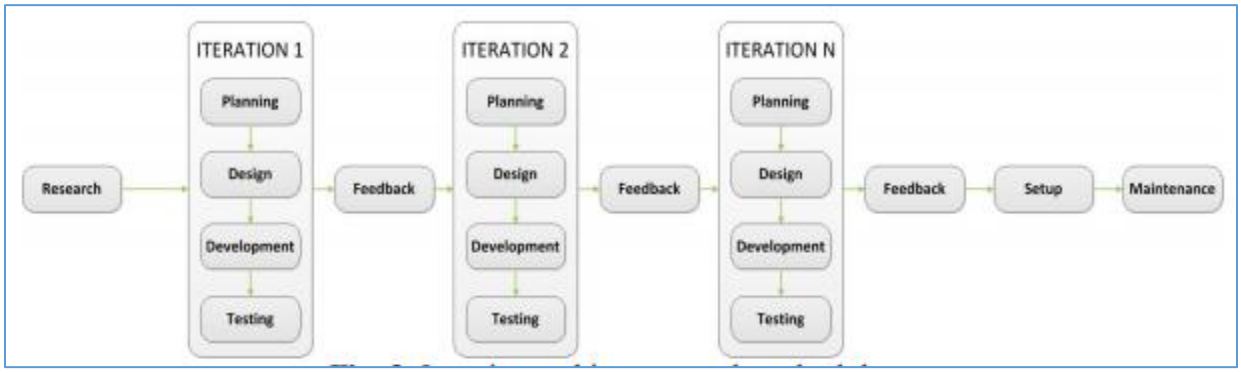

Figure 2.16: Iterative and incremental methodology

The project owner's feedback is received after each iteration is completed. The Iterative and incremental methodology emphasizes design over documentation and is suitable for medium and large projects. Spiral is a methodology that focuses on identifying objectives and analyzing viable alternatives in the context well documented project constrains.

Spiral methodology

The Spiral methodology has 4 major phase[86]: planning, risk analysis, development and evaluation. Project will fallow each phase multiple times in the above mentioned order until the software application is ready to be setup on the live environment. The Spiral methodology emphasizes risk analysis and always evaluates multiple alternatives before proceeding to implementing one. Figure 2.17 is a graphical depiction of the Spiral methodology.

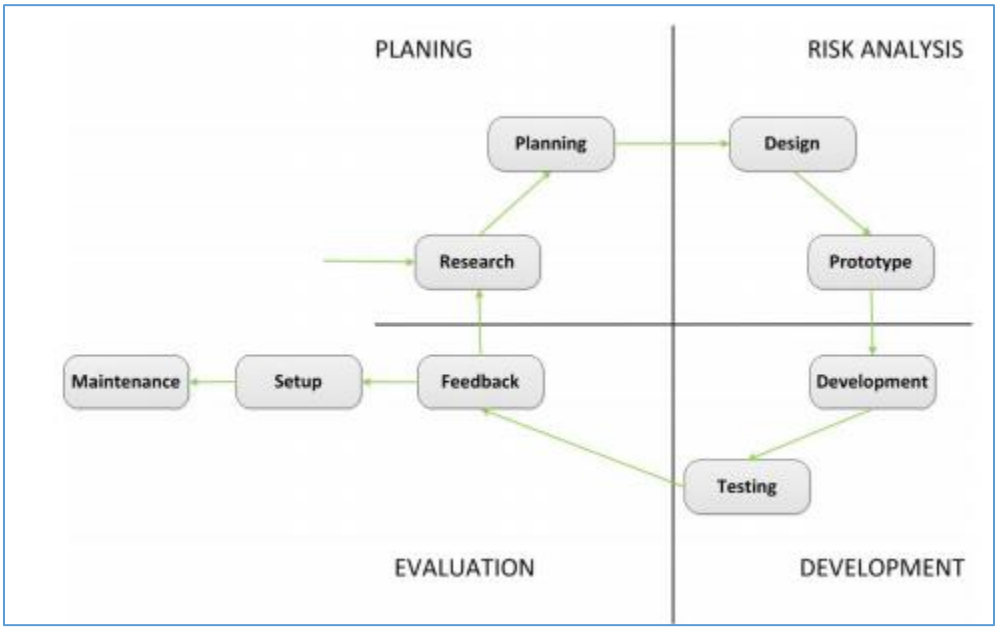

Figure 2.17: Spiral methodology

The project owner's feedback is received after the first iteration of the spiral is completed. The Spiral methodology is suitable for medium and large-scale projects. It has also proven more effective in implementing internal projects as identifying risks proprietary to your own organization is easier.

Rapid application development

Rapid application development is a development lifecycle designed to give much faster development and higher quality results than those achieved with the traditional methodologies. It is designed to take the maximum advantage of powerful development software. Rapid application development imposes less emphasis on planning tasks and more emphasis on development. Development cycles are time boxed and multiple cycles can be developed at the same time. Figure 2.18 graphically demonstrates the Rapid Application development methodology.

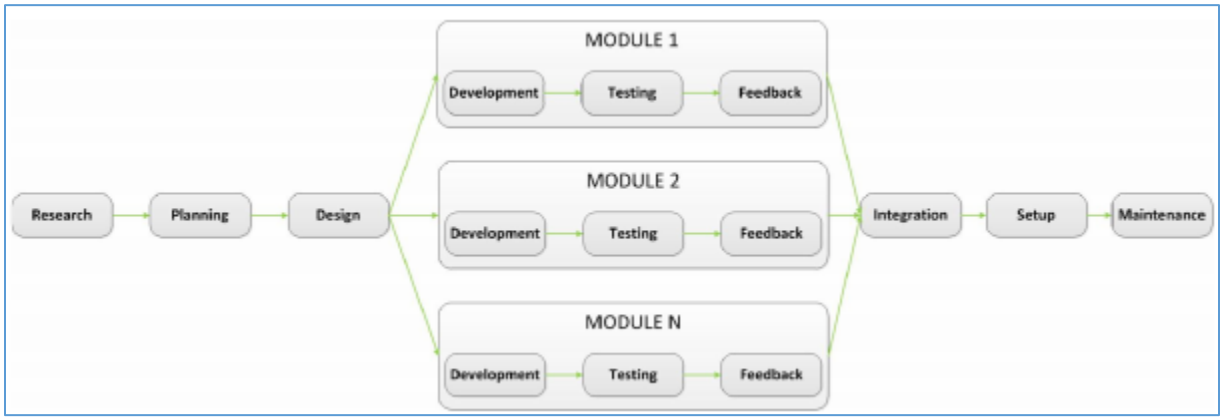

Figure 2.18: Rapid application

The project owner's feedback is received after each module is completed. The Rapid application development methodology is suitable for small, medium and large-scale projects with the constraint that projects have to be broken down into modules.

Summary of Methodologies, Strengths and Weakness

Table 2: methodology, characteristics, strengths and weakness

Methodology	Characteristics	Strengths	Weaknesses
Waterfall	• Comprehensive documentation meticulous planning	• easy to manage	• working code is delivered late in the project
	• linear-sequential process • each phase has its own deliverables	• easy to understand for the project owner and team	• does not cope well with changing requirements • low tolerance for design and planning errors
Prototyping	• build one or more demo versions of the software product	• accurate identification of application requirements -	• leads to unnecessary increase of the application's complexity

		early feedback from the project owner	• increased programming effort
	• project owner is actively involved	• improved user experience • early identification of missing or redundant	• costs generated by building the prototype
	• prototypes are meant to be discarded		
Spiral	• focuses on objectives, alternatives and constrains • has 4 major phase: planning, risk analysis, development and evaluation • emphasizes risk analysis • evaluates multiple alternatives before proceeding to the planning stage	• working code is delivered • early in the project • minimizes risk • strong documentation	• costs generated by risk handling • dependent on accurate risk analysis
Rapid application development	• less emphasis on planning tasks and more focus on development • time box approach	• applications are developed fast • code can be easily reused	• poor documentation • high development costs • code integration issues • application has to be broken into modules

Extreme programming	• pair programming • unit testing • fast consecutive releases • collective ownership • on-site project owner • open workspace • project owner decides the • priority of tasks	• application gets very fast in the production environment • frequent releases of working code • reduced number of bugs • smooth code integration • continuous feedback from the project owner	• lack of documentation • developers reluctance to pair programing • developer's reluctance to write tests first and code later • requires frequent • meetings • lack of commitment to a well-defined product leads to project owner reluctance
V-Model	• introduces testing at every development stage • highlights the importance of maintenance	• low bug rate • easy to understand and use	• vulnerable to scope creep • relies heavily on the initial set of specifications
Scrum	• iterative development • time box approach known as Sprints • daily meetings to assess progress known as Daily • Scrum	• deliver products in short cycles • enables fast feedback • rapid adaptation to change	• lack of documentation • requires experienced developers • hard to estimate at the beginning the overall effort re-

	• self-organizing development team • tasks are managed using backlogs; product backlog and sprint backlog		quired to implement large projects; thus cost estimates are not very precise
Cleanroom	• iterative development • box structure method • using mathematic models in quality control • statistical approach to testing	• considerable reduction in bug rate • higher quality software products	• increased development costs • increased time to market for software product • requires highly skilled highly experienced developers
Dynamic systems development method	• iterative development • Moscow prioritization of tasks • time box approach • non-negotiable deadlines • strict quality standards set at the beginning of the project • project team and project owner share a workplace (physical or virtual) • test early and continually	• focusses on addressing effectively the business needs • post project implementation performance assessment • complete documentation • active user involvement	• requires large project teams at it has multiple roles to cover • requires very skilled developers

Rational Unified Process	• iterative development • prioritize risk handling - adequate business modelling • change management • performance testing	• accurate and comprehensive documentation • efficient change request management • efficient integration of new code • enables reuse of code and software components	• requires highly qualified professionals • development process is complex and poorly organized
Lean software development	• iterative development • discards all components that do not add value to the • product • amplify learning • customer focus • team empowerment • continuous improvement	• reduced project time and cost by eliminating waste • early delivery of working code • motivated project team	• project is highly dependable on individual team members • a team member with strong business analysis skills required
Test-driven development	• unit testing • testing scenarios are developed before actual coding • repeated short development cycles	• less time spent on debugging • higher quality code • by designing tests the devel-	• tests are focused on syntax and overlook actual functionality • requires more code than most methodologies

	• suitable for debugging legacy code developed with other techniques	oper empathizes with the user • less defects get to the end user	• the developer is actually the one doing the testing • writing unit tests increases costs
Behavior-driven development	• focuses on business value • genuine collaboration between business and development • unit testing	• easy to maintain • usability issues are discovered early • reduced defect rate • easy to integrate new code	• project owners are reluctant to write behavior scenarios
Feature-driven development	• iterative development • application is broken down into features • no feature should take longer than two weeks to implement • uses milestones to evaluate progress	• multiple teams can work simultaneously on the project • scales well to large teams • good progress tracking and reporting capabilities • easy to understand and adopt	• individual code ownership • iterations are not well defined

Model-driven engineering	• uses domain model • models are automatically transformed into working code • knowledge is encapsulated in high level models • emphasizes reuse of standardized models	• high degree of abstraction • increased productivity • delivers products with a high degree of compatibility and portability • shorter time to market	• requires considerable technical expertise • documentation is readable only by domain experts • is difficult to implement version control on modelling environment

2.12 Review of Software Development Tools

In choosing a particular programming language for the development of systems that are related to health care. A review was carried out. The figure 2.19 shows the different languages and their popularity on GitHub[87].

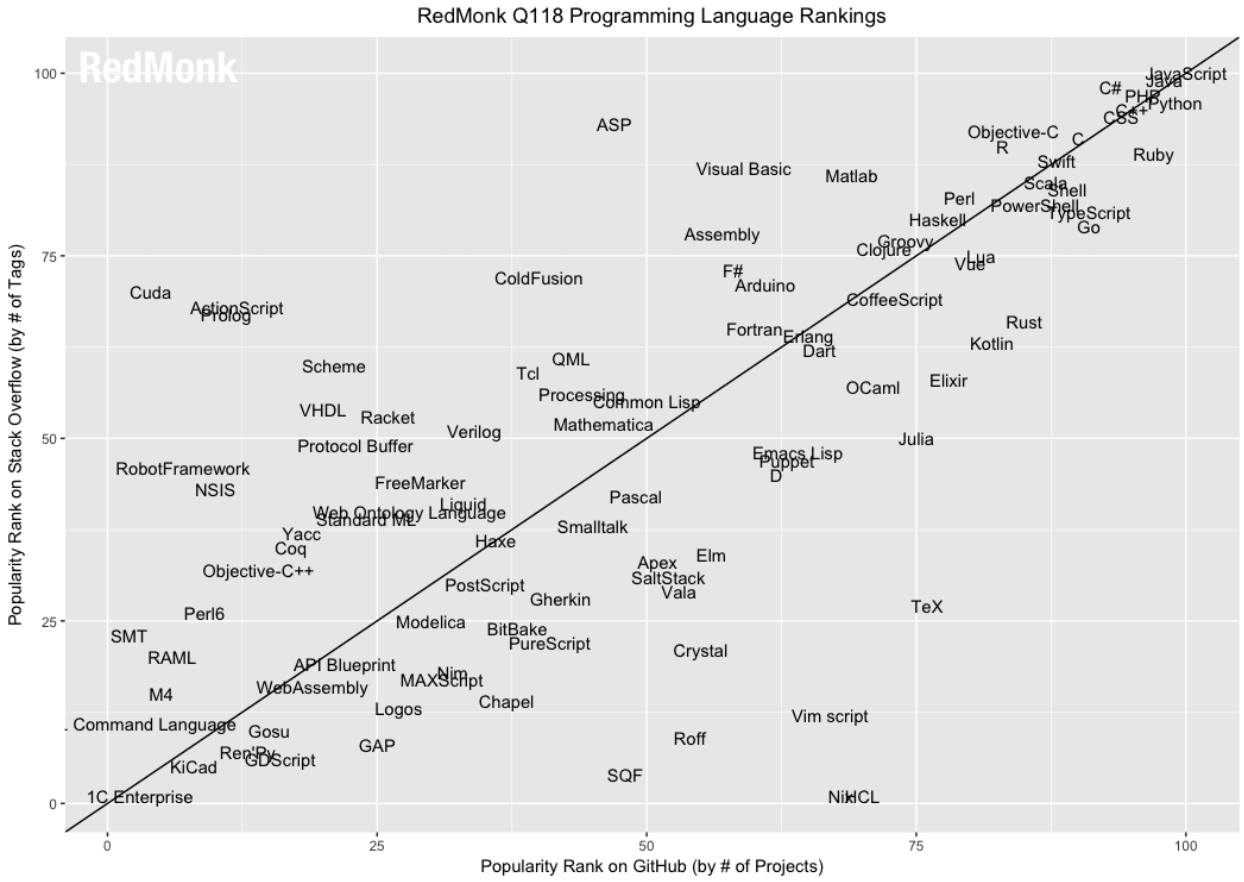

Figure 2.19: different languages and their popularity on GitHub (Source: red-monk.com/sogrady/2018/03/07/language-rankings-1-1 Java programming in healthcare software development)

Java

Java is one of the most commonly used and mature programming languages for building enterprise applications. Over the years, Java development has evolved from small applets run on a Web browser to large enterprise distributed applications run on multiple servers. Now, Java has three different platforms, or flavors, and each addresses certain programming requirements. J2EE (Java 2 Platform, Enterprise Edition) is Sun's preferred Java platform for multi-tier enterprise applications. It simplifies enterprise applications by basing them on standardized, modular components, by providing a complete set of services to those components, and by handling many details of application behavior automatically, without complex programming [88]. J2ee uses a multi-tier distributed application model. There are three tiers in the typical J2EE application model: Web presentation tier, business logic tier and data tier[89]. Java Platform, Enterprise Edition or Java EE is Oracle's enterprise Java computing platform, the platform provides an API and runtime environment for developing and running the enterprise software. It also includes network and web services, and other large-scale, multi-tiered, scalable, reliable, and secure network applications. Java EE extends the Java Platform, Standard Edition (Java SE) which provides an API for object relational mapping, distributed and multi-tier architectures, and web services.

Java’s robustness, ease of use, cross-platform capabilities and security features make it a language of choice for building an EHR system.

MySQL

The MySQL Database powers the most demanding Web, E-commerce and Online Transaction Processing (OLTP) applications. It is a fully integrated transaction-safe, ACID compliant database with full commit, rollback, crash recovery and row level locking capabilities. MySQL delivers the ease of use, scalability, and performance that has made MySQL the world's most popular open source database. Some of the world's most trafficked websites like Facebook, Google, Ticketmaster, and eBay rely on MySQL for their business critical applications [90].

Solutions, to create a Web-based Secure Electronic Health Record management application

ECLIPSE IDE

Eclipse is an open-source platform of extensible software development application frameworks, tools and run times that was initially created as a Java-based integrated development environment (IDE). Eclipse's runtime system is based on a collection of Equinox Open Services Gateway Initiative (OSGi) runtime-built open-source projects covering Java IDE, static/dynamic languages, thick/thin-client and server-side frameworks, modeling/business reporting and embedded/mobile systems[82], [85], [91]–[93].

2.13 Chapter Summary

The review of literature dealt with understanding the importance of health and its contribution to the economy of nations. The application of ICTs to health to produce the same benefits as in the business world. The contribution of record keeping to health management, electronic health records, interoperability and standards. Based on the title of the research, the researcher focused on the sematic interoperability standard using Snomed CT. A review of application of Snomed CT in third world countries was done. The chapter also looked at the implementation, tools and methods of systems. The literature helped to establish gaps in that in Zambia, so far there is not Standardized Electronic Health Record System.

CHAPTER 3: METHODOLOGY

3.1 Introduction

This chapter discusses the methods and approach taken in the collection, analysis and presentation of research data. A review of the methodology used in the baseline study is presented followed by the method used in the design, development and implementation of the prototype.

3.2 Baseline study

This section looks at the methodologies employed in the research. In order to get clear understanding of the challenges faced by the University Teaching Hospital Surgery Department, a baseline study was carried out. A combination of Quantitative and Records/Documents inspection was used for this research. This approach was selected as it allowed detailed data collection which in more insight into the problem[94]–[96].

3.3 Study Setup

The research was conducted in Lusaka University Teaching Hospital Surgery Department. Additional information for better understanding of how the surgery departments work was obtained from Ndola teaching Hospital. This was made possible by one of the Surgeon team members who was transferred to Ndola to head the Department.

In the quest to establish the Patient Record System being used at the University Teaching Hospital in Lusaka, a descriptive quantitative analysis was used. This supported further by formal and informal interviews and focus group discussion of stakeholders. Since the project of implementing and Electronic Health Record System was a corroborative effort between the department of Computer Science at the University of Zambia and the Department of Surgery at the University Teaching Hospital, the focus group were held in form of arranged meeting in which stakeholder made of experts in surgery and IT talked and highlighted the issues with the systems at the Hospital. Record inspection was done by going through all the books and files involved in the patient care management process from the time the patient arrives to the time of discharge.

Convenient sampling was used due to time and the non-accessibility of the target population. The population is made of clinicians who are very busy. A total of 48 questionnaires were administered and only 40 were successfully answered. This represented a response rate of 83%.

The questionnaire data were entered into a statistical package called SPSS version 20 and descriptive statistics showing frequency and percentage were obtained.

After analysing the result of the baseline, record inspection and focus group discussion with the members of staff of the Department of Surgery University teaching Hospital was done.

This was followed by a review from literature of Surgical Operating Room (OR) patient scheduling methods. From the studied methods one was picked and as the solution to be implemented in the system. The important design that was carried out was the application of the sematic standardisation coding system called Snomed CT. For the international version of Snomed CT was adopted for easy access and the design of incorporating it into the system was developed.

3.4 Summary of the baseline survey

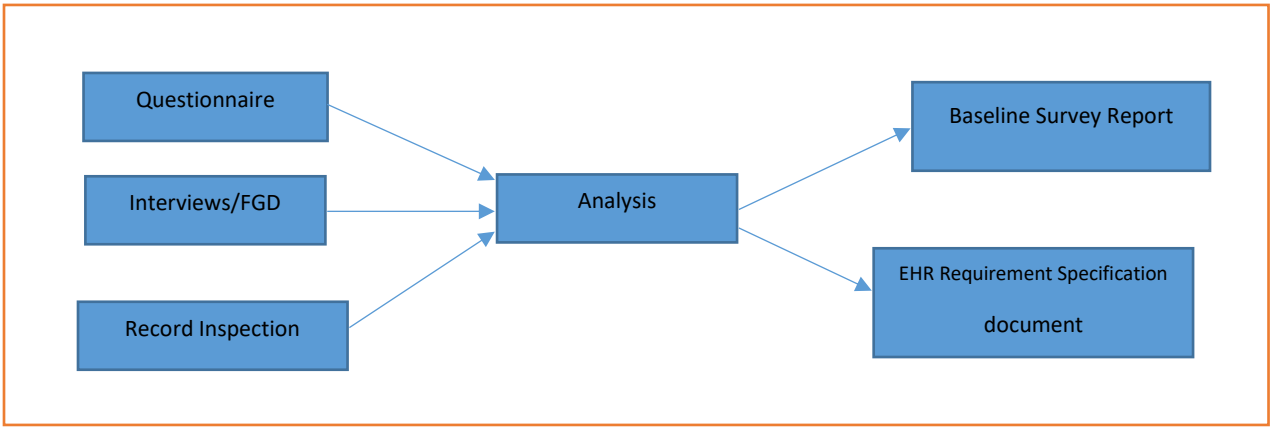

Figure 3.1: summary baseline survey method

3.5 Ethical Consideration

Confidentiality was maintained during the process of data collection. None of the respondent's identity was revealed neither did they reveal their organisations sensitive information in the questionnaire.

3.6 Limitations of the Baseline Study

Questionnaire did not capture the views of the patients on the subject matter. Additionally, not all questions were answered.

3.7 System Automation

System requirement specifications and design model of this research were derived from the quantitative data supplied though the questionnaire and the records which were inspected. The overview of the current business processes was learnt through the above stated interactions. The methodology that was used for analysis, design and development of the software system was prototyping combined with Rapid Application Development to aid the process of coming up with a prototype.

3.8 Current Business Process OR Patient Flow

The University Teaching Hospital of Lusaka is a national referral centre for all cases which the rest of the general Hospitals cannot handle. The Surgery Department plays a key role in that it handles the bulk share of referral patients. Additionally, almost all accidents victims within the province and surrounding areas are directly brought to the Department.

There are two types of patients which arrive at the Hospital. Emergency cases (Polytrauma Patients or mass casualty event victims) and cold cases. The cold cases follow through all the routine processes as outline. The figure 3.2 is the patient flow chart showing how the patients flow.

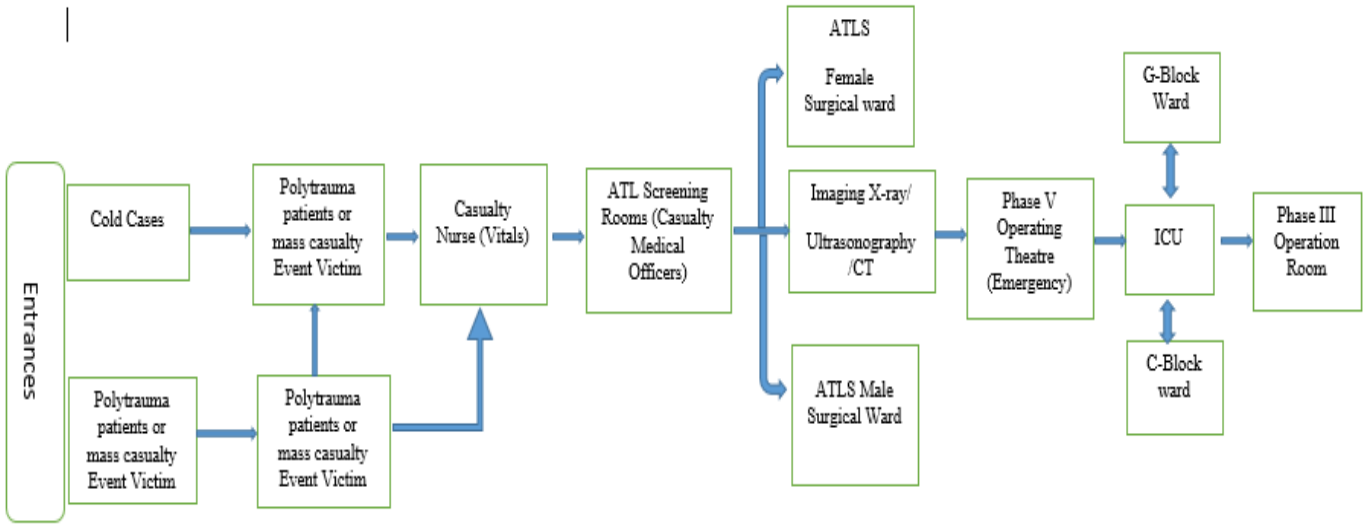

Figure 3.2: Patient flow chart at UTH surgery

The patient traffic chart below indicates two possible points of entry into the Hospital depending on the condition of the patient. Post trauma patient or Mass Casualty event victims will enter through the emergency room, if they have relatives with them, registration is done on their behalf,

the nurse will collect vitals and at the same time screening by the medical officers at casualty is done.

The other entry is for cold cases or non-emergency cases. These start off by first registering their details with the clerk at the reception, from here they proceed to the casualty nurse for collection of the vitals and once they finish here, the proceed to the screening rooms (casualty medical officers.

After the patient has been screened, they can be taken for further imaging (xray, ultrasonography, CT), then proceed to the surgical ward which can either be male or female ward depending on their gender. Or after a patient has been screened, they can proceed to either the male or female surgical ward depending on their gender. The next place they go to is Phase V Operating Theatre (Emergency OR). From this theatre, a patient can be taken to C-block wards or Intensive Care Unit (ICU) or G-block wards for recovery until discharge. If the patient is taken to ICU, it means they mean need to go to Phase III Operating room. The chart below shows the patient flow chart

3.9 System Architecture

The system under development is networked based system with a cloud computing architecture component comprising the server side hosting Snomed CT application and the database. The Network client system is installed on the user machines, which are connected to the network. The system will be accessible through the World Wide Web (www) or a direct link on the network depending on user's preference and location. Secure Access will be implemented through the Secure Socket Layer Communication between clients and server to encrypt data transmitted maintain privacy and confidentiality. Use of digital certificates for authenticating communication between the server and clients will be implemented. A trusted third party agent will handle the management of certificates. To further enhance security, a firewall is going to be implemented in the network especially for those accessing the system within the premises of the Hospital. The figure 3.3 shows the conceptual model of the system architecture.

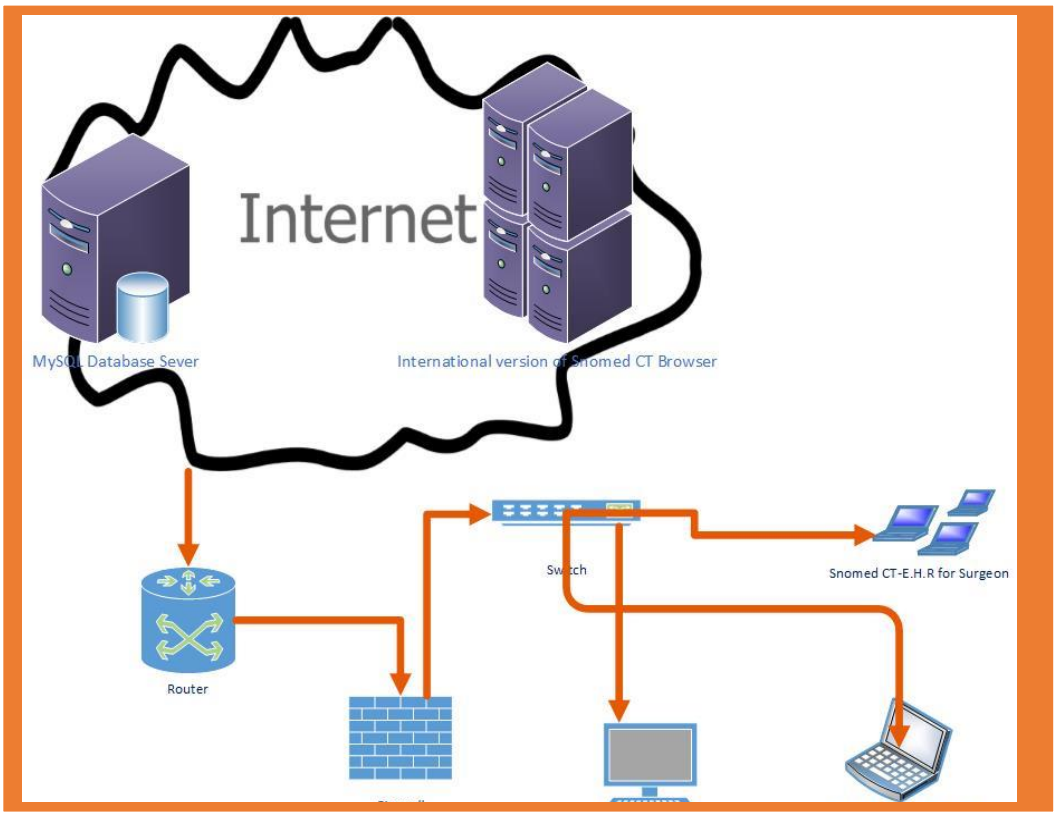

Figure 3.3: General system architecture

The system is implementing a semantic standard called Systematic Nomenclature for Medical and Clinical Terms (SNOMED CT) hence the diagram in figure 3.4 shows the conceptual working of the system in view of system interoperability and semantic interoperability.

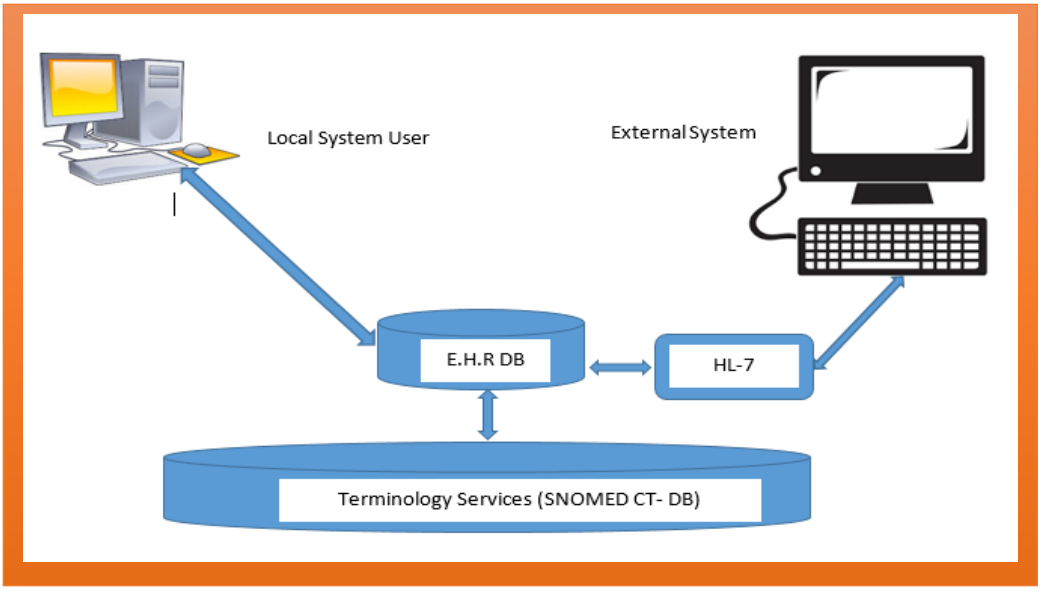

Figure 3.4: conceptual working of the system in view of system interoperability and semantic interoperability

3.10 Functional Specification

In order to come up with the system requirements specification for proposed system, an object oriented approach was used. The requirements derived from the best system operations standards. These requirements show user needs and purpose of the system. requirements are categorised into two, that is, functional and non-functional requirements. Functional requirements are statements or listing of services a system will provide and non-functional requirements are limitations or constraints on services or functions expected from the system. These functions serve as a guide on how far the system can be in terms of implementation.

3.10.1 Requirement determination

Requirements can be categorized in different ways into different types but broadly fall into Functional requirements and Non-functional requirements. Functional requirements describe the system should be able to perform: they are explicit features and functionalities that the users want for the system to work. Non-functional requirements describe what the system should be or specify the criteria for judging the system: they are quality attributes of the system such as usability, stability, security, scalability, maintainability[97]. The functional requirements are further subdivided into high level and low level.

The high- level functional requirements for the system under development are shown in table tabulated below:

Table 3: Low-Level functional requirements

SR No.	Processing Requirement	Required Input	Desired Output
1	User Login	User enters username, password and capture characters	The user authorized profile menu is displayed
2	Create lookup fields	Lookup fields, e.g district are entered using input device	The lookup fields are stored in the database

3	Create users	User details through input device are supplied	The user details are stored in the database
4	Update/Delete User Details	Username, password and profile	Updated database
5	Register Patient	Patient details are entered using input device	Patient details are saved in the database
6	Update/Delete patient details	Username and password according to level of privilege	Updated database
7	Enter Hospital employee details	Hospital employee details are entered	Hospital employee details are stored in the database
8	Update/Delete employee details	Username and password according to level of privilege	Updated database
9	Schedule patient	Patient ID, Doctor or Surgeon Id and Date	Patient details queued for appointment to respective doctor/surgeon
10	Attend to patient	Patient ID, Problem list, Snomed CT Concept ID, Prescription	Patient doctor attendance details are stored in the database
11	Patient Pre-Operative details	Patient Id, Preoperative details, operation scheduled date	Patient Preoperative details and operation schedule date stored in the database
12	View patient details	Query using Patient ID	Correct patient read-only form with details

13	Patient Post-operative details	Patient Id, Operation Id, Post-operation details	Patient post-operation details are stored in the database
14	View Patient Pre-operative details	Query using Patient Id and Pre-operation Schedule ID,	Correct Read-Only form with details Pre-operation
15	View Patient Post-operative details	Query using Patient ID, in the list of post-operation	Correct read-only form with details of patient post-operation details.

The low level Functional requirements given in the table below:

Table 4: low-level functional requirement

HLFR No.	**High Level Functional Requirement (table 3)**	**Low-level requirements expected**
1	User Login	1.1 User must submit login credentials and characters from the capture. 1.2 The system should only permit authenticated users to login
2	Create lookup fields	2.1 The system must permit authenticated user to create lookup fields 2.2 The system should only permit authenticated administrator to create lookup fields
3	Create users	3.1 The system must permit authenticated user to create users 3.2 The system should only permit authenticated administrator to create users
4	Update/Delete User Details	4.1 The system should only allow authenticated staff with administrative access

		rights to make amendments or deletions to information stored in the database
5	Register Patient	5.1 The system should allow authenticated user to Register patients. 5.2 The system must enforce that only authenticated staff access the system
6	Update/Delete patient details	6.1 The system should only allow authenticated staff with administrative access to make amendments to patients details or deletions to information stored in the database.
7	Enter Hospital employee details	7.1 The system should allow authenticated user to Enter Hospital employee details 7.2 The system must enforce that only authenticated staff access
8	Update/Delete employee details	8.1 The system should allow authenticated staff with administrative access to make amendments to employee details or deletions of employee details stored in the database.
9	Schedule patient	9.1 system should allow authenticated user to create a schedule of patients. 9.2 The system must only permit staff with a profile for patient scheduling to access
10	Attend to patient	10.1 The system should allow authenticated users to attend to patients 10.2 The system should permit only patients queued or scheduled up to be attended by a staff

		10.3 the system must permit allow staff authenticated as nurse, doctor or surgeon to attend to patients
11	Patient Pre-Operative details	11. 1 The system should allow authenticated user to enter patient pre-operative details 11.2 The system should allow only staff with doctor, anesthetist or surgeon profile to access
12	View patient details	12.1 The system should allow authenticated users to view patient details. 12.2 The system should only allow authenticated user view patient details only when it is necessary
13	Patient Post-operative details	13. 1 The system should allow authenticated user to enter patient post-operative details 13.2 The system should allow only staff with doctor, anesthetist or surgeon profile to access
14	View Patient Pre-operative details	14.1 The system should allow authenticated users to view patient Pre-operative details. 14.2 The system should only allow authenticated user view patient Pre-operative details only when it is necessary
15	View Patient Post-operative details	15.1 The system should allow authenticated users to view patient Post-operative details. 15.2 The system should only allow authenticated user view patient Post-operative details only when it is necessary

3.11 Nonfunctional Requirements

Table 5 gives details of the nonfunctional requirements the system shall assume.

Table 5: nonfunctional requirements

ID N	Requirements
1	The system must support running in high availability mode (active-passive)
2	Ability to perform full system recovery from scheduled backup files with RTO (recovery time objective) less than 4 hours
3	There should be at least one documents scanning workstation on each clinical floor.
4	The system must be capable of connecting to a centralized enterprise database server.
5	The system must be capable of storing content to a network mapped drive provided by UTH.
6	The system management interface must be accessible via web browser.
7	The system must be capable of storing unlimited number of documents.
8	The system must be capable of performing full documents search in a reasonable amount of time.

3.11.1 System Modelling and Design

The final stage was to identify the functional requirements as shown in table 5. These are software requirements that define the system behaviour or functions that the system is supposed to perform. Use case diagrams which model the interaction between a system and external actors were used to show the functional requirements of the system as would be viewed by users [98]. These are able to represent system functions or goals, which will respond to external inputs, and accomplish a task that will be of use from s user's point of view. Six primary actors and roles in the system were identified before drawing up the high-level model. Figure 23 illustrates a top-level user interaction between system actors and applications. This also shows interaction between actors and their role in the system.

TOP LEVEL USE CASE

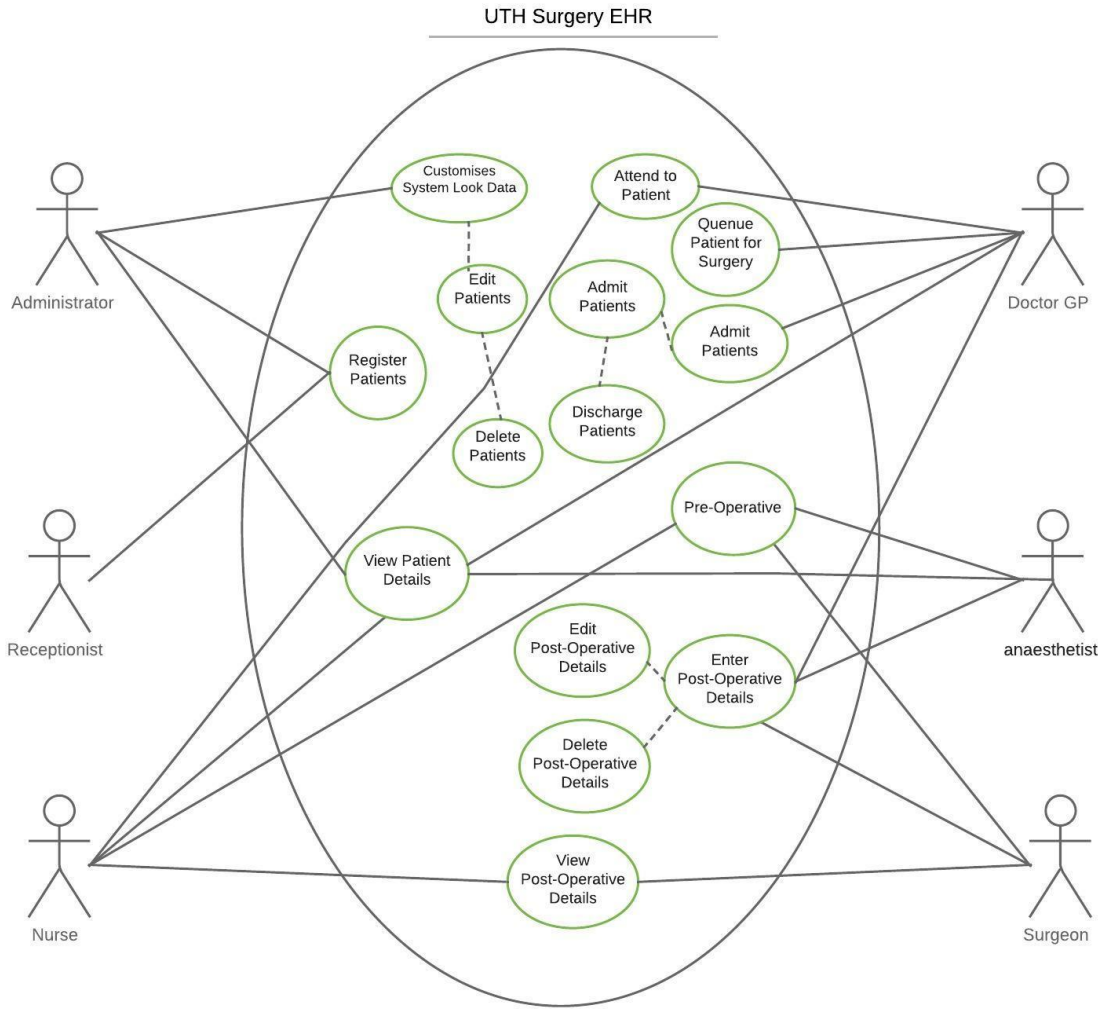

Figure 3.5: System Level use case diagram

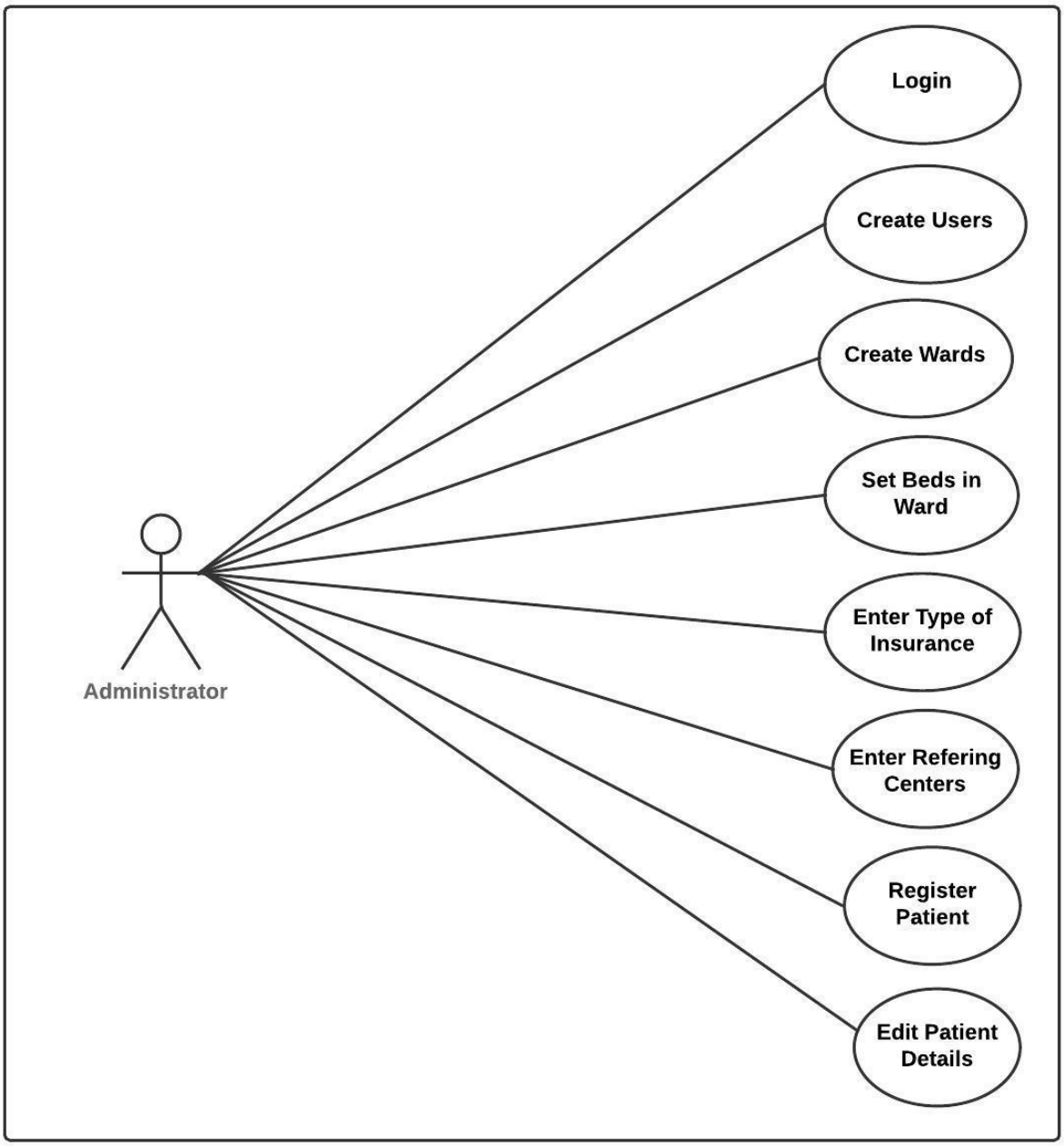

Figure 3.6: Administrator use case diagram

Based on the illustrated use case diagram, the administrator can login to the system, create users, create wards, set beds in wards, set type of insurance, enter referring centres, register patients and edit patient details.

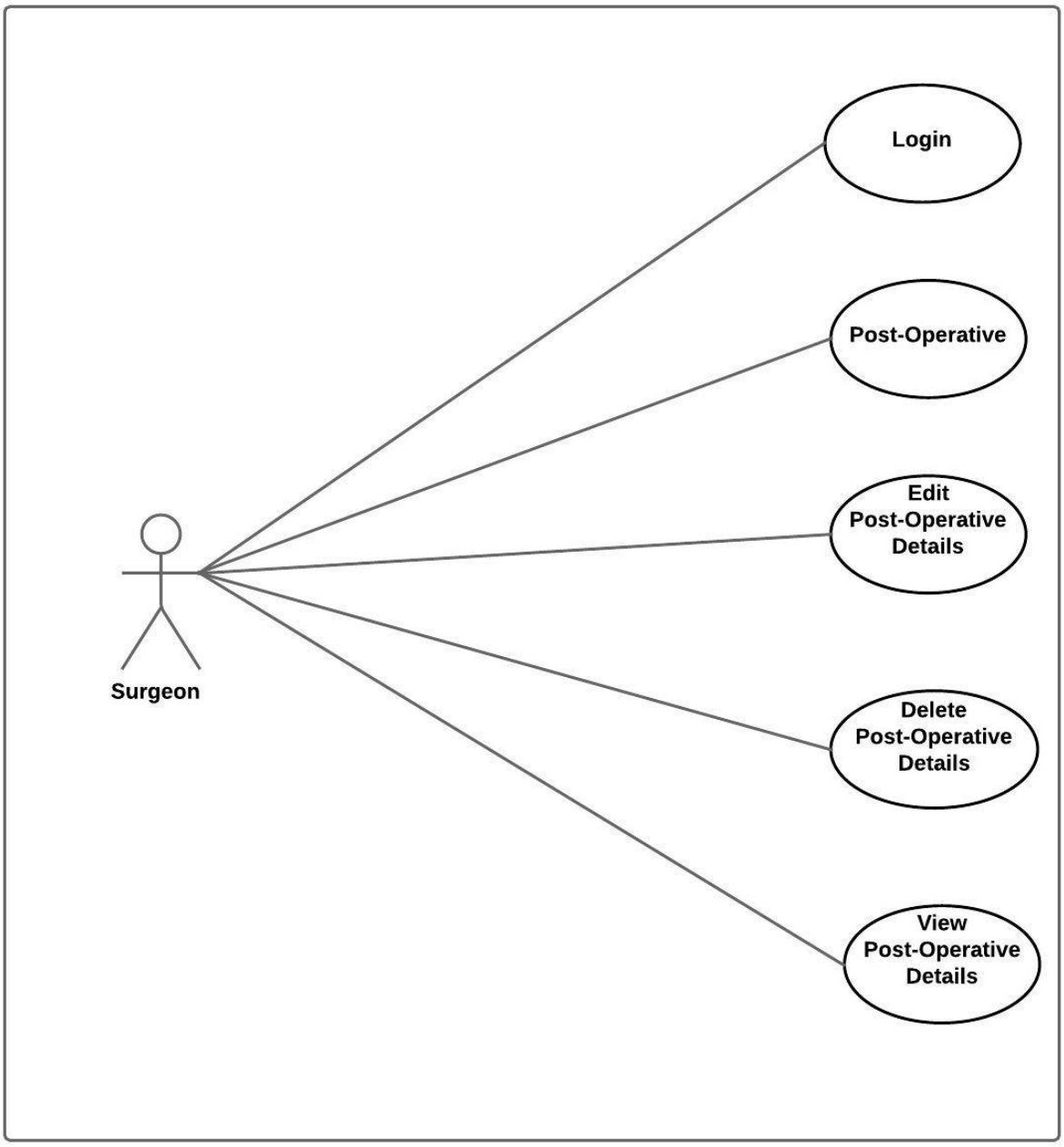

Figure 3.7: Surgeon Use Case Diagram

Based on the illustrated use case diagram, the Surgeon as actor, can login to the system, Enter Pre-Operative Patient data, Edit Post-Operative Patient Data, Delete Post-Operative Patient Data and View Post-Operative Patient Data.

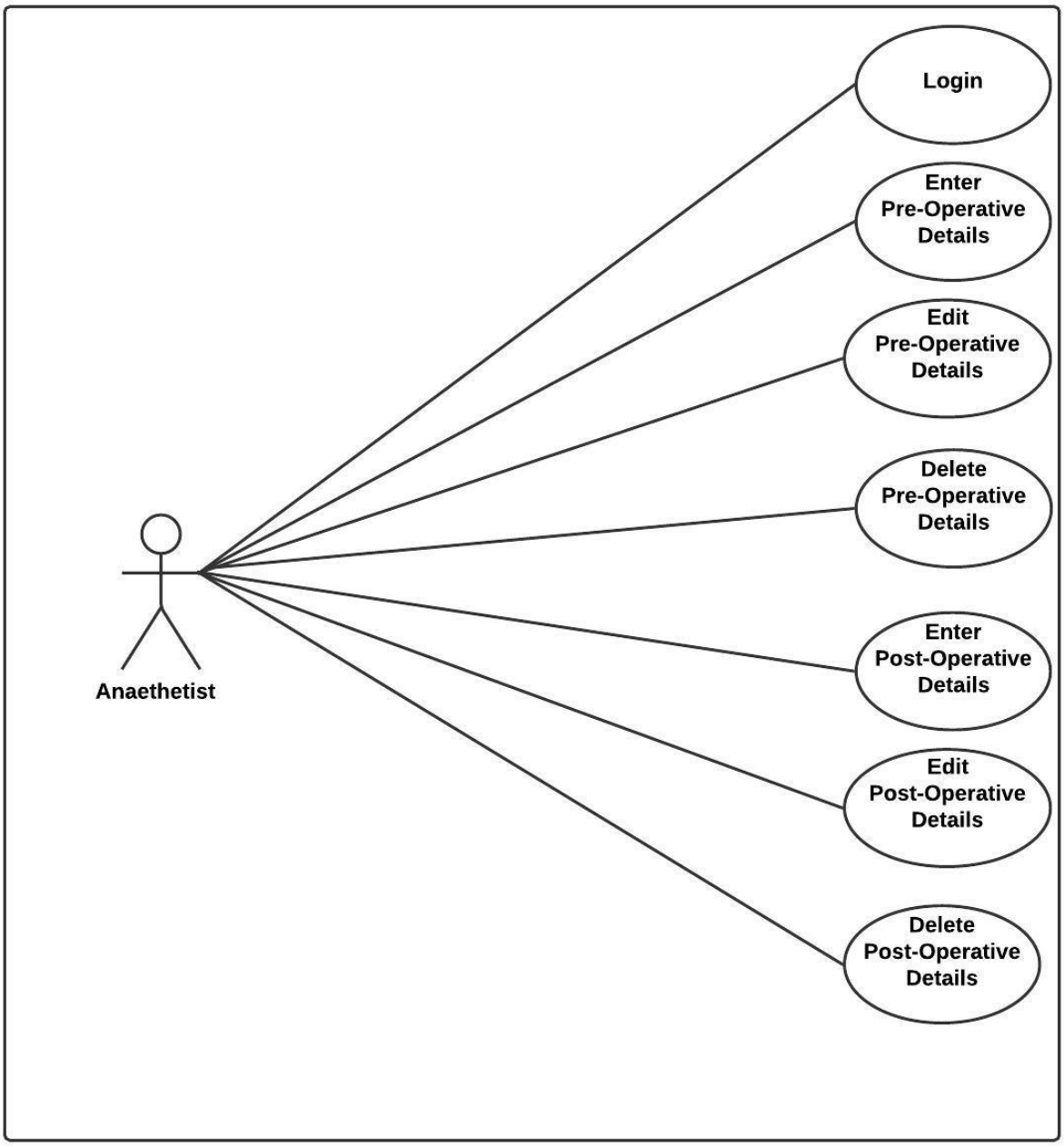

Figure 3.8: Anesthetist use case

Based on the illustrated use case diagram, the Anaesthetist as actor can login to the system, Enter Pre-Operative Patient data, Edit Pre-Operative Patient data, Delete Pre-Operative Patient Data, Enter Post-Operative Patient data, Edit Post-Operative Patient data and Delete Post-Operative Patient Data

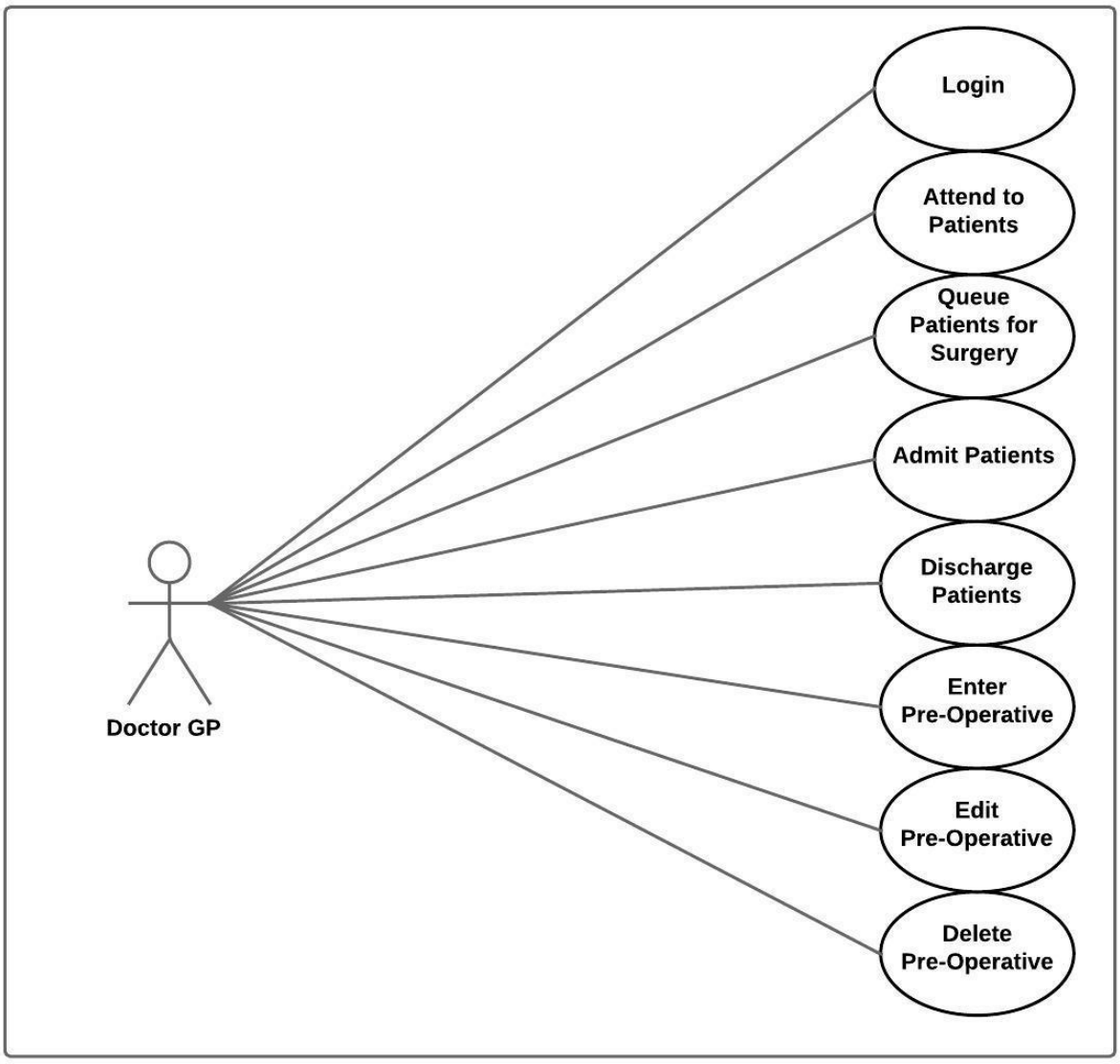

Figure 3.9: Doctor GP use Case Diagram

Based on the illustrated use case diagram, the Doctor DP as actor, can login to the system, Admit patients, Discharge Patients, Enter Pre-Operative Patient data, Edit Pre-Operative Patient Data, Delete Pre-Operative Patient Data.

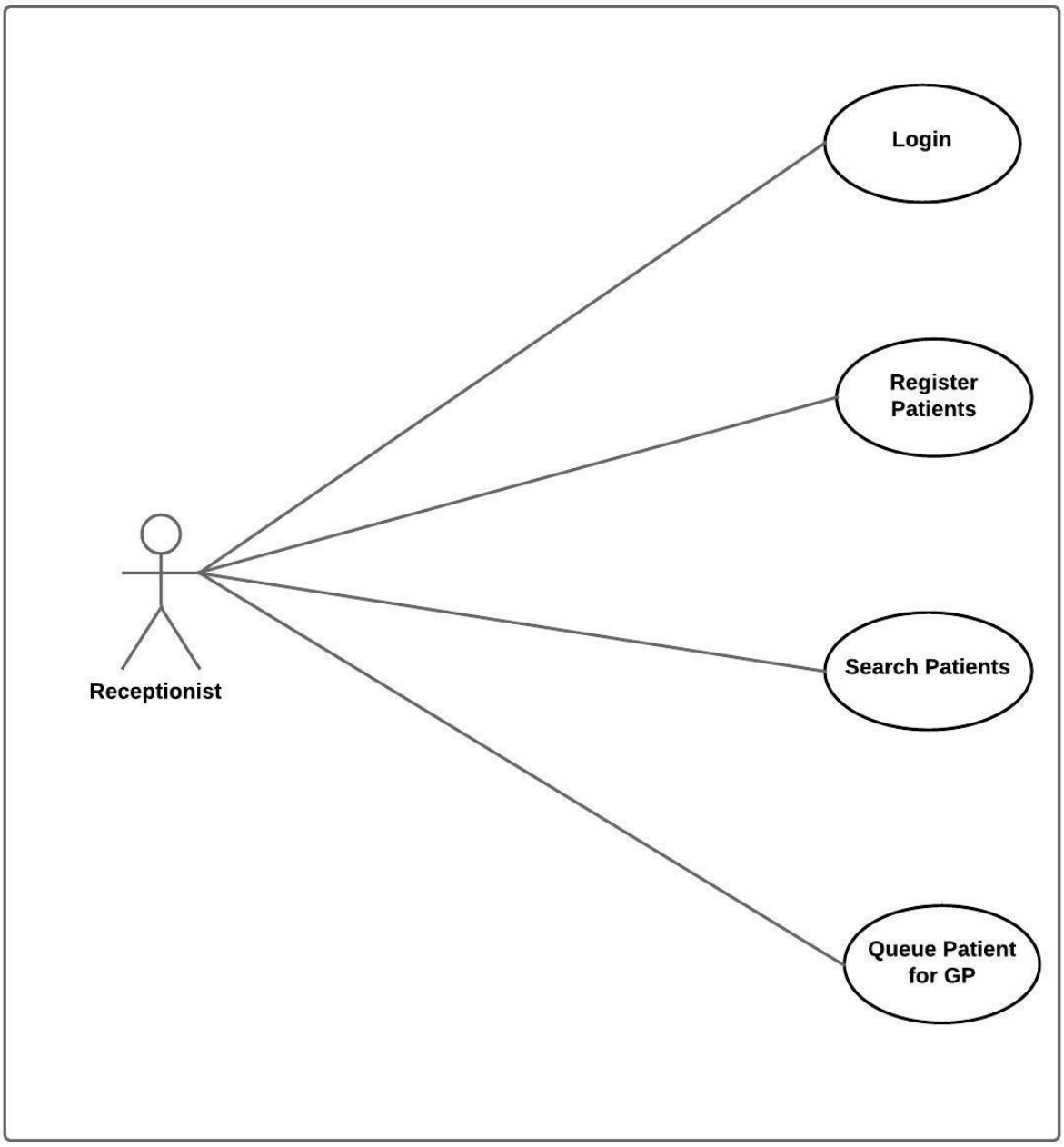

Figure 3.10: Receptionist Use Case Diagram

Based on the illustrated use case diagram, the Receptionist as actor, can login to the system, Register Patients, Search Patients, Queue patients for GP.

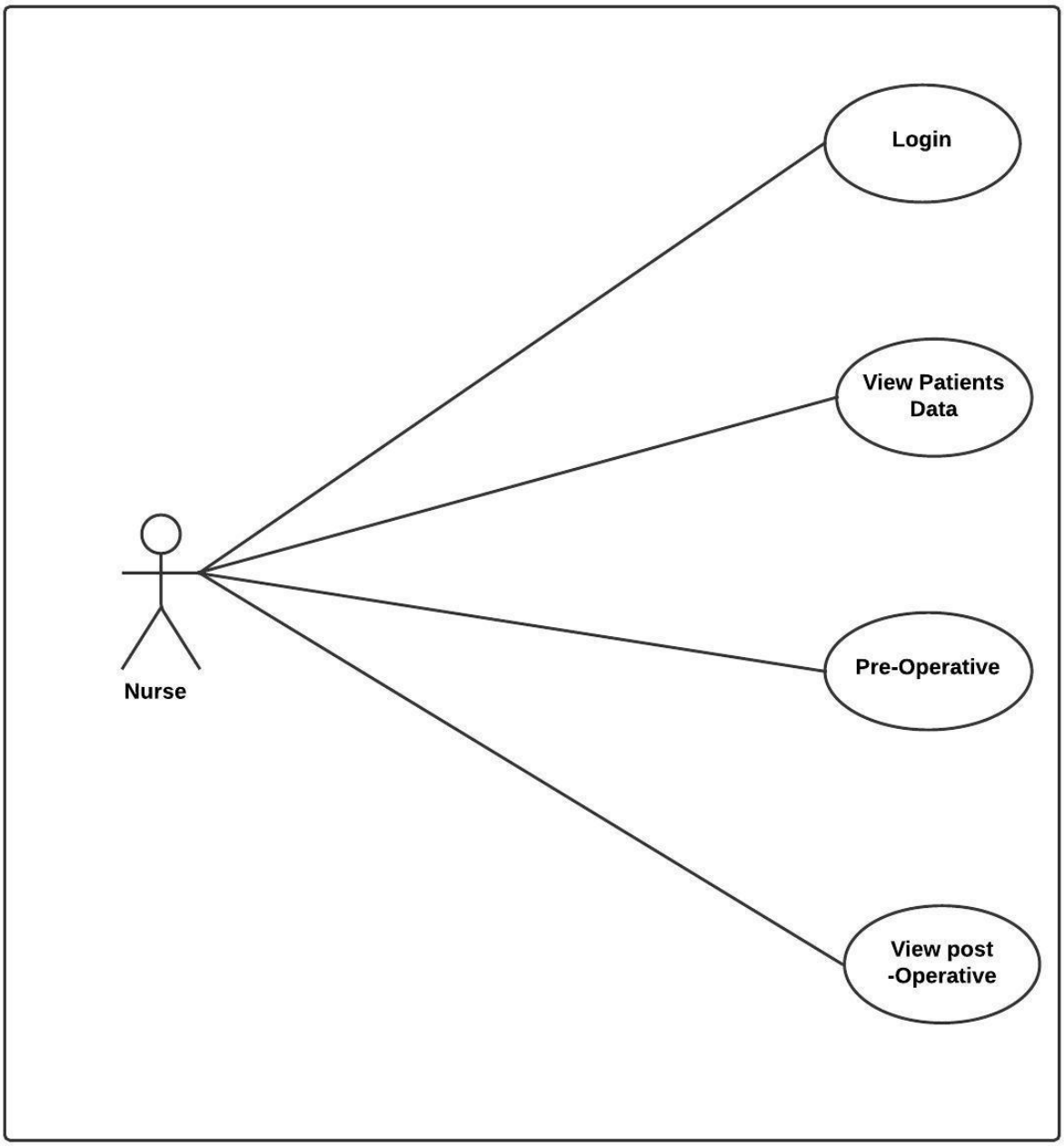

Figure 3.11: Nurse Use Case Diagram

Based on the use case diagram in figure 30, the nurse can login to the system, and view patient details, enter vital signs information in the pre-operative form, and can view post operation information.

Sequence Diagrams and Communication diagrams

Sequence diagrams are not just used to model the interactions between the actor and the objects in a system but also show interaction between the objects themselves[99]. As such, the sequence diagram is interaction diagram which shows the dynamic side of system[98]. In UML, a communication diagram shows the interactions between the objects or roles associated with lifelines and the messages that pass between lifelines[100]. As the matter of fact, the communication and sequence diagrams a have three symbols[101], which are used in the communication diagrams and these, are: Boundary class stereotype, control class and entity class stereotype [102]–[104].

Communication diagram symbols

Icon	Description
Actor	An Actor is a user of the system; user can mean a human user, a machine, or even another system or subsystem in the model.
Object	An Object is a particular instance of a Class at run time.
Boundary	A Boundary is a stereotyped Object that models some system boundary, typically a user interface screen.
Control	A Control element represents a controlling entity or manager that organizes and schedules other activities and elements.
Entity	An Entity is a stereotyped Object that models a store or persistence mechanism that captures the information or knowledge in a system.
Package	Packages are used to organize your project contents, but when added onto a diagram they can be use for structural or relational depictions.

Figure 3.12: symbols of communication diagram

Figure 3.12 shows the symbols, which are used to express communication diagrams.

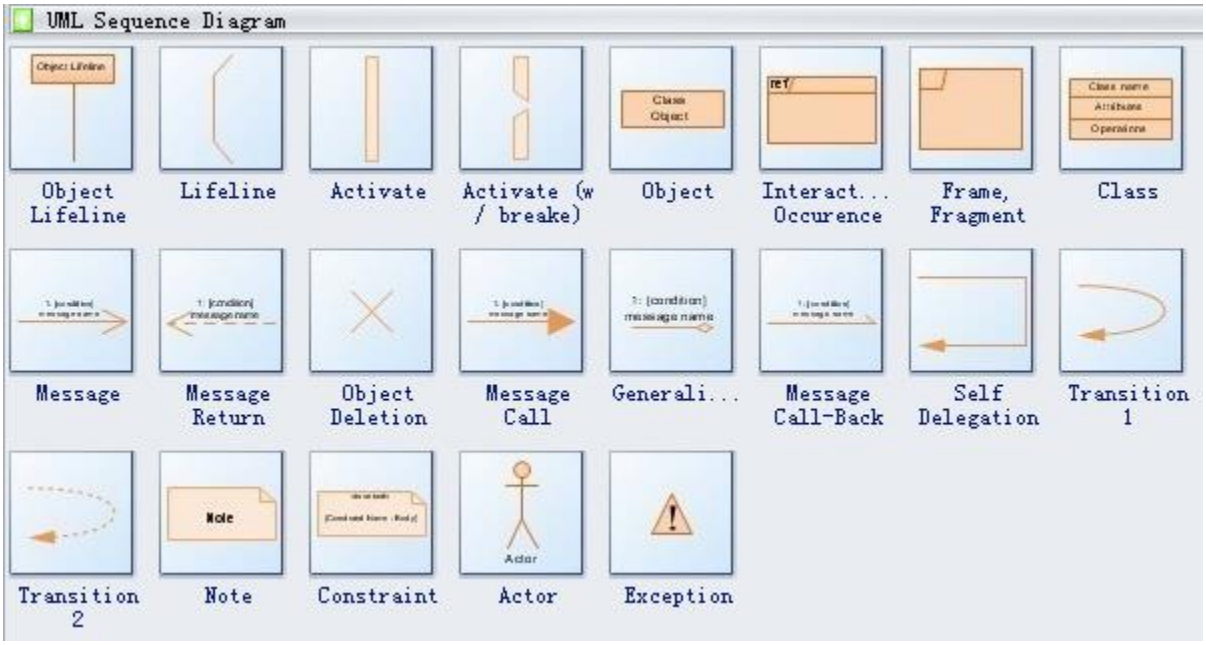

Figure 3.13: Sequence Diagram symbols

Figure 3.13 is display of the mostly used symbols in drawing sequence diagram.

Message	Description
→	**Synchronous:** A synchronous message between active objects indicates wait semantics; the sender waits for the message to be handled before it continues. This typically shows a method call.
→	**Asynchronous:** With an asynchronous flow of control, there is no explicit return message to the caller. An asynchronous message between objects indicates no-wait semantics; the sender does not wait for the message before it continues. This allows objects to execute concurrently.
⇠	**Reply:** This shows the return message from another message.

Figure 3.14: Symbols and description of messages

Figure 3.14 is message type and respective description used in sequence diagrams.

Login

Figure 3.15 shows how the login user interface (UI) is launched by instantiating the control object. As such, the control object prompts the use for login credentials in order for the user to be authenticated upon entering correct login credentials. Subsequently, a request is sent to the user entity object to get the user account from the database. It is at this point that he controls object will finally ask the boundary object to display the landing page to the user at the application interface of the computer being used.

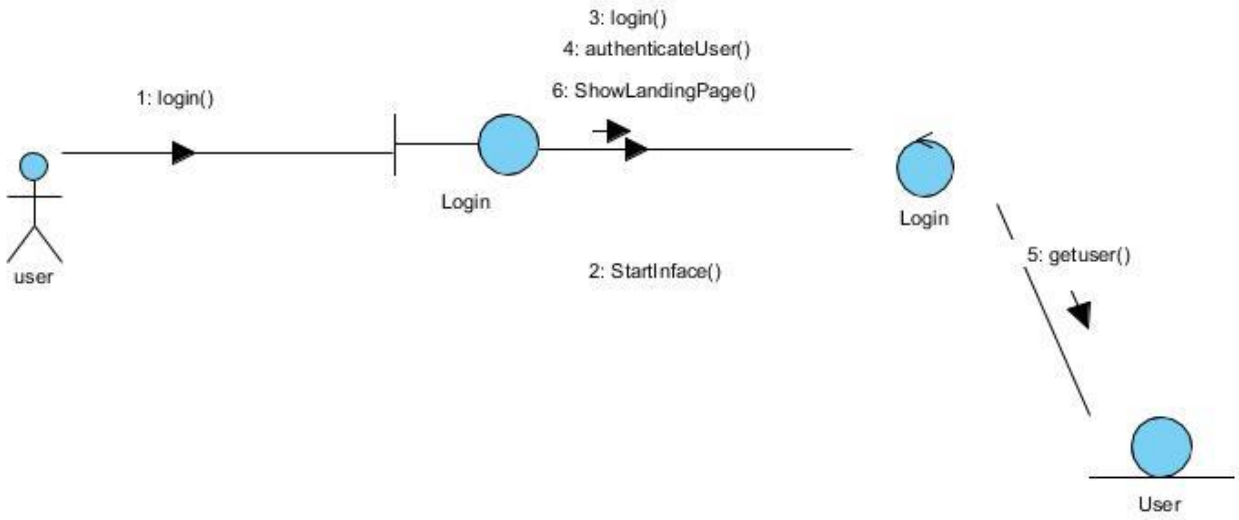

Figure 3.15: Login Communication Diagram

Figure 3.16 is a sequence diagram, which follows the communication diagram shown in figure 3.15. The sequence diagram is the sequence of events for the Login Use case.

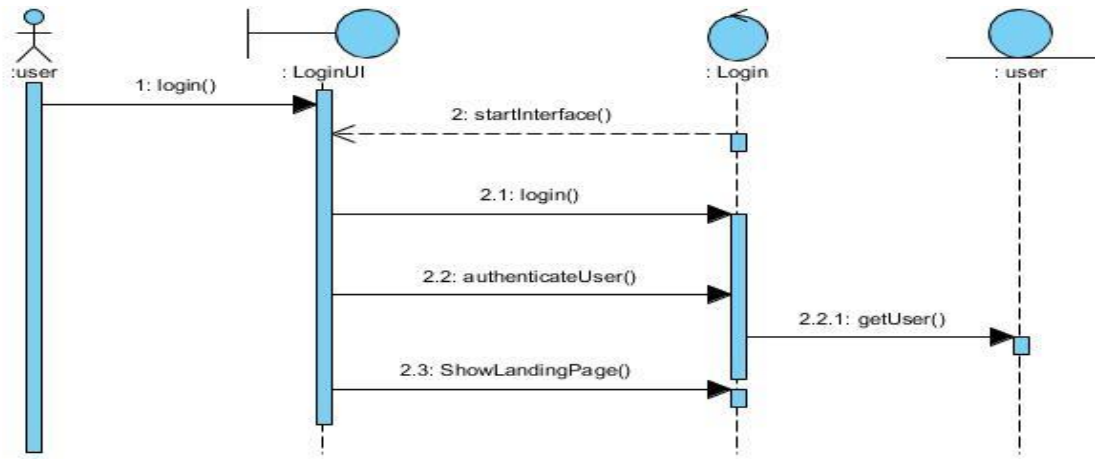

Figure 3.16: Login Sequence diagram

VIEW PATIENT DETAILS

One of the activities the receptionist undertakes is the viewing of the patient's details whenever a patient appears for medical attention. Figure 3.17 shows the receptionist's interaction with the system when viewing the patient's information.

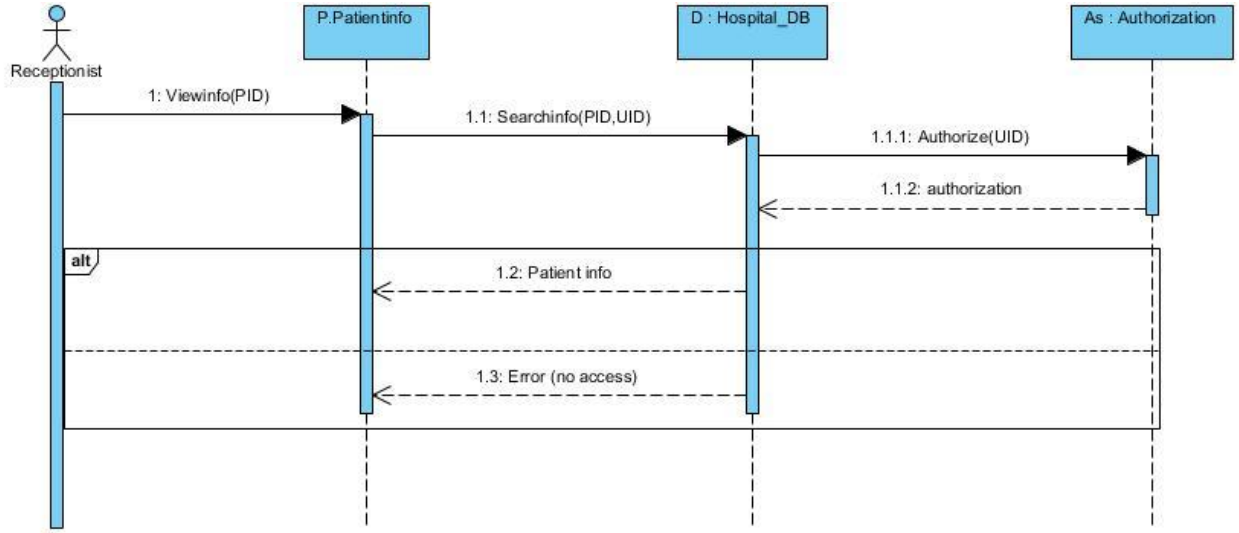

Figure 3.17: Receptionist View

QUEUE PATIENT FOR ATTENTION

The process of queuing a patient is shown in figure 3.18 sequence diagrams.

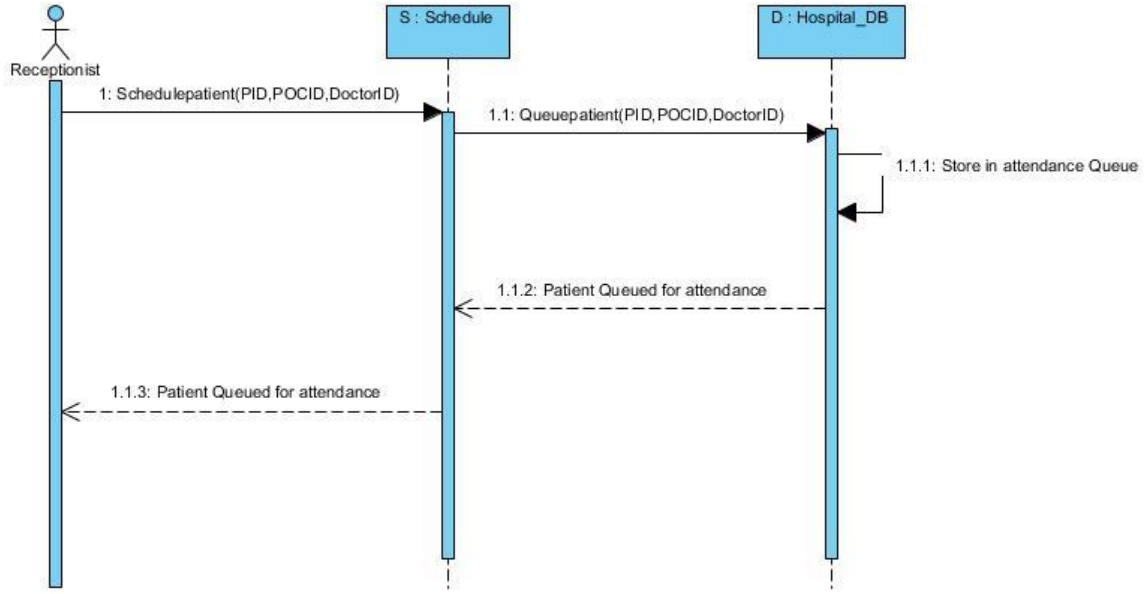

Figure 3.18: Queue Patient

The receptionist performs the queue patient process. The sequence diagram depicts the process and interaction of the receptionist and the system objects.

ATTEND TO PATIENT

The use case for attending to patient involves the Doctor or General practitioner's interaction with the system and the patient.

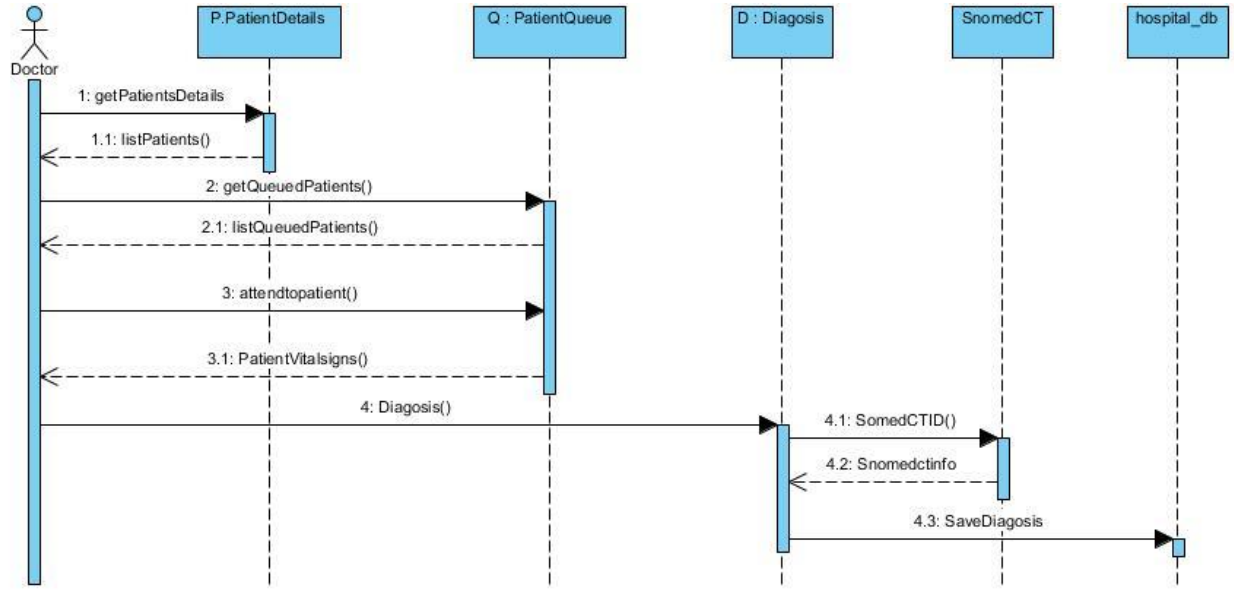

Figure 3.19: Doctor attend to patient

Figure 3.19 shows the sequence diagram for the use case of doctor attending to patient. The doctor views the list of patient's details that in the system, then from their queued patients can be seen. The doctor then picks the first patient to attend. During diagnosis, the doctor interact with Snomed CT to get the Snomed CT Concept Id and description in order to make them part of the diagnosis.

Record Vital Signs

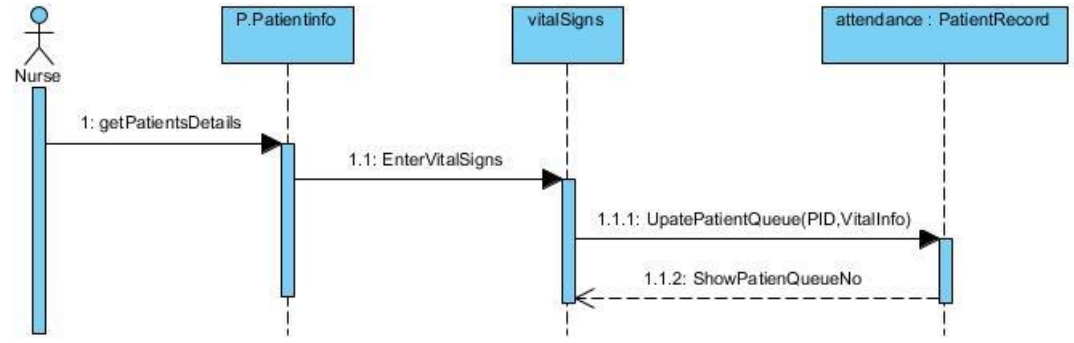

Figure 3.20: Nurse Sequence Diagram

Figure 3.20 is sequence diagram involving the nurse in recording patient's vital signs. The sequence diagram show the interaction of the Nurse with the system.

Pre-Surgery Procedure

The Anaesthetist is involved with the patient prior to surgery. The Anaesthetist sequence diagram is shown in figure 3.21. The sequence of interaction with the system is depicted.

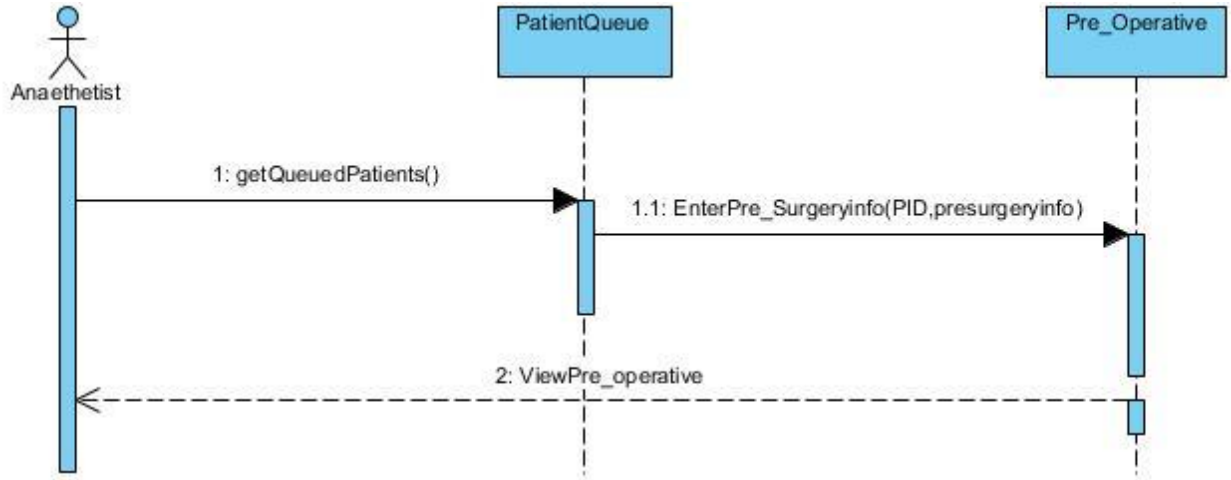

Figure 3.21: Anesthetist sequence diagram

Class Diagram

A UML Class diagram describes the object and its static structure, in particular, the things that exist (such as classes and types), their internal structure, and their relationships to other things. The UML class diagram for the Snomed CT Electronic Health Record system is represented in Figure 3.22.

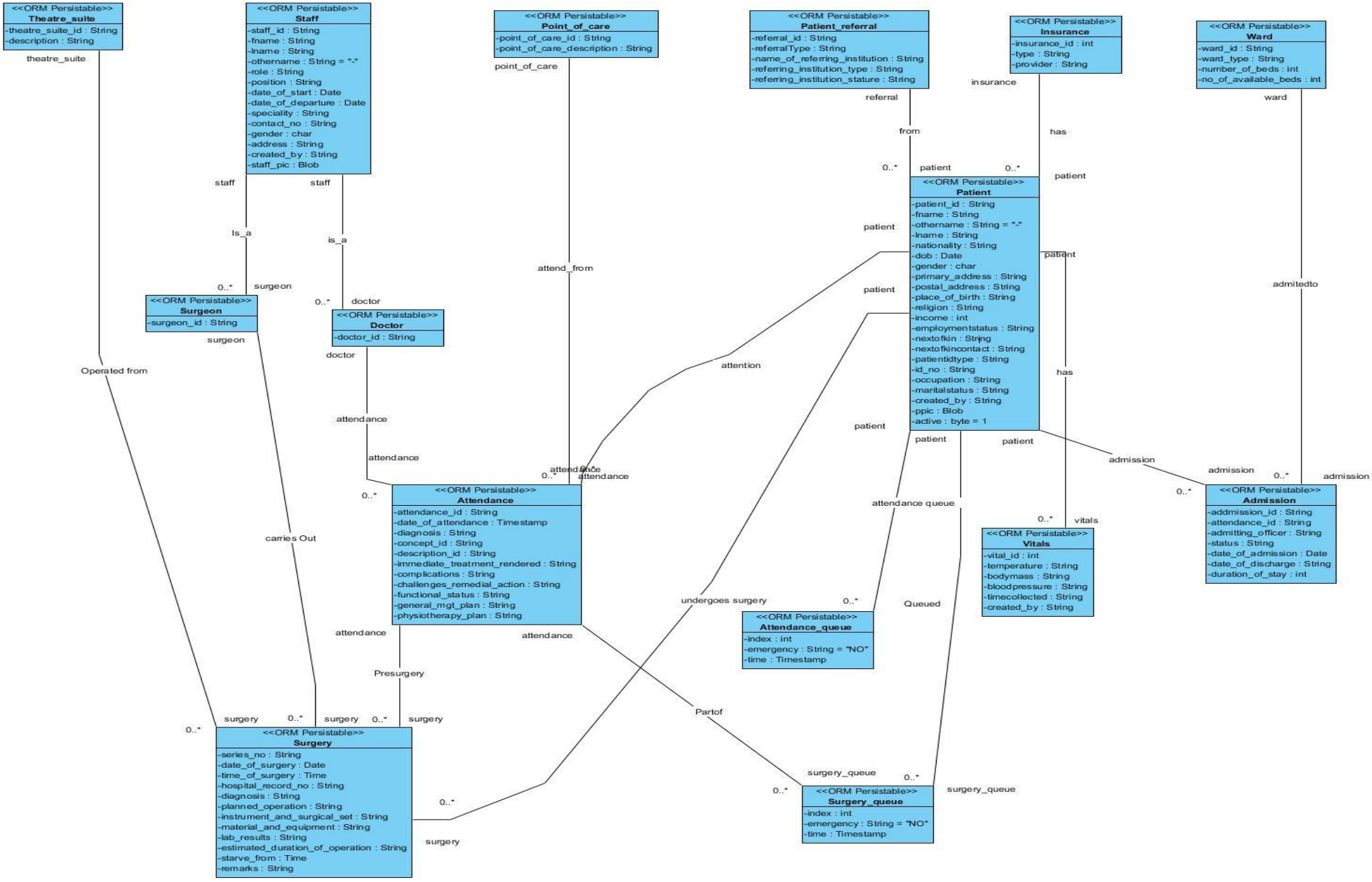

Figure 3.22: Snomed CT Electronic Health Record System Class Diagram

Data Models

The entity-relationship E-R data model uses a collection of basic of basic objects, called entities, and relationships among these objects. Entities are a collection of objects or concepts that are identified by the enterprise as having an independent existence and share common characteristics.

The ER diagram in Figure 3.23 for the Snomed CT standardized Electronic Health Record System is presented. It shows how the system's tables are linked to interact with one another to store and retrieve data from the database.

ENTITY RELATIONSHIP MODEL

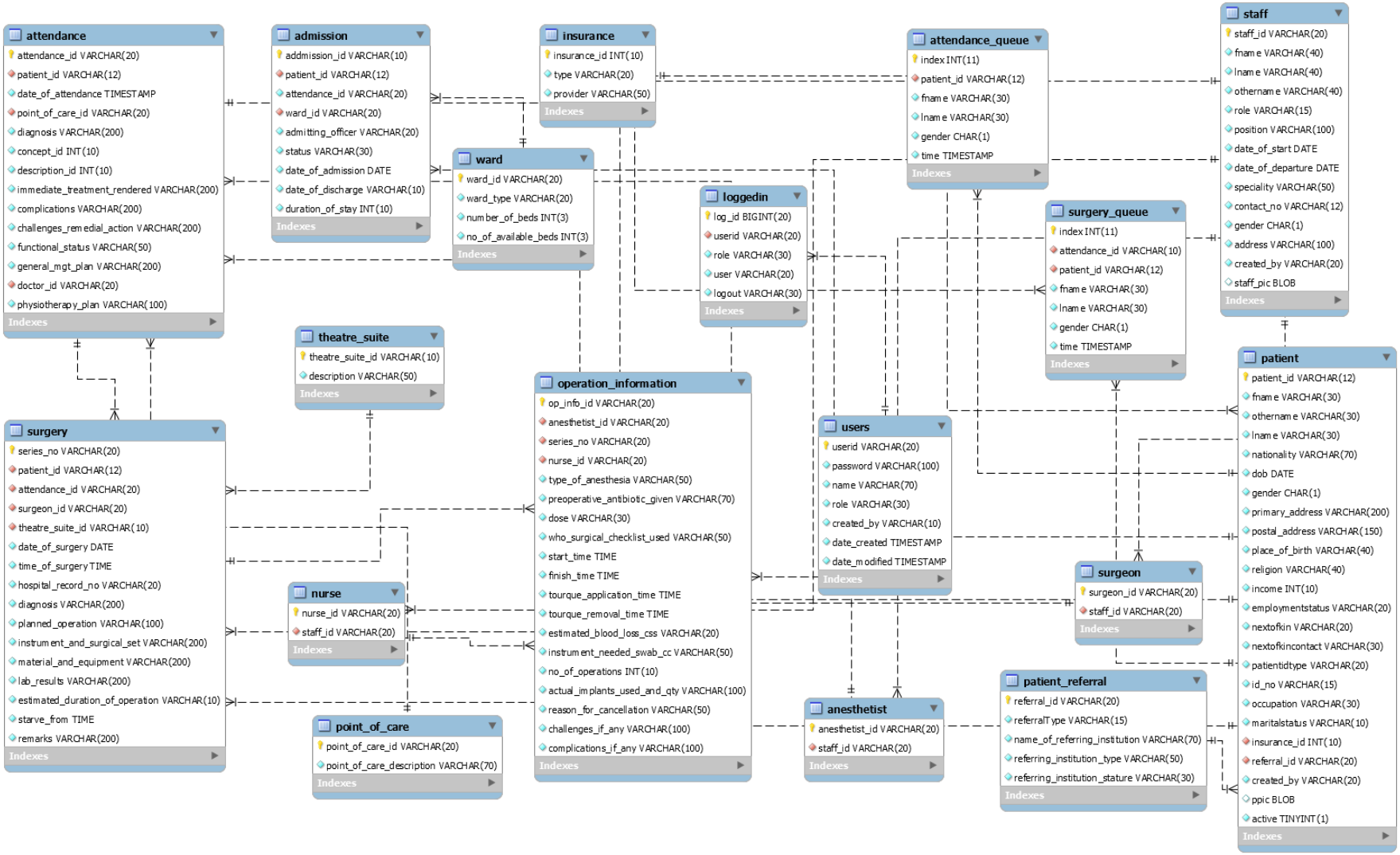

Figure 3.23: ER DIAGRAM

The entities and attributes that constituted the Snomed CT standardized Electronic Health Record System are tabulated in the Data Dictionary.

DATA DICTIONARY

Table 6 below is the Data Dictionary showing Entity name, attribute name, data type, constraints and description.

Table 6: Data Dictionary

	Entity Name	**Field or attributes**	**Data type**	**Constraints**	**Description**
	Login				
		log_id	bigint(20)	Not Null	login unique identification
		userid	varchar(20)	Not Null	User unique identification
		role	varchar(30)	Not Null	User's role
		user	varchar(20)	Not Null	User
		logout	varchar(30)	Not Null	Exit
	Entity name	**Field or attributes**	**Data type**	**Constraints**	**Description**
	Admissionde-tails				
		addmission_id	varchar(10)	Not Null	Unique identification for admission
		patient_id	varchar(12)	Not Null	Unique identification of the patient
		attendance_id	varchar(20)	Not Null	Unique identification of the attendee to the patient
		ward_id	varchar(20)	Not Null	Unique identification for the ward
		admitting_officer	varchar(20)	Not Null	Officer admitting the patient
		status	varchar(30)	Not Null	Patients status
		date_of_admission	date	Not Null	Patients admission date
		date_of_discharge	varchar(10)	Not Null	Discharge date of the patient
		duration_of_stay	int(10)	Not Null	Duration the patient was admission for
	Entity type	**Field or attribute**	**Data type**	**Constraints**	**Description**
	ANESTHE-TIST				
		anesthetist_id	varchar(20)	Not Null	Anesthetist unique identification
		staff_id	varchar(20)	Not Null	Staff unique identification

	Entity Name	Field or attributes	Data type	Constraints	Description
	ATTEND-ANCE				
		attendance_id	varchar(20)	Not Null	Attendance unique identification
		patient_id	varchar(12)	Not Null	patient unique identification
		date_of_attendance	timestamp	Not Null	CURRENT_TIMESTAMP
		point_of_care_id	varchar(20)	Not Null	Point of care unique identification
		diagNot Nullsis	varchar(200)	Not Null	
		concept_id	int(10)	Not Null	Concept unique identification
		description_id	int(10)	Not Null	Description unique identification
		immediate_treat-ment_rendered	varchar(200)	Not Null	Immediate treatment rendered to the patient
		complications	varchar(200)	Not Null	Complications faced
		challenges_reme-dial_action	varchar(200)	Not Null	Challenges
		functional_status	varchar(50)	Not Null	Function status
		general_mgt_plan	varchar(200)	Not Null	General management plan
		doctor_id	varchar(20)	Not Null	Doctor unique identification
		physiotherapy_plan	varchar(100)	Not Null	physiotherapy_plan
	Entity type	**Field or attribute**	**Data type**	**Constraints**	**Description**
	Attendance queue	index	int(11)	Not Null	
		patient_id	varchar(12)	Not Null	Patient unique identification
		fname	varchar(30)	Not Null	Patient first name
		lname	varchar(30)	Not Null	Patient last name
		gender	char(1)	Not Null	Patients gender
		time	timestamp	Not Null	CURRENT_TIMESTAMP

3.11.2 System Implementation

The system was developed using Java, which was installed on a laptop running Windows 10 Operating System. MySQL Database Management System (DBMS) application was installed through XAMPP, which is a suite of programs comprising of Apache, MySQL, Perl and PHP. XAMPP was used because it installs a MySQL graphical user interface called PHPMyadmin, which make it possible to create a database, tables and views in MySQL without using commands. Therefore, using this GUI tool accessed through the browser using localhost as a server, the database was implemented by following the designs, which have been given in Chapter 3.

After implementing the database, the researcher installed eclipse Oxygen that was used as a system development editor. In order to facilitate quick and systematic system component generation, Scene Builder was installed on the same laptop.

By following the system business process, which were mapped in Chapter 3, the researcher implemented the system by creating a package using Java in Eclipse which contained classes that there responsible for connecting to the database. After the database connectivity were completed and tested, next the landing pages for the profile of administrator, receptionist, doctor, surgeon and nurse were developed.

The menus for each of the profile were done using scene builder. The lookup field forms followed this. Respective forms, which registers Hospital details, point of care, districts, specialization, countries, wards, theaters rooms and insurance, were developed.

Next form to be done were staff registration form, followed by system user registration form. The next forms to be created were that capture patient details, ward details, attendance, vital signs, queue forms, surgery details and operations forms. Next the research designed the login for capturing patient's profile picture using the webcam, followed by the logic and code for searching patient using either patient id, or scanning using the barcode which is labelled on the patient Identification card. This was followed by the logic and code for application of Snomed CT. This was done by providing a link with the diagnosis form, which the doctor, nurse, or surgeon can access to browse the Snomed CT concept ID. Within the diagnosis form, a text box was created for entering the Snomed CT concept ID and its description.

Hardware Components for the web based registration system

The table 7 below describes the hardware components for the Snomed CT standardized Electronic Health Record System.

Table 7: Snomed CT standardized Electronic Health Record System hardware components

Component Name	Description
Firewall	Protects server from unauthorized access by filtering incoming and outgoing communication.

Webserver	The server supporting the online Snomed CT international reference browser, which is referenced during patient diagnosis.
Database Server	The server that supports the database software.

3.11.3 Limitations of the Prototype Developed

The Snomed CT standardized Electronic Health Record System was developed but does not have a portal to enable patients have access to their records. The component for enabling patients have access to their record is constituency of an Electronic Health Record System. However, due to time constraint, the research could not accomplish this.

3.12 Chapter Summary

In this chapter, the materials and methods that were used in the baseline study were discussed and the system architecture of the EHR system. The methodology used for the baseline study, which was a mixed method, was discussed. The various methods previous were reviewed in order to have a clear understanding of the process to enable a mapping of the business. The proposed business process model for the UTH Surgery Department were presented after analyzing the previous and current system used.

CHAPTER 4: RESULTS

4.1 Introduction

This chapter discusses the major findings of the research for the development of a Snomed CT Electronic Health Record for the Surgery Department of the University Teaching Hospital. The study focused on the system currently in use for registering patients, admitting patients, scheduling mechanism for surgical operation, the patient flow chart and the storage of information. The research also focused on understanding the challenges experienced by Clinicians in their daily discharge of service in the light of the current system in use. The chapter discusses the findings according to two themes namely Baseline Study Results and Developed Prototype.

4.2 Baseline Study Results.

In this section, the survey sampling, data collection methods are discussed.

4.2.1 Survey Sampling and Data collection

The Baseline study was carried out as a case study of the University Teaching Hospital Department of Surgery. The study's main objective was to understand the system being used, the challenges being experienced and how they can have solved. In carrying out this, interviews were conducted with key stakeholders mainly clinicians. Then records inspection was conducted followed by quantitative analysis in which a questionnaire was administered to 48 clinicians out of which only 40 where responded to. The response rate was 83%. Convenient sampling was used due to time and non-accessibility of the target group of people. The clinicians are very busy people.

4.3 Survey Results and Discussion

The survey results are grouped according to finding from the quantitative results, record inspection and interview discussions with stakeholders.

Demographic Findings

Demographic analysis results have been presented using frequencies and pie charts.

Table 8: Gender of Respondent

VARIABLE NAME		**FREQUENCY**	**PERCENT**
GENDER			
	male	20	50.0
	female	20	50.0
	Total	40	100.0

Table 8 shows frequency statistics of the variable named gender of respondents. Out of the 40 (100%) respondents 20(50%) were female and 20(50%) were male.

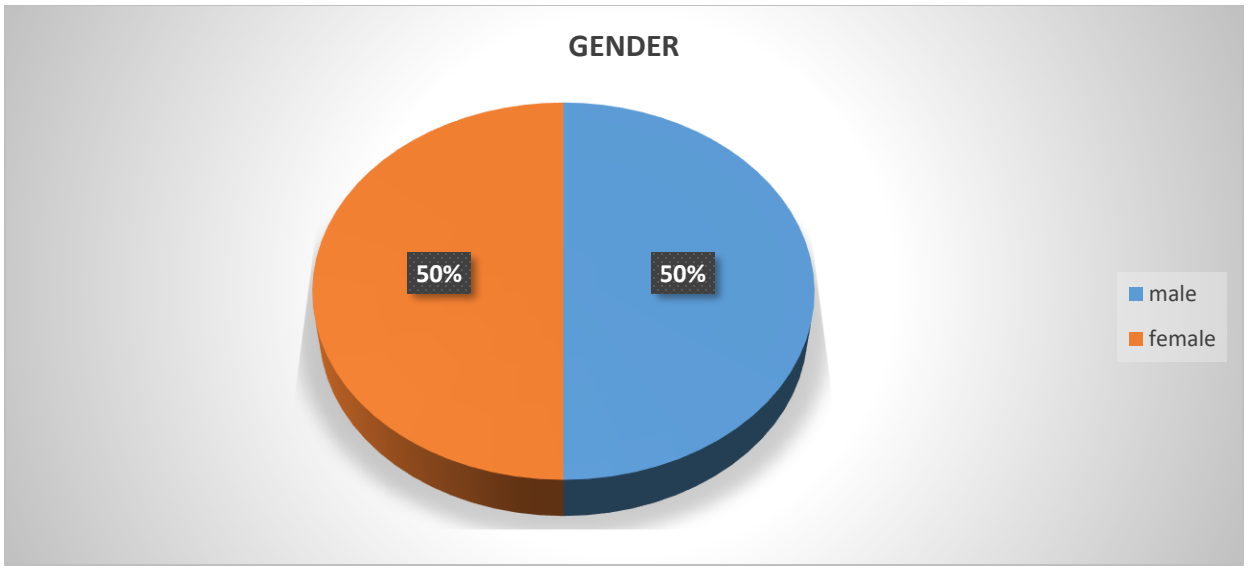

Figure 4.1: Gender of Respondent

Figure 4.1 is pie chart giving a graphic depiction of the results shown in table 8. The results are result rounded off to the nearest whole number percentage.

Table 9: Age Group

VARIABLE NAME	FREQUENCY	PERCENT
AGE GROUP		
29 years and Below	4	10.0
30-39 years	9	22.5
40-49 years	16	40.0
50 years or over	11	27.5
Total	40	100.0

Table 9 shows frequencies results for another demographic results called Age group. Out of 40(100%) respondents who participated in the survey, 4(10%) reported that they were 29 years and below, 9(22.5%) were aged 30 to 39 years, 16(40%) were aged 40 to 49 years old and 11(27.5%) were aged 50 years and above.

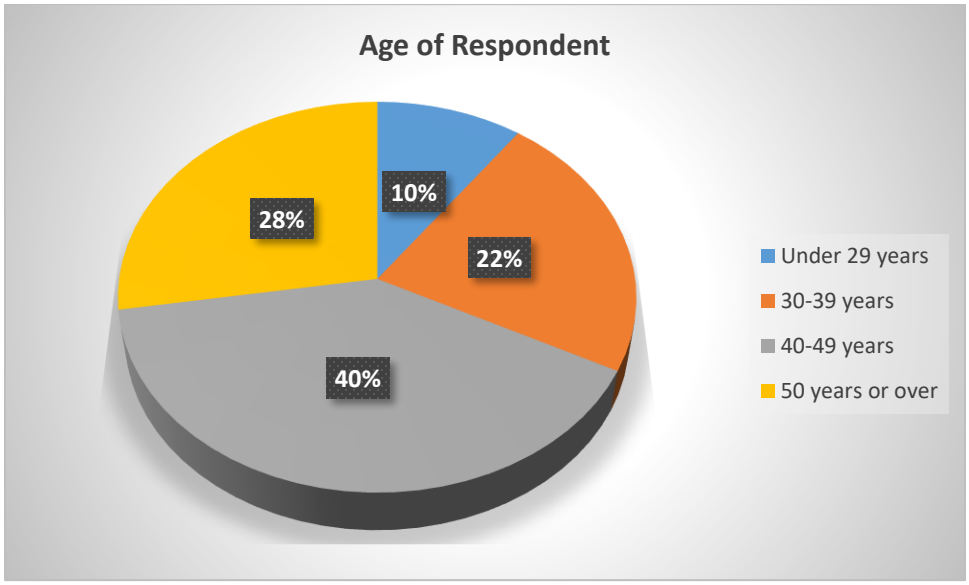

Figure 4.2: Age of respondent

Figure 4.2 is a pie chart showing the distribution of age of respondents in a percentage format. The results are result rounded off to the nearest whole number percentage.

Table 10: Highest level of education

VARIABLE NAME		FREQUENCY	PERCENT
HIGHEST LEVEL OF EDUCATION			
	Certificate	5	12.5
	Diploma	13	32.5
	Graduate	14	35.0
	Post Graduate	8	20.0
	Total	40	100.0

Table 10 shows frequency distribution of the respondent's highest level of education variable. Of the 40(100%) respondents, 5(12.5%) reported that there highest qualification was a certificate, 13(32.5%) reported that there highest education level was a Diploma, 14(35%) reported that there highest education level was a Graduate Degree and 8(20%) reported that there highest education level was Post Graduate Degrees.

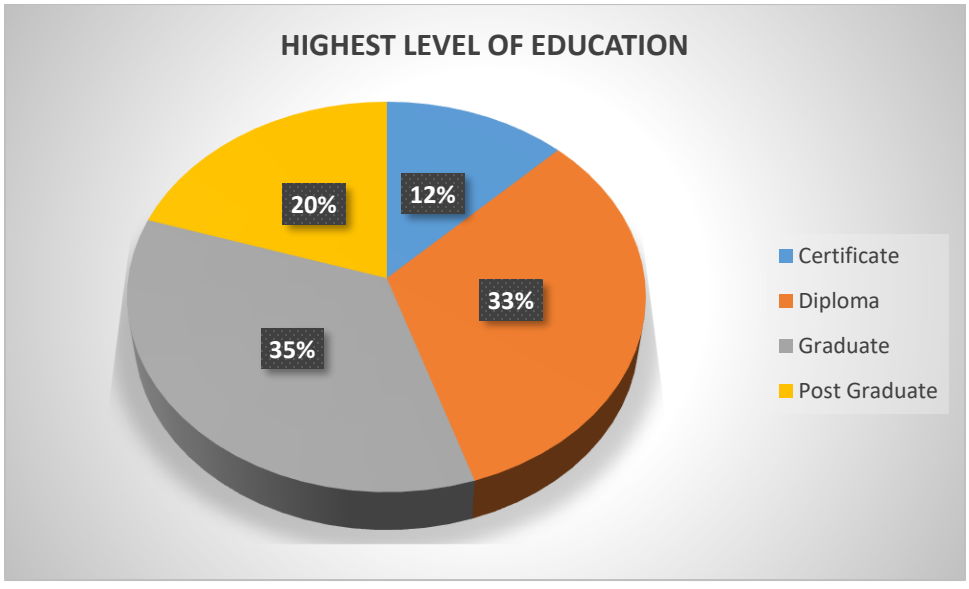

Figure 4.3: Highest level of education

Figure 4.3, shows the results of the questionnaire variable 'Highest level of education'. The pie chart shows the same results tabulate in table 10. The results are result rounded off to the nearest whole number percentage.

Table 11: Job Title

VARIABLE NAME		FREQUENCY	PERCENT
JOB TITLE			
	Nurse	5	12.5
	Doctor	9	22.5
	Surgeon	20	50.0
	anesthetist	5	12.5
	Record Clerk	1	2.5
	Total	40	100.0

Table 11 shows statistical frequency results for job title, which was one of the variables for the survey. Out of 40(100%) respondents, 5(12.5%) of respondents reported that they worked as Nurses, 9(22.5%) of respondents reported that they worked as Doctors, 20(50%) reported that they worked as Surgeons, 5(12.5%) worked as anesthetist and 1 (2,5%) worked as Records clerk.

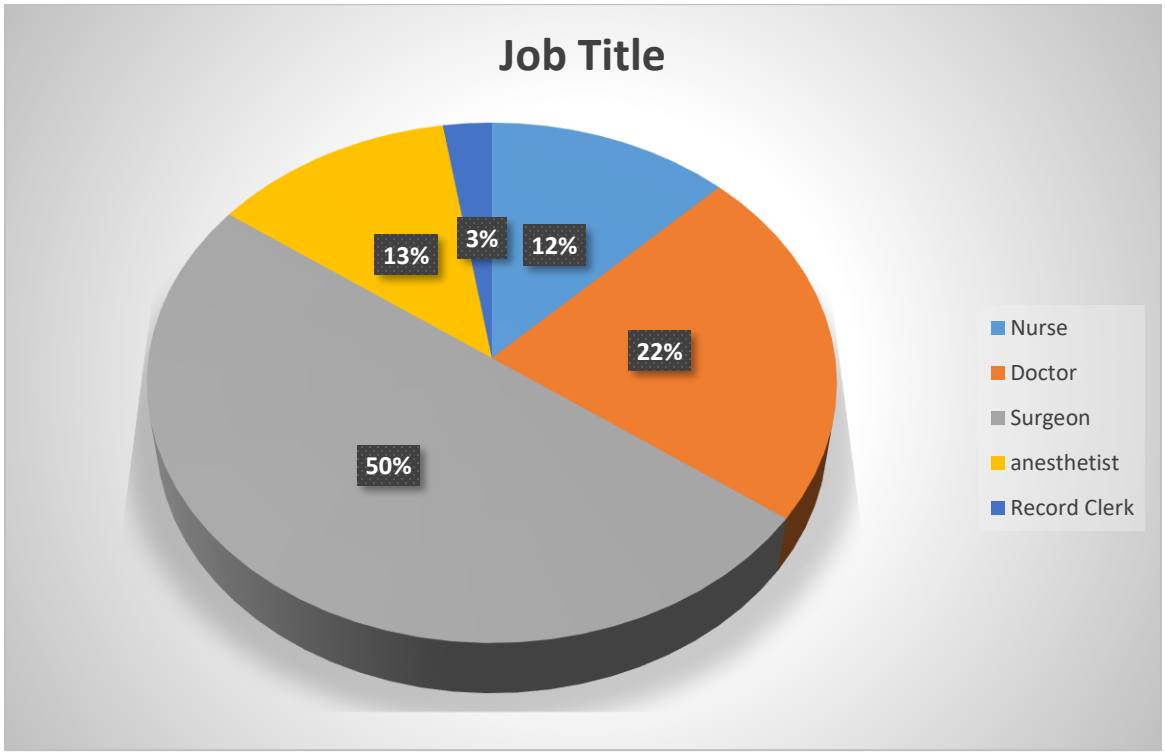

Figure 4.4: Job Title

Figure 4.4, shows frequency results of the variable 'Job title'. The same results are shown in tabular form in table 11. The pie chart has been used to aid quick understanding. The results are result rounded off to the nearest whole number percentage.

Table 12: length of service

VARIABLE NAME		FREQUENCY	PERCENT
FOR HOW LONG HAVE YOU BEEN WORKING IN THIS FACILITY			
	Less than 1 year	3	7.5
	1 - 5 years	13	32.5
	6 – 10 years	13	32.5
	More than 10 years	11	27.5
	Total	40	100.0

Table 12 shows results of frequency statistics for the variable 'For how long have you been working in this facility?' Out of 40 (100%) respondents, 3(7.5%) reported that they had worked for less than 1 year, 13(32.5%) reported that they had worked for 1 to 5 years, 13(32.5%) had worked for 6 to 10 years and 11(27.5%) had worked more than 10 years.

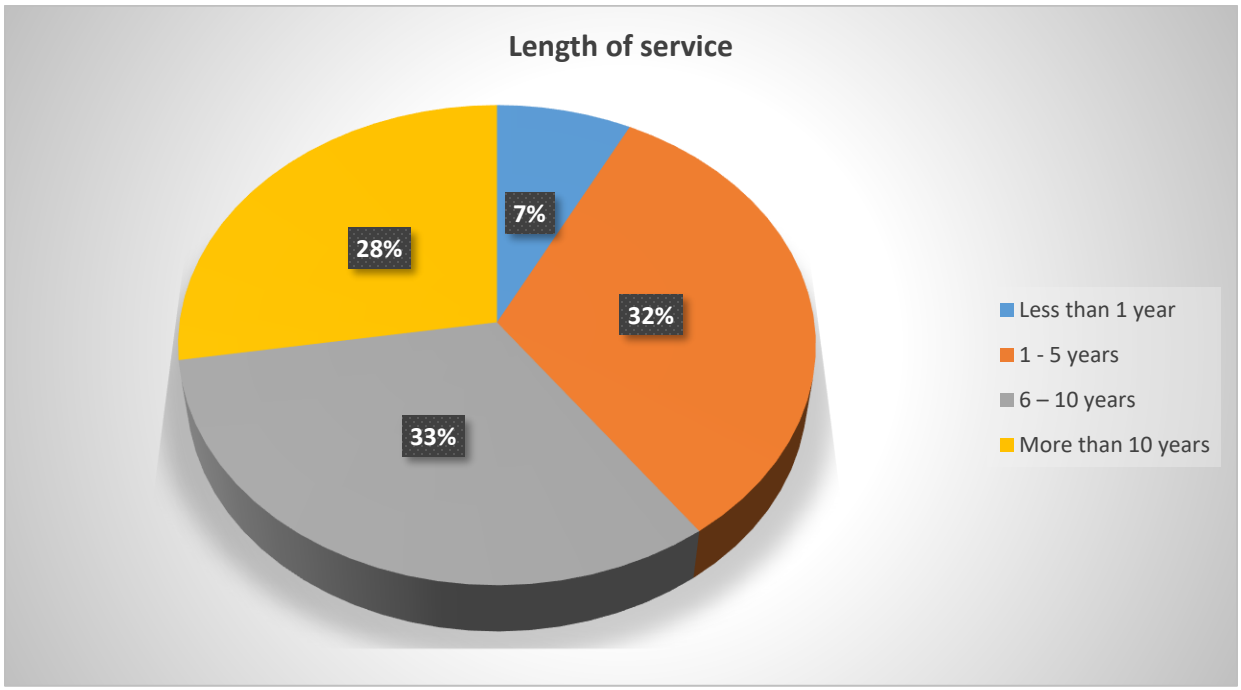

Figure 4.5: length of service

Figure 4.5, shows the same results presented in table 10 but in a pie chat for easy understanding. The results are result rounded off to the nearest whole number percentage.

Table 13: Existence of computer in the department or section

IS THERE A COMPUTER(S) IN YOUR DEPARTMENT?		FREQUENCY	PERCENT
	yes	29	72.5
	no	11	27.5
	Total	40	100.0

The question of infrastructure is very important was important especially the existence of a personal computer. Table 13 shows that 29(72.5%) out of 40 reported that they had a computer in their department or office, and 11(27.5%) out of 40 reported that they did not have a computer.

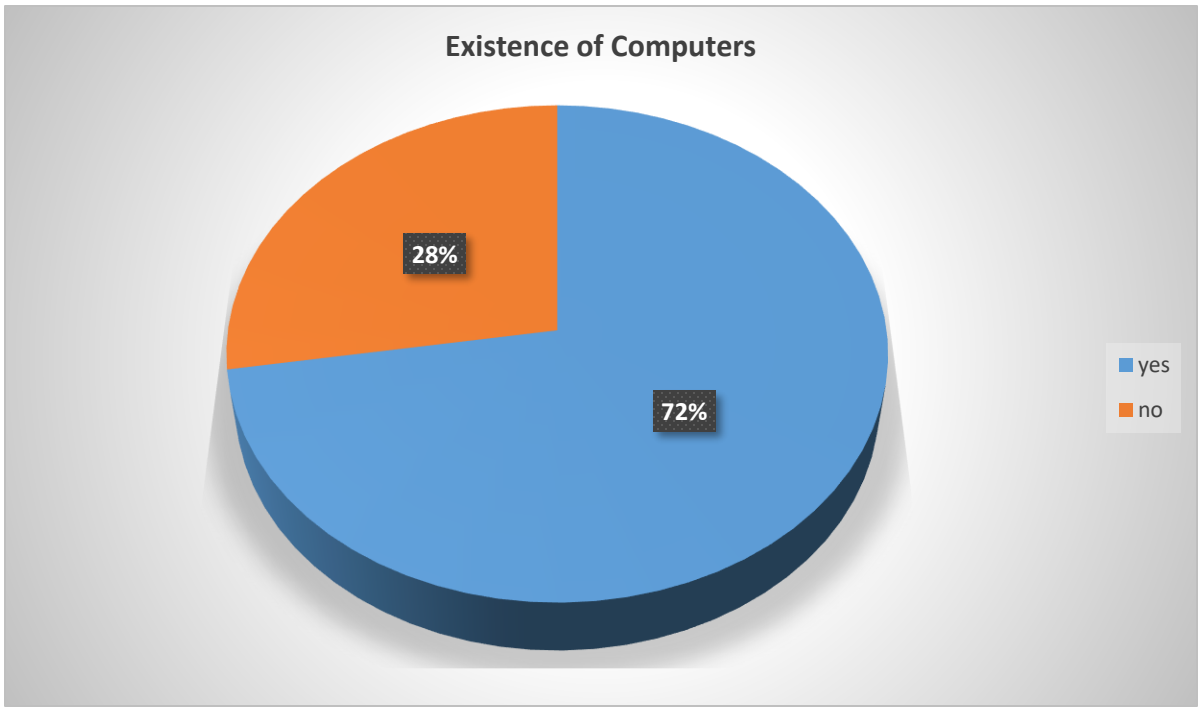

Figure 4.6: Existence of Computers

Figure 4.6 display results which have been presented in table number 13. The results are rounded off to the nearest whole number, therefore, 28% reported that they had computers in their department and 72% reported that they did not have.

Table 14: level of electronic records implementation

CHOOSE WHAT BEST DESCRIBES THE LEVEL OF ELECTRONIC MEDICAL RECORD SYSTEM IN YOUR DEPARTMENT?		FREQUENCY	PERCENT
	Management of health records in this department is fully electronic.	1	2.5
	Management of health records in this department is hybrid (partially electronic and partially paper-based.)	24	60.0
	We do not have electronic medical records in this department.	15	37.5
	Total	40	100.0

Table 14 shows that 1(2.5%) reported that the department had fully implemented electronic record system, 24(60%) reported that the department was using both paper based and electronic health record system, and 15(37.5%) reported that they did not have an electronic record system.

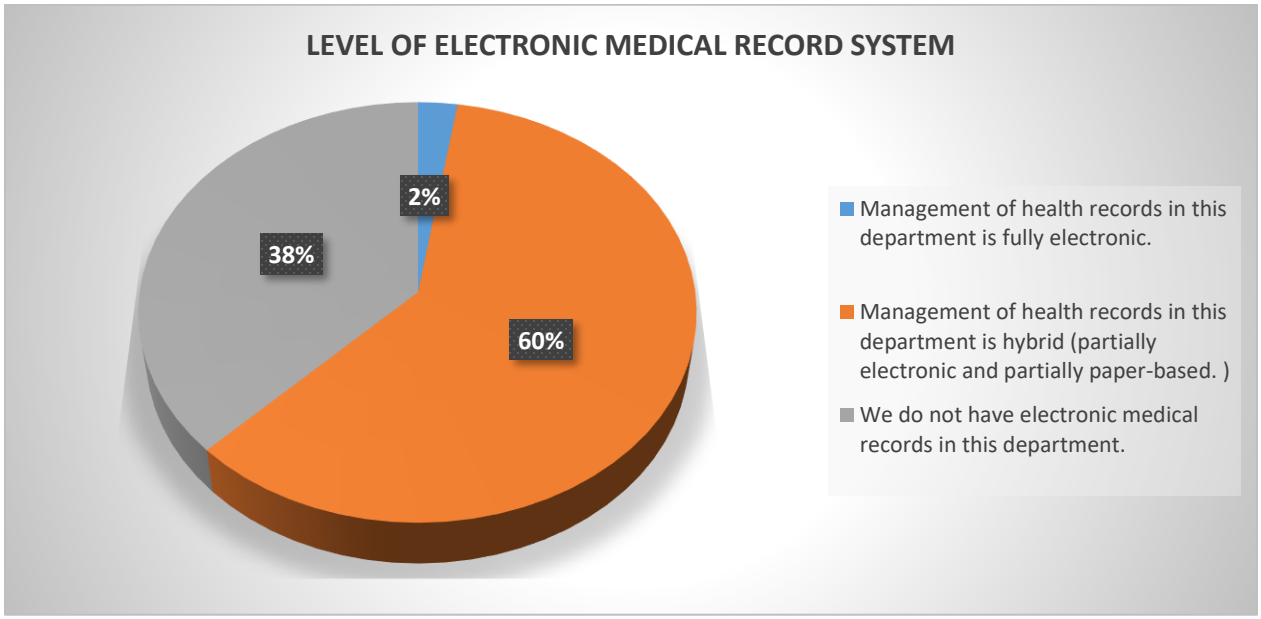

Figure 4.7: level of electronic medical record implementation

Figure 4.7 shows the results of level of electronic health system implementation in graphic format. The pie shows results rounded off to the nearest whole in percentage.

Table 15: scheduling of patients for operation

The Schedule of patients to be operated on is known in advance, except for emergency cases	*Frequency*	*Percent*
disagree	3	7.5
uncertain	16	40.0
agree	17	42.5
strongly agree	4	10.0
Total	40	100.0

From table 15, the response using a Likert scale (1 strongly disagree, 2 disagree, 3 uncertain, 4 agree, 5 strongly agree) to the statement: the scheduling of patients to be operated on is known in advance, except for emergency cases were, 3 (7.5%) out of 40 disagree, 16(40%) were uncertain, 17(42.5%) agree, 4(10%) strongly agree. Combined total of those whose were uncertain and those who disagreed, we have almost 40.75% of respondents who disagree or are not sure about every important statement.

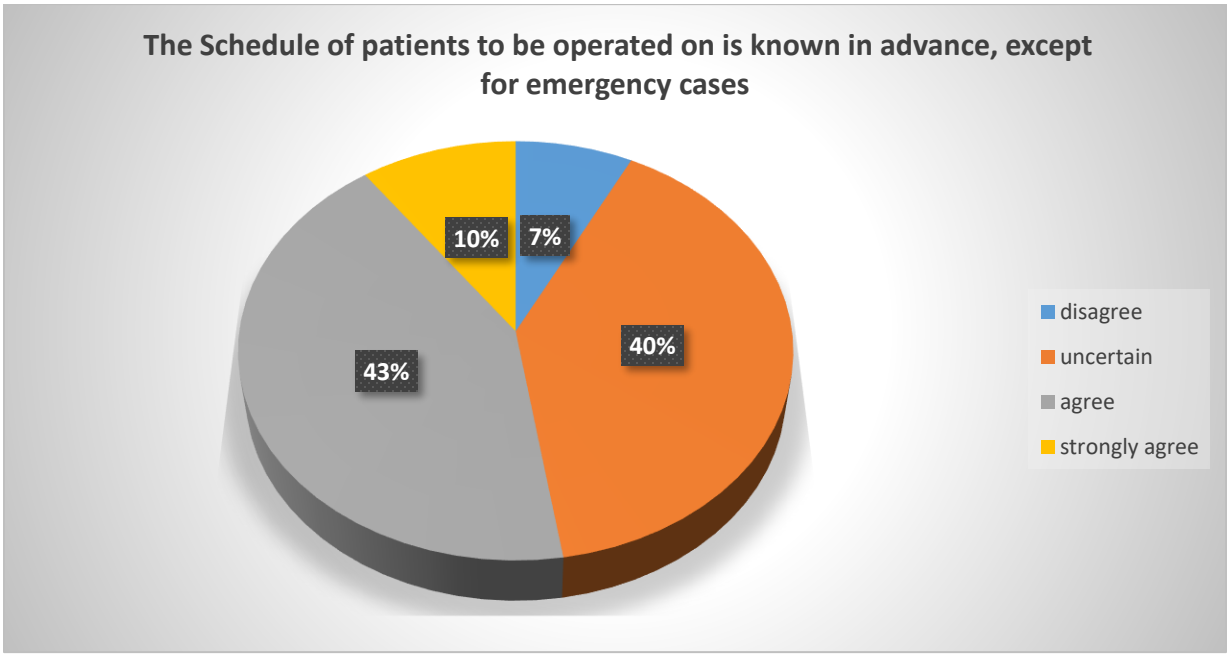

Figure 4.8: Schedule of patients to be operated on

Figure 4.8 shows the results in a pie chart for the variable 'the schedule of patients to be operated on is known in advance, except emergency cases'. The results are rounded to the nearest whole number in percentage format.

Table 16: Information regarding a particular patient before surgery is readily available during operation procedures

Information regarding a particular patient before surgery is readily available during operation procedures			
	strongly disagree	2	5.0
	disagree	6	15.0
	uncertain	15	37.5
	agree	14	35.0
	strongly agree	3	7.5
	Total	40	100.0

Using the Likert scale as explained above, the respondents to the statement; Information regarding a particular patient before surgery is readily available during operation procedures as follows: 2(5%) out of 40 strongly disagreed, 6(15%) out of 40 disagreed, 15(37.5%) were uncertain, 14(35%) out of 40 agreed and 3(7.5%) strongly agreed. Table 16 is a presentation of the results.

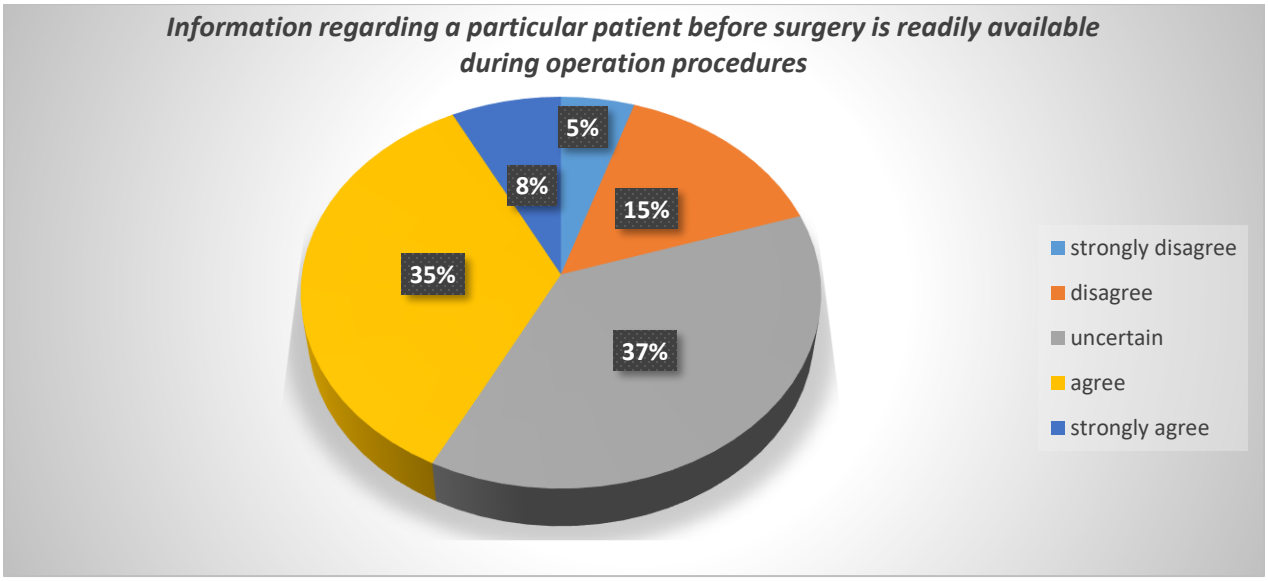

Figure 4.9: Information regarding a particular patient before surgery is readily available during operation procedures

Information in figure 4.9 is the results from table 16 in which the frequency for the variable 'Information regarding a particular patient before surgery is readily available during operation procedures' are presented. The results are rounded off to the nearest whole number in percentage format.

Table 17: Operating room and patient scheduling

All supplies necessary for operation procedure are known in advance via existing system and are made available before commencement of operation procedure			
	strongly disagree	1	2.5
	disagree	7	17.5
	uncertain	14	35.0
	agree	15	37.5
	strongly agree	3	7.5

	Total	40	100.0

Table 17 show the frequency results of using Likert Scale to respond to the statement 'All supplies necessary for operation procedure are known in advance via existing system and are made available before commencement of operation procedure' as follows: 1(2.5%) out of 40 strongly disagreed, 7(17.5%) out of 40 disagreed, 14(35%) were uncertain, 15(37.5%) out of 40 agreed and 3(7.5%) strongly agreed.

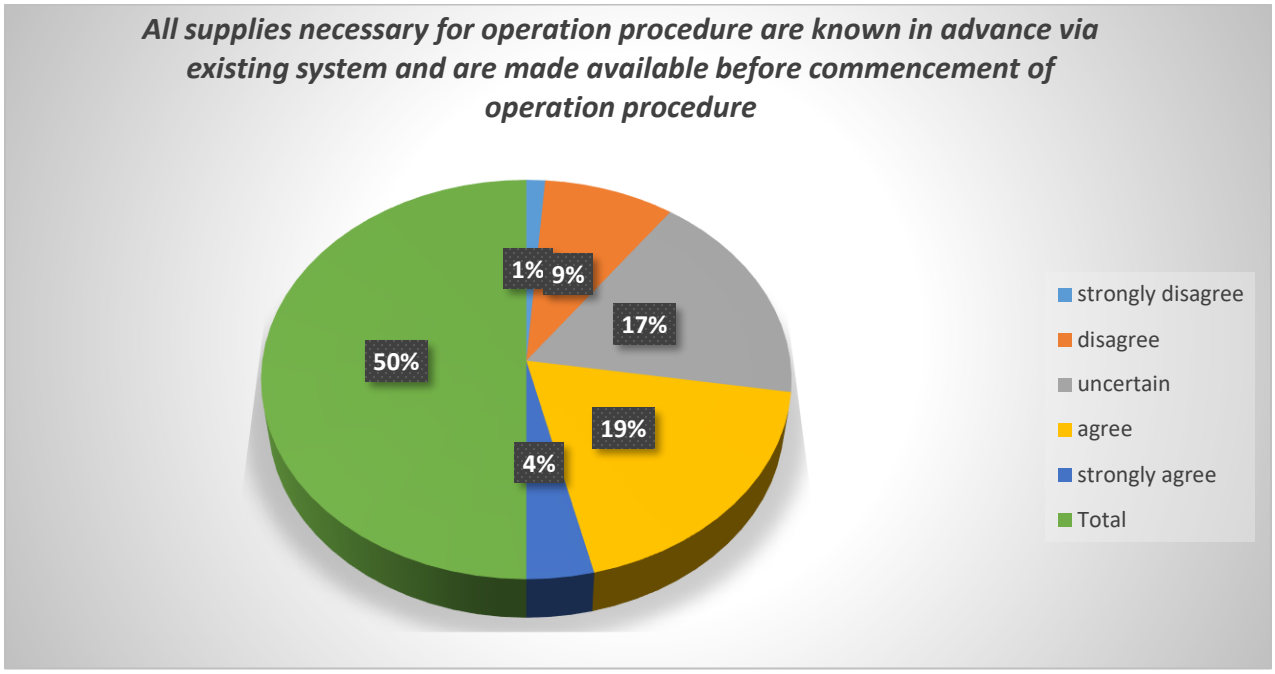

Figure 4.10: Availability of information before Surgery

Figure 4.10 is a pie chart showing results presented in table 17. The results are percentages rounded to the nearest whole number.

Table 18: Rating in terms of privacy and security of patient's medical data

Rating in terms of privacy and security of patient's medical data	*Frequency*	*Percent*

disagree	7	17.5
uncertain	22	55.0
agree	11	27.5
Total	40	100.0

Table 18, shows ratings by respondents on the privacy and security of patient's medical data. The results indicate that 7(17.5%) disagreed, 22(55%) were uncertain, 11(27.5) agreed that the privacy and security exists.

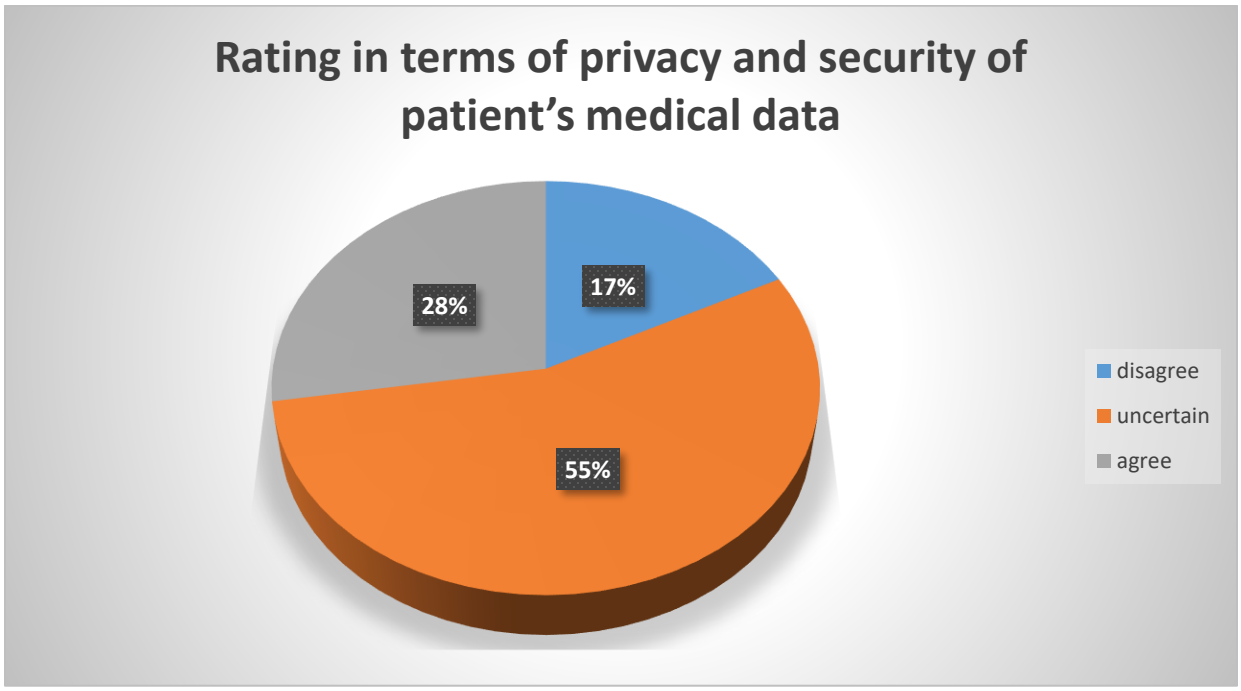

Figure 4.11: Privacy and Security rating

Figure 4.11 is a pie chart of results presented in table 18. The results are formatted in percentage to the nearest whole number.

Table 19: Rating of report generation with the current system

	Response Type	Frequency	Percentage
Rating of report generation with the current system			
	disagree	3	7.5
	uncertain	26	65.0
	agree	10	25.0
	strongly agree	1	2.5
	Total	40	100.0

Table 19, shows ratings by respondents regarding easiness of reports generation under the current system. 3(7.5%) disagreed, 26(65%) were uncertain, 10(25%) agreed, and 1(2.5%) strongly agreed.

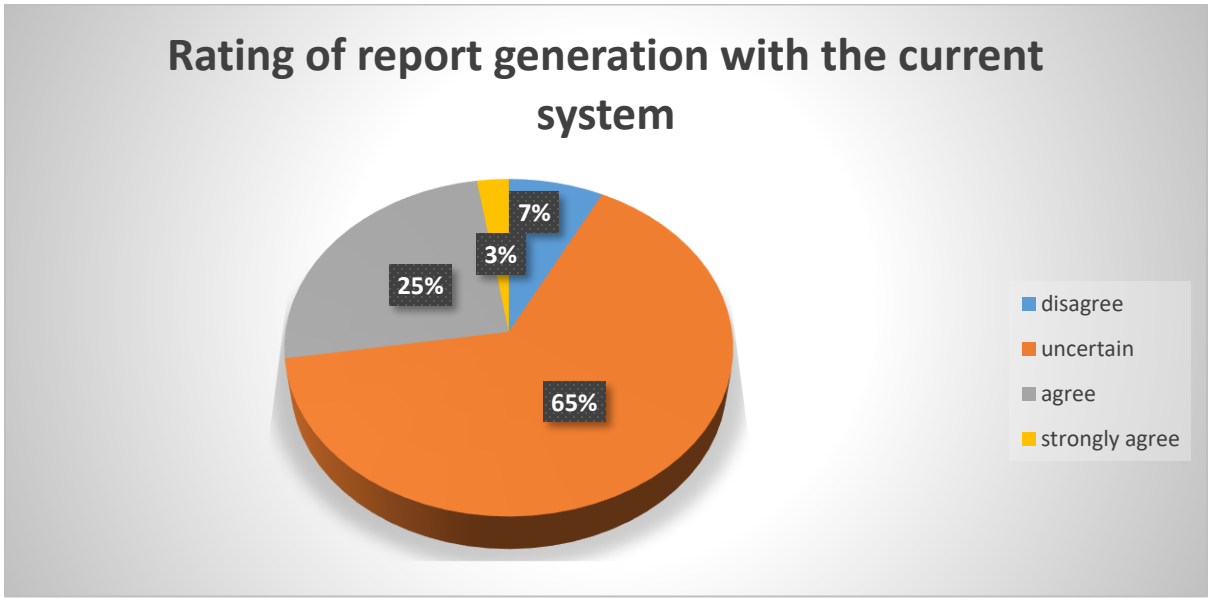

Figure 4.12: Rating of report generation with the current system

Figure 4.12 is a pie chart of results presented in table 19. The results are formatted in percentage to the nearest whole number.

Table 20: Patient scheduling, report generation, level of satisfaction

	Response Type	Frequency	Percentage
Rating of Level of Satisfaction with the current system			
	very dissatisfied	1	2.5
	dissatisfied	20	50.0
	uncertain	17	42.5
	satisfied	2	5.0
	Total	40	100.0

Table 20, shows the respondents rating of their level of satisfaction with the current system being use. 1(2.5%) were very satisfied, 20(50%) were dissatisfied, 17(42.5%) were uncertain with their level of satisfaction, 2(5%) were satisfied.

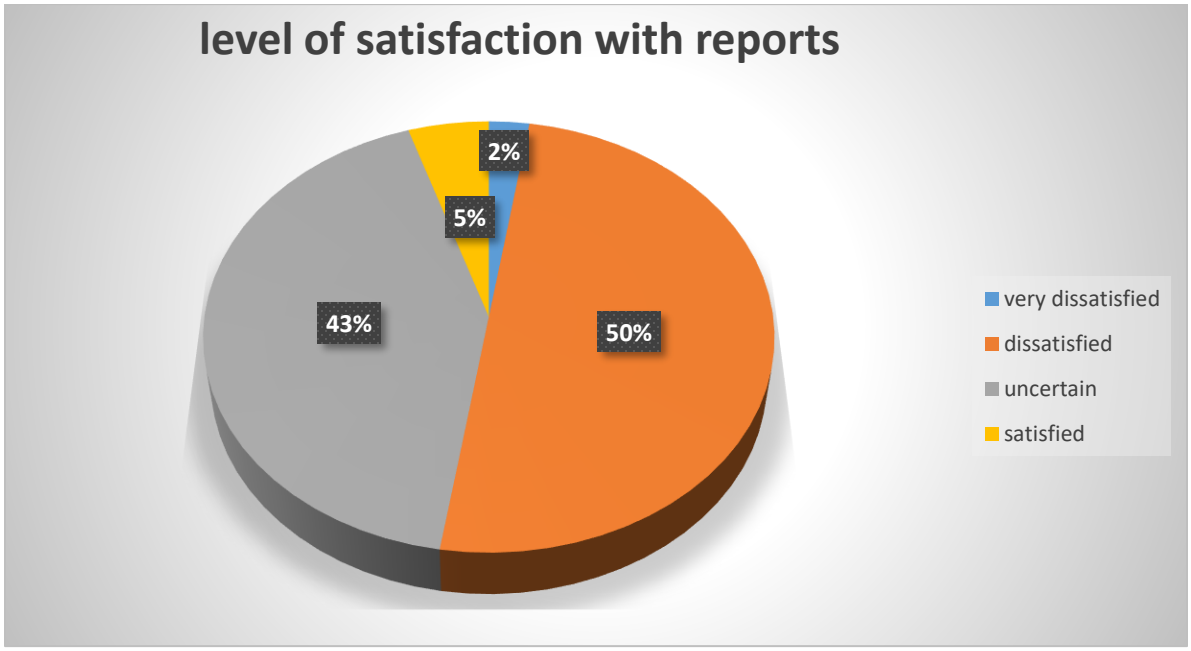

Figure 4.13: Level of satisfaction with the current system

Figure 4.13 is a pie chart of results presented in table 20. The results are formatted in percentage to the nearest whole number.

4.4 Findings from Record Inspection

Despite, what the quantitative results from the questionnaire have revealed, the picture on the ground was better painted by the actual physical inspection of the systems being used by the UTH Surgery Department. Permission was obtained to capture the books which were used to register patients when they arrive at the hospital, admission registered, ward round progress report, operation room scheduling book, shift book report.

Figure 4.14: Pictures of Manual Records in Use

Figure 4.14 is a picture showing some of the books that were captured by the researcher using a camera. Not the books have been shown due to the sensitivity of the data, which they contain.

MS EXCEL DATABASE

GENERAL AND DEMOGRAPHIC INFORMATION

DATE OF ATTENDANCE /ADMISSION	POINT OF ATTENDANCE/ADMISSION (ER/LCC/HCC/MSW/FS	DATE OF DISCHARGE/ DEATH	NUMBER OF CURRENT ADMISSION TO NTH	DURATION OF HOSPITAL STAY (INPATIENT DAYS OF CARE)	PATIENT STAY STATUS (INPATIENT/LODGER)	REFERRAL TYPE (SELF/INSTITUTIONAL)	NAME OF REFERING INSTITUTION	REFERING INSTITUTION (GRZ/PVT)	REFERING INSTITUTION STATURE (HP/HC/DH/CH/GH/TH)	HEALTH INSURANCE COVER TYPE (INSURANCE/HOSPITAL SCHEME/OUT OF POCKET)	HEALTH INSURANCE COVER PROVIDER	EMPLOYMENT STATUS	AVERAGE DAILY INCOME (ESTIMATE)	AVERAGE MONTHLY INCOME (ESTIMATE)

Next of Kin (Name)	Relationship of patient with next of kin	Next of Kin Contact Number	PATIENT'S FULL NAME	DATE OF BIRTH	PATIENT'S IDENTITY NUMBER (NRC/PASSPORT)	PATIENT'S HOSPITAL RECORD NUMBER	PATIENT'S CONTACT NUMBER	PATIENT'S RESIDENTIAL ADDRESS (DOMICILE)	PATIENT'S PLACE OF BIRTH	PATIENT'S NATIONALITY	PATIENT'S OCCUPATION	PATIENT'S RELIGION

PATIENT'S MARITAL STATUS	SEX	AGE	WARD ADMITTED IN	POINT OF CARE	DIAGNOSIS	IMMEDIATE TREATMENT RENDERED	COMPLICATIONS	CHALLENGES AND REMEDIAL ACTION INSTITUTED	FUNCTIONAL STATUS	GENERAL MANAGMENT PLAN

Figure 4.15: Sample Ms Excel Information System

In order to take advantage of the computers which currently exist in the department, the person who are sophisticated computer users have developed a database to help register patient details, schedule patients for operations, plan operation supplies and record outcome. Figure 4.15 shows the sample screen shoots capture for the Ms Excel Database or Record System.

4.5 Development of a Prototype

The prototype was developed using JavaFX, MySQL, ClinicClue Xplore tools and Barcode technology to generate barcode for tagging and reading patient identification numbers.

The prototype is client based with a centralised database, which can be hosted in the cloud. All installed applications will have the database connection pointed to the location of the database. This enables concurrent usage by different clinicians and administrators. The server side also includes an online access to the Snomed CT browser, accessible within the prototype. As an alternative to the online accessible Snomed CT browser, CliniclueXplore application is installed alongside the prototype and become accessible within the system during the Doctor's diagnosis period.

MySQL Database was installed on the machine using an opensource suite of software called XAMPP. XAMPP when installed makes available on the computer, Apache, MySQL, Perl and PHP. From that suite, only MySQL was used in implementing the system. MySQL database was used as a database server for storage of EHR. Database implementation involved translating the relational schema into the physical database by mapping the schema onto the target database management system (DBMS). JAVA and MySQL were programmed to communicate with each other. This was made possible through the connect function which was coded in JAVA for instructing the prototype to connect to the backend (MySQL database).

Algorithm for connecting Java to MySQL is Part of JAVA connection code is shown in figure 4.17.

```
import java.sql.*;
class MysqlCon{
public static void main(String args[]){
try{
Class.forName("com.mysql.jdbc.Driver");
Connection con=DriverManager.getConnection(
"jdbc:mysql://localhost:3306/hospital_db","root","");
//here hospital_db is database name, root is username and pass-
word
Statement stmt=con.createStatement();
ResultSet rs=stmt.executeQuery("select * from emp");
while(rs.next())
System.out.println(rs.get-
Int(1)+"  "+rs.getString(2)+"  "+rs.getString(3));
con.close();
}catch(Exception e){ System.out.println(e);}
```

Figure 4.17: Sample Database Connection

Sample Prototype Screen Shots

1. Login

Navigation to this system starts from a user login. Once the system is initialized, it prompts the user to login. The user provides the username, password and input the auto generated security code as shown below in figure 4.18.

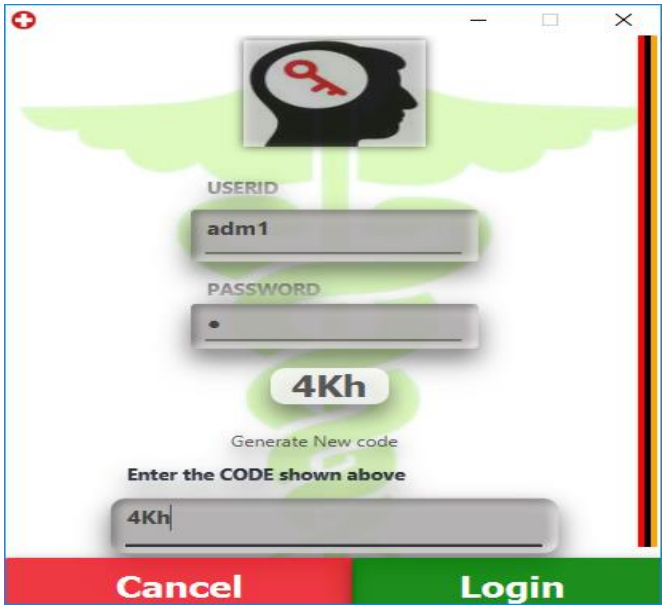

Figure 4.18: Login form

2. Administrator Directory

Once logged in, in this case as administrator, the following screen appears. Depending on the user's level of access (access privilege), the content of the proceeding window after logging in differ.

Admin Initial page

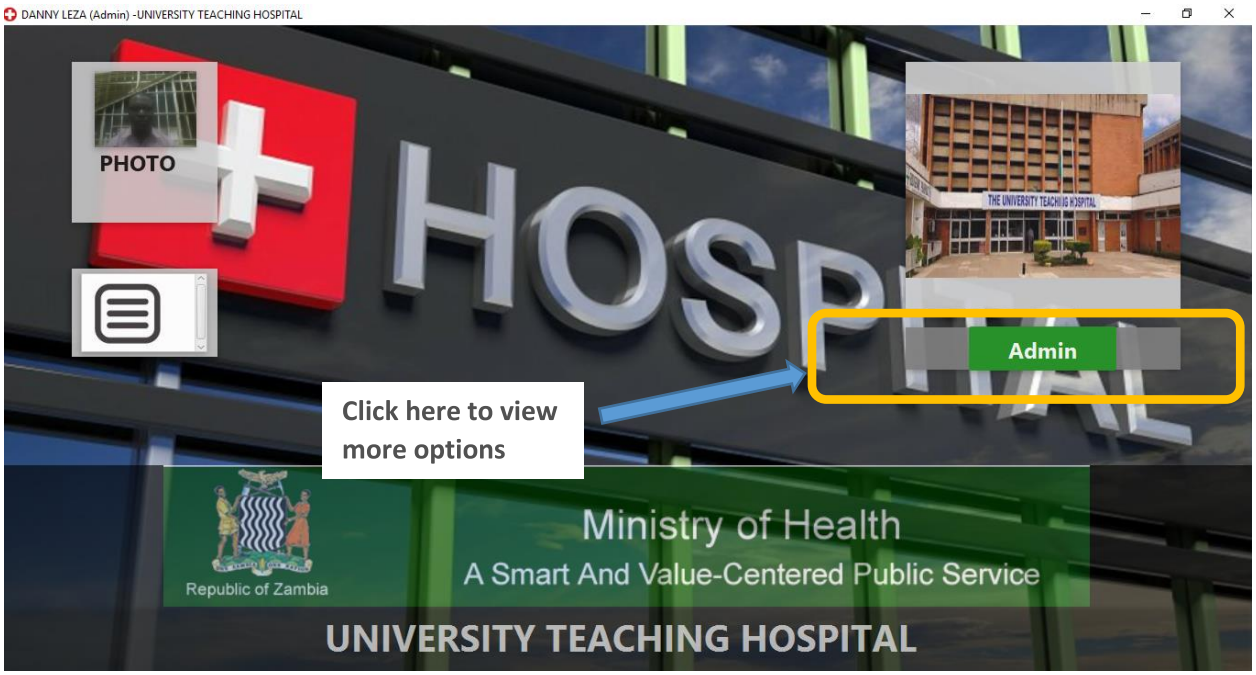

Figure 4.19: landing Page for System Administrator

The administrator would see the system main menu by pressing the Admin button. The admin button is shown in screenshot of figure 4.19. This will result in the display of the window shown in figure 4.20.

Main Administrator directory

Figure 4.20: Administrator Menus

By default, the main menu is the home where all the sub-menus are located as seen in the screen above. If the administrator has logged in for the very first time, the system requires the definition of system variables beginning with the hospital name, images which shown appear like the one which shows hospital. The menu items for configuring this are shown on the left side of figure 4.21.

Figure 4.21: Hospital Creation

Figure 4.21 is the form which enables the administrator to define the Hospital details including browsing the computer to find the location of the Snomed CT offline browser.

Once the Hospital name and images have been created, the administrator will then proceed to define lookup fields, like districts, wards, beds, operating rooms, personnel titles, specializations, insurance types among others. It is only after these fields are setup that is when system user details can be created. This is followed by the use of the system, which is first done by entering patient details.

3. Patient Administration

Patient administration is the main function that the system offers to the users. The first task here is the creation of patient record. This feature is accessible in the administrator and Receptionist user profiles. Figure 4.22 shows where this patient details creation is found in the administrator profile.

a. Create Patient

To create or register a patient, click on the **Create Patient Tab** and wait for the registration form to display as shown below in figure 4.22:

Figure 4.22: Create Patient Button

Patient Registration Form

The Patient Registration Form has four tabs that is the Patient's Biodata, Patient's Biodata (continuation), Other Details and Upload Photo.

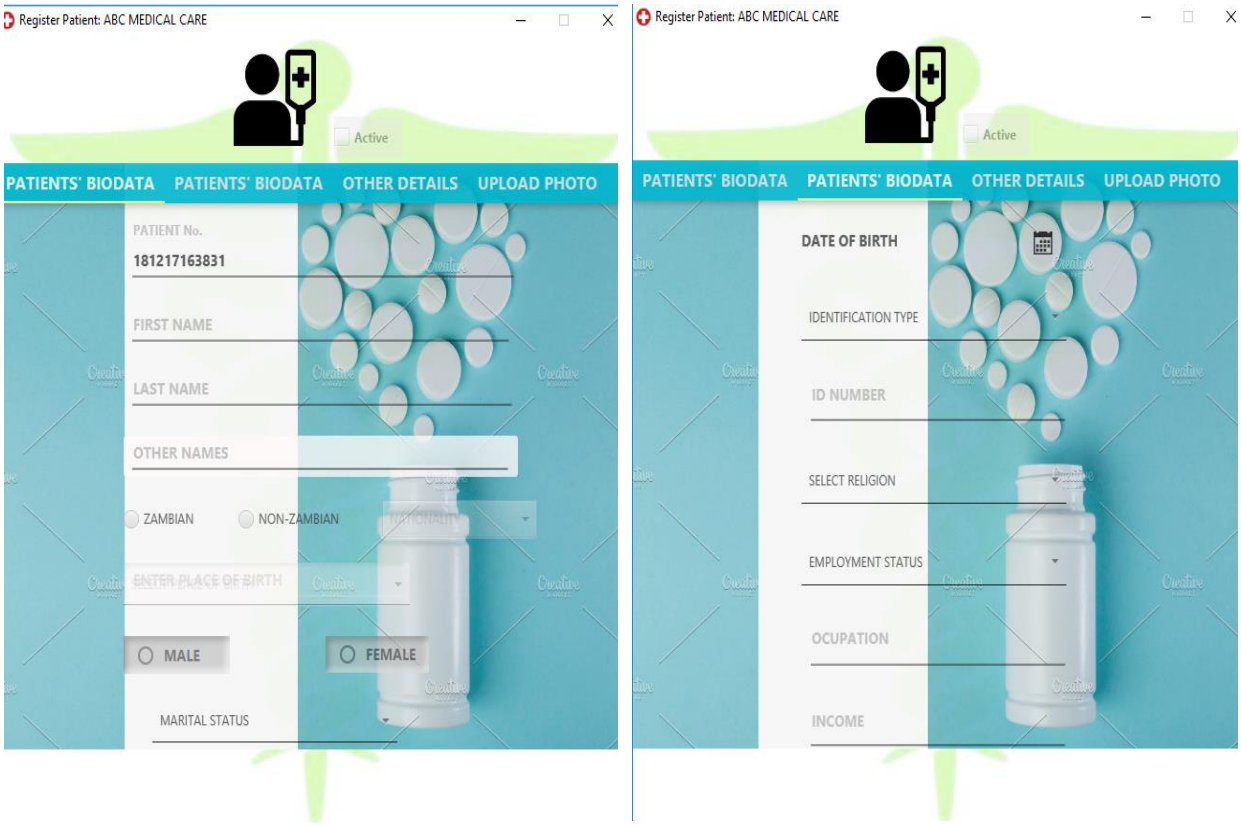

Figure 4.23: Patient Bio Data

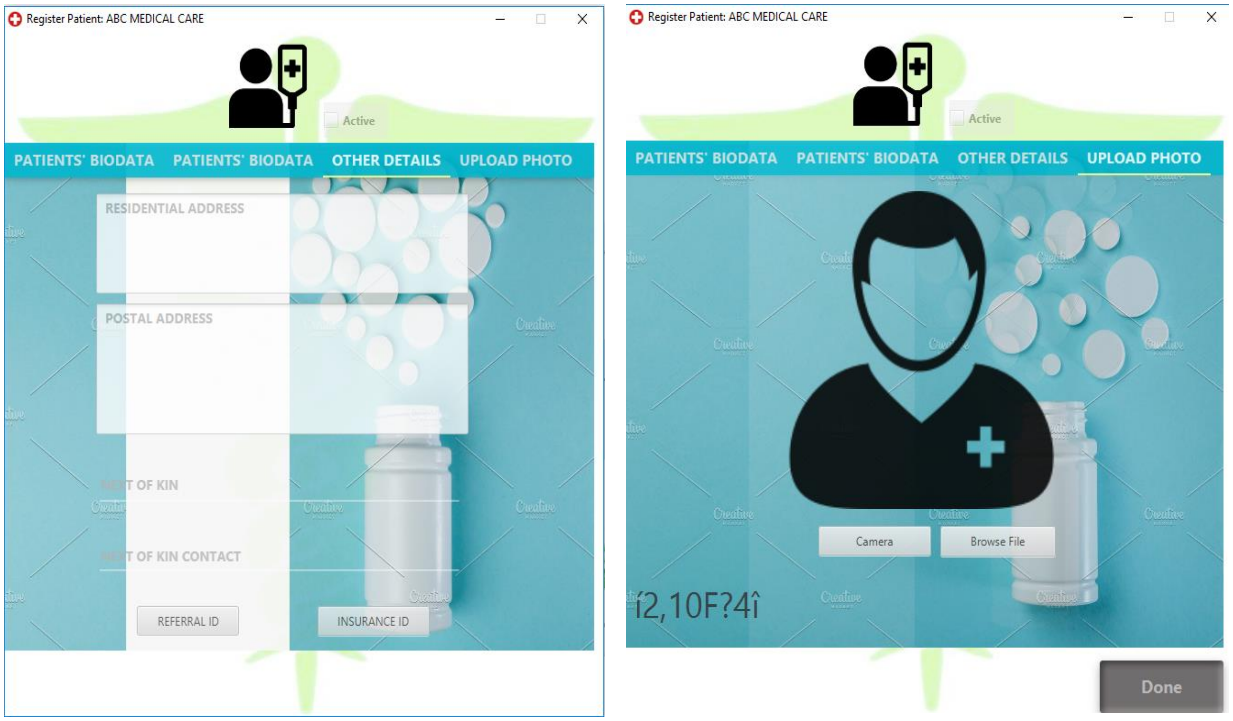

Figure 4.24: Patient's other details

i. Patient's Biodata

This is a form that accepts the Patient's biodata, that is first name, last name, other names, nationality, place of birth, marital status, date of birth, identification as shown in the snippet figure 4.24.

ii. Other Details

Accepts the following details: residential address, postal address, and next of kin as shown in figure 4.24.

iii. Upload Photo

The Upload Photo Tab allows the user to upload a photo, figure 4.25. The method of upload is twofold; the user either can use a camera or can upload an existing image from the file directory (internal or external sources). The following illustrate how an image can be uploaded:

To use the camera, click ok CAMERA button then click the Capture Button to capture image, otherwise click close camera to exit the camera mode.

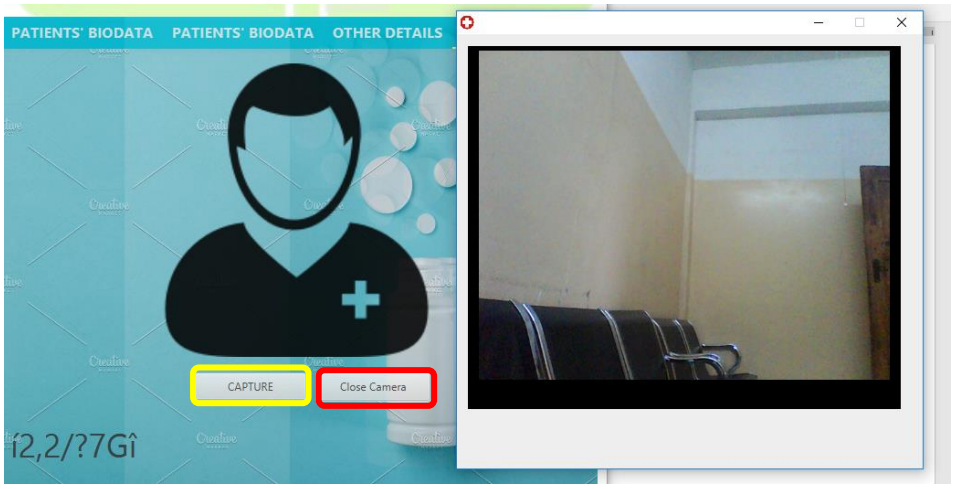

Figure 4.25: Capturing the patient image using webcam

Viewing Patient Details

Patient details can be viewed, queued for surgery, attendance, admission and surgery information. The following screenshots summarizes the different views of patients.

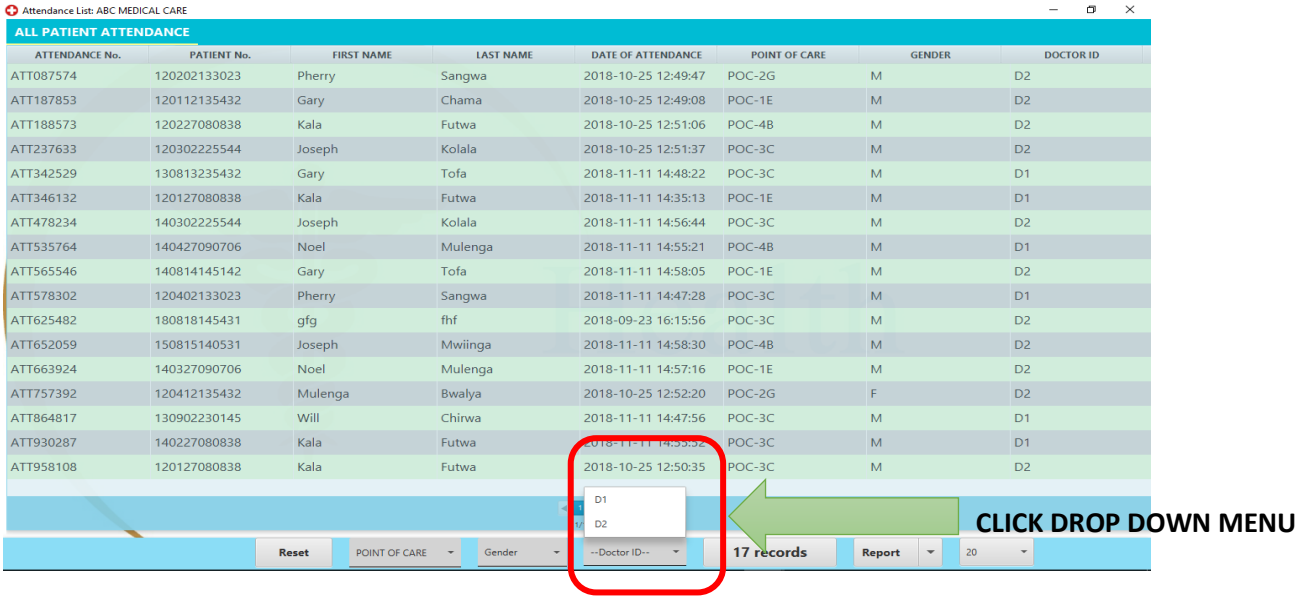

Figure 4.26: View patient details doctor identification number

Figure 4.26 shows display of patient details filtered according to the attending doctor.

Filter by Gender

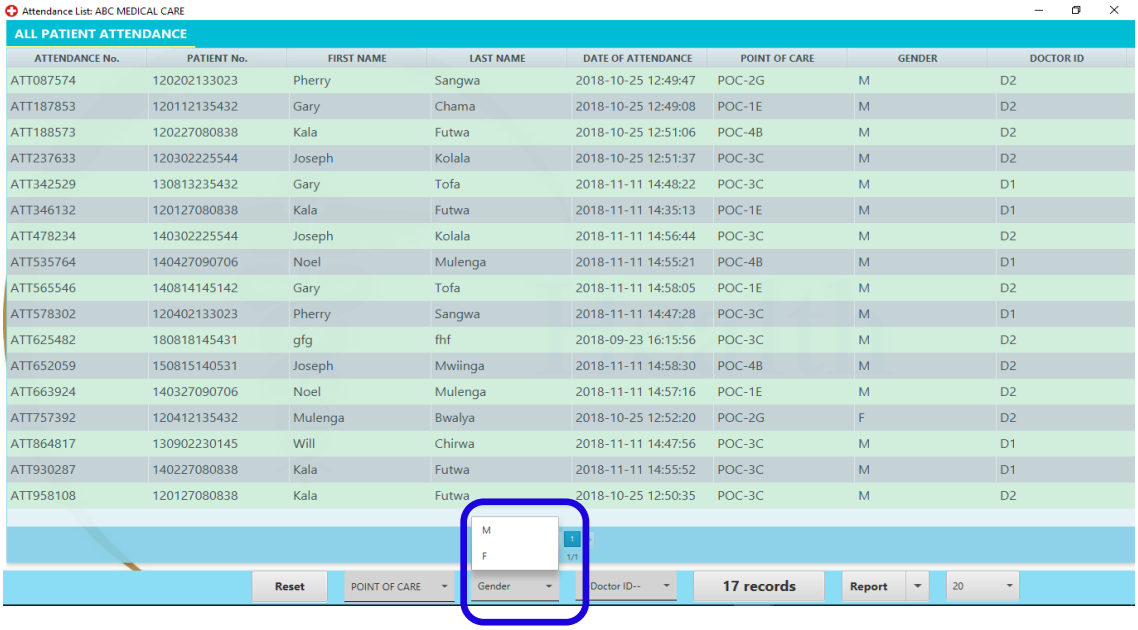

Figure 4.27: View Patient details filtered by Gender

Figure 4.27 shows patient details filtered according to their gender.

Filter by Point of Care (POC)

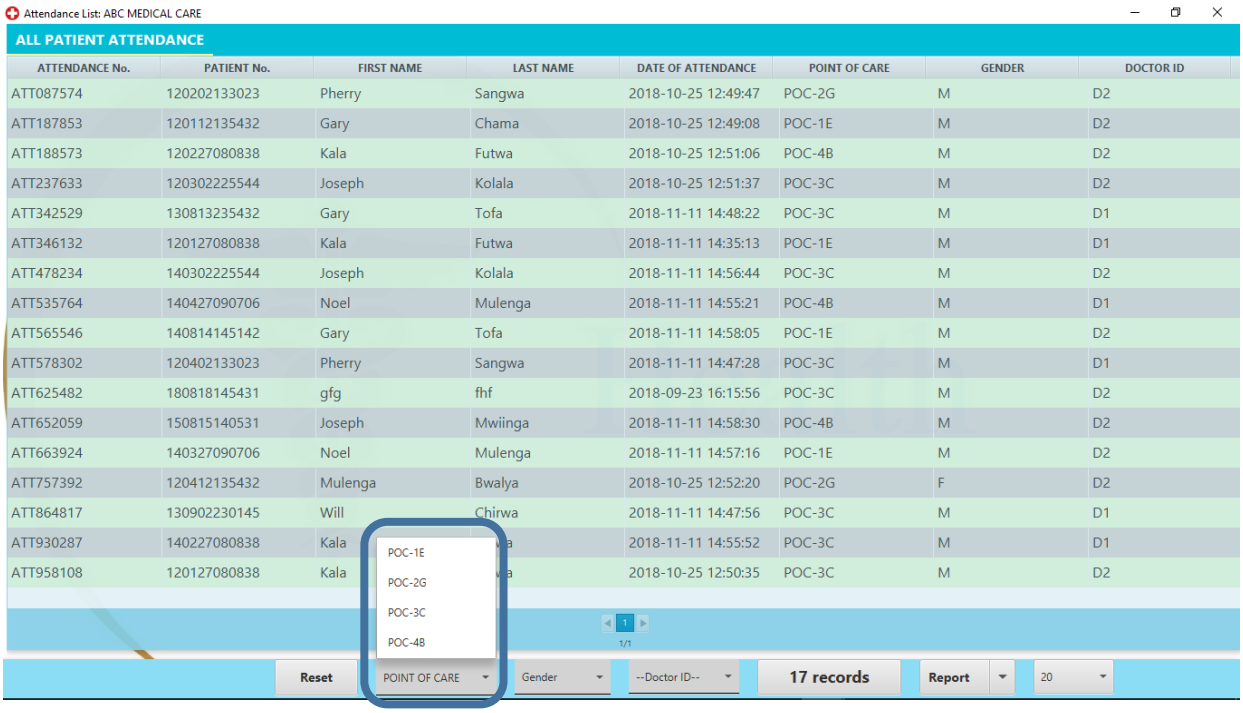

Figure 4.28: View Patient Details by Point of Care

Figure 4.28 shows patient details filtered according to the point of care. Point of Care is the clinic or specialized unit handling the case.

Generating an Excel Sheet from the Patient's Attendance List

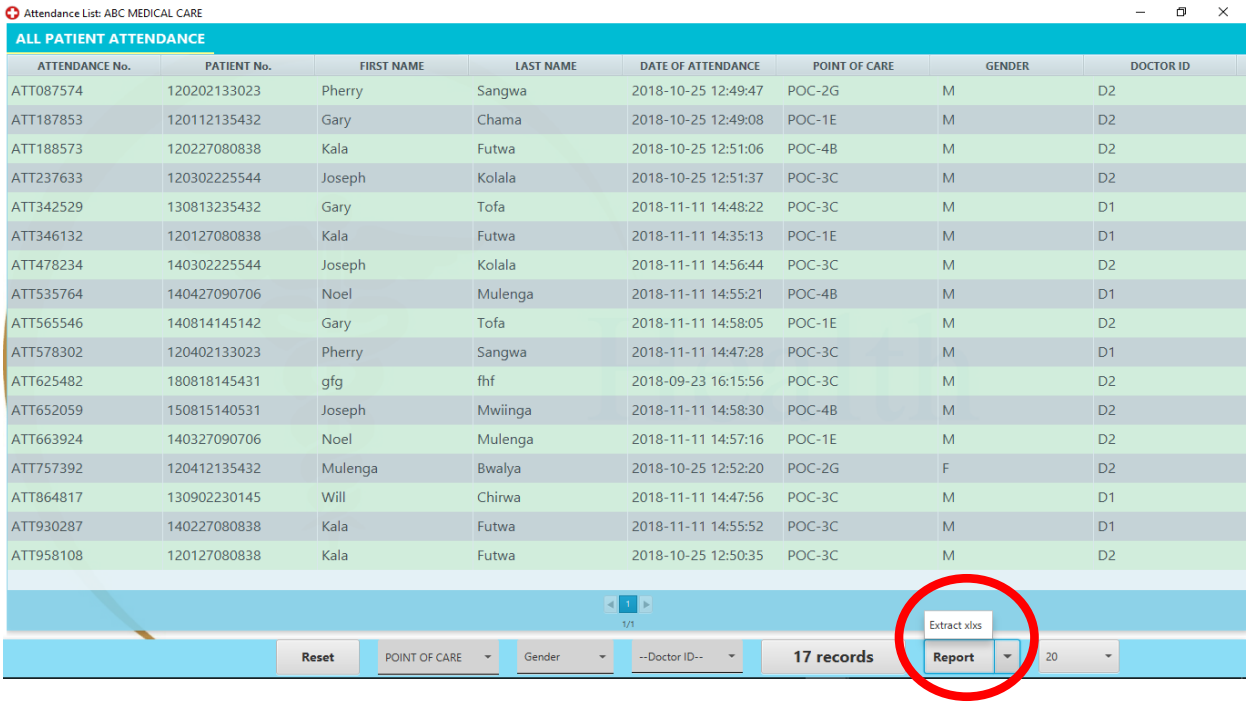

Figure 4.29: Generating Reports from patient list

b. View Queued

The **View Queued** Tab allows the user to check the patients who are awaiting attendance by a doctor. Once clicked, the user will be navigated to the viewing window, which gives the user two views, that is, queued for attendance and queued for surgery. The window also shows the number of records currently entered as shown in the figure below:

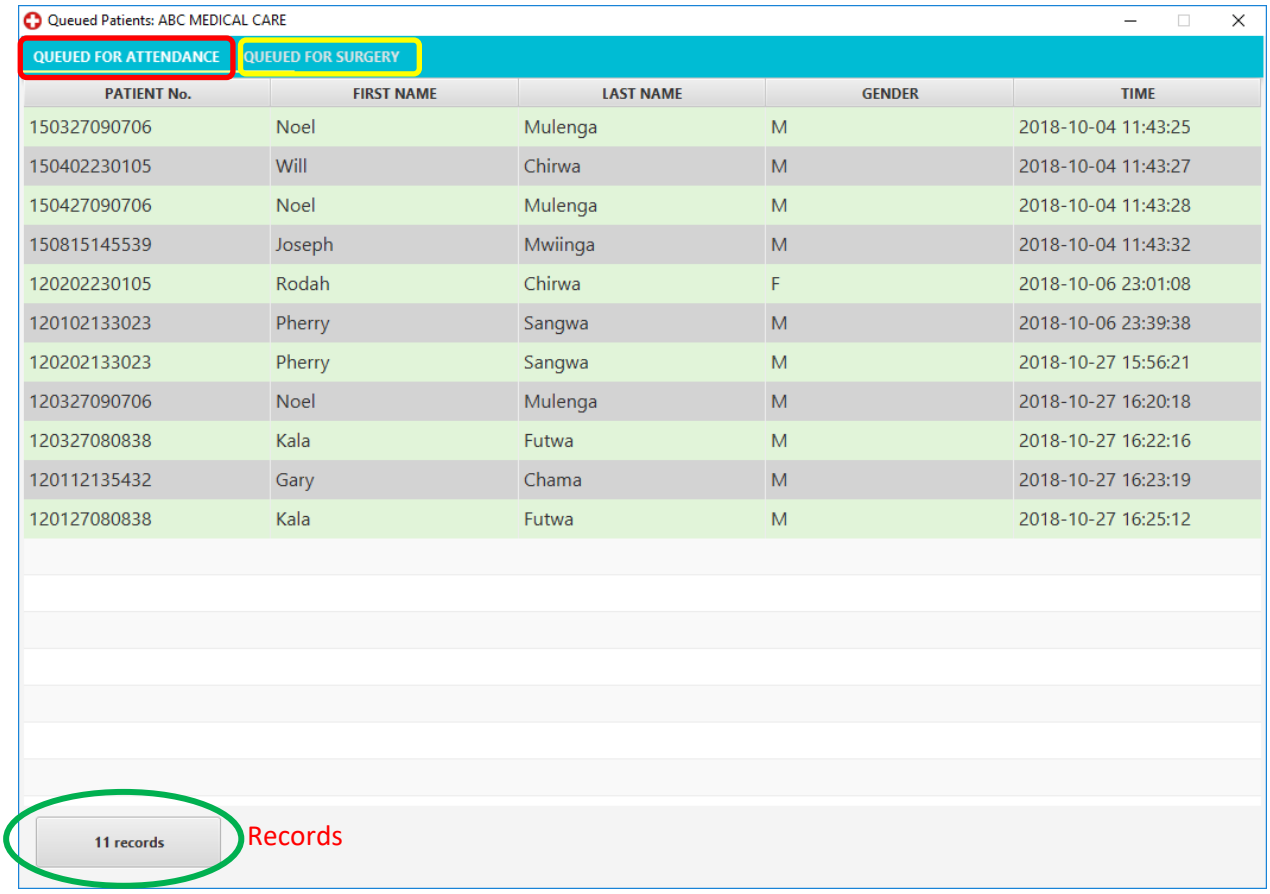

Figure 4.30: Queued Patients

c. View Patient

When clicked, the **View Patient** Tab invokes the window showing the Patients' Details. The main patients viewing window has got four tabs, **Patient Bio**data, **More Patient Details, Referral Details** and **Insurance Details.** These tabs help the user view different details concerning the Patient. It allows the user to perform filtering functions such as filtering patients by **Gender, Employment, Employment and Marital Status.**

Patient Biodata Tab

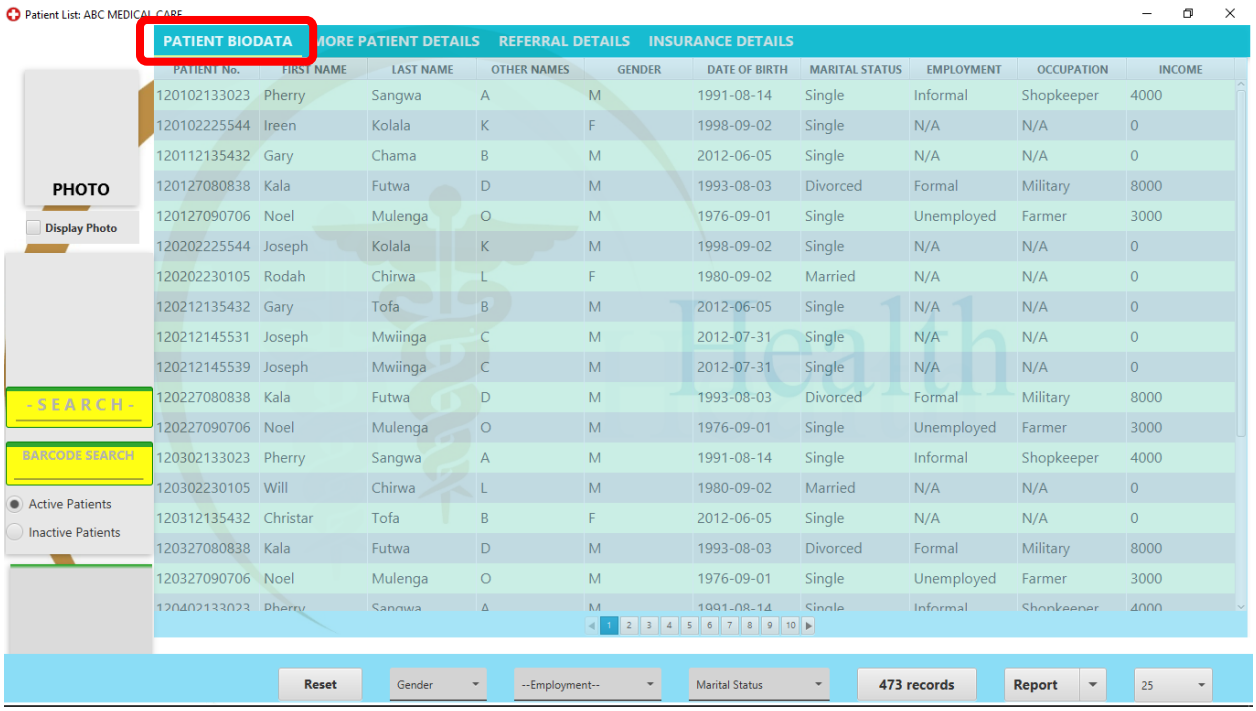

Figure 4.31: Patient biodata

Figure 4.31 shows the form from which the user can see all the details of patients from biodata to insurance particulars. The panel on the left with a photo placeholder and a blank space is used to display summary details of patients when the user clicks on any listed record.

Patient can be searched by entering the patient identification number in the yellow highlighted field shown in figure 4.31 or by placing the curse in the barcode field and using the barcode scanner to retrieve the specific patient details.

Select from the Table rows

As shown in the figure below, selection of a patient from the table row will enable the user to display a particular patient's details in the left side bar. The details shown for this selection include the Patient's ID, Photo and Name. The button below the image **"Change Photo"** allows the user to update the patient's photo.

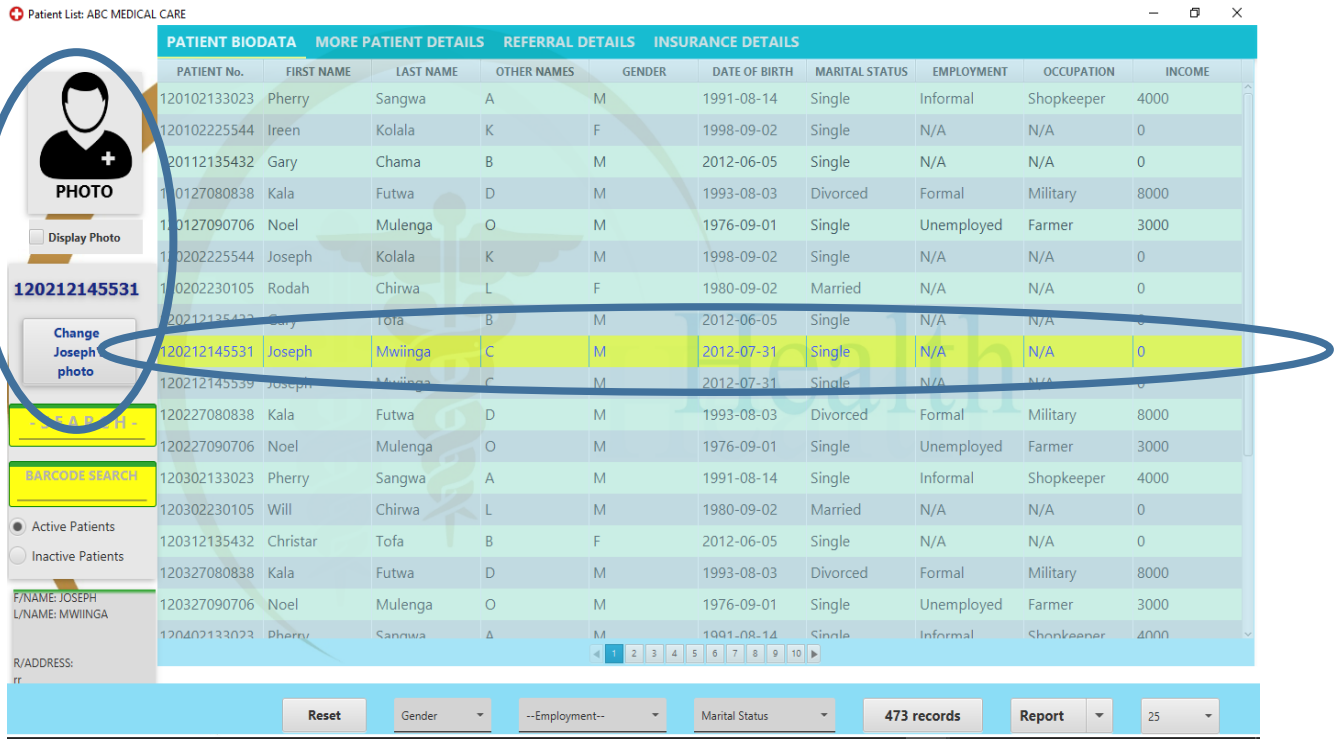

Figure 4.32: Selecting patient

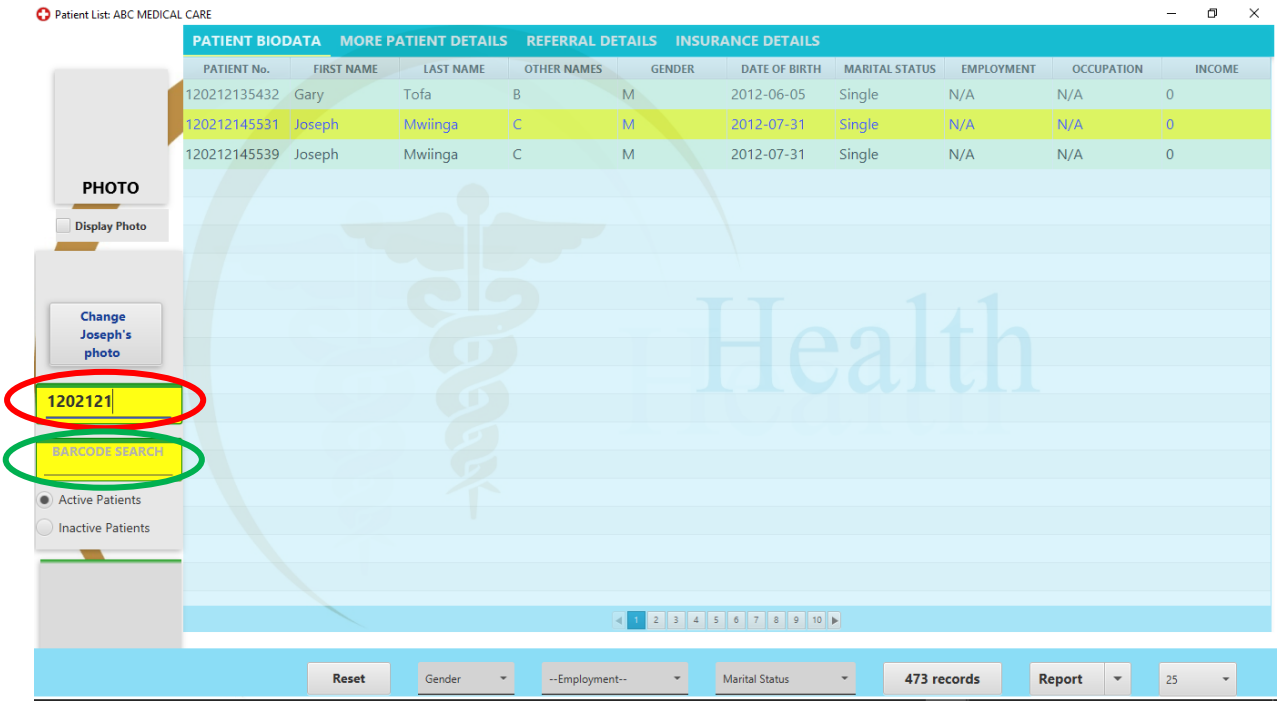

Figure 4.33: Searching for a patient

Doctor Profile

The image in figure 4.34 is the doctor-landing page.

Doctor Home Page

Figure 4.34: Doctor Home page

The doctor's home page has six tabs; these include *Queued Patients, My List of Patients, My Attendance, All Attended Patients, and Attendance Information*.

Queued Patients

This tab enables the doctor to check the queued patients. The doctor can decide to attend to a patient by clicking on the attend button as illustrated below.

To attend to a patient, the doctor needs to select a patient by clicking on the corresponding row, then click on the **attend button**. Once clicked the doctor is to further options as illustrated below.

Figure 4.35: Doctor attending to patient

Attendance Button clicked:

Once the attend button is clicked, the **Patient Attendance** Form shown in figure 4.36 is displayed**.**

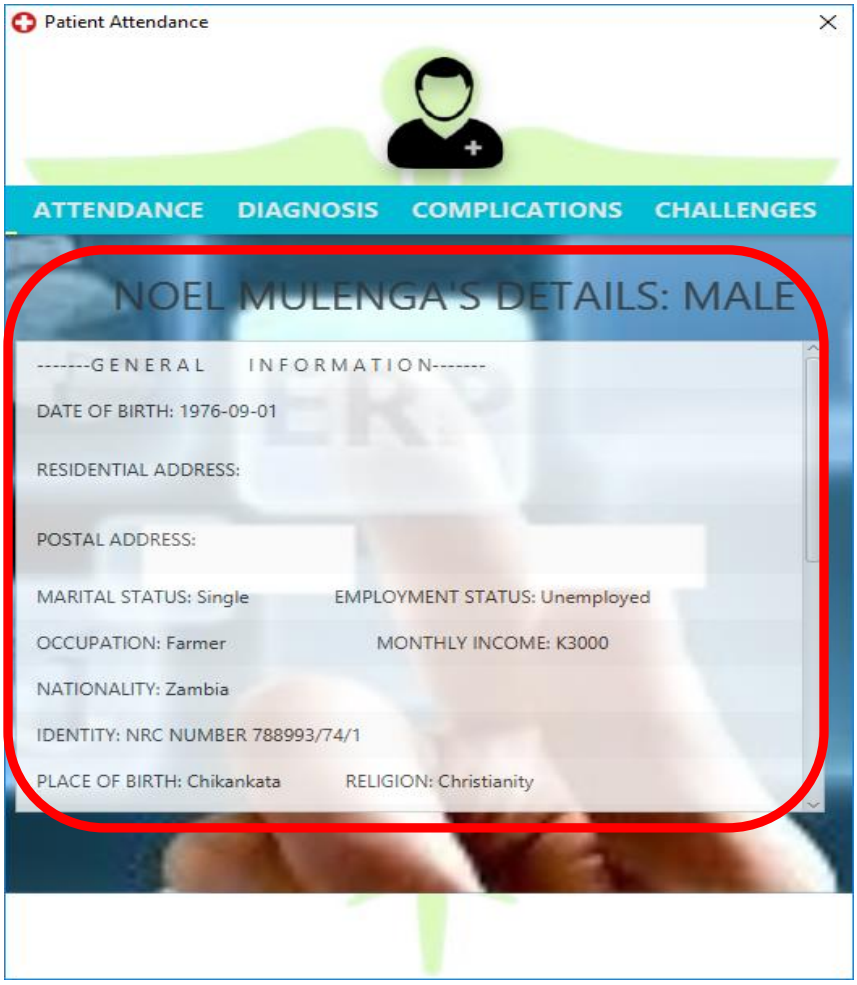

Figure 4.36: Summary patient details as seen by the doctor

Once verification of the Patient's details is done, the doctor can **click on Attendance Tab** to proceed with the attendance process as shown in figure 78. This will generate the attendance Identification number for the diagnosis.

Figure 4.37: Attendance Details

Next, the doctor can click the **Diagnosis Tab**, which provides a form to enter a diagnosis. The diagnosis is aided with the **SNOMED CT application** containing a variety collection of diagnoses. Each diagnosis has a code that is the **Concept ID** and **Description ID** given in the SNOMED app. The SNOMED Concepts can be accessed online or Offline accessed from the file directory on the computer running the system (this means that the application must installed on the machine for it to be accessed).

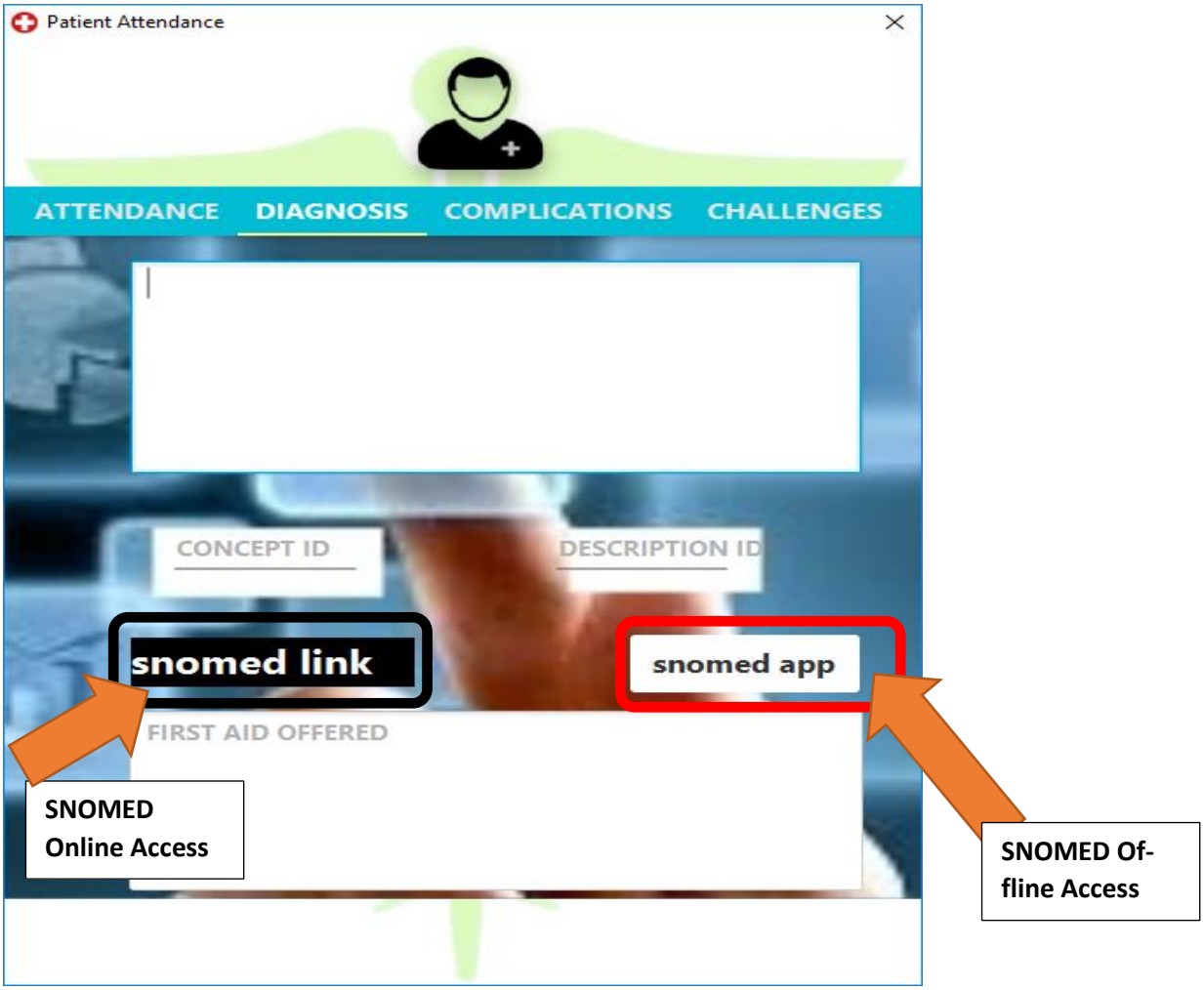

Figure 4.38: Snomed CT Incorporation in diagnose

The diagnosis has the place for entering the diagnosis written in the doctor's own understanding. The other place is where the Snomed CT Concept Id is entered.

Offline SNOMED Access

When the offline SNOMED CT Button is clicked, the screen shown in figure 4.39 is displayed. This figure shows a third part Snomed CT browser called **ClinicClue Xplore.** The doctor can enter the diagnosis in the text box shown. This diagnosis will result in a list of synonyms. When listed diagnosis which described the exact concept that the doctor wants to communicate.

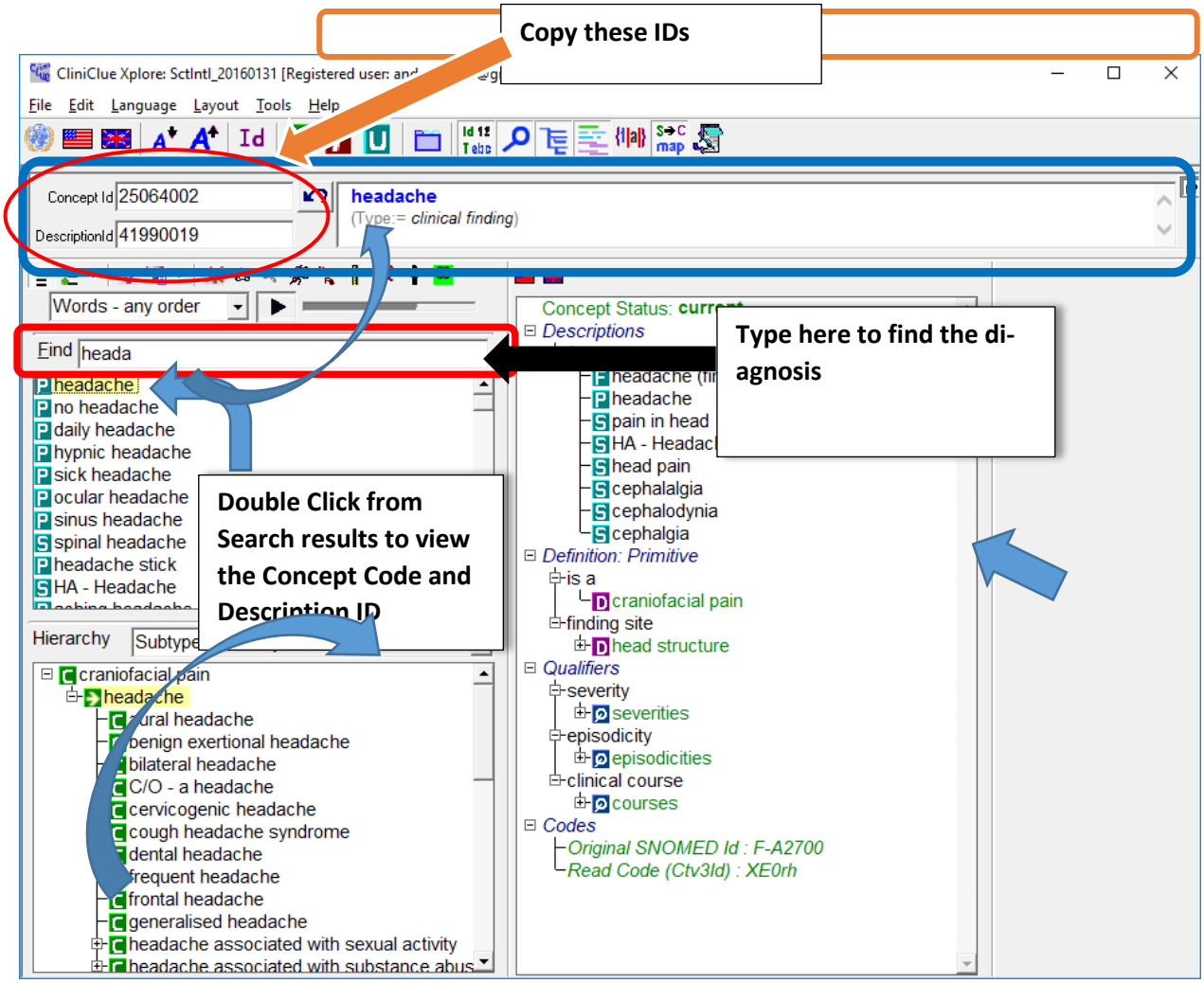

Figure 4.39: Offline Snomed CT Concept ID Access

Once the diagnosis search is done, copy the concept ID and the description ID is paste it in the fields respectively, as shown below:

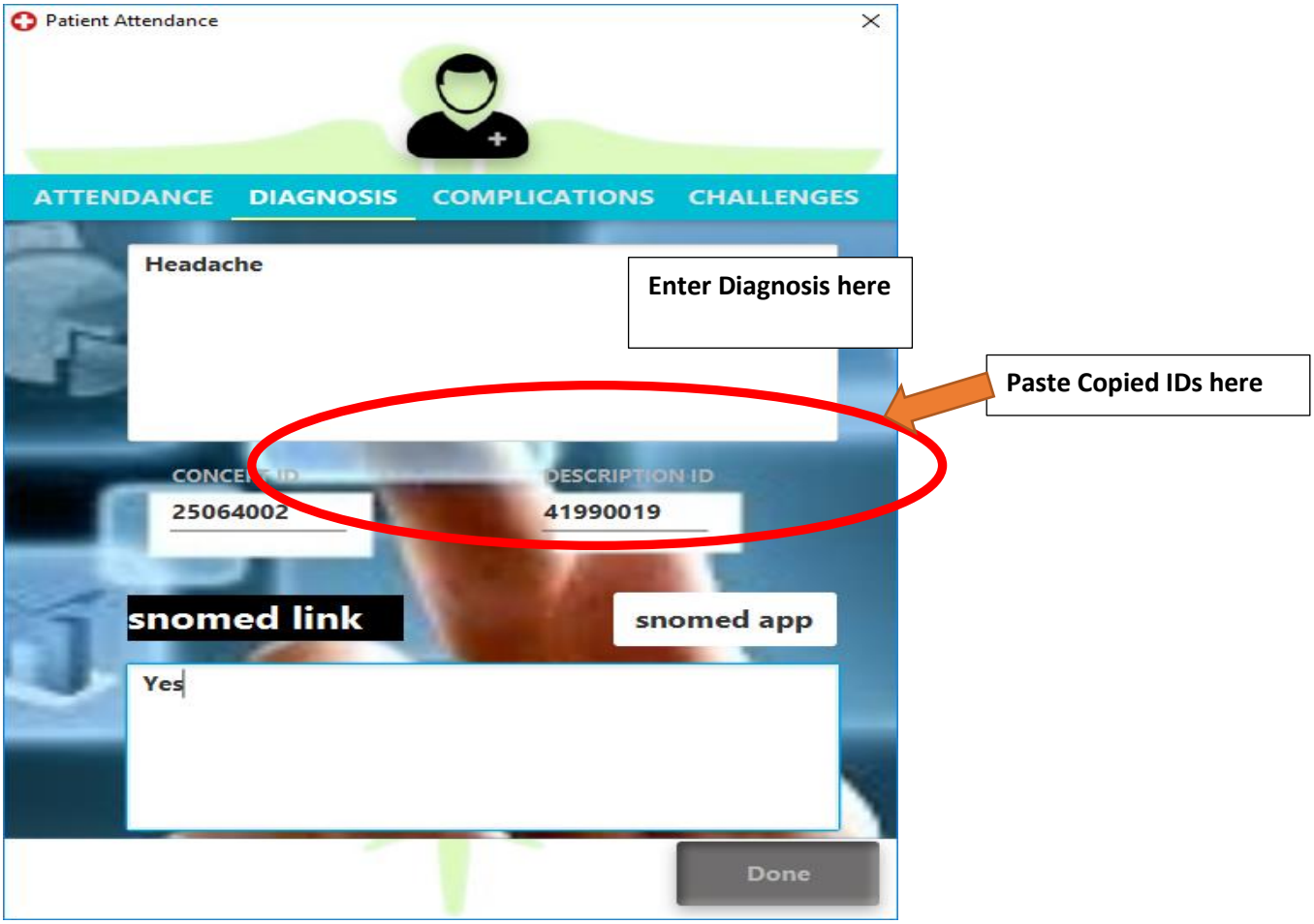

Figure 4.40: Diagnosis with SNOMED CT Concept ID and Description

As a way to plan for a surgery operation, the tabs showing complications and challenges are also filled as in the diagnosis.

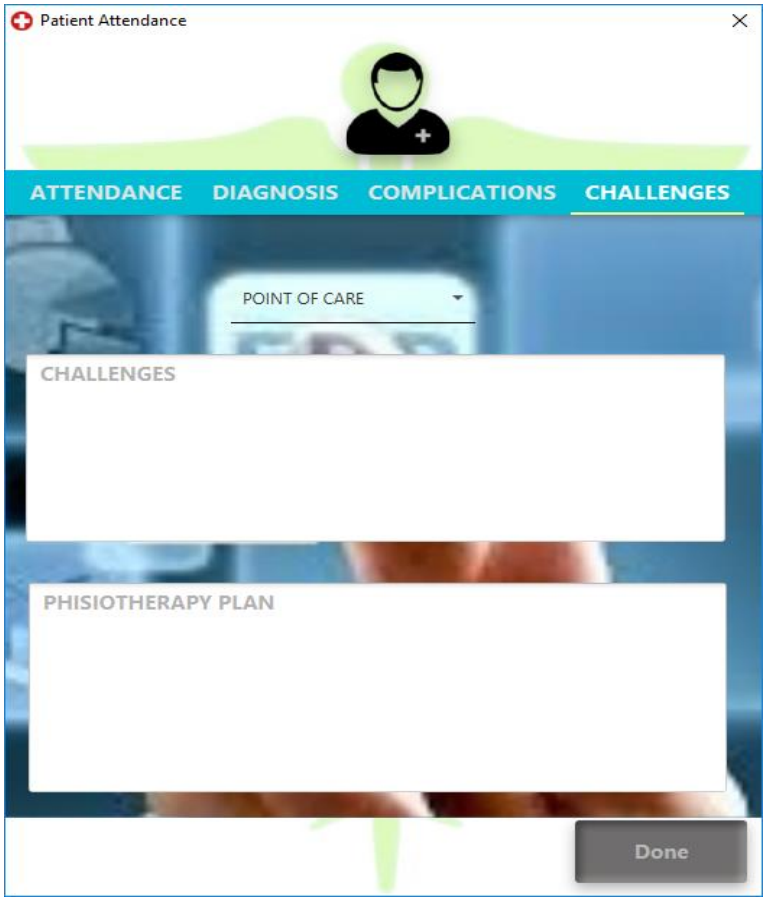

Figure 4.41: Challenges table form

My Patient List Tab

This tab shows a list of a doctor's Patient List. The doctor can decide to admit the patient by selecting a patient and clicking on **admit.**

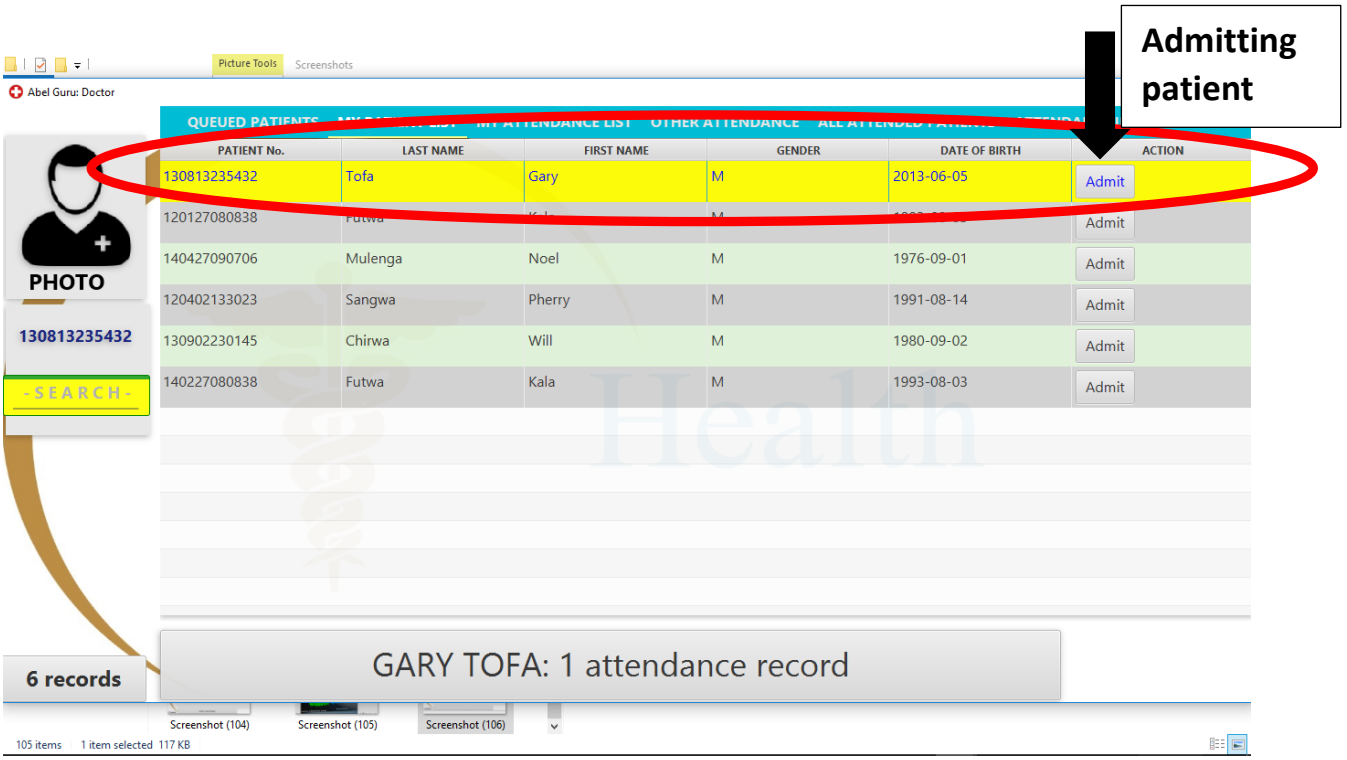

Figure 4.42: admitting a patient

Once the **admit button** is clicked, the user is navigated to the Patient admission window as shown figure 4.43. The admission window has two tabs, the **details and ward details**.

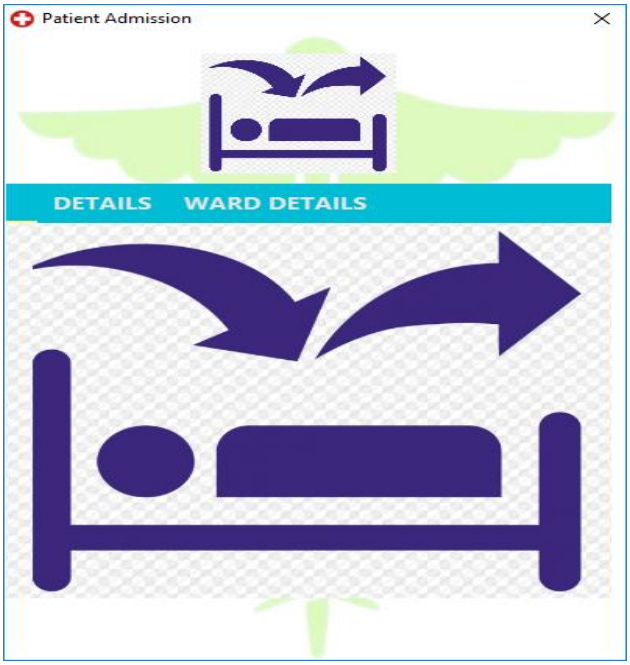

Figure 4.43: patient admission

Figure 4.43 shows the patient admission ward details populated.

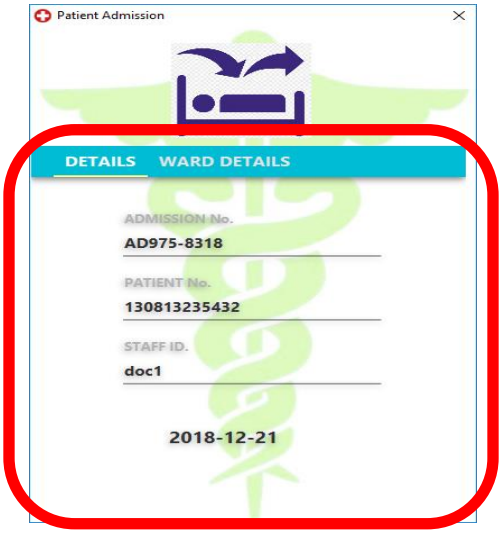

Figure 4.44: Admission Details

This show the Admission Number, Patient Number, the Staff Number (Doctor in this case) and the date of admission.

My Attendance List

This is a doctor's Patient attendance list. The doctor can queue or update attendance.

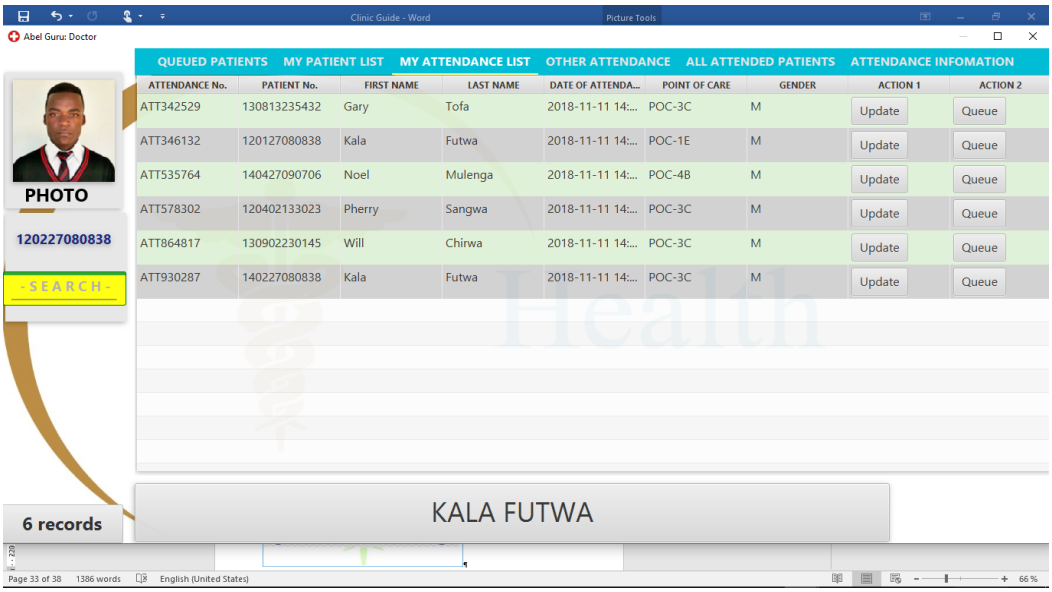

Figure 4.45: attendance list

Figure 4.45 show Doctor's attendance list.

To update, select a patient, click the Update Button, and follow the prompts. The records to update are the diagnosis, complications etcetera as earlier illustrated in the

Queued Patients Tab.

Other Attendance

This also shows the pending attendances as shown below.

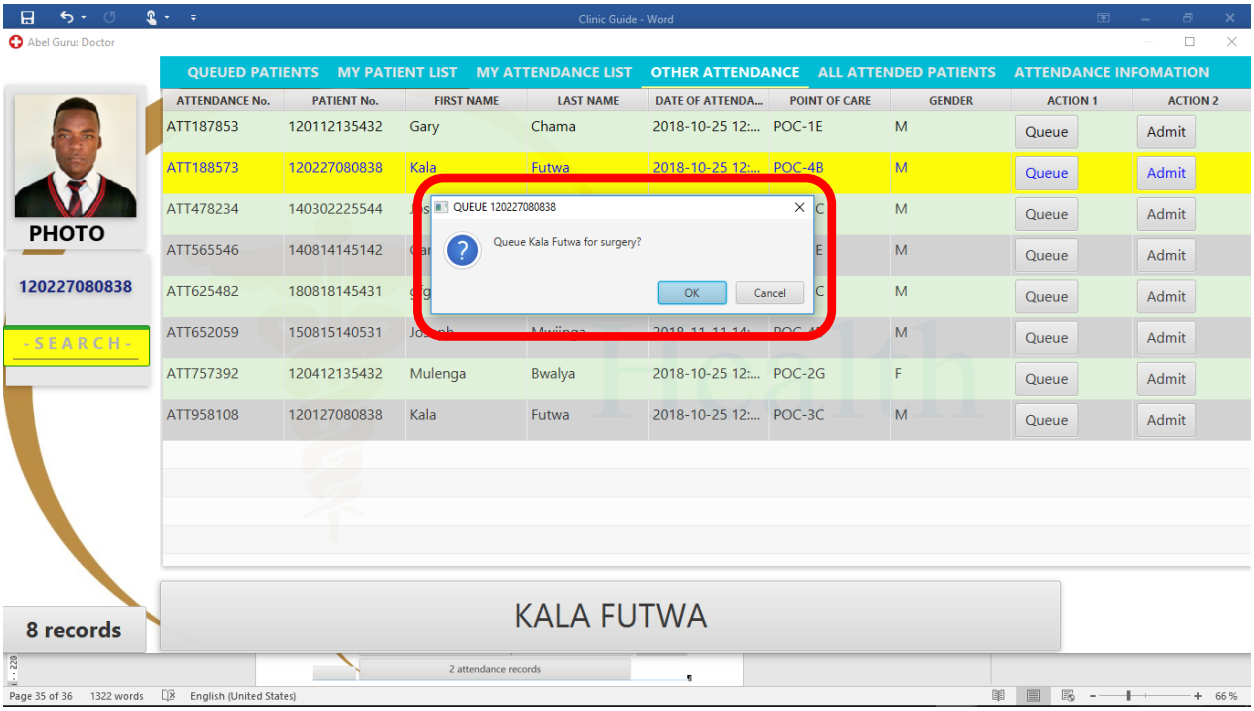

Figure 4.46: other attendance tab form

Once a patient is selected, and queue clicked, a dialog box asking the user to confirm queuing pops up. When **OK** is clicked, the user will be queued successfully as shown in figure 4.46, otherwise, the queuing is **cancelled using the Cancel Button. Note,** the admit button works in like manner as the earlier illustrated.

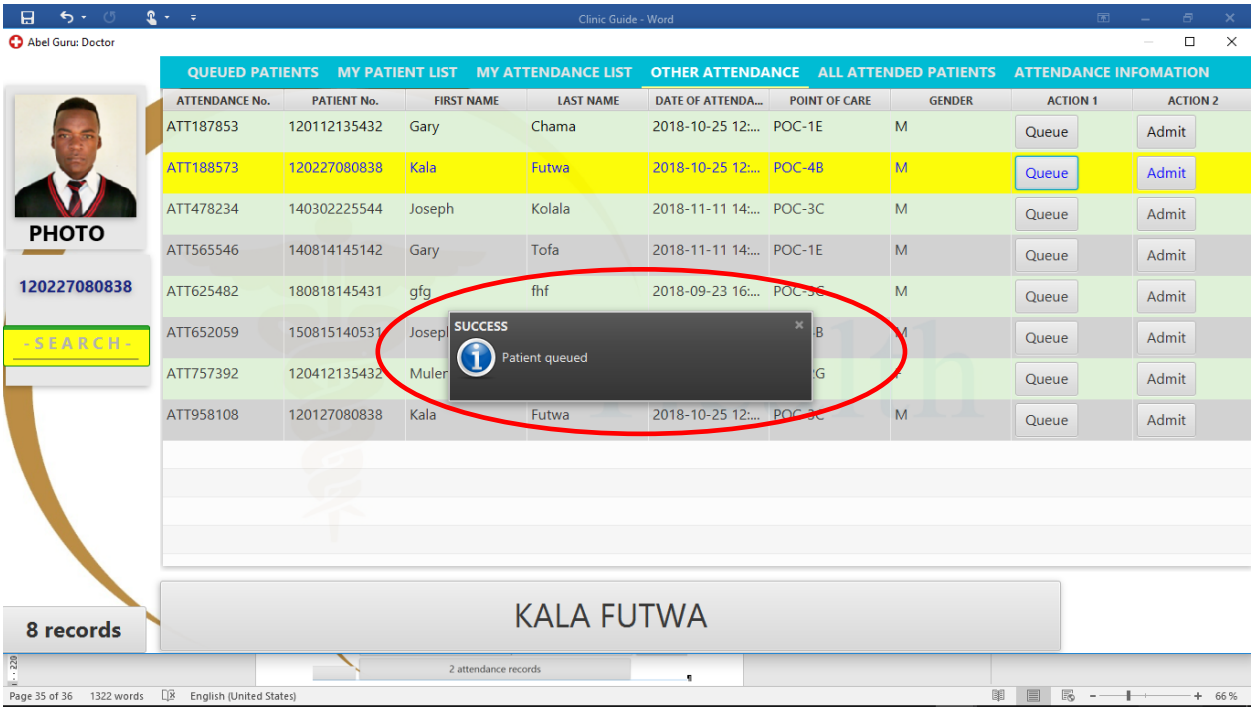

Figure 4.47: Patient queued up for surgery

Attendance Information

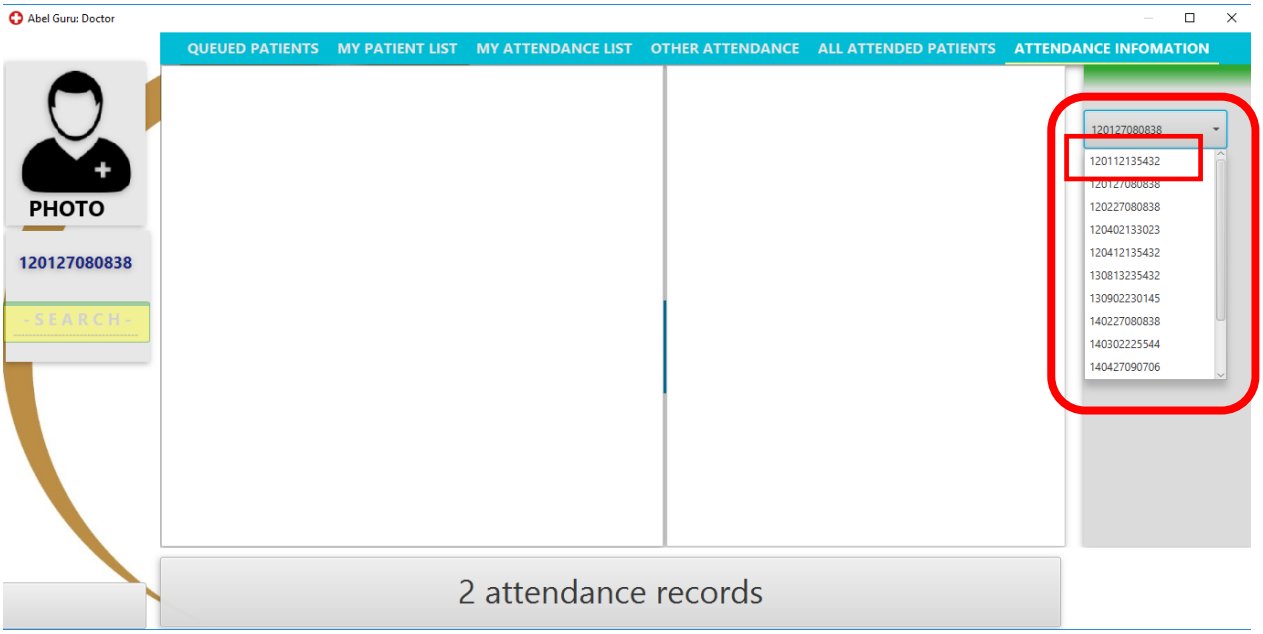

Figure 4.48: patient attendance record

This helps the user to select the Patient Number and display the Patient's Attendance details. Once the Patient's Number is clicked, the attendance information of the patient is displayed in the middle pane as shown below.

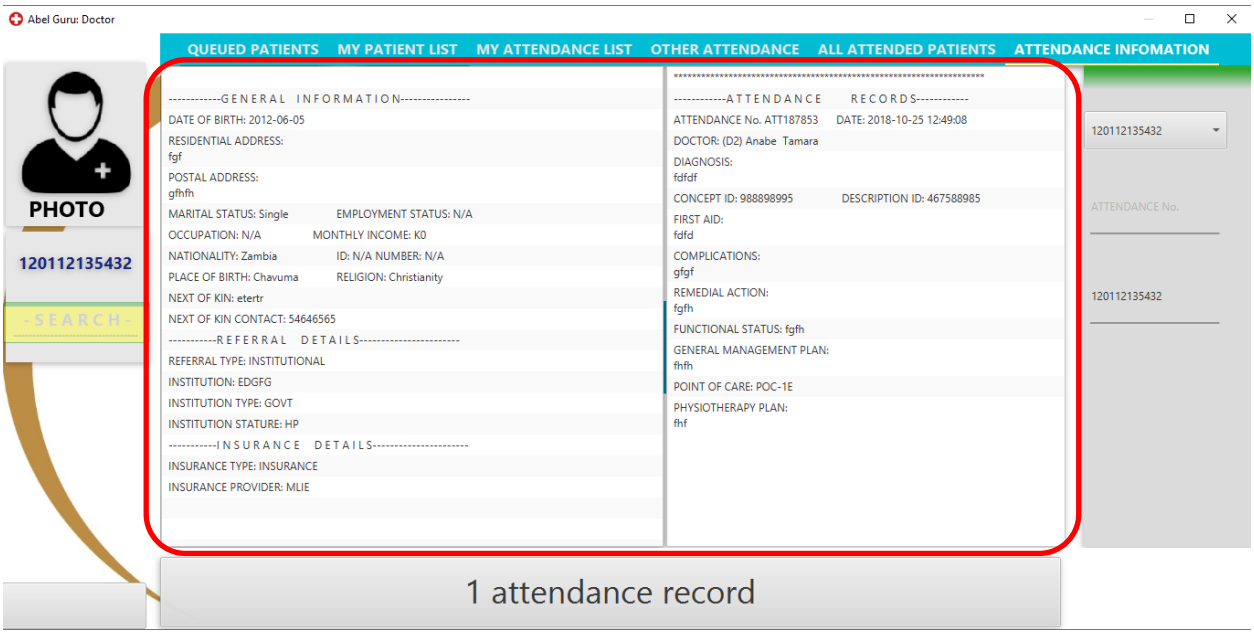

Figure 4.49: Patient attendance record

Possible Application of the System

The aim of the research was to develop a system that enables the clinicians and administrators in a surgery department to automate the record system in use. The system is designed for use at the university Teaching Hospital in Lusaka surgery department, but the same system can be used by any other hospital to automate and standardize their record keeping system.

Summary

This chapter has seen the successful implementation of a Snomed CT Electronic Health Record System prototype that can automate the University Teaching Hospital Surgery Department record system.

CHAPTER 5: DISCUSSION AND CONCLUSION

5.1 Introduction

This chapter present the discussion and conclusion of the research based on the research objectives stated in chapter one.

5.2 Discussion

This section, results presented in chapter 4 are discussed to show how they relate to the objectives of the research.

5.2.1 Baseline study

The baseline study was conducted in order to understand the system, which is use, and the challenges faced in using this system. The study involved the administration of a questionnaire and record inspection.

The looking at the demographic data, the questionnaire focused on understanding the ICT infrastructure. Therefore, the question concerning the existence of personal computers in the sections of the department of surgery. 72.5% of respondents reported that they had computers against 27.5%, which did not have computers. This simply means that the department has enough computers support automation.

The next question was trying to find out if there is any Electronic Record System to help with automating records. 2.5% reported that the department had fully implemented electronic record system, 60% reported that the department was using both paper based and electronic health record system, and 37.5% reported that they did not have an electronic record system. The 2.5% which indicated use of Electronic Health Record System could be those who are involved with Smart care, District Health Management System or are using the excel database to keep information. In addition, this case with those indicated that they are using both manual and electronic health system. The 37.5% reported the real situation on the ground which is that there not using any form of electronic health record system. This is the case because the interviews and discussions with held with the stakeholder reviewed that they do not have any electronic health record system.

The survey sought to know whether information required for surgery was known in advance by those involved, the response given using a Likert scale (1 strongly disagree, 2 disagree, 3 uncertain, 4 agree, 5 strongly agree) to the statement: the scheduling of patients to be operated on is known in advance, except for emergency cases were, 7.5%, 40% were uncertain, 42.5% agree, 10% strongly agree. Combined total of those whose were uncertain and those who disagreed, we have almost 40.75% of respondents who disagree or are not sure about that every important statement.

This lack of information is because there is no system to help enter pre-operating data and to schedule patients. This was one the biggest challenges is being faced by the department.

5.2.2 Record Inspection

To validate the questionnaire results, the author went through department in order to understand the business processes and to check the record are kept.

The images shown in chapter 4 are just of sample of all the records captured. There was also mention discovery by the author that the manual records are destroyed every after ten to fifteen years due to space. This results in loss of patient history data as well as secondary data which could be used by researchers. It is for the afore stated reasons that automation was deemed very necessary. The author, through literature review and interviews with stakeholders of the department, came across the need for developing sematic standardised electronic health record system. In this case, Snomed CT was picked since it is being used over 53 countries of the world.

5.2.3 Business Process Mapping

Major Business Processes were identified in this research and have been shown in chapter 3. Some of the benefits which will be gained from automating the system are have been listed

- Patient details will be retrieved faster with the use of a barcode scanner to retrieve a patient's details.
- Electronic copies of the patient records will eliminate the fear of patients going away with physical patient files.
- Scheduling of patient for doctor attendance and for surgery is now more optimised and resources required for surgery are known in advance through the system.

- The possible misunderstanding leading to misinterpretation of clinicians' diagnosis and prescription will be eliminated by the use of Snomed CT concept codes. This will contribute to patient safe by the elimination of errors due to language ambiguities.
- There is improved security as access to documents or data is restricted to only personnel with access rights. This mechanism makes audit trains easy to manage.

Developed software prototype

The system developed using JavaFX and MySQL as major development tools, is able to create user and assign rights, create patient details, schedule and queue patient for doctor attention as well as for surgery operation, capture diagnosis and prescription, enter pre-operating information and post-operating information and store all this information in MySQL database. In implementing Standardised Interoperability, Snomed CT is implemented by making use of an offline and online browser. The browsers enable the doctor or surgeon to get the Snomed Identification number and save it in addition to their own written diagnosis and prescriptions.

The system is implemented by deploying the database in the cloud or on a computer, which is part of a local area network. At the installation stage, the system will prompt to specify either, the uniform resource locater (URL) or path to the database. This system will help to improve patient care and remove toxicity in the working environment that every clinician faces when encountering the challenges of using manual records.

Conclusion

The University Teaching Hospital Department of Surgery is using manual system in carrying out their major business which surgery operations. Despite efforts of computerisation, no standardised system exists in the department.

It also been observed that the department is faced with problems associated with manual record systems. Scheduling of patients using manual system is rigid and unreliable.

The prototype using Snomed CT as a sematic standard was designed and prototype implemented using JavaFX and MySQL as the Database Management System. Snomed CT was added using the international online Snomed Browser as well as the Offline ClinicClue Xplore system, which has been designed to install together with the system.

Future Works

Despite the achievements stated that this study has scored, there remains a lot of work tobe done. The hospital working environment is much interconnected and as such, the following works can be added to make the system more viable:

- Interconnecting the system with X-ray and laboratory systems
- Creating a mobile system, which is a replica to enable clinicians attend to patients remotely.
- Incorporate HL7 and develop an API to enable other departments and hospitals interconnect with this system.
- Create a patient portal so that patients can have access to their own records.

Summary

In summary, the objective set out in chapter one has have been met, the research question has also been answered. The research hopes that the system developed will be put to use and that all the user will be adequately trained to use the system.

Chapter Summary

The chapter has discussed the results of the research focussing on the objectives set in chapter one and how each one has been met following the materials and methods discussed in chapter 3.

REFERENCES

[1] C. Mwangi, “Computerization of the Kenyan Health Care Records,” no. April, 2013.

[2] X.-Y. Zhang and P. Zhang, “Recent perspectives of electronic medical record systems.,” *Exp. Ther. Med.*, vol. 11, no. 6, pp. 2083–2085, Jun. 2016.

[3] T. MITRE Corporation, “Accelerating and Enhancing Research From Basic Discovery to Improved Patient Care,” 2006.

[4] N. L. George Kokolakis, “The role of Interoperability in global e-Governance: Cloud Computing, Smart Grid and the Internet of Things.”

[5] A. W. Group, “Electronic Documentation Templates Support ICD-10-CM/PCS Implementation (2015 update),” *J. AHIMA*, 2015.

[6] M. K. Rizer, B. Kaufman, C. J. Sieck, J. L. Hefner, and A. S. McAlearney, “Top 10 Lessons Learned from Electronic Medical Record Implementation in a Large Academic Medical Center.,” *Perspect. Heal. Inf. Manag.*, vol. 12, no. Summer, p. 1g, 2015.

[7] P. Mishra, J. C. Kiang, and R. W. Grant, “Association of Medical Scribes in Primary Care With Physician Workflow and Patient Experience,” *JAMA Intern. Med.*, vol. 178, no. 11, p. 1467, Nov. 2018.

[8] University Teaching Hospital, “SURGERY | The University Teaching Hospital,” 2015. [Online]. Available: http://www.uth.gov.zm/?page_id=663. [Accessed: 02-Sep-2018].

[9] F. Williams and S. A. Boren, “The role of the electronic medical record (EMR) in care delivery development in developing countries: a systematic review.,” *Inform. Prim. Care*, vol. 16, no. 2, pp. 139–45, 2008.

[10] D. Lehn, “Improving pre-surgery information with a digital information system,” 2017.

[11] R. I. Mutemwa, “HMIS and decision-making in Zambia: re-thinking information

solutions for district health management in decentralized health systems," *Health Policy Plan.*, vol. 21, no. 1, pp. 40–52, Jan. 2006.

[12] "Zambia: Making the 'SmartCare' Transition to Electronic Healthcare Records | Elizabeth Glaser Pediatric AIDS Foundation." .

[13] T. Schramme and S. D. Edwards, *Handbook of the philosophy of medicine*. .

[14] N. Sartorius, "The meanings of health and its promotion.," *Croat. Med. J.*, vol. 47, no. 4, pp. 662–4, Aug. 2006.

[15] M. Garau, K. K. Shah, P. Sharma, and A. Towse, "IS THE LINK BETWEEN HEALTH AND WEALTH CONSIDERED IN DECISION MAKING? RESULTS FROM A QUALITATIVE STUDY.," *Int. J. Technol. Assess. Health Care*, vol. 31, no. 6, pp. 449–56, 2015.

[16] "HEALTH BUDGET HEADLINE MESSAGES," 2017.

[17] M. Farhadi, R. Ismail, and M. Fooladi, "Information and communication technology use and economic growth.," *PLoS One*, vol. 7, no. 11, p. e48903, 2012.

[18] K. Nuamah-Gyambrah, M. Offei Otu, and F. Agyeiwaa, "Role of Information and Communication Technology (ICT) in the Survival of Small and Medium Scale Enterprises (SME's) in Ghana: Evidence from selected Small and Medium Scale Enterprises in New Juaben Municipality, Koforidua."

[19] "2018 IT (Information Technology) Industry Trends Analysis | CompTIA." [Online]. Available: https://www.comptia.org/resources/it-industry-trends-analysis. [Accessed: 02-Sep-2018].

[20] I. of M. (US) C. on R. A. D. P. H. C. F. and D. Systems, J. M. Corrigan, A. Greiner, and S. M. Erickson, "Information and Communications Technology Infrastructure: A 'Paperless' Health Care System," 2002.

[21] J. I. Westbrook *et al.*, "Use of information and communication technologies to support effective work practice innovation in the health sector: a multi-site study," *BMC Health Serv. Res.*, vol. 9, no. 1, p. 201, Dec. 2009.

[22] P. Shekelle, S. C. Morton, and E. B. Keeler, “Introduction,” 2006.

[23] D. A. Ludwick and J. Doucette, “Adopting electronic medical records in primary care: Lessons learned from health information systems implementation experience in seven countries,” *Int. J. Med. Inform.*, vol. 78, no. 1, pp. 22–31, Jan. 2009.

[24] “Patient record | definition of patient record by Medical dictionary.” [Online]. Available: https://medical-dictionary.thefreedictionary.com/patient+record. [Accessed: 24-Dec-2018].

[25] *The Computer-Based Patient Record*. Washington, D.C.: National Academies Press, 1997.

[26] I. of M. (US) C. on I. the P. Record, R. S. Dick, E. B. Steen, and D. E. Detmer, “Introduction,” 1997.

[27] “RECORDS MANAGEMENT HANDBOOK.”

[28] G. Eysenbach, “What is e-health?,” *J. Med. Internet Res.*, vol. 3, no. 2, p. E20, Jun. 2001.

[29] P.-G. Svensson, “eHealth Applications in Health Care Management.,” *eHealth Int.*, vol. 1, no. 1, p. 5, Sep. 2002.

[30] “electronic-health-record-strategic-assessment.” .

[31] “No Title,” no. November, 2015.

[32] C. P. Stone, “A Glimpse at EHR Implementation Around the World: The Lessons the US Can Learn,” 2014.

[33] J. Braa and C. Hedberg, “The Struggle for District-Based Health Information Systems in South Africa,” *Inf. Soc.*, vol. 18, pp. 113–127, 2002.

[34] L’Oréal S.A., “2017 Annual Report,” *2017 Annu. Rep.*, p. 72, 2017.

[35] H. S. F. Fraser, P. Biondich, D. Moodley, S. Choi, B. W. Mamlin, and P. Szolovits, “Implementing electronic medical record systems in developing countries,” *Inform. Prim. Care*, vol. 13, no. 2, pp. 83–95, 2005.

[36] A. Lee Gonzalez Fanfalone, “Benefits and Costs of the Infrastructure Targets for the Post-2015 Development Agenda,” 2015.

[37] C. Castaneda *et al.*, “Clinical decision support systems for improving diagnostic accuracy and achieving precision medicine.,” *J. Clin. Bioinforma.*, vol. 5, p. 4, 2015.

[38] E. V Murphy, “Clinical decision support: effectiveness in improving quality processes and clinical outcomes and factors that may influence success.,” *Yale J. Biol. Med.*, vol. 87, no. 2, pp. 187–97, Jun. 2014.

[39] S. A. Bantom and R. De La Harpe, “ACCESSIBILITY TO PATIENTS’ OWN HEALTH INFORMATION: A CASE IN RURAL EASTERN CAPE, SOUTH AFRICA,” 2016.

[40] S. Bah, H. Alharthi, A. A. El Mahalli, A. Jabali, M. Al-Qahtani, and N. Al-kahtani, “Annual survey on the level and extent of usage of electronic health records in government-related hospitals in Eastern Province, Saudi Arabia.,” *Perspect. Heal. Inf. Manag.*, vol. 8, no. Fall, p. 1b, 2011.

[41] “Bolstering Use of Smartcare Electronic Health Records System | UNDP in Zambia.” [Online]. Available: http://www.zm.undp.org/content/zambia/en/home/presscenter/articles/2014/02/03/bolstering-use-of-smartcare-electronic-health-records-system.html. [Accessed: 30-Oct-2018].

[42] “Zambia Leads the Way in SmartCare Electronic Health Records System, A Benefit to Both Providers and Patients | Jhpiego.” [Online]. Available: https://www.jhpiego.org/success-story/zambia-leads-the-way-in-smartcare-electronic-health-records-system-a-benefit-to-both-providers-and-patients/. [Accessed: 30-Oct-2018].

[43] R. Dehnavieh *et al.*, “The District Health Information System (DHIS2): A literature review and meta-synthesis of its strengths and operational challenges based on the experiences of 11 countries,” *Heal. Inf. Manag. J.*, p. 183335831877771, Jun. 2018.

[44] “DHIS2 - Transforming Health IT Standards in the Developing World (Part 2) | Open Health News.” [Online]. Available: http://www.openhealthnews.com/articles/2017/dhis2-transforming-health-it-standards-developing-world-part-2. [Accessed: 26-Oct-2018].

[45] HIMSS, “Definition of Interoperability,” *Himss*, p. 2013, 2013.

[46] E. S. Torres, “Conception of a SNOMED CT and HL7 V3 standard : domain analysis model of preoperative anesthesia assessment project Recommended Citation,” 2010.

[47] E. S. Torres, “Conception of a SNOMED CT and HL7 V3 standard : domain analysis model of preoperative anesthesia assessment project,” no. May, 2010.

[48] A. S. Apostol, C. Catu, and C. Vernic, “Electronical Health Record’s Systems. Interoperability,” *An. Ser. Inform.*, vol. I, no. 1, p. 14, 2009.

[49] C. Soguero-ruiz *et al.*, “An Interoperable System toward Cardiac Risk Stratification from ECG Monitoring,” pp. 1–16.

[50] T. MITRE Corporation, “Accelerating and Enhancing Research From Basic Discovery to Improved Patient Care,” 2006.

[51] “Electronic Health Records - Association for Healthcare Documentation Integrity.” [Online]. Available: https://www.ahdionline.org/page/ehr. [Accessed: 25-Dec-2018].

[52] “The International Classification of Diseases, 10th Edition (ICD-10) - American Society for Radiation Oncology (ASTRO) - American Society for Radiation Oncology (ASTRO).” [Online]. Available: https://www.astro.org/Daily-Practice/Coding/The-International-Classification. [Accessed: 15-Jan-2019].

[53] “EICD Main Page.” [Online]. Available: http://www.eicd.com/EICDMain.htm. [Accessed: 15-Jan-2019].

[54] P. W. Gething *et al.*, “Improving imperfect data from health management information systems in Africa using space-time geostatistics.,” *PLoS Med.*, vol. 3, no. 6, p. e271, Jun. 2006.

[55] “Data Analytics with SNOMED CT – Case Studies,” pp. 1–34, 2015.

[56] S. Ct and D. Library, “Using LOINC with SNOMED CT,” no. April, 2017.

[57] A. Masi and A. Agrawal, “Developing a Search Algorithm and a Visualization Tool for SNOMED CT,” vol. 6, no. 4, pp. 19–23, 2015.

[58] “Data Analytics with SNOMED CT,” pp. 1–73, 2015.

[59] Mark D. McDougall, “HL7 - Health Level Seven International.” [Online]. Available: https://www.efmi.org/institutional-members/hl7-health-level-seven-international. [Accessed: 17-Dec-2018].

[60] “Introduction to HL7 Standards | HL7 International.” [Online]. Available: http://www.hl7.org/implement/standards/index.cfm?ref=nav. [Accessed: 17-Dec-2018].

[61] “Kanter- Interface terminologies for Africa,” no. December, 2013.

[62] J. J. Cimino, “In defense of the Desiderata,” *J. Biomed. Inform.*, vol. 39, no. 3, pp. 299–306, Jun. 2006.

[63] F. Shiferaw and M. Zolfo, “The role of information communication technology (ICT) towards universal health coverage: the first steps of a telemedicine project in Ethiopia.,” *Glob. Health Action*, vol. 5, pp. 1–8, 2012.

[64] N. Mbananga, S. Mniki, A. Oelofse, S. Makapan, and M. Lubisi, “A model of developing medical terms in indigenous languages: a step towards consumer health informatics in South Africa.,” *Stud. Health Technol. Inform.*, vol. 107, no. Pt 2, pp. 1216–8, 2004.

[65] “Scalability and Load Balancing in a Security Environment.”

[66] J. Duggan *et al.*, “The BigDAWG Polystore System.”

[67] P. Hepner, “Integrating Heterogeneous Databases: An Overview,” 1995.

[68] “Wide Area Network (WAN).” [Online]. Available: https://erg.abdn.ac.uk/users/gorry/eg3561/intro-pages/wan.html. [Accessed: 15-Jan-2019].

[69] “Wide Area Network - an overview | ScienceDirect Topics.” [Online]. Available: https://www.sciencedirect.com/topics/earth-and-planetary-sciences/wide-area-network. [Accessed: 15-Jan-2019].

[70] D. Craig and G. Farrell, “DESIGNING A PHYSICIAN-FRIENDLY INTERFACE FOR AN ELECTRONIC MEDICAL RECORD SYSTEM.”

[71] “5 dos and don’ts of EHR interface design | Healthcare IT News.” [Online]. Available: https://www.healthcareitnews.com/news/5-dos-and-donts-ehr-interface-design. [Accessed: 15-Jan-2019].

[72] “Pew: EHR Usability Concerns May Still Impact Patient Safety.” [Online]. Available: https://healthitanalytics.com/news/pew-ehr-usability-concerns-may-still-impact-patient-safety. [Accessed: 15-Jan-2019].

[73] C. Heath and P. Luff, “Documents and Professional Practice: ‘bad’ organisational reasons for ‘good’ clinical records,” ACM Press, 1996.

[74] “Zheng K et al. 2009 "An interface-driven analysis of user interactions with an electronic health records system." | AHRQ National Resource Center; Health Information Technology: Best Practices Transforming Quality, Safety, and Efficiency.” [Online]. Available: https://healthit.ahrq.gov/health-it-tools-and-resources/workflow-assessment-health-it-toolkit/research/zheng-k-et-al-2009. [Accessed: 15-Jan-2019].

[75] V. L. Patel, J. F. Arocha, and A. W. Kushniruk, “Patients’ and physicians’ understanding of health and biomedical concepts: relationship to the design of EMR systems,” *J. Biomed. Inform.*, vol. 35, no. 1, pp. 8–16, Feb. 2002.

[76] “Clinician-Patient Communication to Enhance Health Outcomes,” 1989.

[77] J. F. Ha and N. Longnecker, “Doctor-patient communication: a review.,” *Ochsner J.*, vol. 10, no. 1, pp. 38–43, 2010.

[78] H. Laerum, T. H. Karlsen, and A. Faxvaag, “Effects of scanning and eliminating paper-based medical records on hospital physicians’ clinical work practice.,” *J. Am. Med. Inform. Assoc.*, vol. 10, no. 6, pp. 588–95, 2003.

[79] “Now More Than Ever UNIVERSAL COVERAGE REFORMS SERVICE DELIVERY REFORMS LEADERSHIP REFORMS PUBLIC POLICY REFORMS.”

[80] D. Vanderpool, “EHR DOCUMENTATION: How to Keep Your Patients Safe, Keep Your Hard-Earned Money, and Stay Out of Court.,” *Innov. Clin. Neurosci.*, vol. 12, no. 7–8, pp. 34–8.

[81] B. M. Amal Al Aswad and S. K. Susan Baxter, “Issues Concerning the Adoption and Usage of Electronic Medical Records in Ministry of Health Hospitals in Saudi Arabia,” 2015.

[82] M. L. Despa, “Comparative study on software development methodologies,” 2014.

[83] “Software Development Methodologies.”

[84] “The Four Phases of Project Management.” [Online]. Available: https://hbr.org/2016/11/the-four-phases-of-project-management. [Accessed: 15-Jan-2019].

[85] M. L. Despa, “Comparative study on software development methodologies,” 2014.

[86] B. Boehm, “A spiral model of software development and enhancement,” *ACM SIGSOFT Softw. Eng. Notes*, vol. 11, no. 4, pp. 22–42, Aug. 1986.

[87] “TIPS & TRICS – DPIT WEB ENGINEERING.” [Online]. Available: https://dpitwebengineering.wordpress.com/category/tips-trics/. [Accessed: 15-Jan-2019].

[88] A. Saxena, “Web Based Custom Validation Using Framework in Java,” *Int. J. Comput. Sci. Trends Technol.*, vol. 3, 2013.

[89] A. Saxena, “Understanding Architecture and Framework of J2EE using Web Application.” .

[90] K. I. Satoto, R. R. Isnanto, R. Kridalukmana, and K. T. Martono, “Optimizing MySQL database system on information systems research, publications and community service,” in *2016 3rd International Conference on Information Technology, Computer, and Electrical Engineering (ICITACEE)*, 2016, pp. 1–5.

[91] E. Merah, “Design of ATL Rules for Transforming UML 2 Sequence Diagrams into Petri Nets,” *Int. J. Comput. Sci. …*, vol. 8, no. 1, pp. 1–21, 2014.

[92] K. McClements, R. O. Dendy, R. Bingham, J. G. Kirk, and L. O. Drury, *Acceleration of cosmic ray electrons by ion-excited waves at quasi perpendicular shocks*, vol. 291. 1997.

[93] E. Chen, “Master Thesis Study on Electronic Health Record and its Implementation,” 2012.

[94] R. B. Johnson and A. J. Onwuegbuzie, “Mixed Methods Research: A Research Paradigm Whose Time Has Come,” *Educ. Res.*, vol. 33, no. 7, pp. 14–26, Oct. 2004.

[95] R. B. Johnson, A. J. Onwuegbuzie, and L. A. Turner, “Toward a Definition of Mixed Methods Research,” *J. Mix. Methods Res.*, vol. 1, no. 2, pp. 112–133, Apr. 2007.

[96] A. Shorten and J. Smith, “Mixed methods research: expanding the evidence base.,” *Evid. Based. Nurs.*, vol. 20, no. 3, pp. 74–75, Jul. 2017.

[97] J. K. Kabukye, “User requirements for an electronic medical records system for a cancer hospital in Uganda,” *User Requir. an Oncol. EMR LMIC*, no. June, pp. 1–66, 2016.

[98] *Software and systems modeling.* Springer, 2002.

[99] D. Harel and S. Maoz, “Assert and negate revisited: Modal semantics for UML sequence diagrams,” *Softw. Syst. Model.*, vol. 7, no. 2, pp. 237–252, May 2008.

[100] “IBM Knowledge Center - Communication diagrams.” [Online]. Available: https://www.ibm.com/support/knowledgecenter/SS8PJ7_9.6.0/com.ibm.xtools.sequence.doc/topics/ccommndiag.html. [Accessed: 13-Jan-2019].

[101] A. Cavarra and J. Küster-Filipe, “Combining Sequence Diagrams and OCL for Liveness,” *Electron. Notes Theor. Comput. Sci.*, vol. 115, pp. 19–38, Jan. 2005.

[102] R. France, A. Evans, K. Lano, and B. Rumpe, “The UML as a formal modeling notation,” *Comput. Stand. Interfaces*, vol. 19, no. 7, pp. 325–334, Nov. 1998.

[103] R. France and B. Rumpe, Eds., *«UML»'99 — The Unified Modeling Language*, vol. 1723. Berlin, Heidelberg: Springer Berlin Heidelberg, 1999.

[104] X. Li and J. Lilius, "Timing Analysis of UML Sequence Diagrams," Springer, Berlin, Heidelberg, 1999, pp. 661–674.

Appendices

Figure 92 show coding structure of folders containing classes for the code.

Sample Code

The following code was used for the package which has classes dealing with database connections.

```
package dbconnection;
import java.io.PrintWriter;
import java.io.StringWriter;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.sql.Statement;
import java.util.Optional;
import java.util.Properties;

import com.jfoenix.controls.JFXTextArea;

import index.Index;
import javafx.application.Platform;
import javafx.geometry.Pos;
import javafx.scene.control.Alert;
import javafx.scene.control.ButtonBar.ButtonData;
import javafx.scene.control.ButtonType;
import validations.Validator;
public final class DBConnection {
   private static DBConnection connect=null;
   public static final String SERVER="jdbc:mysql://localhost:3306/";
   public static final String DB_NAME="hospital_db";
   public static final String user ="root";
   public static final String password ="";
```

```
public static final String DB_URL_LIVE="jdbc:mysql://db4free.net:3306/";
public static final String user_LIVE ="hospital_name";
public static final String password_LIVE ="H05P1TAL";
public static Connection conn = null;
private static Statement stmt = null;
private static boolean connectivity;
private static Properties properties;
private static final String MAX_POOL = "250";

private DBConnection(){
   createConnection();
}

// create properties
private static Properties getPropertiesOffline() {
   if (properties == null) {
      properties = new Properties();
      properties.setProperty("user", user);
      properties.setProperty("password", password);
      properties.setProperty("MaxPooledStatements", MAX_POOL);
   }
   return properties;
}

private static Properties getPropertiesOnline() {
   if (properties == null) {
      properties = new Properties();
      properties.setProperty("user", user_LIVE);
      properties.setProperty("password", password_LIVE);
      properties.setProperty("MaxPooledStatements", MAX_POOL);
   }
   return properties;
```

```
}

 public static DBConnection mysqlConnection(){
                if(connect==null){
                        connect=new DBConnection();
                } return connect;
        }

 public void disconnect() {
    if (conn != null) {
        try {
           conn.close();
           conn = null;
        } catch (SQLException e) {
                 Alert error = new Alert(Alert.AlertType.ERROR);
                        error.setTitle("SQLEXCEPTION");
                        error.setHeaderText(null);
                        StringWriter sw=new StringWriter();
                        PrintWriter pw = new PrintWriter(sw);
                        e.printStackTrace(pw);
                        JFXTextArea area = new JFXTextArea(sw.toString());
                        error.getDialogPane().setExpandableContent(area);
                        error.showAndWait();
        }
    }
 }
/*public Connection createConnection(){
          Alert conf = new Alert(Alert.AlertType.CONFIRMATION);
          Alert info = new Alert(Alert.AlertType.INFORMATION);
          if(conn==null){
 try{
    Class.forName("com.mysql.jdbc.Driver").newInstance();
```

```
                conf.setTitle("CONNECT");
                conf.setHeaderText(null);
                conf.setContentText("Connect to Database");
                ButtonType offline= new ButtonType("Offline");
                ButtonType online= new ButtonType("Online");
                ButtonType cancel= new ButtonType("Cancel");
                conf.getButtonTypes().setAll(offline,online,cancel);
                Optional<ButtonType> connect= conf.showAndWait();
                        info.setTitle("CONNECTION");
                        info.setHeaderText(null);
                if(connect.get()==offline) {
                        connectivity=false;
                        info.setContentText("Connection to localhost");
                        info.show();
                        conn=DriverManager.getConnection(SERVER+DB_NAME,getPropertie-
sOffline());
                }else if(connect.get()==online) {
                        connectivity=true;
                        info.setContentText("Connection to remote database... ");
                        info.show();
                        conn=DriverManager.getConnec-
tion(DB_URL_LIVE+DB_NAME,getPropertiesOnline());
                }else{
                        System.out.println("Exit");
                        System.exit(0);
                }
        info.close();
    }catch(Exception e){
            info.close();
            Alert error = new Alert(Alert.AlertType.ERROR);
                    error.setTitle("EXCEPTION");
                    error.setHeaderText(null);
```

```
                StringWriter sw=new StringWriter();
                PrintWriter pw = new PrintWriter(sw);
                e.printStackTrace(pw);
                JFXTextArea area = new JFXTextArea(sw.toString());
                error.getDialogPane().setExpandableContent(area);
                error.showAndWait();
   }
   }return conn;
  }
  */
  public Connection createConnection(){
          Alert info = new Alert(Alert.AlertType.INFORMATION);
          if(conn==null){
   try{
      Class.forName("com.mysql.jdbc.Driver").newInstance();
                        info.setTitle("CONNECTION");
                        info.setHeaderText(null);
                        info.setContentText("Connection to localhost");
                        info.show();
                        conn=DriverManager.getConnection(SERVER+DB_NAME,getPropertie-
sOffline());
                        info.close();
   }catch(Exception e){
         info.close();
         Alert error = new Alert(Alert.AlertType.ERROR);
                error.setTitle("EXCEPTION");
                error.setHeaderText(null);
                StringWriter sw=new StringWriter();
                PrintWriter pw = new PrintWriter(sw);
                e.printStackTrace(pw);
                JFXTextArea area = new JFXTextArea(sw.toString());
                error.getDialogPane().setExpandableContent(area);
```

```
                error.showAndWait();
  }
  }return conn;
}
  public ResultSet executeQuery(String select){
          ResultSet result;
        try{
                stmt = conn.createStatement();
                result=stmt.executeQuery(select);
        }catch(SQLException ex){
                 Alert error = new Alert(Alert.AlertType.ERROR);
                        error.setTitle("SQLEXCEPTION");
                        error.setHeaderText(null);
                        StringWriter sw=new StringWriter();
                        PrintWriter pw = new PrintWriter(sw);
                        ex.printStackTrace(pw);

                        JFXTextArea area = new JFXTextArea(sw.toString());
                        error.getDialogPane().setExpandableContent(area);
                        error.showAndWait();
                return null;
        }return result;
  }

  public boolean executeAction(String sqlQuery) {
                try {
                        stmt = conn.createStatement();
                        stmt.execute(sqlQuery);
                        return true;
                } catch (SQLException ex) {
                         Alert error = new Alert(Alert.AlertType.ERROR);
                         error.setTitle("SQLEXCEPTION");
```

```
                error.setHeaderText(null);
                StringWriter sw=new StringWriter();
                PrintWriter pw = new PrintWriter(sw);
                ex.printStackTrace(pw);
                JFXTextArea area = new JFXTextArea(sw.toString());
                error.getDialogPane().setExpandableContent(area);
                error.showAndWait();
                return false;
            }
        }
    public boolean isLogin(String user, String pass,String role) throws SQLException{
            String query = "SELECT * FROM users WHERE userid='"+user+"' AND pass-
word='"+pass+"';";
            ResultSet result = connect.executeQuery(query);
            try{
                if(result.next()){
                return true;
                }else{
                    return false;
                }
            }catch(Exception e){
                Alert error = new Alert(Alert.AlertType.ERROR);
                error.setTitle("EXCEPTION");
                error.setHeaderText(null);
                StringWriter sw=new StringWriter();
                PrintWriter pw = new PrintWriter(sw);
                e.printStackTrace(pw);

                JFXTextArea area = new JFXTextArea(sw.toString());
                error.getDialogPane().setExpandableContent(area);
                error.showAndWait();
                return false;
```

```
            } finally{
                result.close();
            }
        }

    public boolean isDBConnection(){
            try{
                return !conn.isClosed();
            }catch(SQLException e){
                Alert error = new Alert(Alert.AlertType.ERROR);
                error.setTitle("SQLEXCEPTION");
                error.setHeaderText(null);
                StringWriter sw=new StringWriter();
                PrintWriter pw = new PrintWriter(sw);
                e.printStackTrace(pw);

                JFXTextArea area = new JFXTextArea(sw.toString());
                error.getDialogPane().setExpandableContent(area);
                error.getDialogPane().setExpanded(true);
                error.showAndWait();
                return false;
            }
        }
}
```

The following code is used for registering a patient

```
package addpatient;
import java.awt.image.BufferedImage;
import java.io.ByteArrayInputStream;
import java.io.ByteArrayOutputStream;
import java.io.File;
```

```
import java.io.FileNotFoundException;
import java.io.FileOutputStream;
import java.io.IOException;
import java.io.InputStream;
import java.io.OutputStream;
import java.io.PrintWriter;
import java.io.StringWriter;
import java.net.URL;
import java.sql.Connection;
import java.sql.PreparedStatement;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.text.SimpleDateFormat;
import java.util.Date;
import java.util.Optional;
import java.util.ResourceBundle;
import java.util.logging.Level;
import java.util.logging.Logger;
import java.util.regex.Matcher;
import java.util.regex.Pattern;
import javax.imageio.ImageIO;
import com.jfoenix.controls.JFXComboBox;
import com.jfoenix.controls.JFXDatePicker;
import com.jfoenix.controls.JFXRadioButton;
import com.jfoenix.controls.JFXTabPane;
import com.jfoenix.controls.JFXTextArea;
import com.jfoenix.controls.JFXTextField;
import com.jfoenix.effects.JFXDepthManager;
import add_details.AddDetailsController;
import dbconnection.DBConnection;
import dbconnection.SystemUtils;
import dbconnection.Source;
```

```
import display_details.DisplayDetailsController.Referral;
import display_patient.DisplayPatientController;
import display_patient.DisplayPatientController.Patient;
import display_patient.DisplayPatientController.PatientInsurance;
import displaystaff.DisplayStaffController;
import javafx.collections.FXCollections;
import javafx.collections.ObservableList;
import javafx.event.ActionEvent;
import javafx.fxml.FXML;
import javafx.fxml.FXMLLoader;
import javafx.fxml.Initializable;
import javafx.geometry.Pos;
import javafx.scene.Parent;
import javafx.scene.Scene;
import javafx.scene.control.Alert;
import javafx.scene.control.Button;
import javafx.scene.control.ButtonType;
import javafx.scene.control.CheckBox;
import javafx.scene.control.ComboBox;
import javafx.scene.control.Label;
import javafx.scene.control.RadioButton;
import javafx.scene.control.Tab;
import javafx.scene.control.TextArea;
import javafx.scene.control.ToggleGroup;
import javafx.scene.image.Image;
import javafx.scene.image.ImageView;
import javafx.scene.layout.AnchorPane;
import javafx.stage.FileChooser;
import javafx.stage.Modality;
import javafx.stage.Stage;
import javafx.stage.StageStyle;
import main.Login;
```

```
import javafx.stage.FileChooser.ExtensionFilter;
import net.connectcode.Code128Auto;
import net.coobird.thumbnailator.Thumbnails;
import validations.Validator;
import webcam.CamCap;

public class RegisterPatientController implements Initializable{
   DBConnection connect;
        Alert error = new Alert(Alert.AlertType.ERROR);
        Alert info = new Alert(Alert.AlertType.INFORMATION);
        Alert conf = new Alert(Alert.AlertType.CONFIRMATION);
        @FXML private Tab tab0;
        @FXML private Tab tab1;
        @FXML private Tab tab2;
        @FXML private Tab tab3;
        @FXML private AnchorPane tabUpload;
        @FXML private AnchorPane anchorPane;
        @FXML private ToggleGroup GENDER;
        @FXML private CheckBox check;
        @FXML private ToggleGroup NATIONALITY;
        @FXML public Label lblGender;
   @FXML public JFXTextField txtOccupation;
   @FXML public JFXTextField txtIncome;
   @FXML public JFXTextField txtNextOfKinCont;
   @FXML public JFXTextField txtFname;
   @FXML private Button btnChooseFile;
   @FXML public Button btnAddPatientBiodata;
   @FXML public TextArea txtPrimaryAddress;
   @FXML public JFXTextField txtIDNumber;
   @FXML public JFXTextField txtPatientID;
   @FXML public JFXTextField txtPOB;
   @FXML public JFXTextField txtNextOfKin;
```

```
@FXML public JFXTextField txtLname;
@FXML public TextArea txtPostalAddress;
@FXML public JFXTextField txtNationality;
@FXML public JFXTextField txtMname;
@FXML private JFXDatePicker dpDOB;
@FXML public Label lblDate;
@FXML private Label lblDOB;
@FXML private Label lblCode;
@FXML private JFXRadioButton radBtnFemale;
@FXML private JFXRadioButton radBtnMale;
@FXML public JFXComboBox<String> comboBoxEmpStatus;
@FXML public JFXComboBox<String> comboBoxMaritalStatus;
@FXML public JFXComboBox<String> comboBoxPOB;
@FXML private JFXComboBox<String> comboBoxPOBOther;
@FXML public JFXComboBox<String> comboBoxReligion;
@FXML public JFXComboBox<String> comboBoxIDType;
@FXML private ImageView patientImage;
@FXML private ImageView patientIcon;
@FXML private ImageView btnImage;
@FXML private ImageView btnImage2;
@FXML private ImageView btnImage3;
@FXML private ImageView btnImage4;
@FXML private Label lbl1;
@FXML private Label lbl2;
@FXML private Label lbl3;
@FXML private Label lbl4;
@FXML private Label lbl5;
@FXML private Label lbl6;
@FXML private Label lbl7;
@FXML private Label lbl8;
@FXML private Label lbl9;
@FXML private Label lbl10;
```

```
@FXML private Label lbl11;
@FXML private Label lbl12;
@FXML private Label lbl13;
@FXML private Label lbl14;
@FXML private Label lbl15;
@FXML private Label lbl16;
@FXML private Label lbl17;
@FXML private Label lbl18;
@FXML private Label lbl19;
@FXML private Label lbl20;
@FXML private Label lbl21;
@FXML public Label lblInsuID;
@FXML public Label lblRefID;
@FXML private Label lblE;
@FXML private Label lblFile;
@FXML private ComboBox<String> comboEdit;;
@FXML public ComboBox<String> comboBoxCountry;
@FXML private RadioButton radBtnNonZambian;
@FXML private RadioButton radBtnZambian;
@FXML private JFXTabPane tpane;
@FXML public Button btnRefID;
@FXML public Button btnInsuID;
@FXML private Button btnCamera;
    private File selectedFile;
    private Referral referral;
    PatientInsurance insurance;
    private Image image;
    String filename=null;
    Date currentDate;
    CamCap camera;
    private byte[] patient_pic=null;
Boolean isEditableMode = Boolean.FALSE;
```

```
public String query,insuID,refID,patID,gend,mnam,fnam,lnam,nat,pob,dob,raddr,paddr,relig,incom,emp,nxtkn,nxtknMobil,IDTyp,natID,occ,maritl,yy,mm,dd,hr,min,sec;
private int active;
ObservableList<String> state=FXCollections.observableArrayList();
ObservableList<String> emp_stat=FXCollections.observableArrayList("N/A","Formal","Informal","Unemployed");
ObservableList<String> mar_stat=FXCollections.observableArrayList("Married","Single","Divorced","Widowed");
ObservableList<String> district=FXCollections.observableArrayList();
ObservableList<String> religion=FXCollections.observableArrayList("Christianity","Islam","Hinduism","Buddhism","Judaism","Other");
ObservableList<String> ref_id=FXCollections.observableArrayList();
ObservableList<String> id_typ=FXCollections.observableArrayList("VOTER'S CARD","D/LICENCE","PASSPORT","NRC","N/A");
ObservableList<String> id_insu=FXCollections.observableArrayList("INSURANCE ID");
ObservableList<String> id_ref=FXCollections.observableArrayList("REFERRAL ID");
ObservableList<String> insu_id=FXCollections.observableArrayList();
@Override
    public void initialize(URL location, ResourceBundle resources) {
    JFXDepthManager.setDepth(tpane, 1);
    connect = DBConnection.mysqlConnection();
    Source.setDateTime();
    txtPatientID.setText(Source.patientID);
    fillComboBoxCountry();
    setPatientIcon();
    btnAddPatientBiodata.setVisible(false);
    txtNationality.setVisible(false);
    //lblDate.setVisible(false);
    lblGender.setVisible(false);
    comboBoxCountry.setDisable(true);
    txtPOB.setDisable(true);
```

```
        comboBoxPOB.setDisable(true);
        comboBoxEmpStatus.setItems(emp_stat);
        comboBoxMaritalStatus.setItems(mar_stat);
        comboBoxReligion.setItems(religion);
        comboBoxIDType.setItems(id_typ);
        lblFile.setVisible(false);
        check.setDisable(true);
        handleCamera();
}

public void fillComboBoxCountry() {
        state.clear();
                DBConnection connect = DBConnection.mysqlConnection();
                query = "SELECT country FROM country_list;";
                        try {
                                ResultSet result = connect.executeQuery(query);
                                while(result.next()){
                                        state.add(result.getString("country"));
                                }
                        } catch (SQLException e) {
                                        error.setHeaderText(null);
                                        error.setContentText("Error:"+e.getLocalizedMessage());
                                        error.showAndWait();
                                        e.printStackTrace();
                        }
                        comboBoxCountry.getItems().setAll(state);
                }

public void fillComboBoxDistrict() {
        district.clear();
                DBConnection connect = DBConnection.mysqlConnection();
                query = "SELECT place_of_birth FROM district;";
```

```
                try {
                        ResultSet result = connect.executeQuery(query);
                        while(result.next()){
                                district.add(result.getString("place_of_birth"));
                        }
                } catch (SQLException e) {
                                error.setHeaderText(null);
                                error.setContentText("Error:"+e.getLocalizedMessage());
                                error.showAndWait();
                                e.printStackTrace();
                }
                comboBoxPOB.getItems().setAll(district);
        }

    @FXML
    void setReferalDetails(ActionEvent event) {
                patID=txtPatientID.getText();
                fnam=txtFname.getText();
                lnam=txtLname.getText();
                try {
                        FXMLLoader loader = new FXML-
Loader(getClass().getResource(Source.DETAILS));
                        Parent root = loader.load();
                        Scene scene = new Scene(root);
                        Stage stage = new Stage(StageStyle.DECORATED);
                        AddDetailsController ref = (AddDetailsController) loader.getController();
                        ref.setReferralDetails(referral);
                        scene.getStylesheets().add(getClass().getResource(Source.CSS_1).toEx-
ternalForm());
                        stage.setTitle("P"+patID+" referral details");
                        ref.tabRef.setText(fnam.toUpperCase()+"  "+lnam.toUpperCase()+"'S RE-
FERRAL DETAILS");
```

```
                    stage.setScene(scene);
                    stage.initModality(Modality.APPLICATION_MODAL);
                    stage.setResizable(false);
                    stage.show();
                    SystemUtils.setIcon(stage);
                    stage.setOnCloseRequest((e)->{
                        lblRefID.setVisible(false);
                        refID = ref.txtReferralID.getText();
                        lblRefID.setText(refID);
                        btnRefID.setText(refID);
                        btnRefID.setDisable(true);
                    });
        } catch (IOException ex) {
            error.setHeaderText(null);
                    error.setContentText("Error: "+ex.getLocalizedMessage());
                    error.showAndWait();
            }
    }

    @FXML
    void setInsuranceDetails(ActionEvent event) {
            patID=txtPatientID.getText();
            fnam=txtFname.getText();
            lnam=txtLname.getText();
            try {
                FXMLLoader loader = new FXMLLoader(getClass().getResource("/add_details/AddDetails.fxml"));
                Parent root = loader.load();
                Scene scene = new Scene(root);
                Stage stage = new Stage(StageStyle.DECORATED);
                AddDetailsController insu = (AddDetailsController) loader.getController();
```

```
                    insu.setInsuranceDetails(insurance);
                    scene.getStylesheets().add(getClass().getResource(Source.CSS_1).toEx-
ternalForm());
                    stage.setTitle("P"+patID+" insurance details");
                    insu.tabInsu.setText(fnam.toUpperCase()+"      "+lnam.toUpperCase()+"'S
INSURANCE DETAILS");
                    stage.setScene(scene);
                    stage.initModality(Modality.APPLICATION_MODAL);
                    stage.setResizable(false);
                    stage.show();
                    SystemUtils.setIcon(stage);
                    stage.setOnCloseRequest((e)->{
                            lblInsuID.setVisible(false);
                            insuID = insu.txtInsuranceID.getText();
                            lblInsuID.setText(insuID);
                            btnInsuID.setText(insuID);
                            btnInsuID.setDisable(true);
                    });
        } catch (IOException ex) {
                error.setHeaderText(null);
                    error.setContentText("Error: "+ex.getLocalizedMessage());
                    error.showAndWait();
                }
    }

    private boolean fieldsValidated(){
                Validator.f1 = Validator.isFieldEmpty(txtPatientID, lbl1, "*");
                Validator.f2 = Validator.isFieldEmpty(txtFname, lbl2, "*");
                Validator.f3 = Validator.isFieldEmpty(txtLname, lbl3, "*");
                Validator.f5 = Validator.isFieldEmpty(txtNationality, lbl5, "*");
                Validator.f6 = Validator.isFieldEmpty(txtPOB, lbl6, "*");
                Validator.f7 = Validator.isLabelEmpty(lblGender, lbl7, "*");
```

```
        Validator.f8 = Validator.isComboBoxNull(comboBoxMaritalStatus, lbl8, "*");
        Validator.f9 = Validator.isLabelEmpty(lblDate, lbl9, "*");
        Validator.f10 = Validator.isComboBoxNull(comboBoxIDType, lbl10, "*");
        Validator.f11 = Validator.isFieldEmpty(txtIDNumber, lbl11, "*");
        Validator.f12 = Validator.isComboBoxNull(comboBoxReligion, lbl12, "*");
        Validator.f13 = Validator.isComboBoxNull(comboBoxEmpStatus, lbl13, "*");
        Validator.f14 = Validator.isFieldEmpty(txtOccupation, lbl14, "*");
        Validator.f15 = Validator.isFieldEmpty(txtIncome, lbl15, "*");
        Validator.f16 = Validator.isAreaEmpty(txtPrimaryAddress, lbl16, "*");
        Validator.f17 = Validator.isAreaEmpty(txtPostalAddress, lbl17, "*");
        Validator.f18 = Validator.isFieldEmpty(txtNextOfKin, lbl18, "*");
        Validator.f19 = Validator.isFieldEmpty(txtNextOfKinCont, lbl19, "*");
        Validator.f20 = Validator.isLabelEmpty(lblRefID, lbl20, "*");
        Validator.f21 = Validator.isLabelEmpty(lblInsuID, lbl21, "*");
        if(txtPatientID.getText().isEmpty() ||
        comboBoxMaritalStatus.getValue()==null ||
txtOccupation.getText().isEmpty() ||
comboBoxReligion.getValue()==null ||
comboBoxEmpStatus.getValue()==null ||
txtIncome.getText().isEmpty() ||
lblGender.getText().isEmpty() ||
txtNextOfKinCont.getText().isEmpty() ||
txtFname.getText().isEmpty() ||
txtLname.getText().isEmpty() ||
txtPrimaryAddress.getText().isEmpty() ||
comboBoxIDType.getValue()==null ||
lblRefID.getText().isEmpty() ||
txtIDNumber.getText().isEmpty() ||
txtNextOfKin.getText().isEmpty() ||
txtPOB.getText().isEmpty() ||
txtPostalAddress.getText().isEmpty() ||
txtNationality.getText().isEmpty() ||
```

```
        lblInsuID.getText().isEmpty() ||
        lblDate.getText().isEmpty()){
                lblE.setText(Validator.errEmpty);
                return false;
                }else{
                        return true;
                }
        }

@FXML
public void registerUpdatePatient(ActionEvent event){
        if(fieldsValidated() &&  dataValidatedInteger() && dataValidatedNames()){
        if(isEditableMode) {
                                updatePatientDetails();
                }else {
                                registerPatient();
                        }
        }
}

private boolean dataValidatedNames(){
        nxtkn=txtNextOfKin.getText();
        nxtkn=nxtkn.replaceAll("\\s", "");
                txtNextOfKin.setText(nxtkn);

                Pattern ptn = Pattern.compile("[a-zA-Z]+");
                Matcher match = ptn.matcher(txtFname.getText());
                Matcher match2 = ptn.matcher(txtMname.getText());
                Matcher match3 = ptn.matcher(txtLname.getText());
                Matcher match4 = ptn.matcher(txtNextOfKin.getText());
                if((match.find() && match.group().equals(txtFname.getText())) &&
                                (match2.find() && match2.group().equals(txtMname.getText()) ||
```

```
                                        txtMname.getText().isEmpty()) && (match3.find()
&&
        match3.group().equals(txtLname.getText())) &&
                                                (match4.find()                    &&
match4.group().equals(txtNextOfKin.getText()))){
                        txtFname.getStyleClass().remove("error2");
                        txtMname.getStyleClass().remove("error2");
                        txtLname.getStyleClass().remove("error2");
                        txtNextOfKin.getStyleClass().remove("error2");
                        lbl2.setText("");
                        lbl3.setText("");
                        lbl4.setText("");
                        lbl18.setText("");
                        return true;
                }
                else{
                        lblE.setText(Validator.errAlpha);
                        txtFname.getStyleClass().add("error2");
                        txtMname.getStyleClass().add("error2");
                        txtLname.getStyleClass().add("error2");
                        txtNextOfKin.getStyleClass().add("error2");
                        Validator.f_2 = Validator.isFieldAlpha(txtFname, lbl2, "$");
                        Validator.f_3 = Validator.isFieldAlpha(txtLname, lbl3, "$");
                        Validator.f_4 = Validator.isFieldAlpha(txtMname, lbl4, "$");
                        Validator.f_5 = Validator.isFieldAlpha(txtNextOfKin, lbl18, "$");
                        return false;
                }
        }

   private boolean dataValidatedInteger(){
                Pattern ptn = Pattern.compile("[0-9]+");
```

```
                Pattern ptn2 = Pattern.compile("[0-9]+");
                Pattern ptn3 = Pattern.compile("[0-9]+");
                Matcher match = ptn.matcher(txtPatientID.getText());
                Matcher match2 = ptn2.matcher(txtIncome.getText());
                Matcher match3 = ptn3.matcher(txtNextOfKinCont.getText());
                if((match.find() && match.group().equals(txtPatientID.getText())) &&
(match2.find() && match2.group().equals(txtIncome.getText())) && (match3.find() &&
match3.group().equals(txtNextOfKinCont.getText()))){
                        txtPatientID.getStyleClass().remove("error2");
                        txtIncome.getStyleClass().remove("error2");
                        txtNextOfKinCont.getStyleClass().remove("error2");
                        lbl1.setText("");
                        lbl15.setText("");
                        lbl19.setText("");
                        return true;
                }       if(txtNextOfKinCont.getText().length()<10    ||    txtNextOfKin-
Cont.getText().length()>15){
                        txtNextOfKinCont.getStyleClass().add("error2");
                        lbl19.setText("#");
                        lblE.setText("invalid contact number #");
                        return false;
                }else{
                        lblE.setText(Validator.errNum);
                        Validator.f_1 = Validator.isFieldNumeric(txtPatientID, lbl1, "#");
                        Validator.f_5 = Validator.isFieldNumeric(txtIncome, lbl15, "#");
                        Validator.f_6 = Validator.isFieldNumeric(txtNextOfKinCont, lbl19, "#");

                        return false;
                }
        }

   public void registerPatient(){
```

```
patID=txtPatientID.getText();
        maritl=comboBoxMaritalStatus.getValue();
occ=txtOccupation.getText();
relig=comboBoxReligion.getValue();
emp=comboBoxEmpStatus.getValue();
incom=txtIncome.getText();
gend=lblGender.getText();
nxtknMobil=txtNextOfKinCont.getText();
fnam=txtFname.getText();
mnam=txtMname.getText();
lnam=txtLname.getText();
raddr=txtPrimaryAddress.getText();
IDTyp=comboBoxIDType.getValue();
refID=lblRefID.getText();
natID=txtIDNumber.getText();
nxtkn=txtNextOfKin.getText();
pob=txtPOB.getText();
paddr=txtPostalAddress.getText();
nat=txtNationality.getText();
insuID=lblInsuID.getText();
dob=lblDate.getText();
query = "INSERT INTO patient (patient_id, fname, othername, lname, nationality,
dob, gender, primary_address, postal_address, place_of_birth, religion, income, employmentsta-
tus, nextofkin, nextofkincontact, patientidtype, id_no, occupation, maritalstatus, insurance_id, re-
ferral_id,created_by, ppic) VALUES ('"+patID+"', '"+fnam+"', '"+mnam+"', '"+lnam+"', '"+nat+"',
'"+dob+"', '"+gend+"', '"+raddr+"', '"+paddr+"', '"+pob+"', '"+relig+"', '"+incom+"', '"+emp+"',
'"+nxtkn+"', '"+nxtknMobil+"', '"+IDTyp+"', '"+natID+"', '"+occ+"', '"+maritl+"', '"+insuID+"',
'"+refID+"','"+Login.user+"',?);";

try {
        Validator.f22 = Validator.isLabelEmpty(lblFile, lblE, Validator.err-
Pic );
```

```
                    PreparedStatement stmt = connect.createConnection().pre-
pareStatement(query);
                        stmt.setBytes(1, patient_pic);
                        btnChooseFile.getStyleClass().remove("error");
                        if(Validator.f22) {
                            System.out.println(query);
                        stmt.execute();
                        lblE.setText("");
                        Validator.showInformationNotification("SUCCESS", 2,
Pos.TOP_CENTER, "Patient registered");
                        clearFields();
                        }else {
                            btnChooseFile.getStyleClass().add("error");
                        }
                        } catch (SQLException e) {
                            error.setHeaderText(null);
                            error.setContentText("Exception:"+e.getMes-
sage());
                            error.showAndWait();
                    }
            }


  void handleCamera() {
        if(CamCap.webcam.isOpen()) {
            btnChooseFile.setText("Close Camera");
            btnCamera.setText("CAPTURE");
        }
  }

  @FXML
        public void openCamera(ActionEvent event){
```

```
        if(btnCamera.getText().contains("Camera")){
                camera = new CamCap();
                camera.setVisible(true);
                btnChooseFile.setText("Close Camera");
                btnCamera.setText("CAPTURE");

        }else if(btnCamera.getText().contains("CAPTURE")){
                CamCap.btnCaptureImage();
                btnCamera.setText("Open");
                btnChooseFile.setText("Browse file");
        } else if(btnCamera.getText().contains("Open")){
                CamCap.startCamera();
                btnCamera.setText("CAPTURE");
                btnChooseFile.setText("Close Camera");
        }
        camera.setDefaultCloseOperation(0);
        camera.btnStart.setVisible(false);
        camera.btnCapture.setVisible(false);
        }

@FXML
        public void chooseFile(ActionEvent event){
        if(btnChooseFile.getText().contains("Close Camera")) {
                btnCamera.setText("Camera");
                btnChooseFile.setText("Browse file");
                camera.setVisible(false);
                CamCap.closeCamera();

                }else {
                        browsePhoto();
                }
}
```

```
  void browsePhoto() {
        FileChooser fileChooser =new FileChooser();
            fileChooser.getExtensionFilters().addAll(
            new ExtensionFilter("Images","*.jpg","*.jpeg","*.png"));//reading images
            fileChooser.setInitialDirectory(new File(Source.CAPTURED_IMAGE));
            selectedFile=fileChooser.showOpenDialog(null);
            filename=selectedFile.getAbsolutePath();
            try {
                Image image = null;
                File file=new File(filename);
                BufferedImage bImage=ImageIO.read(file);
                BufferedImage thumbNail = Thumbnails.of(bImage).size(150, 150).as-
BufferedImage();
                ByteArrayOutputStream baos=new ByteArrayOutputStream();
                ImageIO.write(thumbNail, "png", baos);

                InputStream is= new ByteArrayInputStream(baos.toByteArray());

                ByteArrayOutputStream baos2 = new ByteArrayOutputStream();

                byte[] buff=new byte[1024];
                try {
                    for(int readNum;(readNum=is.read(buff))!=-1;) {
                        baos2.write(buff,0,readNum);
                        //System.out.println("Read "+readNum+"bytes");
                    }
                    image = new Image(selectedFile.getAbsolute-
File().toURI().toString(),
                            patientImage.getFitWidth(),patientImage.get-
FitHeight(),true,true);
                    patientImage.setImage(image);
```

```
                    patientImage.setPreserveRatio(true);
                }catch(IOException ex) {
                    Validator.showErrorNotification("FAILURE", 2, Pos.BOT-
TOM_RIGHT, ""+ex);
                }
                lblFile.setText("`");
                patient_pic=baos2.toByteArray();
            }catch(Exception e) {
                Validator.showErrorNotification("FAILURE", 2, Pos.BOTTOM_RIGHT,
""+e);
            }
    }

    //Generating a Barcode from patient ID
    public void generateBarCode() {
            Code128Auto code128=new Code128Auto();
        String barCode = code128.encode(txtPatientID.getText());
        lblCode.setText(barCode);
            }

     private void clearFields() {
            comboBoxCountry.setDisable(true);
            patientImage.setImage(null);
                comboBoxMaritalStatus.setValue(null);
            txtOccupation.setText("");
            comboBoxEmpStatus.setValue(null);
            txtIncome.setText("");
            lblGender.setText("");
            txtFname.setText("");
            txtMname.setText("");
            txtMname.setText("");
            lblDate.setText("");
```

```
txtPrimaryAddress.setText("");
comboBoxCountry.setValue("-NATIONALITY-");
txtLname.setText("");
txtPatientID.setText("");
comboBoxReligion.setValue(null);
txtNextOfKinCont.setText("");
comboBoxIDType.setValue(null);
txtIDNumber.setText("");
txtNextOfKin.setText("");
comboBoxPOB.setValue("");
txtPOB.setText("");
txtPostalAddress.setText("");
txtNationality.setText("");
btnInsuID.setText("REFERRAL ID");
btnRefID.setText("INSURANCE ID");
lblInsuID.setText("");
lblRefID.setText("");
radBtnMale.setSelected(false);
radBtnFemale.setSelected(false);
radBtnNonZambian.setSelected(false);
radBtnZambian.setSelected(false);
lblInsuID.setVisible(true);
lblRefID.setVisible(true);
lblE.setText("");
lblDOB.setVisible(true);
Source.setDateTime();
txtPatientID.setText(Source.patientID);
}

private void updatePatientDetails() {
        maritl=comboBoxMaritalStatus.getValue();
occ=txtOccupation.getText();
```

```
relig=comboBoxReligion.getValue();
emp=comboBoxEmpStatus.getValue();
incom=txtIncome.getText();
gend=lblGender.getText();
nxtknMobil=txtNextOfKinCont.getText();
fnam=txtFname.getText();
mnam=txtMname.getText();
raddr=txtPrimaryAddress.getText();
IDTyp=comboBoxIDType.getValue();
refID=btnRefID.getText();
natID=txtIDNumber.getText();
patID=txtPatientID.getText();
nxtkn=txtNextOfKin.getText();
pob=txtPOB.getText();
lnam=txtLname.getText();
paddr=txtPostalAddress.getText();
nat=txtNationality.getText();
insuID=btnInsuID.getText();
dob=lblDate.getText();
query = "UPDATE patient SET fname='"+fnam+"',other-
name='"+mnam+"',lname='"+lnam+"',nationality='"+nat+"',dob='"+dob+"',gen-
der='"+gend+"',primary_address='"+raddr+"',postal_ad-
dress='"+paddr+"',place_of_birth='"+pob+"',religion='"+relig+"',income='"+incom+"',employ-
mentstatus='"+emp+"',nextofkin='"+nxtkn+"',nextofkincontact='"+nxtknMobil+"',patien-
tidtype='"+IDTyp+"',id_no='"+natID+"',occupation='"+occ+"',maritalstatus='"+maritl+"',insur-
ance_id='"+insuID+"',referral_id='"+refID+"' WHERE patient_id='"+patID+"';";
        DBConnection connect=DBConnection.mysqlConnection();
        lblE.setText("");
        if(connect.executeAction(query)){
            Validator.showInformationNotification("SUCCESS", 2, Pos.TOP_CEN-
TER, "Patient details updated");
        }
```

```
	}

	@FXML
private void setDate(ActionEvent event){
	dpDOB = (JFXDatePicker) event.getSource();
	lblDate.setText(dpDOB.getValue().toString());
	lblDOB.setVisible(false);
	}

	@FXML
public void setGenderMale() {
		GENDER.setUserData('M');
	lblGender.setText("");
	lblGender.setText(""+(char)
			GENDER.getUserData());
}
	@FXML
public void setGenderFemale() {
	GENDER.setUserData('F');
	lblGender.setText("");
	lblGender.setText(""+(char)
			GENDER.getUserData());
}
	@FXML
public void setZambian(ActionEvent event) {
		if(radBtnZambian.isSelected()) {
			fillComboBoxDistrict();
			txtPOB.setText("");
			comboBoxCountry.setValue("-NATIONALITY-");
			txtNationality.setText("Zambia");
			comboBoxPOB.setValue("SELECT PLACE OF BIRTH");
			comboBoxCountry.setDisable(true);
```

```
                comboBoxPOB.setVisible(true);
                txtPOB.setVisible(false);
                comboBoxPOB.setDisable(false);
                txtPOB.setDisable(true);
        }
}
        @FXML
public void setNonZambian(ActionEvent event) {
        if(radBtnNonZambian.isSelected()) {
                txtPOB.setText("");
                txtNationality.setText("");
                comboBoxCountry.setDisable(false);
                comboBoxPOB.setVisible(false);
        txtPOB.setVisible(true);
        comboBoxPOB.setDisable(true);
        txtPOB.setDisable(false);
        }
}
        @FXML
public void selectCountry(ActionEvent event) {
        nat = comboBoxCountry.getValue();
        txtNationality.setText(nat);
        if(txtNationality.getText().contains("Zambia")) {
                txtPOB.setVisible(false);
                comboBoxPOB.setVisible(true);
        }else {
                txtPOB.setVisible(true);
                comboBoxPOB.setVisible(false);
        }
}

public void getGender() {
```

```
        gend = lblGender.getText();
                if(gend.contains("M")) {
                        radBtnMale.setSelected(true);
                }else {
                        radBtnFemale.setSelected(true);
                }
}
public void getNationality() {
        nat = txtNationality.getText();
                if(nat.contains("Zambia")) {
                        radBtnZambian.setSelected(true);
                        txtPOB.setVisible(false);
                        comboBoxPOB.setDisable(false);
                        comboBoxCountry.setDisable(true);
                        comboBoxCountry.setValue("-NATIONALITY-");
                }else {
                        radBtnNonZambian.setSelected(true);
                        txtPOB.setVisible(true);
                        txtPOB.setDisable(false);
                        comboBoxPOB.setVisible(false);
                        comboBoxCountry.setDisable(false);
                }
}

        @FXML
public void selectPOB(ActionEvent event) {
                pob = comboBoxPOB.getValue();
                txtPOB.setText(pob);
                if(pob=="Other") {
                        comboBoxPOB.setVisible(false);
                        txtPOB.setText("");
                        txtPOB.setVisible(true);
```

```
                txtPOB.setDisable(false);
                txtPOB.setPromptText("PLEASE SPECIFY PLACE OF BIRTH");
        }
}
    @FXML
public void setIDNo(ActionEvent event) {
        natID = comboBoxIDType.getValue();
        if(natID=="N/A") {
                txtIDNumber.setText(natID);
                txtIDNumber.setEditable(false);
        } else {
                txtIDNumber.setText("");
                txtIDNumber.setEditable(true);
        }
}

    @FXML
public void setEmployment(ActionEvent event) {
        emp = comboBoxEmpStatus.getValue();
        if(emp=="N/A") {
                txtOccupation.setText(emp);
                txtOccupation.setEditable(false);
                txtIncome.setText("000");
        } else {
                txtOccupation.setText("");
                txtOccupation.setEditable(true);
                txtIncome.setText("");
        }
}

public void setPatientImage() {
    image=new Image("/images/pat.png");
```

```
            patientIcon.setImage(image);
}
public void setPatientIcon() {
      setPatientImage();
      image=new Image("/images/ppic.png");
            patientImage.setImage(image);
}

      public void updatePatientDetails(String patient) {
            btnInsuID.setDisable(true);
            btnRefID.setDisable(true);
            tab3.setDisable(true);
            tab3.setText("");
            lblDOB.setVisible(false);
      lblInsuID.setVisible(false);
      lblRefID.setVisible(false);
      patientIcon.setImage(DisplayPatientController.image);
      isEditableMode=Boolean.TRUE;
      }

      public void updatePatientDetails(Patient patient) {
            //btnImage.setVisible(false);
            btnInsuID.setDisable(true);
            btnRefID.setDisable(true);
            tab3.setDisable(true);
            tab3.setText("");
            lblDOB.setVisible(false);
            txtNationality.setText(patient.getNat());
            getNationality();
            comboBoxPOB.setValue(patient.getPob());
      txtPOB.setText(patient.getPob());
            comboBoxCountry.setDisable(false);
```

```
            btnChooseFile.setVisible(false);
            btnAddPatientBiodata.setVisible(true);
            btnAddPatientBiodata.setText("Done");
            txtPatientID.setEditable(false);
            comboBoxMaritalStatus.setValue(patient.getMaritl());
txtOccupation.setText(patient.getOcc());
comboBoxEmpStatus.setValue(patient.getEmp());
txtIncome.setText(patient.getIncom());
lblGender.setText(patient.getGend());
txtFname.setText(patient.getFnam());
txtMname.setText(patient.getMnam());
txtMname.setText(patient.getMnam());
lblDate.setText(patient.getDob());
txtPrimaryAddress.setText(patient.getRaddr());
comboBoxCountry.setValue(patient.getNat());
dpDOB.setPromptText(patient.getDob());
txtLname.setText(patient.getLnam());
txtPatientID.setText(patient.getID());
comboBoxReligion.setValue(patient.getRelig());
txtNextOfKinCont.setText(patient.getNxtknMobil());
comboBoxIDType.setValue(patient.getIDTyp());
txtIDNumber.setText(patient.getNatID());
txtNextOfKin.setText(patient.getNxtkn());
txtPostalAddress.setText(patient.getPaddr());
btnInsuID.setText(patient.getInsuID());
btnRefID.setText(patient.getRefID());
lblInsuID.setText(patient.getInsuID());
lblRefID.setText(patient.getRefID());
lblInsuID.setVisible(false);
lblRefID.setVisible(false);
getGender();
checkForActivity(patient.getID());
```

```
patientIcon.setImage(DisplayPatientController.image);
isEditableMode=Boolean.TRUE;
}

public void viewPatientDetails(Patient patient) {
        check.setDisable(true);
        btnInsuID.setVisible(false);
        btnRefID.setVisible(false);
        tab3.setDisable(true);
        tab3.setText("");
        lblDOB.setVisible(false);
radBtnMale.setDisable(true);
radBtnFemale.setDisable(true);
radBtnZambian.setDisable(true);
radBtnNonZambian.setDisable(true);
comboBoxPOB.setDisable(true);
        comboBoxCountry.setDisable(true);
        comboBoxMaritalStatus.setDisable(true);
txtOccupation.setEditable(false);
comboBoxEmpStatus.setDisable(true);
txtIncome.setEditable(false);
lblGender.setText(patient.getGend());
txtFname.setEditable(false);
txtMname.setEditable(false);
txtMname.setEditable(false);
txtPrimaryAddress.setEditable(false);
dpDOB.setDisable(true);
txtLname.setEditable(false);
comboBoxReligion.setDisable(true);
txtNextOfKinCont.setEditable(false);
comboBoxIDType.setDisable(true);
txtIDNumber.setEditable(false);
```

```
txtNextOfKin.setEditable(false);
comboBoxPOB.setDisable(true);
txtPostalAddress.setEditable(false);
        comboBoxPOB.setVisible(true);
        txtPatientID.setEditable(false);
        comboBoxMaritalStatus.setValue(patient.getMaritl());
txtOccupation.setText(patient.getOcc());
comboBoxEmpStatus.setValue(patient.getEmp());
txtIncome.setText(patient.getIncom());
lblGender.setText(patient.getGend());
txtFname.setText(patient.getFnam());
txtMname.setText(patient.getMnam());
txtMname.setText(patient.getMnam());
lblDate.setText(patient.getDob());
txtPrimaryAddress.setText("RESIDENTIAL ADDRESS: \n"+patient.getRaddr());
comboBoxCountry.setValue(patient.getNat());
dpDOB.setPromptText(patient.getDob());
txtLname.setText(patient.getLnam());
txtPatientID.setText(patient.getID());
comboBoxReligion.setValue(patient.getRelig());
txtNextOfKinCont.setText(patient.getNxtknMobil());
comboBoxIDType.setValue(patient.getIDTyp());
txtIDNumber.setText(patient.getNatID());
txtNextOfKin.setText(patient.getNxtkn());
comboBoxPOB.setValue(patient.getPob());
txtPOB.setText(patient.getPob());
txtPostalAddress.setText("POSTAL ADDRESS: \n"+patient.getPaddr());
txtNationality.setText(patient.getNat());
btnInsuID.setText(patient.getInsuID());
btnRefID.setText(patient.getRefID());
lblInsuID.setText(patient.getInsuID());
lblRefID.setText(patient.getRefID());
```

```
lblInsuID.setVisible(false);
lblRefID.setVisible(false);
getGender();
getNationality();
checkForActivity(patient.getID());
patientIcon.setImage(DisplayPatientController.image);
txtPOB.setDisable(true);
comboBoxCountry.setDisable(true);
btnAddPatientBiodata.setVisible(true);
btnAddPatientBiodata.setText("Close");
btnAddPatientBiodata.setOnAction(e->{
        closeStage();
});
}

public void updatePatientPhoto(Patient patient){
        tab0.setText("");
        tab1.setText("");
        tab2.setText("");
        tab0.setDisable(true);
        tab1.setDisable(true);
        tab2.setDisable(true);
        btnInsuID.setVisible(false);
        btnRefID.setVisible(false);
        tab3.setText("UPDATE PHOTO");
        lblDOB.setVisible(false);
radBtnMale.setVisible(false);
radBtnFemale.setVisible(false);
radBtnZambian.setVisible(false);
radBtnNonZambian.setVisible(false);
comboBoxPOB.setVisible(false);
        comboBoxCountry.setVisible(false);
```

```
        comboBoxMaritalStatus.setVisible(false);
txtOccupation.setEditable(false);
comboBoxEmpStatus.setDisable(true);
txtIncome.setEditable(false);
lblGender.setText(patient.getGend());
txtFname.setVisible(false);
txtMname.setVisible(false);
txtPrimaryAddress.setEditable(false);
dpDOB.setDisable(true);
txtLname.setVisible(false);
comboBoxReligion.setDisable(true);
txtNextOfKinCont.setEditable(false);
comboBoxIDType.setDisable(true);
txtIDNumber.setEditable(false);
txtNextOfKin.setEditable(false);
txtPostalAddress.setEditable(false);
        txtPatientID.setVisible(false);
        comboBoxMaritalStatus.setValue(patient.getMaritl());
txtOccupation.setText(patient.getOcc());
comboBoxEmpStatus.setValue(patient.getEmp());
txtIncome.setText(patient.getIncom());
lblGender.setText(patient.getGend());
txtFname.setText(patient.getFnam());
txtMname.setText(patient.getMnam());
lblDate.setText(patient.getDob());
txtPrimaryAddress.setText("RESIDENTIAL ADDRESS: \n"+patient.getRaddr());
comboBoxCountry.setValue(patient.getNat());
dpDOB.setPromptText(patient.getDob());
txtLname.setText(patient.getLnam());
txtPatientID.setText(patient.getID());
comboBoxReligion.setValue(patient.getRelig());
txtNextOfKinCont.setText(patient.getNxtknMobil());
```

```
        comboBoxIDType.setValue(patient.getIDTyp());
        txtIDNumber.setText(patient.getNatID());
        txtNextOfKin.setText(patient.getNxtkn());
        comboBoxPOB.setValue(patient.getPob());
        txtPOB.setText(patient.getPob());
        txtPostalAddress.setText("POSTAL ADDRESS: \n"+patient.getPaddr());
        txtNationality.setText(patient.getNat());
        btnInsuID.setText(patient.getInsuID());
        btnRefID.setText(patient.getRefID());
        lblInsuID.setText(patient.getInsuID());
        lblRefID.setText(patient.getRefID());
        lblInsuID.setVisible(false);
        lblRefID.setVisible(false);
        getGender();
        getNationality();
        checkForActivity(patient.getID());
        patientIcon.setImage(DisplayPatientController.image);
        txtPOB.setDisable(true);
        comboBoxCountry.setDisable(true);
        btnAddPatientBiodata.setVisible(true);
        btnAddPatientBiodata.setText("Update");
        btnAddPatientBiodata.setOnAction(e->{
                updatePatientPhoto(patient.getID());
        });
        }

        @FXML
    void patientActivate(ActionEvent event) {
                patID=txtPatientID.getText();
        if(active==0) {
                conf.setTitle("ACTIVATE");
                        conf.setHeaderText(null);
```

```
                conf.setContentText("Activate patient No. "+patID+"?");
                Optional<ButtonType> action= conf.showAndWait();
                if(action.get()==ButtonType.OK) {
            activatePaitient(patID);
                }else {
                check.setSelected(false);
            }
        }else if(active==1) {
            conf.setTitle("DEACTIVATE");
                conf.setHeaderText(null);
                conf.setContentText("Deactivate patient No. "+patID+"?");
                Optional<ButtonType> action= conf.showAndWait();
                if(action.get()==ButtonType.OK) {
            deactivatePaitient(patID);
            }else {
                check.setSelected(true);
            }
        }else {
        }
    }

    void activatePaitient(String ID) {
        query = "UPDATE patient SET active = 1 WHERE patient_id= '"+ID+"';";
            if(connect.executeAction(query)) {
            Validator.showInformationNotification("SUCCESS", 2, Pos.CENTER, "patient
activated");
            }
        }
    void deactivatePaitient(String ID) {
        query = "UPDATE patient SET active = 0 WHERE patient_id= '"+ID+"';";
            if(connect.executeAction(query)) {
```

```
                Validator.showInformationNotification("SUCCESS", 2, Pos.CENTER, "patient deactivated");
                }
    }

        public void checkForActivity(String ID){
                try{
                        query = "SELECT active FROM patient WHERE patient_id='"+ID+"';";
                        ResultSet result = connect.executeQuery(query);
                        while(result.next()){
                                active=result.getInt("active");
                        }
                        if(active == 1){
                                check.setSelected(true);
                        }else if(active == 0){
                                check.setSelected(false);
                        }else {
                        }
                        }catch(Exception ex){
                                Validator.showErrorNotification("FAILURE", 4, Pos.BOT-TOM_RIGHT, "Exception: "+ex);
                        }
        }

        public void updatePatientPhoto(String ID){
                try{
                        query = "UPDATE patien SET ppic = ? WHERE patient_id= '"+ID+"';";
                                PreparedStatement stmt = connect.createConnection().pre-pareStatement(query);
                                stmt.setBytes(1, patient_pic);
                                if(lblFile.getText().isEmpty()){
                                        lblE.setText(Validator.errPic);
```

```
                    }else{
                    stmt.execute();
                    lblE.setText("");
                    Validator.showInformationNotification("SUCCESS", 2, Pos.CEN-
TER, "patient photo updated");
                    stmt.close();
                    }
            }catch(SQLException ex){
                Alert error = new Alert(Alert.AlertType.ERROR);
                 error.setTitle("SQLEXCEPTION");
                 error.setHeaderText(null);
                 StringWriter sw=new StringWriter();
                 PrintWriter pw = new PrintWriter(sw);
                 ex.printStackTrace(pw);

                 JFXTextArea area = new JFXTextArea(sw.toString());
                 error.getDialogPane().setExpandableContent(area);
                 error.showAndWait();
            }
}

    private void closeStage() {
        ((Stage)btnAddPatientBiodata.getScene().getWindow()).close();
    }

    @FXML
    public void setTab0() {

  }
    @FXML
  public void setTab1() {
        btnInsuID.setDisable(false);
```

```
            btnRefID.setDisable(false);
}

    @FXML
public void setTab2() {

}

    @FXML
public void setTab3() {
            btnAddPatientBiodata.setVisible(true);
            btnInsuID.setDisable(false);
            btnRefID.setDisable(false);
            generateBarCode();
}
```

APPENDIX

Appendix B: Questionnaire

QUESTIONNAIRE FOR HEALTH CARE PRACTITIONER

My name is Danny Leza. I am a Postgraduate Student at the University of Zambia, carrying out a study on the Implementation of Snomed CT enabled Electronic Health Record system. The questionnaire is a survey whose goal is to understand the system(s) being used in the surgery department at the University Teaching Hospital in Lusaka Zambia. I kindly request you to spare a few minutes and answer the attached questionnaire. The information provided will be used for academic purposes only and will be treated with utmost confidentiality. Please do not write your name anywhere on the questionnaire. I would appreciate your voluntary participation in completing the questionnaire.

Thank you.

(Please read the instructions carefully before every question and provide your response appropriately.)

SECTION ONE: GENERAL INFORMATION

(Please tick (√) the appropriate answer.)

1. Your Gender? Male ( ) Female ( )

2. Your Age group?

Under 29 years [ ] 30-39 years [ ]

40-49 years [ ] 50 years or over [ ]

3. What is your highest level of education?

Certificate [ ] Diploma [ ]

Graduate [ ] Post Graduate [ ]

4. What is your Job title?

Nurse [ ] Doctor [ ]

Surgeon [ ] anesthetist[ ]

Record Clerk [ ] Other [ ] Specify……

5. For how long have you been working in this facility?

a) Less than 1 year [ ]

b) 1 - 5 years [ ]

c) 6 – 10 years [ ]

d) More than 10 years [ ]

SECTION TWO: Infrastructure

6. Is there a computer(s) in your department?

Yes [1] ☐

No ☐

[2]

If your answer is No in the above question, skip to SECTION THREE, otherwise answer questions below.

Please point out the extent to which you agree with each of the following statements. Please tick
(√) the appropriate answer.

Use the scale of: 1= strongly disagree, 2= disagree, 3= neutral 4= agree and 5= strongly agree

		1	2	3	4	5
7.	I have full access to a computer every day					
8.	I have full access to a printer whenever I need					

9. Are computers in this department linked via local area networks?

Yes [1] ☐

No [2] ☐

10. Computers in this department have access to internet.

Yes [1] ☐

No [2] ☐

11.Choose what best describes the level of electronic medical record system in your department? (Tick only one box)

a) [ ] Management of health records in this department is fully electronic.

b) [ ] Management of health records in this department is hybrid (partially electronic and partially paper-based.

c) [ ] We do not have electronic medical records in this department.

SECTION THREE: Scheduling of Patient in the Operating Room.

Rate the following in terms of scheduling of patients. Please tick (√) the appropriate answer.

Use the scale of: 1 = strongly disagree, 2 = disagree, 3 = uncertain, 4 = agree, 5 = strongly agree

		1	2	3	4	5
12.	The Schedule of patients to be operated on is known in advance, except for emergency cases					
13.	Information regarding a particular patient before surgery is readily available during operation procedures					
14.	All supplies necessary for operation procedure are known in advance via existing system and are made available before commencement of operation procedure					
15.	The current system makes it easy to produce reports of each surgical procedure undertaken					
16.	The current system is capable of optimal scheduling and utilization of operation both, doctor schedule and resource availability					

SECTION FOUR: Privacy and Security.
Rate the following in terms of privacy and security of patient's medical data. Please tick (√) the appropriate answer.

Use the scale of: 1 = strongly disagree, 2 = disagree, 3 = uncertain, 4 = agree, 5 = strongly agree

		1	2	3	4	5
17.	The current system make it very difficult to prevent patients and their bedsiders from reading medical diagnosis					
18.	The current patient record system make it very difficult for patients to walk out of the hospital premises with a copy or actually record of a patient					
19.	Patient Records seldom go missing					
20.	Patient records are securely protected from theft					
21.	Patient records are inaccessible by unauthorised people					
22	Patient are kept in a fireproof location					

SECTION FIVE: Reports.

Rate the following in terms of Reports. Please tick (√) the appropriate answer.

Use the scale of: 1 = strongly disagree, 2 = disagree, 3 = uncertain, 4 = agree,

5 = strongly agree

		1	2	3	4	5
23	The current System is very difficult to generate reports					
24	The Current system produces accurate report					
25	It is very easy to produce statistics with the current system					
26	The Produced report can easily be used in future					
27	The produced reports are only accessible to authorise personnel					
28	The Management tire of the Hospital has access to report as when needed					
29	The current system allow for reports to be processed within ten minutes					

SECTION SIX: Level of Satisfaction

Rate the following in terms of level of Satisfaction. Please tick (√) the appropriate answer.

Use the scale of: 1 = very dissatisfied, 2 = dissatisfied, 3 = uncertain, 4 = satisfied, 5 = very satisfied

		1	2	3	4	5
30	The flow of patients and work					
31	The system of record keeping					
32	The safety of patient records					

33	The semantic standards being used for diagnosis					
34	Level of efficiency in report production					