Development of a Health Management Information System Using Agile Software-Engineering Methods

Amir Kror Shahidzay

Thesis presented in fulfilment
of the requirements for the degree of
Master of Science
at the University of the Western Cape

Supervisor: Michael Norman

December 2013

UNIVERSITY of the
WESTERN CAPE

Declaration

I, Amir Kror Shahidzay, declare that this thesis "*Development of a Health Management Information System Using Agile Software-Engineering Methods*" is my own work, that it has not been submitted before for any degree or assessment at any other university, and that all the sources I have used or quoted have been indicated and acknowledged by means of complete references.

Signature:  Date: 28th February 2014.

Amir Kror Shahidzay.

UNIVERSITY of the
WESTERN CAPE

Abstract

The purpose of this thesis is to discuss the development of a web application from scratch. It serves to build a *health management information system* from basic principles and covers all the software engineering activities starting from the gathering of requirements, evaluating these and eventually implementing a health management information system by applying several iterations of the Agile-extreme-programming-software-engineering approach to develop a Health Management System for the Kabul University Poly-clinic located at Kabul University campus in order to computerize clerical activities at the hospital.

Questionnaires were used to uncover the clerical problems experienced by the hospital staff. Attempts to address these problems by designing and implementing software and refine the software after some iterations of feedback-redesign-and-implementation following the guidelines of *Agile extreme programming*.

The previous Health management systems at the hospital were paper based. The new computerized system has eased the burdens of tracking the files of patients at the hospital, leading to easier and more efficient access to information by the health-care professionals at the hospital. An assessment of the impact this has had on the medical and clerical staff and the smoother administration of the hospital by repeated user acceptance testing by means of questionnaires confirms the success of the project.

UNIVERSITY of the
WESTERN CAPE

Key words

Agile extreme programming

Design

Health management information system

Participatory design

Unified modeling language

Interface

Patient management

Dari

Use case

User Story

UNIVERSITY of the
WESTERN CAPE

Acknowledgments

This thesis is a compilation of the efforts of many people that helped and me through the years. I would first like to thank my supervisor Mr Michael Norman and Dr Jan Plane for supporting and encouraging me during my Masters study. Without our weekly meetings, this work would not have been possible. Without their help I would certainly not be where I am today.

I would also like to thank USAID for their unwavering financial assistance without which our efforts would have been impossible.

Contents

List of Tables

List of Figures

Glossary

AUP	Agile unified process
Dari	An official language of Afghanistan
DHIS	District Health Information System
DSDM	Dynamic System Development Method
EMR	Electronic Medical Records
FDD	Feature Driven Design
HEW	Health Extension Worker
HISP	Health Information System Program
HIV	Human Immunodeficiency Virus
HMIS	Health Management Information System
ICTPH	Indira Kranthi Patham Centre for Technologies in Public Health
IKP	Indira Kranthi Patham
IPD	I
Java	Java programming language
LSD	Lean Software Development
MIS	Management Information System
OO	Object Oriented
RDBMS	Relational Database management system
SSL	Secure Socket Layer
UML	Unified Modeling Language

UP	Unified Process
USAID	United States Agency for International Development.
XP	Extreme Programming

Chapter 1

Introduction

The goal of this thesis is to discuss the entire development of an enterprise application from scratch starting with the elicitation of requirements and gathering of information and evaluating the needs that eventually lead to the final implementation of a *health management information system* (HMIS) using Agile *extreme programming* (XP).

The thesis discusses the conception of a web application for the Kabul University Poly-clinic[1] located on the Kabul University campus.The Poly-clinic treats patients without any remuneration. Since the clinic is very well equipped it has a surfeit of clients such as the teaching staff, other employees and students of the university visiting the clinic for treatment.

1.1 The old manual patient management system

The nine doctors and nine nurses working in this clinic are funded by the Afghanistan Health Ministry and the Higher Education Ministry. The doctors and nursing staff come from different medical universities of Afghanistan.

In the paper-based system each new patient had to be registered first using forms which were then manually recorded in books. This was tedious and thus the necessary paper work led to long delays and waiting time for patients before getting a turn to be attended to medically. Repeat visits by a patient had to be preceded by a search for the patient's paper work which took an inordinate amount of time. In the paper-based system the patient records were stored in the archive and it took some time to find the patient's records. At times the patient's records were lost and all the data had to be recaptured.

Initially we had no idea of the system requirements, but were led to believe that the management of the poly-clinic felt that there was a dire need for

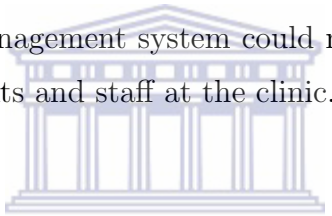
[1]Its full name is the "Kabul University Poly-clinic and Organization for Health Affairs of Higher Education in Afghanistan".

modernizing the administration of the patient health records at the poly-clinic. So it was decided to develop an application to overcome these administrative problems at the clinic by providing a computer-based patient-management system that centralizes the patient records and information of the clinic and enables access to the system from terminals in different parts of the clinic.

At the outset, we had to understand the exact requirements of such a system in order to be able to take steps toward the system implementation.

Information was gathered from all the stakeholders by means of interviews discussed in Appendix A. The interviews were done during the initial requirements gathering or so-called requirements engineering. The information that was extracted from the customer interviews, forms and discussions that were captured during meetings and sessions informed the developer of the requirements.

A simple computer-based management system could manage the paper work and save time for both patients and staff at the clinic.



1.2 Current challenges

The Poly-clinic employees had to process and handle many tasks and deal with issues manually under time pressure. The manual performance of daily routines could be quite time consuming and this often caused backlogs of activities and administrative tasks that could not be completed timeously and led to long waiting times for patients.

1.2.1 Instability and inconsistency

Instability and inconsistency was one of the main problems in the Poly-clinic, as they had no stable software management system available. Staff needed to perform all activities manually in hand-written documents stored inside specific folders and in bound record books.

Unavailability of previous patient information, albeit historical patient information, was another kind of problem. This historical information was stored in an archive room with thousands of filed folders that often made it virtually impossible to trace the details of particular patient. On visiting the Poly-clinic archive room, it was obvious that folders and documents could easily be lost or in any case were difficult to find in the warehouse-like archive

room. It was not always easy to determine that a folder was still on the desk of a doctor or in the pharmacy or elsewhere, if it could not be found in the archive.

Since it was so difficult to trace the folder of a patient, the clerks, instead of searching for the old folder, simply opened a new file for the patient when a patient returned for follow-up treatment—this led to redundant data being generated.

1.2.2 Poly-clinic problems

To solve the above mentioned problems a number of high-priority actions needed to be addressed, and solved by our application.

- Data redundancy was one of the main problems, where multiple copies of data for one patient were stored in separate folders.
- Since a patients records were written up in books, searching, looking up data, or updating data for a particular patient could be spread over several different books and folders was thus an almost insurmountable task.
- There was no stable software to keep the patients information, thus clearly patient data was often lost.
- When a patient lost a prescription for medicine, it was difficult or sometimes impossible to determine what medication had already been prescribed.

In summary, information was not readily available for repeat patients who visited the clinic for follow-up treatment.

1.3 Software system responsibilities

The HMIS application was tackled. It consists of Use cases, which we discuss in detail in Chapter 3, such as *patient management*, *employee management* and *user management*. *Patient management* is responsible for taking care of all patient treatments and information. *Employee management* handles the administrative activities of the poly-clinic where employees acquire and store

information, and *user management* copes with the handling of HMIS user privileges.

1.4 What is a health information system?

A *health information system* is "a set of components and procedures organized with the objective of generating information which will improve health care management decisions at all levels of the health system" [Lippeveld et al., 2000]. A health information system records and stores health data—input—for later retrieval and processing for decision-making—output. Decision-making mostly includes managerial aspects such as planning, organizing and control of health care facilities.

1.5 Design of a health information system

Health workers needed to learn computer basics, assisted by developers. Given the lack of computer skills among the health workers, the first task was to train them in computer basics as follows:

- Computer hardware basics, such as using the keyboard, mouse, monitor, and printer.
- The basics of using an operating system, for example, using Microsoft Windows, drop-down menus, command buttons, manipulating the mouse and with single and double clicks, printing a document, backing up data, and so forth.
- Using a word processor such as Microsoft Word for capturing a document and how to save, close, open or print it.
- Applying a spreadsheet like Microsoft Excel for simple tasks such as manipulating data and representing data graphically.
- Microsoft Access, for example how to define a field and set its properties, create tables and how to create a simple database[Kimaro and Titlestad, 2005]

The reason for letting users participate in the system design is to ensure that the system works as they desire. This participation from users using the XP

software engineering approach is important for good design and acceptance by users. There are significant attempts and literature based on European and western society but very little on implementing systems using XP in developing countries.

We now focus on a statement of the research problem.

1.6 The research problem / hypothesis

The problem of this study is to answer the following questions:

1. Does the HMIS fulfill the current requirements of the Poly-clinic?
2. Which procedures are needed to implement the HMIS application?
3. What are the problems after system implementation?
4. How should we implement a health management information system to be efficient using an XP Agile software engineering approach?
5. Does the HMIS fulfill the computing needs of the Poly-clinic?

The following research methods: interviews, observation, surveys, participate design, prototypes, focus groups etc. will be used for developing and designing the HMIS:

- Data collection and analysis using interviews and questionnaire.
- Test the prototype.
- Design a prototype in English.[2]
- Improve prototype will be based on user feedback.

For collecting and analyzing the data we first interviewed doctors, nurses and the administrative staff at the clinic. The patient record can be considered as an important source for the data. The patient record includes the *date*, *patient card number*, *name*, *job*, *age*, *diagnosis*, and *prescribed medicine*, *type of illness/complaint*. In this phase the correctness of data is important.

We used the MySQL *database management system* (DBMS) to develop the system. It is popular for web-based applications and implements the

[2]We needed to use English to enable our supervisors to give us feedback.

database component. A relational database approach is used for recording and manipulating the data. Distributing the database over the local network and the Internet enables us to address scalability and accessibility issues. This system has been designed so that it can be ported to other hospitals in Afghanistan.

1.7 Conclusion

In this project we design and implement a HMIS which will store health Information in such a way that, the users, i.e., the medical and administrative staff, will be able to easily use the system. The HMIS will record, store, retrieve and process health data for decision-making.

The thesis has five chapters. This first chapter provides an introduction to the thesis. The second chapter discusses the literature review. Chapter 3 discusses the design and methodology that have been used in the HMIS application and describes the technologies and the reasons why they have been chosen. The fourth chapter discusses the acceptance of the application by the users and summarizes its impact on their productivity. Chapter 5 discusses our Conclusions and gives an overview of the thesis and suggests possible future work which may be undertaken.

UNIVERSITY of the
WESTERN CAPE

UNIVERSITY of the
WESTERN CAPE

Chapter 2

Literature Review

In this chapter we attempt to identify what a *health management information system* (HMIS) is and then consider a few of these systems found in developing countries facing similar challenges to those in Afghanistan where resources are constrained. An information system is the basis of planning, management and evaluation of a health system. The data could be entered into a HMIS system for the daily running of the clinic and for planning and management.

Agile software development is a concept framework for several software engineering development iterations throughout the life of the project.

2.1 What is a HMIS?

In order to understand what a HMIS system is, we first consider some definitions given by authors concerning this topic.

2.1.1 Definitions

1. The definition of Mendoza and Chong [2004] emphasizes the management application of a HMIS rather than its use in health care: "A HMIS is an information system particularly designed to help in the management and planning of programs, as opposed to delivery of care".

2. On the other hand, Lippeveld et al. [2000] see the system as a database that enables the processing of information for health care: "A HMIS is principally a database system in which 'raw data' are stored and transformed into information. The words, information and data, are used interchangeably in many situations. However, they are not synonyms. Raw data is numbers, characters, pictures or other outputs. Such data are typically further processed by a human or input into a computer,

stored and processed there, and/or convey output to another human or computer".

3. A more comprehensive view is given by [Streveler and Sherlock, 2004]: "A HMIS uses contemporary computerization, telecommunications and systems methods to develop health care processes".

Researching another HMIS, [Sahay et al., 2008] identified four areas of competency, namely, technical, public health, implementation and use context. The use context of a HMIS is *management*. Like any other MIS, understanding the data needs skills in numeracy. "A *health management information system* (HMIS) can be stated as the specialty of a *management information system* MIS and the disciplines of HMISs are derived substantially from the more general field of MIS" [Wajid et al., 2002].

Braa et al. suggest "that action research based on the rule of "network of action is a useful way to improve and implement HMIS in developing countries" [Braa et al., 2004]. They expand this argument to a wider area known as a 'health information system program' or HISP, which covers the planning, improvement, and development of free and open source software known as *district health information software* (DHIS)" [Sahay et al., 2008].

2.2 Purpose of HMIS

The aim of a HMIS should ensure that the needed information is readily available to all users in the health arena for managing information and that all the information is adequately and appropriately applied in its predefined and intended objective. [Ministry of Health and Social Welfare, Uganda, 2009] The goal of the HMIS is to "optimize the performance of health services at all levels of administration through the provision of necessary and sufficient information needed by the health managers to monitor, evaluate and plan their activities" [Smith et al., 2008].

2.3 Software engineering

"Agile is a software development methodology used in the present scenario of information technology. This methodology is principally concerned with how

the price and the time of the industrial people are utilized efficiently. How did this methodology evolved from the prior methodology" [Amudha, 2010]?

2.4 Agile methods

"Agile methodology is an approach to project management, typically used in software development. It helps teams respond to the unpredictability of building software through incremental, iterative work cadences, known as sprint" [Nathan-Regis and Balaji., 2012].

2.5 Agile processes

"Agile processes place value in simple design, with a focus on constructing a working product based on the customer stories for the current iteration only. An Agile catch phrase declaring 'You Ain't Gonna Need It', or YAGNI, is used to prevent inclusion by anticipation. This puts a focus on building software for today, rather than preparing for possibilities in future iterations. Value is also placed on a working version of the product, maintained on a weekly or daily basis. Incremental or evolutionary designer [Fowler, 2004] is also practiced. Rather than spending time planning, analyzing, and designing for a big perfect release at the end of a long development cycle, Agile developers do each of these activities, a little at a time, as development progresses" [Fowler, 2004].

2.6 Different Agile methods

The Agile methods approach has seven sub methods:

- Extreme programming (XP) [Beck and Adams, 1999]
- The dynamic system development method (DSDM) [Stapleton, 1999]
- Scrum [Schwaber and Beedle, 2002]
- Crystal [Cockburn, 2004]
- Agile modeling [Ambler, 2004]
- Feature driven design (FDD) [Coad et al., 1999]
- Lean software development (LSD) [Poppendieck and Poppendieck, 2003; Conboy et al., 2009]

2.7 Agile method techniques

"The basic principles of Agile methods comprise an unforgiving honesty of working code, effectiveness of people working together with goodwill, and focus on teamwork. A set of commonsense approaches emerging from Agile software development processes have been suggested by as follows:

- People matter
- Less documentation is possible
- Communication is a critical issue
- Modeling tools are not as useful as usually thought
- Big up-front design is not required

" [Abrahamsson et al., 2002].

2.8 Extreme programming

Extreme programming (XP) is a collection of well-known software engineering practices [Abrahamsson et al., 2003].

Extreme programming (XP) has emerged as one of the most popular and controversial Agile methods. "As its name suggests it is a programmer-centric methodology that give emphasis to technical practices to promote skillful development through repeated delivery of working software. Although XP and Agile methods as a whole are frequently characterized as less rigorous than traditional techniques, this could not be farther from the truth. XP got its name when its founders asked the questions, "What would happen if we took each technique / practice and performed it to the extreme?" "How would that affect the software process?" An example of this is the practice of code reviews. If code reviews are good, then doing stable code reviews would be extreme; but would it be better? This led to practice such as pair programming and refactoring, which pursue the development of simple, effective designs, oriented in a way that optimizes business value. In fact, full adoption of all of XP's practices needs a high degree of discipline, teamwork, and ability. One of the characteristic differences between XP and other methodologies is its cycle time and "level of ceremony". XP proposes very short iterations between

one and four weeks. XP is also a very low ceremony methodology. Minimal artifacts in an XP project include story cards, code, and part tests" [Coffin and Lane, 2010].

2.9 The Agile process

When designing a project that uses XP methodology, specific guidelines should to be adhered to. XP provides for a set daily routines that should be followed. Some of the practices XP prescribes are: "planning game, simple design, customer tests, small releases, whole team, pair programming, test driven development, continuous integration, design improvement, sustainable pace, collective ownership, coding standard, and metaphor" [Coffin and Lane, 2010].

2.10 Object oriented approaches in Agile methods

"While object-oriented (OO) approaches provide a viable method to develop information systems incrementally, a host of new methods called Agile development go a step further in overcoming the limitations of traditional plan-driven ones. [Nerur et al., 2005].

2.11 A HMIS in Tajikistan

"In the Republic of Tajikistan, there is an increasing demand for good quality health information for decision-making and planning, driven by the move towards indicator-driven planning, performance-based resource allocation, a significant argument in the resource for health, mobilized in the newest years, and the move toward a sector wide approach (SWAP) in the health unit. The existing health information system, inherited from the soviet time, is characterized by an oversupply of data coexisting with great unmet need of information. The HIS assessment will support the HMIS working established under Governance group to develop the national health strategy 2010–2020 and the next HIS development plan, structure upon existing initiatives and system" [Ministry of Health of the Republic of Tajikistan, 2009].

2.12 An evaluation of the District Health Information System in rural South Africa

The District Health Information System (DHIS), developed to gather aggregated routine data from all public health facilities in a country, is intended to support decentralized decision making and health service management. Introduced in South Africa in 1996, it was extended to the entire country by 2001. It is used in several other developing countries in Africa and Asia. The DHIS allows health care workers to analyze their levels of service provision, predict service needs, and assess performance in meeting health service targets [Garrib et al., 2008].

The DHIS has been implemented in all 10 clinics, and the supporting organizational infrastructure was in place. "Evaluation of the implementation of the DHIS in 10 primary health care clinics in rural northern KwaZulu-Natal within a health sub-district which had 15 fixed clinics and several mobile service points. The clinics included were convenience sampled and included 6 that were intended intervention clinics for the larger study, and 4 chosen randomly. The evaluation was designed around the information cycle framework and was structured to assess how well each step within this cycle was working. Interviews were conducted with key informants in each clinic, clinic supervisors, district information officers and other primary health care and district management staff" [Garrib et al., 2008].

2.13 HMIS in Uganda

"In Uganda, a health information system was designed to capture and analyze morbidity data for selected communicable and non-communicable diseases, and other services like immunization and family planning (MoH, 1985). Information was collected in the health facilities, summarized at the district level and later forwarded to the ministry of health at the center where data analysis would be done.

The core function of the Uganda HMIS is to establish and maintain a comprehensive source of health and management information for planning, monitoring and evaluation of the health sector strategic plan" [Kintu et al., 2005].

"Computerization of the HMIS in Uganda has been a slow process due to financial and technical limitations. The central health databank at the ministry of health has been computerized using MS-Access (1997–2001) and Epi-Info software (2002 to date). Also, a training program on use of EpiInfo software for district-level health workers has been started in 10 selected districts which received computer units from a WHO/USAID grant in 2003" [Kintu et al., 2005].

2.14 Tanzania

Data integration has been an important priority within the organizational reform agenda, particularly within the field of information systems, for several decades. This was the case first with inter-organizational systems research using EDI technologies to transmit electronic message to improve the interface between customers and suppliers within an organization, and later with BPR strategies which aim for a radical change and restructuring of business processes. Integration is considered to be an important concept in the current public sector reform agenda. The politics of integration gains even more relevance in the domain of health information systems as evidenced by [Monteiro, 2003] who provides a critical assessment of the logic of health systems integration in the Norwegian context. Integration from a management point of view requires that all different stakeholders of the information system be given a voice to express their priorities. The quest for managerial integration is often at odds with the priorities of local health workers as revealed in the case study. Several advantages are thought to accrue from integrating such data specifically: 1) development of common treatment strategies, e.g. better targeting of multi-agent drugs to needed populations, 2) addressing of common exposures and risk factors, e.g. vector control to simultaneously reduce diseases such as malaria and lymphatic filariasis, malnutrition, and environmental exposures, 3) better incorporation of the externalities of treating one disease, e.g. where treatment of one disease may lead to both positive and negative effects on other co-occurring disease, e.g. reducing parasitic disease leading to both increased allergic disease and negative effective on HIV and vaccination, and finally 4) reduction of redundancy in health activities by bet-

ter integration of specific disease control programs leading to more efficient healthy delivery and strengthening of the health system [Smith et al., 2008].

Data integration at the district level will help localized evidence-based health planning and hence the trend towards decentralization in health [Smith et al., 2008].

2.15 Health management information systems in the ICTPH health systems model

"The IKP centre for Technologies in public health (ICTPH) is piloting a nurse-managed, physician-supervised, technology-enabled, comprehensive primary health care delivery model in rural Thanjavur in Tamil Nadu. Technology is pivotal to the success of this model which is being implemented by ICTPH's partner" [Rajanna and Kapila, 2009].

2.15.1 Role of technology in the ICTPH model

"The technology needs of this model are captured broadly beneath the following categories:

- Electronic medical records (EMR),
- Supply chain and inventory management,
- Diagnostics,
- Technology assistance for health extension workers (HEW),
- Managed care delivery and cash management,
- Monitoring and evaluation and
- Member enrolment and identification.

" [Rajanna and Kapila, 2009]

2.15.2 Architecture

"The ITCPH HMIS used technology Use Cases; the ITCPH HMIS is a web-based application that is accessed by the user through a browser. At the backend, the system is built on the LAMP platform, which uses Linux as the

operating system, Apache as the web server, MySQL as the database and PHP as the server-side programming language, all of them open-source platforms. The code organization uses the MVC (model-view-controller) pattern facilitated by an open-source framework called *Code Igniter*" [Rajanna and Kapila, 2009].

2.16 Developing a systems approach for health information systems

The need for improved routine health information system is unequivocal and well documented.

While there is a general consensus that health information systems should be restructured, very few publications have focused on how to develop such systems. "It has even been argued that health information systems are idiosyncratic to the countries that develop them, and no appropriate models exist that can be applied to all countries." [Lippeveld and Sauerborn, 1992].

2.17 Participatory process of defining the HIS

"In November 2004, the Ministry of health and social Welfare in Zanzibar decided to improve the information system within the hospitals (referral hospital, district hospitals and cottage hospital)" [Igira et al., 2005].

"The decisions to improve these problems embarked on the HISP project which as well is contracted to improve the health management information system in Zanzibar as a whole (i.e. data collection, information flows and use between various health care administrative level- community, health facilities, districts, zones and national levels).

The design and implementation process which started in June 2005 and still on- going, involves agreeing and establishing a standardized reporting format for data collection, customizing various reports and analysis tools for data analysis, customizing the DHIS to meet the hospital's information requirements, and extensive training of users" [Igira et al., 2005].

The introduction of web technology has resulted in a somewhat modified picture. "The use of web browsers for clinical workstations is attractive, as browsers offer a simple and intuitive user front end. The feasibility of building

web interfaces to clinical information system has been shown in 1996, and reports of successful projects followed. WebCIS is an example of implementing a web server atop a clinical data repository" [Kuhn and Giuse, 2001].

2.18 WebCIS: Large scale deployment of a web-based clinical information system

Columbia University has implemented WebCIS, which is an Internet-based information system running on their clinical information system architecture. It has a clinical repository, a medical entities dictionary, an interface engine currently maintained by the Health Level Seven (HL7) organization, and it provides automated decision support with a clinical event monitor using Arden syntax, which is widely used by clinical decision support systems [Samwald et al., 2012]. The security features of WebCIS include secure token authentication, where the authorization is controled by a lightweight directory access protocol (LDAP) server, using SSL encryption, permanent audit logs, and application time outs. WebCIS was initially used by 810 doctors at the Columbia hospital for reviewing and recording medical data [Hripcsak et al., 1999]. It replaced the previous "DHIS" system, was extended to many other hospitals in the network. Deployment began in 1998 and by the end of 1999 it was expected to support more than 4300 users.

2.18.1 Application

We show the main screen of WebCIS in Figure 2.1. It is a comprehensive browser for accessing clinical data stored on repository that displays information from ancillary, registration, and ambulatory system. Retrieval by users can be sorted and aggregated according to the requirements of the user. The output can be displayed as spreadsheets, summaries and in graphical format. The physicians can enter their clinical data and sign it electronically.

2.18.2 Architecture

The architecture of the clinical information system should enable the institution to adapt to change. which defines all coded data stored in database, translates between application coding system, and provides a classification hierar-

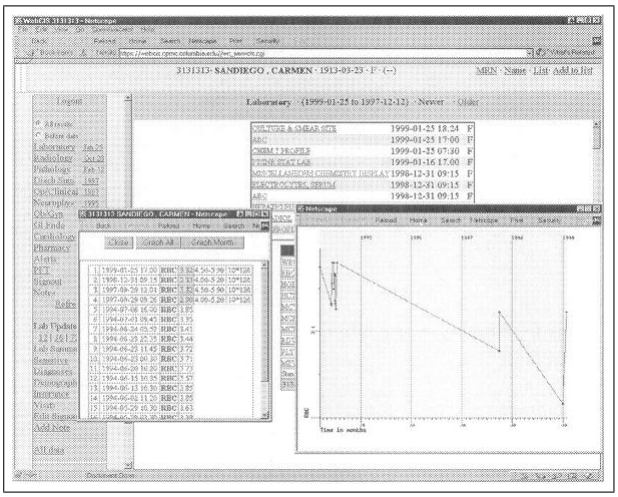


Figure 2.1: WebCIS main screen

chy and semantic relationships that simplify coding and vocabulary maintenance. An overview of the WebCIS architecture shown in Figure 2.2 is similar to that of two other web-based applications, PolyMed and PatCIS [Hripcsak et al., 1999]. WebCIS is implemented as a suite of common gateway interface (CGI) code programmed in C and runs on a UNIX web server. The web server communicates with the mainframe-based repository using the TCP/IP protocol. The CGI code generates HTML pages and JavaScript programs, which then execute on the web browser on the client-side [Hripcsak et al., 1999].

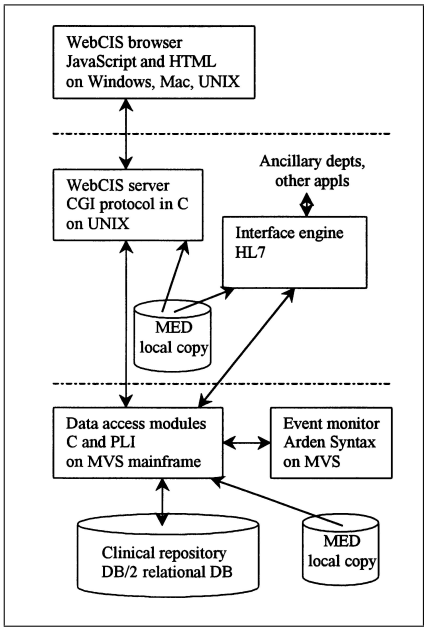


Figure 2.2: WebCIS architecture

2.19 Unified modeling language (UML) and Use cases

The Unified Modeling Language (UML) is a standard language for specifying, visualizing, constructing, and documenting the artifacts of software systems, as well as for business modeling and other non-software systems. The UML represents a collection of best engineering practices that have proven successful in the modeling of large and complex systems. The UML is very important parts of developing object oriented software and the software development process. The UML uses mostly graphical notations to express the design of software projects. Using the UML helps project teams communicate, explore potential designs, and validate the architectural design of the software [Fowler and Scott, 2000].

"Use case is a set of scenarios that describing an interaction between a user and a system. A Use case diagram displays the relationship among actors and Use cases. The two main components of a Use case diagram are Use cases and actors" [Fowler and Scott, 2000].

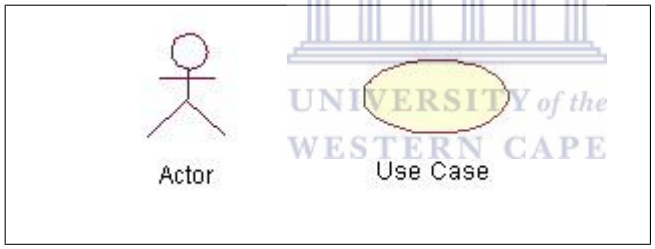


Figure 2.3: Use case and actor

"An actor represents a user or another system that will interact with the system you are modeling. A Use case is an external view of the system that represents some action the user might perform in order to complete a task" [Fowler and Scott, 2000].

2.20 HMIS and its role, features and functionality

2.20.1 HMIS features

The features of the HMIS fall into one of the following six buckets. Here is a summary of these features: (www.ictph.org.in)

- Demographic

- Out-patient
- Supply Chain Management
- In patient department Module
- Health extension worker module
- Monitoring and Evaluation

2.20.2 User responsibilities

"Each user within a participating agency is responsible for maintaining client privacy and protecting each client's protected personal information. Client information shall include, but not be limited to : client's name, address, telephone number, social security number, type of medical care provided, medical condition or diagnosis, veteran status, employment information, and any all other information relating to the client's programming" [Thiessen, 2008].

2.20.3 HMIS functionality

As there is no existing software or data that the HMIS software should be integrated to that, this software needs to meet some particular hardware and software.

Functional requirements are the main features which define what the system supposed to accomplish, actually the functional requirements includes the most common cases that are written descriptions in documents and Use cases [Eidee, 2005].

2.21 Conclusion

In this chapter we gave some definitions and overviewed HMIS in some developing countries. An information system is the basis of planning, management and evaluation of a health system. The data could be entered into a HMIS system for for the daily running of the clinic and for planning and management. A clear definition of a HMIS is a system which provides information to support health management. Agile software development is a concept framework for several software engineering development iterations throughout the life of the project. We applied Agile methods during the development of this

project. Agile methods have several approaches. We concentrated on using the XP method, also described UML and Use cases in this chapter. In Chapter 3 we next discuss our research methodology and design.

Chapter 3

Research Design and Methodology

3.1 Introduction

This chapter discusses the research design and methodology for creating the (HMIS) for Kabul University Poly-clinic. The research methods are interviews, surveys creating the prototypes and focus group. For designing the (HMIS) we use *Hypertext Preprocessor*(PHP) for design the web-based interface; MySQL for system database and the *Unified modeling language* (UML) is design specifically to represent object-oriented system. The project was started by first clarifying what kind of prototype was envisaged to solve the problems of the Poly-clinic by means of gathering concepts from a focus group by conducting interviews and surveys. The analysis of this data led to the first prototype. The HMIS we came up with manages data systematically in order to ease the burden of accessing and manipulating the health records of patients. This chapter discusses and analyzes the technical aspects of the HMIS web-based application. It covers the kind of technologies used and why they were chosen, and in what way the considered solutions could contribute to the project.

3.2 Research hypothesis

A hypothesis is a particular, testable prediction about what you think to happen in the study.We decided to use Agile methods—especially concentrating on *extreme programming* (XP)— as our software engineering approach in order to arrive at a prototype solution quickly, and to enable the rapid convergence of successive implementations of software prototypes to a realistic software solution.

3.3 Research instruments

3.3.1 Sample design and sample techniques

"A sample of convenience is the terminology used to explain a sample in which elements have been selected from the target population on the basis of their accessibility or convenience to the researcher" [Dawson, S and Manderson, L, 1993].

"A sample survey differs from an experiment in several important ways. A sample survey is characterized by: a clearly specified population, a sample selected by a random process from that population, and the goal of estimating some population parameters " [Scheaffer, 1999]. Our survey did *not* sample the population. *All* the personnel participated in the surveys. Based on interviews and questionnaires filled in by the Poly-clinic doctors and other staff, we deduced the first version of the software prototype. Subsequent feedback from them was applied to improve the next version of the software leading to the first *revision* of the software.

3.3.2 Interviews and questionnaire

An interview is a series of questions directed by a researcher by personally engaging with respondents. "A questionnaire is a formalized set of questions for obtaining information from respondents. The overriding objective is to translate the researcher's information needs into a set of specific questions that respondents are willing and able to answer" [Malhotra, 2004]. "Questionnaires are usually paper based or delivered online and consist of a set of questions which all participants are asked to complete" [Adams and Cox, 2008]. We first interviewed the director of the Kabul University Poly-clinic and then with his permission proceeded to collect data using a survey.

3.3.3 Data collecting by interviews or other methods

Data collection through questionnaires operates by bringing together the ideas of the target people in one place to exchange views and answer specific questions. This method can be a good way for collecting necessary data in short time. We used three questionnaires for gathering data.

"Data can be divided into two main types: (1) primary data—which is collected for a specific purpose, and (2) secondary data—information that

is already collected and available. The benefit of primary data is that it is collected to answer your own particular questions. Secondary data has the advantage of being available immediately, is a often cheap and easy to obtain but the disadvantage is that it may not answer all the needs of the researcher. There are several ways to collect primary data, namely, (1) questionnaires or surveys, (2) in-depth interviews or group discussions, and (3) observations". [Scout Association, 2000]. We used questionnaires for gathering data and we conducted interviews where we questioned the Poly-clinic doctors, nurses and other staff. The direct feedback from them was first used to reach an initial design of the first version of the software and then later these scrum sessions were used to improve the first version of prototype. After that we developed improved versions of prototype in response to the feedback we received from the Poly-clinic users.

3.4 Agile methods

According to a definition of Jorge Diaz-Herrera [Jorge, 2001], "software engineering is the application of a systematic, disciplined, quantifiable approach to the development, operation and maintenance of software, that is, the application of engineering to software". "Agile methods are methods that try to be responsive to the requirements—especially changing requirements—of the software development process, that is based on practice and skill, and that focuses on being successful and sufficient" [Hunt, 2004].

Agile methods have dissimilar approaches like "*extreme programming* (XP), Scrum, Lean Software Development, *feature driven development* (FDD), *dynamic systems development method* (DSDM), Crystal and *Agile unified process* (Agile UP or AUP)" [Cordeiro et al., 2007]. We use XP, which is more commonly known and used to address the needs of developers who operate in domains characterized by changing requirements [Newkirk, 2002]. XP was originally designed as a method of supporting small development teams working within uncertain and changing requirements. XP does provide a software development life cycle model as well as guidelines on the organization of a software development group". The *user story* is most important unit of XP [Hunt, 2004]. We extracted information from the user stories of the Poly-clinic staff

to discover the functions and the features they saw as the most needed at the Poly-clinic. We used interviews and a series of questions to extract more information. These questions will be described in Chapter 4 and the responses are tabulated in Appendix A.

3.5 Using Unified modeling language diagrams

The *Unified modeling language* (UML) is designed specifically to represent *object-oriented* systems. Object-oriented development methods describe software as a set of cooperating blocks of information and behavior [Pender, 2003]. The UML has various diagrams such as Use Case diagrams, Sequence diagrams, and Class diagrams. We use Use Case diagrams.

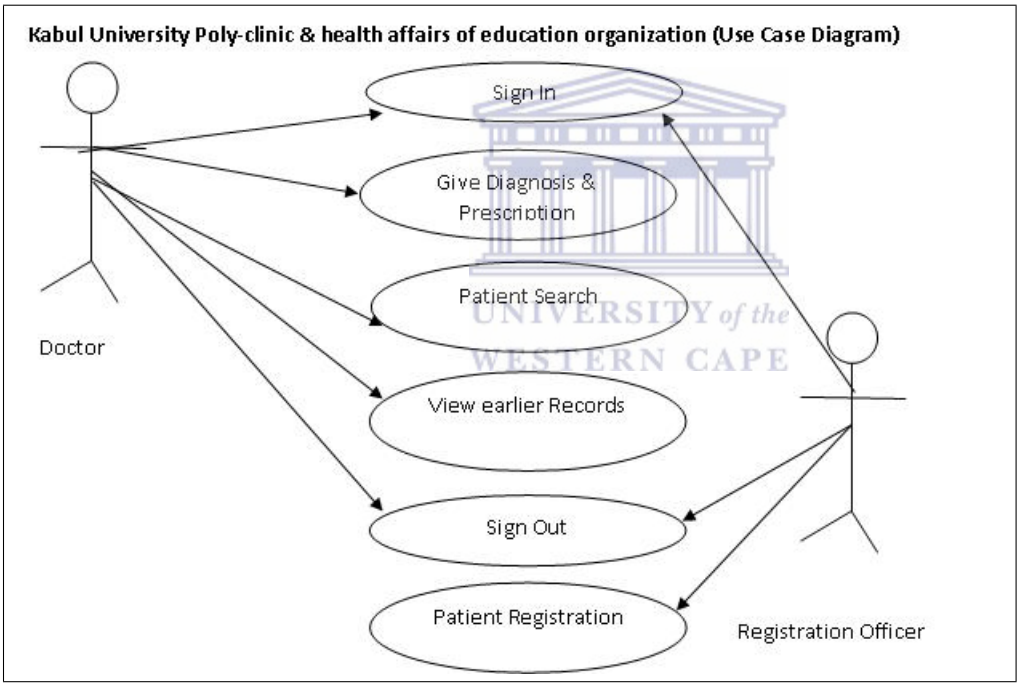


Figure 3.1: Kabul university Poly-clinic—Use case diagram

3.6 Using PHP scripting language for design interface

We used PHP5 (Hypertext Preprocessor) for designing the web-based interface for the Poly-clinic software. This interface must be friendly and easy to use. PHP is a powerful server-side scripting language for creating dynamic and interactive websites. Also PHP is open source software and free to download. PHP is a robust language, which means that it supports a wide variety of

properties, methods, and interactions. PHP's success can be accredited to a number of reasons: It

- is extremely easy to learn.
- is fast.
- has excellent integration with Apache.
- has excellent integration with MySQL, a popular open source relational database.
- runs on various platforms such as Linux, UNIX, Windows, etc.
- is compatible with most web servers used today such as Apache, IIS, etc.

It is compatible with most popular databases, it provides a high security using *secure socket layers* (SSL), and has other features that makes it an obvious choice [Carnaghi, 2004].

3.7 Using MySQL for development database

A good system for a health management information system needs a reliable database system. MySQL is a popular choice of database for use in web applications. It is a fast and robust relational database management system enabling persistent storage, manipulation and retrieval of data [Welling and Thompson, 2009]. A database is essential for storing persistent data. The database must be able to store patient data such as names, addresses, treatments, medications prescribed, appointments, etc. We selected MySQL for the Poly-clinic data, because it provides for our needs by efficiently storing and retrieving data. A strong advantage is that MySQL is free open source software.

3.8 Research prototype

"In software development, a prototype is a rudimentary working model of a product or information system, usually built for demonstration purposes

or as part of the development process" [Moggridge, 2007, Chapter 10, Pages 643–735].

In the systems development life cycle we first used the prototyping model. In this model a simple prototype is developed and thoroughly tested and in subsequent iterations the application reworked until an acceptable design has been achieved, from which the final version is developed. In our research the prototype had two main parts, the database and the application with its web interface.

3.8.1 The database

An expository interview with Dr Ahmadullah Faqiri the head of the Kabul University Poly-clinic helped us to extract enough of an idea of the initial software requirements to lead us to a questionnaire to all the staff. It became obvious that a centralized web application for managing patient information was needed. It should be possible to share information that would be accessible anywhere in the Poly-clinic. The questions and interview with Dr Faqiri provided a clear guideline. After the interview we formulated a survey and the responses to the survey led us to develop the first version of prototype. Using the prototype we could illustrate its functions more vividly to the users. Some of these functions were:

- Elements for adding patient information to the database

If the user clicks on the save button, the user can input information for a patient. This function adds information about the patient, for example the patient's ID number, name, position, contact number, etc.

- Elements for adding diagnostic information to the database

The user can add the patient's diagnosis to the database using the patient's ID number.

- Elements for adding the prescription to database

After diagnozing the patient's state, the user, i.e., the doctor, can add a prescription for the related patient.

3.8.2 Main application and interface

The HMIS for the Poly-clinic is a web-based application; the prototype application was created in PHP using a MySQL database. The tool has several windows with components such as buttons and fields.

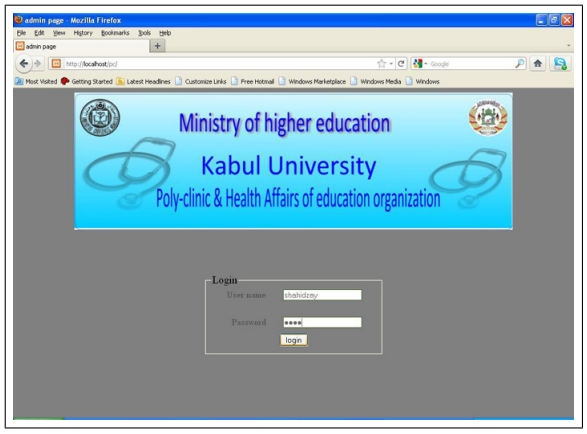


Figure 3.2: Prototype interface for Login Screen of Version 1

- Figure 3.2 shows the User Name and Password buttons to login into the main page

If the correct user name and its accompanying password are entered, then the user gains access to the system. In order to use the system the doctor's user name and password must be entered to unlock the system and connect it to the database. The user can then add new patient patient information and fill in medical advice given to the patient that will be filed into the database.

The database is used to store, adjust, retrieve and otherwise manipulate patient information. The database has tables and each table has related fields.

When the login button is clicked, the user name and password are checked in the database, if the user name and password are correct, the user is logged on and the main page is displayed. In Figure 3.3 we show links for the patient's registration and the doctor's diagnosis and prescription. When the patients come to the Poly-clinic, they first have to register. The registration officer fills in the registration fields shown in Figure 3.4 and issues an ID card to the patient, who can then visit the doctor.

The doctor can check the patient's ID and name in Figure 3.5 and then click on the related ID and enter his diagnosis and prescription as in Figure 3.6.

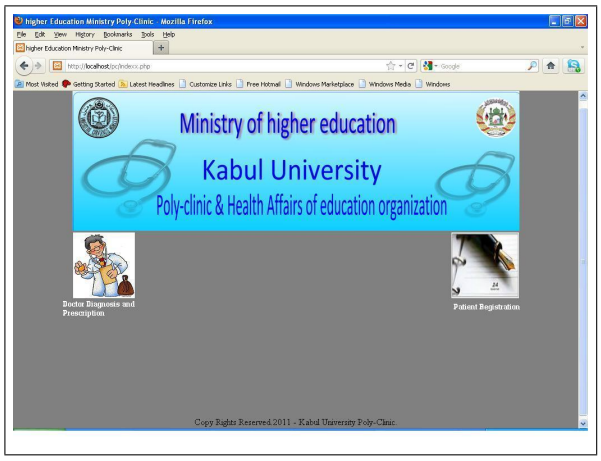


Figure 3.3: Prototype interface for the Main screen of Version 1

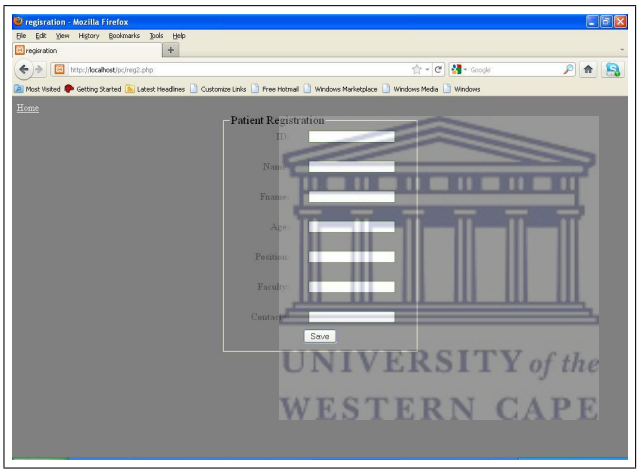


Figure 3.4: Prototype interface for Version 1—Patient registration screen

In Figure 3.6 the doctor just can check the patient's state and give the diagnosis and prescription and he doesn't have access to change or edit the patient name, ID and date. The Submit button stores the patient diagnosis and prescription into the database and print button prints the hard copy of the Figure3.6 for the patient.

Figure 3.7 illustrates the fields in the `patient` table of the database. The fields are: `name`, `fname`, `age`, `position`, `faculty`, and `contact`.

3.9 Prototype development

We applied extreme programming methods from the family of Agile methods to design the HMIS for the Poly-clinic. The first step in developing the HMIS

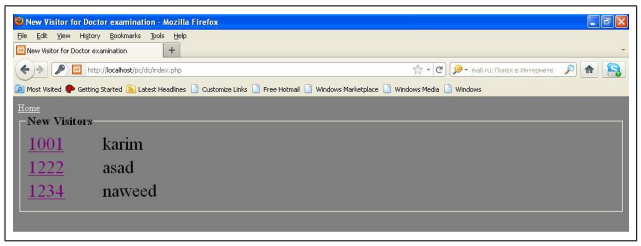


Figure 3.5: Prototype interface for Version 1—New patient for doctor examination screen

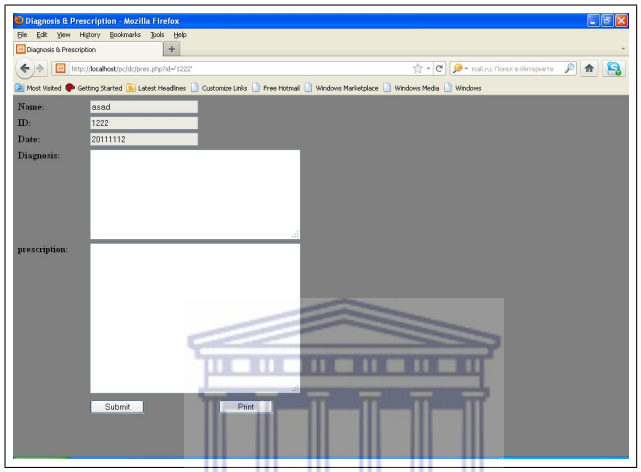


Figure 3.6: Prototype interface for version 1—Diagnosis and prescription screen

system using the extreme programming process was the planning step. Our planning was based on an exploratory interview with Dr Faqiri, the head of he Poly-clinic.

The next steps are coding and testing. We used UML to do the designing. The user story that we evolved is shown in the *Use case diagram* in Figure 3.1 on Page 26. The code was written using PHP and the database was done in MySQL. In the second prototype we designed an interface connected to the database with some buttons to perform various commands.

3.9.1 The first prototype

This prototype was used to test and uncover design problems. Figure 3.3 on Page 30 shows the prototype main screen that has two links—doctor diagnosis and patient registration—each link has a specific functions and features.

Selecting a link on a window by clicking a button executes the action which belongs to it. As is illustrated in Figure 3.3, one of two buttons on the

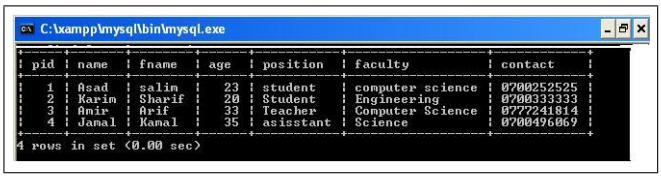


Figure 3.7: Prototype database for Version 1

main screen links to (1) the *doctor's diagnosis* window where the doctor can enter the diagnosis and the other button links to (2) the *patient registration* window that enables code to insert patient information into the database. The

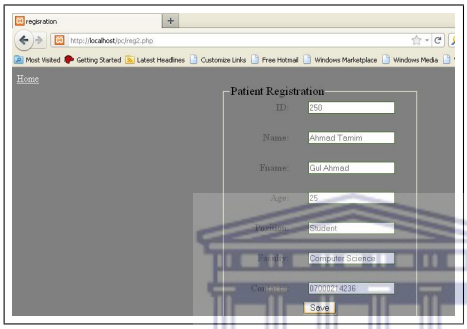


Figure 3.8: Prototype interfaces for Version 1 patient registration fields

latter window is used for inserting the patient ID, Name, Father Name, Age, Position, Faculty and Contact Number. This form has a Save button to store information in the database. The other link in the main screen is for entering the diagnosis of the patient's complaint and the prescription for drugs that the doctor can enter for the treatment of the patient.

3.9.2 The second prototype

We tested the first version of the prototype on the users and obtained their feedback in order to improve the next version. In the first step of the development we recorded user stories as Use case diagrams, then designed the prototype. The prototype shown to the users for feedback. After the user feedback we redesigned the prototype to apply those comments. Next we discuss the second version of the application.

The prototype had a single login screen, as shown in Figure 3.2 on Page 29, where users can login with a username and password. After entering the user name and password on the main screen a new window appeared

with two links, as shown in Figure 3.3 on Page 30. In the second version, we added a separate username and password for for each class of user, i.e., the medical personnel and administrators, and we also changed the background. Figure 3.9 shows the new main screen after adding the new changes. In the

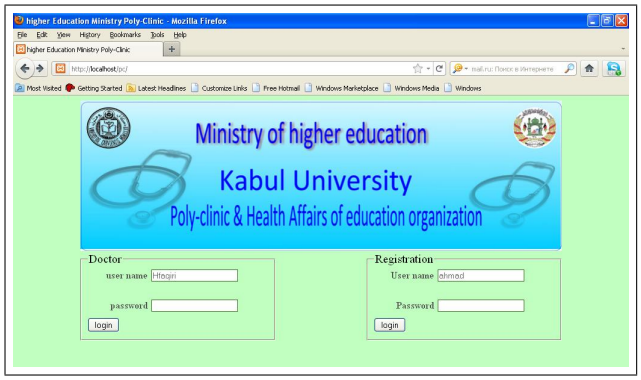


Figure 3.9: prototype interface for version 2 main screen

first version prototype the patient registration window had seven fields. We added two fields: Sex and Place of Work. We also added validation for the the ID, Age and Contact fields so that the registering official can only insert decimal digits, and is compelled to enter letters in the Name, F/name and Place of Work fields. Figure 3.10 shows the patient registration window with the changes.

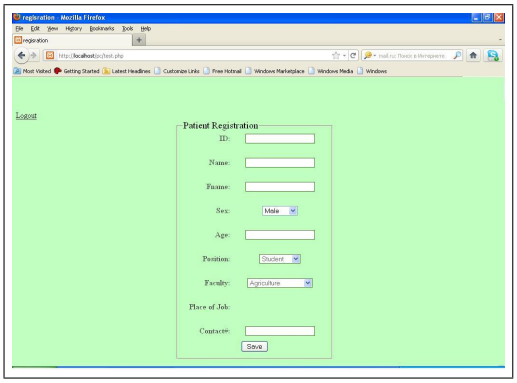


Figure 3.10: Prototype interface for Version 2—patient registration screen

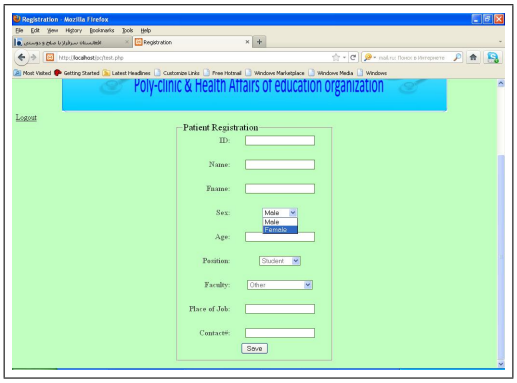


Figure 3.11: Prototype interface for Version 2—Patient registration screen

We inserted a drop down menu with {`Male`|`Female`} options for the the Sex option. Figure 3.11shows these options. Also we changed the Position

option to a drop down menu with the options:

{Student|Professor|Non-teaching Staff}.

. Figure 3.12 shows the Position field options.

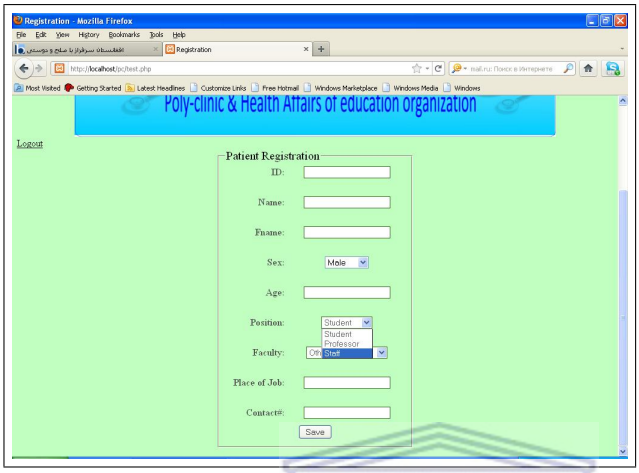


Figure 3.12: Prototype interface for Version 2—Patient registration screen

We also changed the Faculty option to a drop down menu listing all the faculties as options. Figure 3.13 on Page 34 shows the Faculty field options. When a patient is a visitor from outside of the university of Kabul, the registering official must select the Other option, and a Place of Work field will be presented that must be filled in. Figure 3.14 on Page 34 shows the Faculty and Place of Work fields. As a result of user feedback after using the prototype

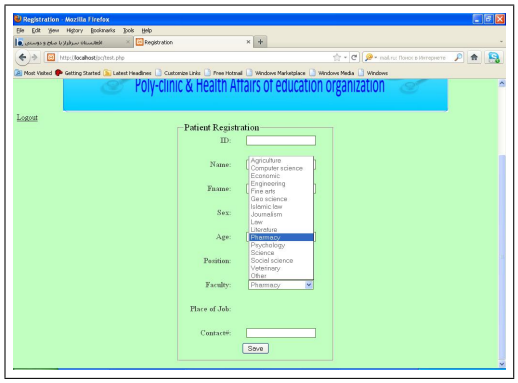


Figure 3.13: Prototype interface for Version 2—Patient registration screen

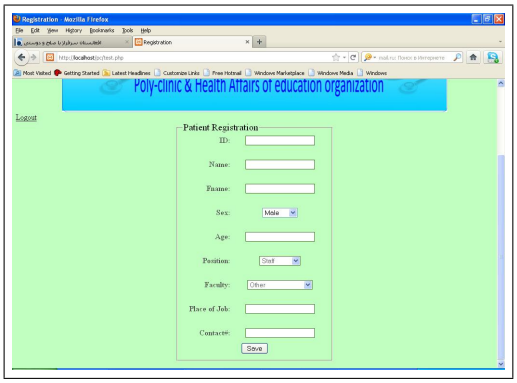


Figure 3.14: Prototype interface for version 2—patient registration screen

we changed the background for the "new patients for doctor examination win-

dow". We added the Poly-clinic logo and we inserted a search button which enabled the user to search for returning patients, so that if a patient had already been registered, re-registration was unnecessary. Figure 3.15 shows the "new patients for doctor window" after this change. When the doctor checks

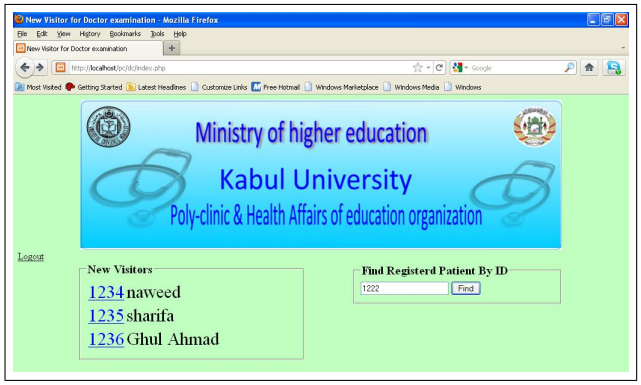


Figure 3.15: Prototype interfaces for Version 2—new patients for doctor examination screen

the ID of a patient that has already been registered, then control goes to the diagnosis and prescription window where that patient's details are displayed. Figure 3.16 shows the diagnosis and prescription window. Links that show the previous dates that the patient has visited the Poly-clinic are displayed on the right-hand side of the window. When such a link is clicked it is followed and the details of any previous prescription and diagnosis for the patient can be seen for any selected date.

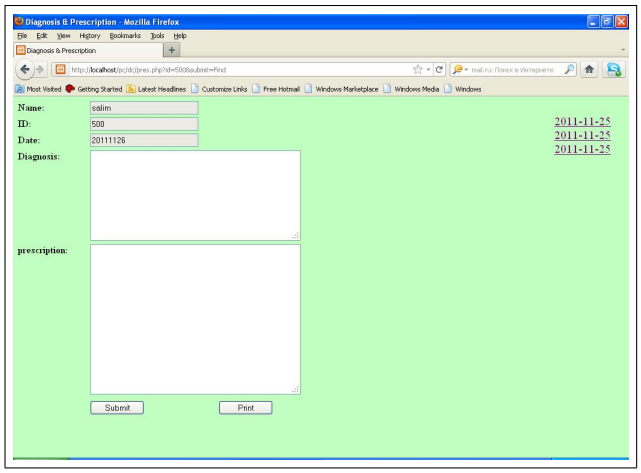


Figure 3.16: Prototype interface for Version 2—Diagnosis and prescription screen

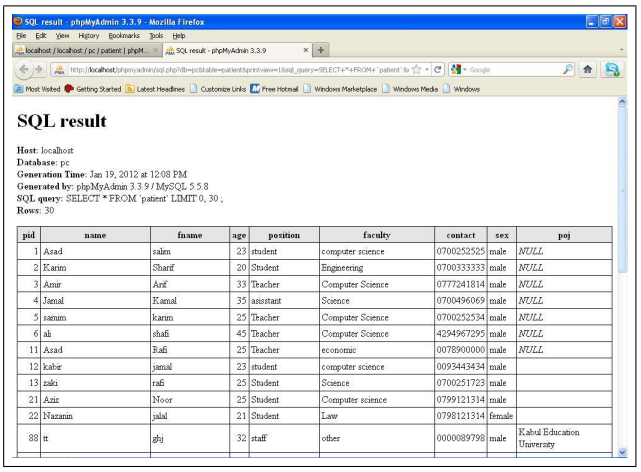


Figure 3.17: Prototype database for Version 2—Patient's registration

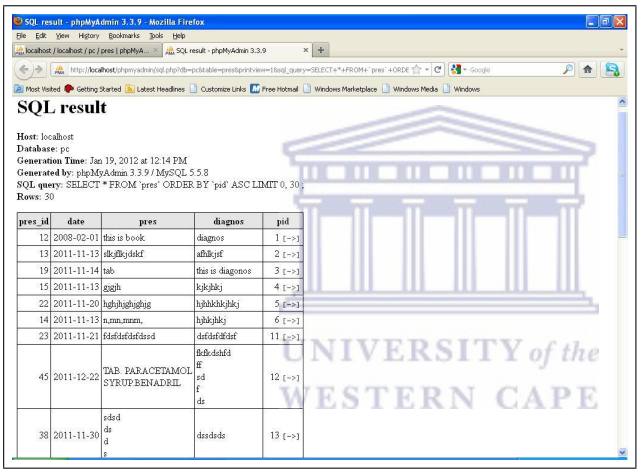


Figure 3.18: Prototype Database for version 2—Prescription and diagnosis

After using the prototype in feedback user is requested some improvements to the diagnosis and prescription window. Some database queries were added and Figure 3.16 shows the result of such a query after the change.

We presented the second version of the software to Poly-clinic staff and received further feedback and we eventually applied that user feedback in the third version of the software.

3.9.3 The third version of the software

After using the second version of the software, users requested us to replace the English interface with a Dari interface. The entire software interface has now been translated to Dari which is an official language in Afghanistan. Pashto is the other official language and a Pashto interface will be essential if we

distribute the software to provinces where Pashto is the dominant language. The main screen of the Dari version is shown in Figure 3.19. The functions of the third version prototype are the same, the only difference is the language.

3.10 HMIS window in Dari

The second version was also presented to the same group of people for three weeks and we created a second set of questionnaires. The results of that questionnaire appear in Appendix A.2.

The third version of the software is the final version. This prototype has all the features and functions described above. In Figure 3.19 and Figure 3.20 on Page 38, the third version prototype interfaces and database was also re-designed into Dari. The users in Poly-clinic do not know enough English. After getting feedback from the users that they wanted a Dari interface, we

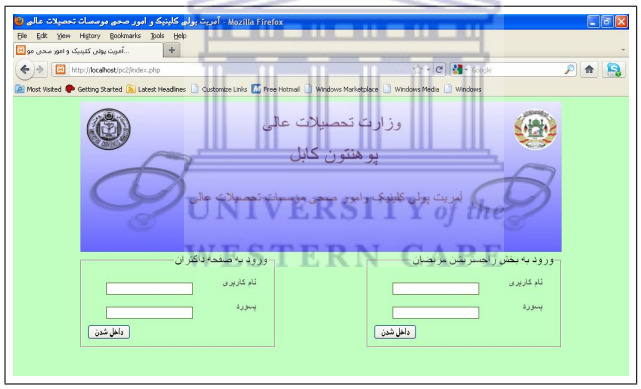


Figure 3.19: HMIS window in Dari

also changed all the windows and their buttons and fields to their respective Dari versions.

3.10.1 Differences between the English and Dari versions of the application

The Dari version of the software is same as the English or second version, but it only differs in the language of the interface. Since Dari reads from left-to-right like Arabic script, all the buttons that were on the left-hand side of the entry boxes they are now on the right-hand side in a mirror-like transposition of all the windows. Everything that was on the left is now on the right and vice versa. Figure 3.20 shows the Dari version of patient registration screen.

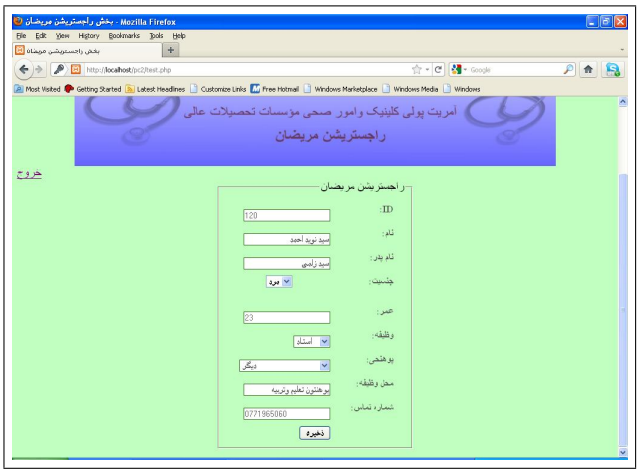


Figure 3.20: Patient registration window in Dari

3.11 Data collection

Three surveys were organized corresponding to the three versions of our software.

3.11.1 First survey

The aims of this survey were to collect the requirements for designing a HMIS for storing information for the Kabul University Poly-clinic.

Initially data was collected in an interview from 12–17 August 2011 with the Poly-clinic director Dr Faqiri. He is involved in health management and had the background to help us. We received useful feedback during this interview which gave us a clear idea of the main problems at the clinic. After the feedback we designed the first version of prototype. A full transcription of the interview with Dr Faqiri is given in Appendix A on Page 63.

3.11.2 Second Survey

The users tried out the prototype for a week from 2–12 November 2011. Following this we did a follow-up second survey and collected data from 13–15 November 2011. The aims of this survey were to capture new ideas about the prototype. We did the second survey with 20 persons. They knew the HMIS and we used their comments and suggestions to develop the software. The second survey was conducted using Questionnaire 2 that is described in Chapter 4. After receiving feedback from the users, we could develop the

Figure 3.21: Prototype interface for Version 3—New patient for doctor examination screen

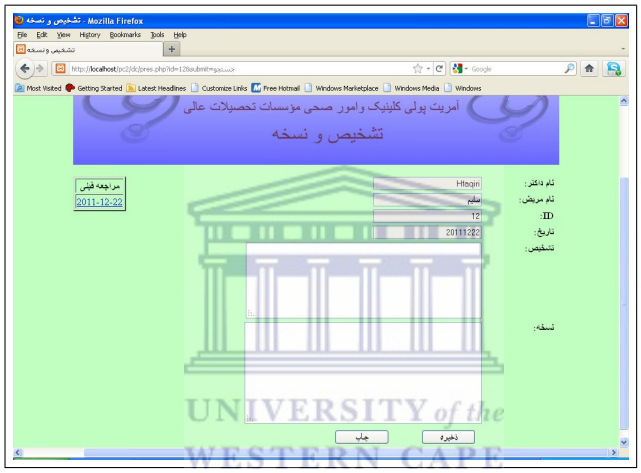


Figure 3.22: Prototype interface for Version 3—Diagnosis and prescription screen

second version of the software.

3.11.3 Third Survey

The purpose of the third survey was to get new ideas about the second version of the software. The participants used the second version for three weeks from 23 November–13 December 2011. We did the third survey from 14–18 December 2011. After we received the feedback from them. We used Questionnaire 3 to develop the third version of the software. The questionnaire and its responses are discussed in Chapter 4.

3.12 Data editing

We used computer based questionnaires for collecting the data. Questionnaires gathered data with more detail. Comments were typed by interviewers. The same questionnaires were used for male and female participants.

For editing the questionnaires were divided in three parts. The first part about background information; the second part of the questionnaire were about computer knowledge. The third parts of the questionnaires were about the HMIS system. For recovering more information on the suitability of our prototype, we designed the questionnaires in our local language (Dari) and then translated the answers to English. After the coding process, we inserted the data in a spreadsheet to extract some simple statistics.

3.13 What is directed interview?

The directed or directive interview involves the interviewer using an outline and asking specific questions within a certain time frame. The interviewer works from a checklist and takes notes. This type of interview is impersonal and seeks to reveal facts.

3.14 Conclusion

In this chapter discussed our research design and methodology. In the first part of this chapter we discussed the research hypotheses in terms of five research questions, the research instruments, and how our surveys led to the design of the prototype, then how user feedback let us improve the design of the prototype to create a second version of the software After some user feedback after the users had experienced the second version of the software we redesigned the third version of the software in Dari.

In the next chapter we will discuss the results of the interviews and questionnaires.

UNIVERSITY of the
WESTERN CAPE

UNIVERSITY of the
WESTERN CAPE

Chapter 4

Presentation and Discussion

4.1 Report and Results

We ran three formal surveys to extract the user requirements for the software we intended to deliver at the Poly-clinic. Next, we discuss this process.

4.1.1 First survey summary

According to the first interview with the head of the Poly-clinic Dr Ahmadullah Faqiri to obtain his views of the requirements for software for the Poly-clinic, a number of problems that were ranked as a high priority were mentioned and regarded as being necessary to be fulfilled by this application. Data redundancy was one of the main problems—the same information for a single patient was stored in different places. Patient records were kept in hand-written books; so that searching, or editing a particular patient via many different books was tedious and sometimes proved to be impossible. There was no clear rule stipulating where to keep patient information, thus it was obvious that data could be lost. It was difficult to follow patients' medication trails in cases where patients lost their doctor's prescription and their records often could not be traced when patients came for repeat treatments. We also gathered some paper patient records that the poly-clinic used to register patients before visiting the doctor.

From the information gathered from the first survey, we developed the initial version of the software to serve as a prototype to familiarize users with our idea of what they wanted.

4.1.2 Second survey summary

The main purpose of the second survey was to get feedback based on the experience gained by the Poly-clinic staff during their use of our first prototype.

In the second survey we received responses from 20 people. Here 45% (9) were doctors and 45% (9) were nurses working in the Poly-clinic and 10% (2) were administrative officers working in the patient registration office of the Poly-clinic. They have skills in Microsoft Windows operating system. Fifteen participants (75%) have word processing skills and can use Microsoft Word; some respondents also have experienced using a spreadsheet and have used Microsoft Excel. Four (20%) of the respondents have actually used a database and have skills in the use of Microsoft Access and 40% (8) have knowledge of Internet Explorer. The report is summarized in Table A.1 on Page 66. The questions used in this survey can be found in Appendix A Section A.2 on Page 66.

4.1.3 Third survey summary

The purpose of the third survey was to receive feedback on our second version prototype. The third survey was given to the same 20 respondents, who are all the staff members of the Poly-clinic. Here 80% (16) responses preferred the HMIS to be in Dari. Eighteen (90%) respondents felt that all necessary database functions are present". From the third survey, we developed the third version of our software using a Dari language interface. All the questions and results used in this survey have been tabled in Appendix B.

The survey results are tabulated in Appendix A.

In the first part of the second survey we learned that the poly-clinic doesn't have Internet connection. Fifteen (75%) have a computer at home. Six (30%) have a computer at the poly-clinic. Some respondents have knowledge of computer use. One (5%) respondents has uses email.

In Table 4.13 Seems 20% (4) have Microsoft Access skills and Table 4.14 Shows 40% (8) have skill of Internet Explorer.

In the second survey part 3 we found in Table 4.15 about the kind of HMIS 55% (11) response that HMIS present information about patients. In Table 4.16 "What type of tool is necessary for HMIS?" 85% (17) response said the HMIS easy to use and more attractive. Table 4.17 shows the 50% (10) responses the HMIS should in local language. In Table 4.18, 90% (18) responses said that all database functions are present. In Table 4.19 questions about "What do you think about the database's fields? (Name, father name, age, position, faculty, contact no)" Responses 65% (13) said adequate.

In the third survey we found the question in Table 4.29 "What is your position?" 45% (9) responses were doctors, 45% (9) were nurses and 10% (2) were administrative officer. Table 4.30 shows the poly-clinic employees work experience. "Table A.30(55%) (11) responses are working from 1-10 years, 30% (6) working from 11-20 years and 15% (3) working 21 or more years". Table 4.31 shows the responses gender, 85% (17) responses are Male and 15% (3) are female.

In the second part of third survey we found that in Table 4.40, 80% (16) responses have knowledge of Microsoft word. Table 4.42 shows 20% (4) responses have knowledge of Microsoft Access. Table 4.43 shows the knowledge of Internet Explorer which 40% (8) responses uses Internet Explorer.

In the third part of third survey the question "Do you prefer the HMIS in English or local language?" revealed that the majority of the users wanted the interface of the software to be in Dari. In Table 4.44, 80% (16) responses were that the HMIS must in Dari. Table 4.46 shows the database functionality which 90% (18) responses were all functions are present. Table 4.51 shows the functionality of login button which 100% (20) responses were "It works well".

4.2 Comparing the research findings / results with the hypothesis

In Section 3.2 our hypothesis stated that if we use Agile methods as the software engineering approach, we will be able to develop a HMIS for Kabul University Poly-clinic.

For designing and improving the prototype through the development of newer versions, we used extreme programming (XP), which is a further process model for the arrangement of the software development process. Extreme programming is an excellent subject for studying internal process dependencies [Vandenburg, 2005]. One important thing is between XP and other methodologies are the cycle time. XP recommends very short iterations between one and four weeks. After the user feedback we designed the third version of prototype in Dari (Dari is a local language in Afghanistan). We did three survey use (interview and questionnaire), the respondents said the interface and the database is appropriate for HMIS to the Kabul University Poly-clinic. The interface, registration page is used for patient registration (data entry) searching the particular patient from the database and displaying the data on screen. The main result of the project is an HMIS for the Kabul University Poly-clinic in Dari—our predominant local language.

4.3 Main trends and patterns

All participate in their comments said that the interface should be in Dari. All respondents answered in their comments about the login screen, patient registration screen, new visitors for doctor examination screen, search the second time patient buttons, and diagnosis and prescription screen were correct. Respondents said in their comments the background color is good and all buttons work well.

4.4 Summary

The surveys and prototypes showed that we successfully designed a HMIS for the Kabul University Poly-clinic using the XP Agile method/process. The HMIS has two main parts, first its interface and second a database. The interface has been done in *Dari* to make it more accessible to users.

UNIVERSITY of the
WESTERN CAPE

UNIVERSITY of the
WESTERN CAPE

Chapter 5

Conclusion and Recommendations

5.1 Introduction

The patient management information system at the Kabul University Poly-clinic system has been paper-based since its inception. The Poly-clinic has been running its patient admission system using a clumsy and error-prone manual system that has led to many frustrations such as missing or duplicated patient records and time has been wasted searching for documents that have not been filed correctly. After intensive interviews with the Poly-clinic personnel and a written questionnaire we determined the greatest needs in order to produce a prototype patient management system.

Following the first round of interviews and questionnaires regarding the requirements of users at the Poly-clinic a usable prototype of the system was installed on a few computers in the patient admissions office so that the clerks and medical personnel could try out the system and familiarize themselves with computers. In this process they also gained invaluable experience in the use of computers to ease their clerical burden of keeping track of patient records in the Poly-clinic.

After engaging with the first prototype for a few weeks, the administrative and medical personnel gained some experience and familiarity with our prototype computing system that assisted them in their daily duties. These users and could now respond with more authority and confidence to questions of how they saw the future growth of the system.

The administrative staff and the medical personnel using the system were subsequently interviewed and a survey was done using a second questionnaire to determine requirements and improvements to the system. Users could now identify problems because they had a better understanding and much more familiarity with the capabilities of a computerized administrative system.

This feedback was used to implement an improved version of the soft-

ware. After feedback from the Poly-clinic staff based on their experience using the first prototype we upgraded the prototype to create a second version of the software.

In order to communicate with the supervisor the first prototype and second version of the software were done using an English interface that the users could use with some effort but with which they were not comfortable. A final version of the software has been implemented in Dari which is the main language used in the Kabul Province and an official language of Afghanistan. The final version has proven to be satisfactory to the users and the clients of the Poly-clinic.

5.2 Overview

A patient management information system using the Agile process to build a HMIS for the Poly-clinic has been implemented. A system that manages the paperwork previously done manually has now been implemented to handle all patient records electronically.

Chapter 1 described the aims of the research and discussed the motivation for the project. The Poly-clinic did not have a database for storing patient information. An overview of the previous state of affairs of the Poly-clinic administration was described. In order to design a new patient management system we needed to understand the shortcomings of the current paper-based system and discuss our research objectives of determining what and how to tackle the implementation of a computerized system.

Chapter 2 focused on a review of the literature covering health management information systems in some developing countries. We discussed definitions, and the purpose of HMIS, Agile methods, Agile method techniques and processes, extreme programming and Unified Modeling language. In particular we overviewed HMIS in Tajikistan, Uganda, Tanzania, Zanzibar, India and an application of HMIS in South Africa.

Chapter 3 described the methodology and design of the research project. This chapter described the design of the HMIS. We illustrated the requirements with Use cases in UML. We explained the research instrument and discussed research design and methods. We did three surveys using inter-

views and questionnaires and created three prototypes based on findings from the surveys. We released the final version for the purposes of this thesis by replacing the English interface of the software with one in Dari.

Chapter 4 discussed the results and reports from the questions covered by the surveys. Three surveys gave us a found clear idea about the role of the HMIS needed by the users at the Poly-clinic.

5.3 Research findings

The literature review led us to conduct focus group interviews and to run a survey to determine the most pressing requirements of an initial system. We did a follow-up survey that gave us feedback from the clinic staff exposing the problems and shortcomings the first prototype. Responses by the Poly-clinic staff led us to improve the prototype and add some innovations. The second prototype prototype was handled similarly.

The third prototype was created in Dari, because it is the official language of Afghanistan and is generally understood by everybody in the country. The patient registration process benefited greatly by using Dari. The respondents found that the interface and the database were appropriate for the HMIS of Poly-clinic. The registration page is used for patient registration, data entry, searching the particular patient from the database and displaying the data on screen. The main result of the project is a HMIS for the Poly-clinic in Dari.

5.3.1 Data collection

Data collection is an important aspect of the research. We used interviews and questionnaires to collect the requirements for designing the HMIS for storing information for the Poly-clinic. We interviewed Dr Ahmadullah Faqiri the head of the Poly-clinic to determine his view of requirements for the system. We did focus group surveys using interviews and questionnaires as our main instruments and acquired three prototypes from the surveys and developed the final version using Dari.

5.3.2 Storing the information

The system needs a suitable database system for storing information used by the HMIS of the Poly-clinic. Patient information is stored in a database

secured by username-and-password system for access.

5.3.3 Agile Method

Extreme programming originated because plan-driven methods do not tend to work well in an environment where there is little prior computerization or even user knowledge of how a software system can improve productivity. In typical plan-driven method such as the waterfall method the user-software-developer cast initially play the roles the users are interrogated by the software developer in order to discover the underlying business rules. In plan-driven methods the cast rigidly determines the software requirements and work forward to the final acceptance of the application in long and time consuming strides.

When using XP, however, very short steps are made and a small development team can determine short term goals and implement discovered requirements as soon they become clear. This in turn causes the users to learn about what they can achieve with the software while it is being developed and with this knowledge and new experience actually improve the target application long before the final product is signed off. In an XP driven project the user-software-developer cast changes so that the developer and users also can play the roles of one another, because the users learn what can be done and the developer learns what has to be implemented.

Extreme programming practices require resourcefulness and teamwork in order to establish fast iterations that both convince the users that the software is beneficial to their efficiency is useful. It also incrementally drives the software to converge rapidly to a satisfactory solution. XP differs from water-fall style methodologies by having short cycles of less than a month and a less rigid "level of ceremony".

The user story is an important aspect of XP. From the user stories we received pertinent information about the functions and features needed at the Poly-clinic. The development steps were clear at each stage of the project.

5.4 Research Questions

This research study was reduced to answering five research questions.

1. *What are the shortcomings of the current paper-based system?*

A high priority demand as a sub part for this web application will be the concept of patient's appointments. For the time being there is no such a tool in the Poly-clinic to let the Poly-clinic administration to manage appointments for the referral of patients.

Another demand of patients is submitting their laboratory results via system to the relevant doctor.

2. *Which procedures needed to be fulfilled to implement the HMIS application?*

   Designing and implementing a health information system using the XP Agile software engineering method requires intensive participation by users. The Poly-clinic staff assisted by developers first need to learn the basics of using computer. The first task thus was to train them how to use the system.

   The following objectives were met while developing and designing the HMIS:

   - Collect data and analyse it using interviews and questionnaires.
   - Develop the system to include a database.
   - Develop a user interface.
   - Develop a prototype.
   - Allow medical staff to use the prototype and provide feedback.
   - Improved prototype based on user feedback.

3. The third research question *What are the incoming problems after system implementation?*

   Actually we faced in two main problems after implementing the system:

   1. Identifying and finding errors
   2. Improve some functionality

   Some of the errors appeared after the system implementation. These problems were reported by the users and corrected. On the other hand, the suggestions of end-users also had to be attended to in order to increase the usability and efficiency of the system.

4. *Did the Agile approach produce a health management information system which fulfilled the needs of the clinic?* For implementing the HMIS we used extreme programming (XP). XP is a sub method of Agile methods. XP has multiple steps. Planning is the first step in developing the HMIS. Our planning was based on user stories achieved from a survey and questionnaires. The other steps are design, code and test. The design used UML and the code was written using the XAMPP (test environment) application and the database was done in MySQL.

5. *Does the HMIS fulfill the computing needs of the Poly-clinic?*

   After installing the HMIS the application solved many serious problems of the poly-clinic:

   - There is no need to register patients manually by entering their details into the patient record book.
   - Managing and manipulating patient data can be done efficiently directly on the system.
   - The Poly-clinic registration officer has complete control over patient data and is now able to search for or look up patient data using their ID number.
   - The system can archive and restore the backup data to the system.

   The software used for implementing the HMIS was

   - PHP (Hypertext Preprocessor) web development scripting language,
   - MySQL server,
   - Firefox web browser,
   - An operating system and
   - PHPMyAdmin.

5.5 Other findings

5.5.1 Computers

In 2010 the Poly-clinic had no computers—now it has 8 computers. Based on our recommendations all of the administrative staff and medical doctors are now equipped with computers connected to the Kabul University network.

5.5.2 Internet connection

Initially the Poly-clinic was not connected to the Internet but this has been remedied enabling staff to communicate with one another, the patients, and the Ministry of Health via the Internet. The main resources of our application consist of

- A centralized web/database based on an Apache and MySQL server with a 500 Giga Byte hard disk and 1GB RAM.
- A switch connected to the university network.
- Network cabling—The network wiring was done with category 6 cabling which permits 10-gigabit speeds.
- PCs for for the administration officers and doctors.
- Backup power supply—the resources include a generator and UPS.

The generator and power supply are an essential component of the system because of the erratic power supply in Kabul.

Besides training the users to use and understand the system they were also familiarized with the use of email, spreadsheets, word processing and the Internet. The administrative officer has also been trained to make backups of the system.

5.6 Implications for policy or practice

Reports about the patients can now be created immediately and correctly without delay. Before the advent of the HMIS, when a patient has been referred to the doctor, the doctor had to find the past records of his patient from archive room, but by implementing this system the doctors able to see all past records of patients throughout the system.

The main advantage of the system is that the Poly-clinic staff do not needs write up patient information record books or loose sheets of paper. The paper based-system led to delays, inefficiency and loss of data.

5.7 Further research and development

Since the HMIS is an ongoing project we intend distributing the system to more hospitals and to the Ministry of Health. The system will be expanded to provide for the storage of digital imagery, for example X-rays, E.C.G and ultrasound scans. We plan to include an appointment system whereby the patient can be reminded by SMS of an appointment. Since Pashto is the prevalent language in some provinces we plan to expand the interface to cater for it.

5.8 Conclusion

The thesis started with a software problem that is a health management information system for Kabul University Poly-clinic. Following interviews and questionnaires we determined the requirements to implement an HMIS to manage patient information. Important elements are data storing, data collecting, and data entry. The main concept of the system were formulated by applying a Use Case diagram to describe its main elements. The UML Use Case diagram not only assisted the developer to creating the prototype but was also used to communicate the functionality of the system to the end users. We used Agile method for developing the HMIS, since the Agile method is accountable for the needs of end-user, as well as for rapid deployment of the system.

Bibliography

Abrahamsson, P., Salo, O., Ronkainen, J., and Warsta, J. (2002). Agile software development methods. review and analysis. *Espoo*, 478.

Abrahamsson, P., Warsta, J., Siponen, M. T., and Ronkainen, J. (2003). New directions on agile methods: a comparative analysis. In *Proceedings of the 25th International Conference on Software Engineering*, ICSE '03, pages 244–254, Washington, DC, USA. IEEE Computer Society.

Adams, A. and Cox, A. L. (2008). Questionnaires, in-depth interviews and focus groups. Technical report, Cambridge University Press, Watervliet Ave Albany NY 12206, `www.caresny.organization`.

Ambler, S. W. (2004). *Agile Modeling: Effective Practices for eXtreme Programming*. John Wiley and Sons.

Amudha, T. (2010). Agile-software development methodology. *Advances in Computing Science Tech.*, 3(2):127–135.

Beck, K. and Adams, C. (1999). *Extreme Programming Explained: Embrace Change*. Addison-Wesley, 2nd edition.

Braa, J., Monteiro, E., and Sahay, S. (2004). Networks of action: Sustainable health information systems across developing countries. *MIS Quarterly*, 28(3):337–362.

Carnaghi, B. (2004). What is PHP server side scripting? `www.webpointmopheus.combines`.

Coad, P., Lefebvre, E., and Luca, J. D. (1999). *Java Modeling in Color with UML*. Prentice Hall.

Cockburn, A. (2004). *Crystal Clear: A Human-powered Methodology for Small Teams*. John Wait.

Coffin, R. and Lane, D. (2010). A practical guide to seven Agile methodologies, part 1. `www.devx.com/architect/Article/32761/1954/`.

Conboy, K., Acton, T., and Halonen, R. (2009). Presenting data for team-based decision-making in agile information system projects. In *17th European conference on information system, ECIS '09*, pages 1–12, National University of Ireland.

Cordeiro, L., Barreto, R., Barcelos, R., Oliveira, M., Lucena, V., and Maciel, P. (2007). Agile development methodology for embedded systems. In *Proceedings of the 14th Annual IEEE international conference and workshops on the engineering of computer-based system*. IEEE.

Dawson, S and Manderson, L (1993). A manual for the use of focus groups. Technical report, United Nations University, Boston, MA.

Eidee, P. L. H. (2005). Quantification and traceability of requirement. In *TDT4735 Software Engineering Depth Study Fall 2005*. NTNU,Norwegian University of Science and Technology.

Fowler, M. (2004). *Extreme programming explained*. Addison-Wesley, Boston, Ma.

Fowler, M. and Scott, K. (2000). Uml distilled.

Garrib, A., Stoops, N., McKenzie, A., Dlamini, L., Govender, T., Rohde, J., and Herbst, K. (2008). An evaluation of the district health information system in rural South Africa. *South African Medical Journal*, 98(7):541–552.

Hripcsak, G., Cimino, J. J., and Sengupta, S. (1999). WebCIS: Large scale deployment of a Web-based clinical information system. In *Proceedings of the American Medical Information Association Symposiium, 1999*, pages 804–808.

Hunt, J. (2004). Agile methods and the Agile manifesto.

Igira, F. T., Titlestad, O. H., Lungo, J. H., Makungu, A., Khamis, M. M., Sheikh, Y., Mahundi, M., Ngeni, M. J., Suleimanand, O., and Bra, J. (2005). Designing and implementing hospital management information systems in developing countries: case studies from Tanzania Zanzabir. Technical report, University of Oslo, Ministry of Health Zanzabir, Department of com-

puter science and information technology, state University of Zanzabir, Dar es Salaam, Tanzania.

Jorge, L. D.-H. (2001). Engineering design for software: On defining the software engineering profession. In *31st ASEE/ IEEE Frontiers in Education Conference*. Southem Polytechnic State University, School of Computing and Software Engineerinig/IEEE.

Kimaro, H. C. and Titlestad, O. H. (2005). Challenges of user participation in the design of a computer based system: the possibility of participatory customization in low income countries. *Journal of Health Informatics in Developing Countries*, 2(1):1–9.

Kintu, P., Nanyunja, M., Nzabanita, A., and Magoola, R. (2005). Development of HMIS in poor countries: Uganda as a case study. *Health Policy and Development*, 3(1).

Kuhn, K. and Giuse, D. (2001). *From Hospital Information System to Health Information System*. Institute of Medical Informatics, Philipps-University Marburg, Germany, 2Informatic center, Vanderbilt University Medical Center, Nashville, TN, USA.

Lippeveld, T. and Sauerborn, R. (1992). *A framework for designing health information systems*, chapter 2, pages 15–32. WHO, NY.

Lippeveld, T., Sauerborn, R., and Bodart, C. (2000). Design and implementation of health information systems. Technical report, WHO.

Malhotra, N. K. (2004). Questionnaire design and scale development. `www.scoutbase.org.uk`.

Mendoza, O. and Chong, Y. (2004). Developing health management information system: a practical guide for developing countries. Technical report, World Health Organization.

Ministry of Health and Social Welfare, Uganda (2009). National health management information system strategy and implementation plan. Technical report, Ministry of Health and Social Welfare, Uganda.

Ministry of Health of the Republic of Tajikistan (2009). Assessment of the health information system in Tajikistan.

Moggridge, B. (2007). *People and prototypes*, chapter 10, pages 643–735. MIT Press.

Monteiro, E. (2003). Integrating health information systems. *Methods of Information in Medicine*, 42(4):428–432.

Nathan-Regis, B. N. and Balaji., V. (2012). Evaluation of the most used Agile methods (XP,LEAN,SCRUM). *International Journal of Engineering Science and Technology*, 4(1):23–29.

Nerur, S., Mahapatra, R., and Mangalaraj, G. (2005). Challenges of migrating to Agile methodologies. *CACM*, 48(5):72–78.

Newkirk, J. (2002). Introduction to Agile processes and extreme programming. In *Proceedings of the 24th International Conference on Software Engineering, ICSE'02*, pages 695–696.

Pender, T. (2003). *UML 2 Bible*. Wiley Publishing Inc, USA.

Poppendieck, M. and Poppendieck, T. (2003). *Lean Software Development: An Agile Toolkit*. Addison-Wesley.

Rajanna, D. and Kapila, S. (2009). Health management information systems in the ICTPH health systems model. `www.ictph.org.in`. IKP centre for technologies in public health, Chennai, India.

Sahay, S., Sæbø, J., Molla, S., and Asalefew, A. (2008). Interplay of institutional logics and implications for deinstitutionalization: Case study of HMIS implementation in Tajikistan. Technical report, University of Oslo, Norway.

Samwald, M., Fehre, K., de Bruin, J., and Adlassnig, K.-P. (2012). The Arden syntax standard for clinical decision support: Experiences and directions. *Journal of Biomedical Informatics*, 45:711–718.

Scheaffer, R. L. (1999). Sampling methods and practice. *NCSSM statistics leadership Institute*.

Schwaber, K. and Beedle, M. (2002). *Agile Software Development with Scrum*. Prentice Hall.

Scout Association (2000). Questionnaire, focus groups and interview.

Smith, M., Madon, S., Anifalaje, A., Lazarro-Malecela, M., and Michael, E. (2008). Integrated health information systems in Tanzania. *The Electronic Journal on Information Systems in Developing Countries*, 33(1):1–21.

Stapleton, J. (1999). *DSDM, Dynamic Systems Development Method: The Method in Practice*. Cambridge University Press.

Streveler, D. J. and Sherlock, S. M. (2004). Health management information systems for resource allocation and purchasing in developing countries. Technical report, The International Bank for Reconstruction and Development / The World Bank. A Health Nutrition and Population (HNP) discussion paper.

Thiessen, A. (2008). *HMIS New User Training*. CARES Inc, Watervliet Ave Albany NY 12206, `www.caresny.organization`.

Vandenburg, G. (2005). A simple model of agile software processes – or – Extreme programming annealed. In *Proceedings of the 20th annual ACM SIGPLAN conference on Object-oriented programming, systems, languages, and applications*, pages 539–545.

Wajid, G., Al Zarouni, A., and Al Massoud, H. A. (2002). Information management as a corner stone for improving the quality of healthcare services in Pakistan. In *ICQI 2002—Pakistan 7th International Convention on Quality Improvement, Karachi*, pages 1–7. Department of curative Medicine, Ministry of Health, Abu Dhabi, United Arab Emirates, ICQI.

Welling, L. and Thompson, L. (2009). *Php and MySQL web development*. Sams Publications, 4th edition.

UNIVERSITY of the
WESTERN CAPE

Appendix A

The Results of the Questionnaires

Section A.1 is a transcription of an interview held with Dr Faqiri the head of the Kabul University Poly-clinic. Section A.2 shows the questionnaire we used to poll users after they gained some experience using the prototype of the software. Section A.3 is the questionnaire used in the third survey. Finally Section B shows the original Dari questionnaire used for the third Survey.

A.1 The pilot interview with the Poly-clinic director

The University of Kabul is financed by the Ministry of Higher Education. A public health clinic is also run by the university and is financed by the Ministry of Health. This clinic is located on the campus of Kabul University near to the National Women's Dormitory. This clinic was selected in order to investigate what software it needed and then to implement an application as a masters degree research project. So I visited the Poly-clinic during 10–12 August 2011. I went there to pose some questions to Dr Ahmadullah Faqiri the Poly-clinic director. The script that follows was the pilot interview interview on which we based our initial ideas.

Question: *Could you please give a short introduction to the Kabul University Poly-clinic, and explain what sort of activities are going on here?*

Answer: The Kabul University Poly-clinic is a part of a unified national health system of Afghanistan, which provides necessary curative services. It serves various governmental and university staff and students. We treat day patients or out-patients on a walk-in basis and do not any beds for patient. It is however planned to add some rooms with a few beds in future. Regarding patient information, unfortunately we don't have a management system, we register our patients using forms archive these forms in our archive room.

Question: *Is this a public or private clinic?*

Answer: It is a public and out-patient clinic.

Question: *What are your main medical services?*

Answer: Pharmacy, nursing and treatment of minor ailments.

Question: *How about medicines?*
Answer: We have a pharmacy which caters for various ailments.

Question: *Which kind of patients come to the Poly-clinic?*
Answer: All related universities students and staff may use this Poly-clinic. When a patient visits us here, the patient is examined and we can do various laboratory tests. After diagnosis we prescribe medicine or treatment where it is possible, otherwise we refer the patient to a more comprehensive hospital for further treatment.

Question: *How you are organizing your services to the patients? Are you using any sort of software applications in order to have better services?*
Answer: Unfortunately, we don't have any software application. All our services are paper-based and we have many problems.

Question: *Do you wish to have such software for the Poly-clinic?*
Answer: Of course, it would be a great help for the Poly-clinic, and will solve too many of our problems at the Poly-clinic.

Question: *Actually, I am going to start developing a software solution for handling patients in the Poly-clinic but before I can start developing this application some sort of information are required to be available. How can the Poly-clinic help in delivering this information?*
Answer: Thanks, the Poly-clinic is ready to assist you in getting this information; I and my colleagues will help to you in each step to give you what you need. Later I will give you copies of the patient registration form and the patient prescription form.

Question: *Good, do you reusing the forms if a previous patient returns to the Poly-clinic?*
Answer: I can say no/yes, because we are very busy, and if it's not a serious case we do not reusing that because it becomes too time consuming to search for archived information of a particular patient.

Question: *How many patients are referring to the Poly-clinic daily?*

Answer: Almost about 40 patients per each day.

Question: *What sort of software would you think be suitable for managing the patient information?*
Answer: We really wish to have a kind of software system in the Poly-clinic that could be able to store our patients' information.

Question: *What kinds of information do you keep in forms and folders?*
Answer: Well, mostly it's about the patient's identification, and also we are registering the type of disease, then we add the form in a folder.

Question: *Do you also keep the reports of treatment for each patient?*
Answer: Yeah sure, each doctor is responsible to write a summary report for the current situation of the patient, and the kinds of medication that he is prescribing.

Question: *Before you said we are not using the patient's information, then what you do if a patient comes for second period of treatment?*
Answer: Well, if he has a new type of illness, or the patient illness is not too seriously ill, we simply made a new folder. This is problematic for us, because we have to keep redundant folders of a certain patient.

Question: *Are you charging the patients for treatment?*
Answer: It's before I said Kabul University Poly-clinic is a governmental Poly-clinic related to the Higher Education Ministry. This Poly-clinic gives free treatment and medication for all related governmental university students and staff.

Question: *OK, Any other question left?*
Answer: Not for the time being, but I am sure we will have further demands and question, I will inform when they come to my mind.

Thanks for your time.

A.2 The second survey findings

A.2.1 Survey 2 Part 1—Background information

Table A.1: Survey 2 Part 1 Question 1: Position in poly-clinic

1. What is your position?		
Response	N	%
Doctor	9	45%
Nurse	9	45%
Administrative Officer	2	10%
Total	20	100%

Table A.2: Survey 2 Part 1 Question 2: Work experience

2. How long have you been employed in this poly-clinic?		
Response	N	%
1-10 years	11	55%
11-20 years	6	30%
21 or more years	3	15%
Total	20	100%

Table A.3: Survey 2 Part 1 Question 3: Gender

3. What is your gender?		
Response	N	%
Male	17	85%
Female	3	15%
Total	20	100%

Table A.4: Survey 2 Part 1 Question 4: Computers

4. Does your poly-clinic have computers?		
Response	N	%
Yes	6	30%
No	14	70%
Total	20	100%

Table A.5: Survey 2 Part 1 Question 5: Connected to the Internet

5. Does your poly-clinic have an Internet connection?		
Response	N	%
Yes	0	0%
No	20	100%
Total	20	100%

Table A.6: Survey 2 Part 1 Question 6: Own cellphone

6. Do you own a cell phone?		
Response	N	%
Yes	20	100%
No	0	0%
Total	20	100%

Table A.7: Survey 2 Part 1 Question 7: Have own computer

7. Do you have a computer at home?		
Response	N	%
Yes	15	75%
No	5	25%
Total	20	100%

Table A.8: Survey 2 Part 1 Question 8: Computer experience

8. How often have you used a computer before?		
Response	N	%
Many times	4	30%
Several times	6	70%
Few times	9	45%
Never	1	5%
Total	20	100%

Table A.9: Survey 2 Part 1 Question 9: Used E-mail

9. How often have you written E-mails?		
Response	N	%
Many times	1	5%
Several times	2	10%
Few times	5	25%
Never	12	60%
Total	20	100%

A.2.2 Survey 2 Part 2—Computer knowledge of users

Table A.10: Survey 2 Part 2 Question 1: Computer knowledge

1. Microsoft Windows		
Response	N	%
Yes	15	75%
No	4	20%
Use another program	1	5%
Total	20	100%

Table A.11: Survey 2 Part 2 Question 2: Used word processor

2. Microsoft Word		
Response	N	%
Yes	15	75%
No	4	20%
Use another program	1	5%
Total	20	100%

Table A.12: Survey 2 Part 2 Question 3: Used spreadsheet

3. Microsoft Excel		
Response	N	%
Yes	9	45%
No	10	50%
Use another program	1	5%
Total	20	100%

Table A.13: Survey 2 Part 2 Question 4: Used database

4. Microsoft Access		
Response	N	%
Yes	4	20%
No	16	80%
Use another program	0	0%
Total	20	100%

Table A.14: Survey 2 Part 2 Question 5: Used Internet

5. Internet Explorer		
Response	N	%
Yes	8	40%
No	12	60%
Use another program	0	0%
Total	20	100%

A.2.3 Survey 2 Part 3—What to store in the HMIS

Table A.15: Survey 2 Part 3 Question 1: What information should be stored in the HMIS

1. What type of information would you like a HMIS to present for you?		
Response	N	%
Information about patients	11	55%
Information about Doctors	0	0%
Both information about patients and doctors	9	45%
Total	20	100%

Table A.16: Survey 2 Part 3 Question 2: Which tools are necessary for the HMIS

2. What type of tools are necessary for the HMIS?		
Response	N	%
Easy for use	3	15%
More attractive in shape	0	0%
Information both A & B	17	85%
Total	20	100%

Table A.17: Survey 2 Part 3 Question 3: Language of HMIS

3. Do you prefer the HMIS in English or local language?		
Response	N	%
In English	10	50%
In local language	10	50%
Total	20	100%

Table A.18: Survey 2 Part 3 Question 4: Database functionality

4. What do you think about the database functionality?		
Response	N	%
All functions are present	18	90%
Functionality incomplete	2	10%
Total	20	100%

Table A.19: Survey 2 Part 3 Question 5: Database fields

5. What do you think of the database's fields? (Name, Father name, Age, Position, Faculty, Contact No)		
Response	N	%
Adequate	13	65%
Inadequate	7	35%
Total	20	100%

Table A.20: Survey 2 Part 3 Question 6: Save button

6. What do you think about the function of the save button?		
Response	N	%
It works well	19	95%
It does not work well	1	5%
Total	20	100%

Table A.21: Survey 2 Part 3 Question 7: Functionality of doctor page

7. What do you think about the function of doctor page?		
Response	N	%
All functions needed are present	13	65%
Additional functions needed	7	35%
Total	20	100%

Table A.22: Survey 2 Part 3 Question 8: Login button

8. What do you think about the function of the login button?		
Response	N	%
It works well	20	100%
It does not work well	0	0%
Total	20	100%

Table A.23: Survey 2 Part 3 Question 9: Submit button

9. What do you think about the function of the submit button?		
Response	N	%
It works well	20	100%
It does not work well	0	0%
Total	20	100%

Table A.24: Survey 2 Part 3 Question 10: Color of homepage

10. Do you like the color of interface homepage?		
Response	N	%
Yes	17	85%
No	3	15%
Total	20	100%

Table A.25: Survey 2 Part 3 Question 11: Background color

11. A good color for the prototype background is:		
Response	N	%
Blue	17	85%
Gray	3	15%
Yellow	0	0%
Total	20	100%

Table A.26: Survey 2 Part 3 Question 12: Buttons

12. Buttons should be:		
Response	N	%
Small	0	0%
Medium	18	90%
Large	2	10%
Total	20	100%

Table A.27: Survey 2 Part 3 Question 13: Navigation—exit

13. Is it easy to exit the program:		
Response	N	%
Yes	19	95%
No	1	5%
Total	20	100%

Table A.28: Survey 2 Part 3 Question 14: Navigation between pages

14. Is it easy to move between pages:		
Response	N	%
Yes	19	95%
No	1	5%
Total	20	100%

A.3 Third survey findings

A.3.1 Survey 3 Part 1—Background information

Table A.29: Survey 3 Part 1 Question 1: Position in poly-clinic

1. What is your position?		
Response	N	%
doctor	9	45%
Nurse	9	45%
Administrative Officer	2	10%
Total	20	100%

Table A.30: Survey 3 Part 1 Question 2: Work experience

2. How long have you been employed in this poly-clinic?		
Response	N	%
1–10	11	55%
11–20	6	30%
21 or more	3	15%
Total	20	100%

Table A.31: Survey 3 Part 1 Question 3: Gender

3. What is your gender?		
Response	N	%
Male	17	85%
Female	3	15%
		%
Total	20	100%

Table A.32: Survey 3 Part 1 Question 4: Age

4. How old are you?		
Response	N	%
20–25	0	0%
26–35	7	35%
36 or more	13	65%
Total	20	100%

Table A.33: Survey 3 Part 1 Question 5: Does poly-clinic have computers?

5. Does your poly-clinic have computers?		
Response	N	%
Yes	6	30%
No	14	70%
Total	20	100%

Table A.34: Survey 3 Part 1 Question 6: Does poly-clinic have access to the Internet

6. Does your poly-clinic have an Internet connection?		
Response	N	%
Yes	0	0%
No	20	100%
Total	20	100%

Table A.35: Survey 3 Part 1 Question 7: Own cell phone

7. Do you own a cell phone?		
Response	N	%
Yes	20	100%
No	0	0%
Total	20	100%

Table A.36: Survey 3 Part 1 Question 8: Have home computer

8. Do you have a computer at home?		
Response	N	%
Yes	15	75%
No	5	25%
Total	20	100%

Table A.37: Survey 3 Part 1 Question 9: Computer experience

9. How often have you used a computer before?		
Response	N	%
Many times	3	15%
Several times	5	25%
Few times	5	25%
Never	7	35%
Total	20	100%

Table A.38: Survey 3 Part 1 Question 10: Have you written E-mails

10. How often have you written E-mails?		
Response	N	%
Many times	0	0%
Several times	2	10%
Few times	5	25%
Never	13	65%
Total	20	100%

A.3.2 Survey 3 Part 2—Computer knowledge of users

Table A.39: Survey 3 Part 2 Question 1: Computer knowledge

1. Microsoft Windows		
Response	N	%
Yes	15	75%
No	4	20%
Use another program	1	5%
Total	20	100%

Table A.40: Survey 3 Part 2 Question 2: Used a word processor

2. Microsoft Word		
Response	N	%
Yes	16	80%
No	3	15%
Use another program	1	5%
Total	20	100%

Table A.41: Survey 3 Part 2 Question 3: Used a spreadsheet

3. Microsoft Excel		
Response	N	%
Yes	9	45%
No	10	50%
Use another program	1	5%
Total	20	100%

Table A.42: Survey 3 Part 2 Question 4: Used a database

4. Microsoft Access		
Response	N	%
Yes	4	20%
No	16	80%
Use another program	0	0%
Total	20	100%

Table A.43: Survey 3 Part 2 Question 5: Used Internet

5. Internet Explorer		
Response	N	%
Yes	8	40%
No	12	60%
Use another program	0	0%
Total	20	100%

A.3.3 Survey 3 Part 3—Some information about the kind of HMIS

Table A.44: Survey 3 Part 3 Question 1: HMIS in English or Dari

1. Do you prefer the HMIS in English or local language?		
Response	N	%
In English	4	20%
In local language	16	80%
Total	20	100%

Table A.45: Survey 3 Part 3 Question 2: Tools needed in HMIS

2. What type of tools is necessary for the HMIS?		
Response	N	%
Easy for use	11	55%
More attractive in shape	0	0%
Information both A & B	9	45%
Total	20	100%

Table A.46: Survey 3 Part 3 Question 3: Functionality of database

3. What do you think about the database functionality?		
Response	N	%
All functions are present	18	90%
Functionality incomplete	2	10%
Total	20	100%

Table A.47: Survey 3 Part 3 Question 4: Tools are user friendly

4. Are the tools user friendly?		
Response	N	%
Yes	19	95%
No	1	5%
Total	20	100%

Table A.48: Survey 3 Part 3 Question 5: Database fields

5. What do you think about the database fields? (Name, Father name, Age, Position, Faculty, Contact No)		
Response	N	%
Adequate	19	95%
Inadequate	1	5%
Total	20	100%

Table A.49: Survey 3 Part 3 Question 6: Save button

6. What do you think about the function of the save button?		
Response	N	%
It works well	19	95%
It does not work well	1	5%
Total	20	100%

Table A.50: Survey 3 Part 3 Question 7: Search button

7. What do you think about the function of search button?		
Response	N	%
All functions needed are present	18	90%
Additional functions needed	2	10%
Total	20	100%

Table A.51: Survey 3 Part 3 Question 8: Login button

8. What do you think about the function of the login button?		
Response	N	%
It works well	20	100%
It does not work well	0	0%
Total	20	100%

Table A.52: Survey 3 Part 3 Question 9: Submit button

9. What do you think about the function of the submit button?		
Response	N	%
It works well	20	100%
It does not work well	0	0%
Total	20	100%

Table A.53: Survey 3 Part 3 Question 10: Color of background

10. A good color for the prototype background is:		
Response	N	%
Pale green	17	85%
Gray	3	15%
Yellow	0	0%
Total	20	100%

Table A.54: Survey 3 Part 3 Question 11: Color of homepage

11. Do you like the color of interface homepage?		
Response	N	%
Yes	19	95%
No	1	5%
		%
Total	20	100%

Table A.55: Survey 3 Part 3 Question 12: Size of buttons

12. Buttons should be:		
Response	N	%
Small	0	0%
Medium	18	90%
Large	2	10%
Total	20	100%

Table A.56: Survey 3 Part 3 Question 13: Exit

13. Is it easy to exit the program:		
Response	N	%
Yes	19	95%
No	1	5%
Total	20	100%

Table A.57: Survey 3 Part 3 Question 14: Movement between pages

14. Is it easy to move between pages:		
Response	N	%
Yes	19	95%
No	1	5%
Total	20	100%

UNIVERSITY of the
WESTERN CAPE

Appendix B

Third Survey in Dari

For the interest of the reader we have also included the Third Survey in its original form in Dari.

سوالنامه

سوالنامه جهت جمع آوری معلومات برای پروژه ماستری
اشتراک کننده گان عزیز و گرامی!
اظهار تشکر و امتنان از اینکه این سوالنامه را جواب میدهید. این سوالنامه برای انکشاف نمونه دوم نرم افزاری که برای آمریت پولی کلینیک صحی ایجاد شده است، میباشد. بعد از نظرات شما تغیرات ضروری در نمونه بعدی اصلاح خواهد شد. در حقیقت این یک وسیله ایست برای پروژه های تحقیقی. جوابات شما برای بهبود نمونه بعدی قابل ملاحظه خواهد بود.
لطفآ جواب سوالات را در سوالنامه هذا حلقه نمایید.

1. وظیفه شما چیست؟

کارمند اداری	معاون داکتر	داکتر
C	B	A

2. از چه مدتی در این پولی کلینیک ایفای وظیفه مینمایید؟

1 – 10	A
11 – 20	B
یا زیادتر 21	C

مرد	A
زن	B

3. جنس شما چیست ؟

4. چند ساله هستید؟

20 – 25	A
26 – 35	B
یا بزرگتر 36	C

5. آیا پولی کلینیک شما برای اجرای امور یومیه کمپیوتر دارد؟

بلی	A
نخیر	B

6. ایا پولی کلینیک شما انترنیت دارد؟

بلی	A
نخیر	B

بلی	A
نخیر	B

7. آیا شما تیلیفون موبایل دارید؟

8. آیا در خانه کمپیوتر دارید؟

بلی	A
نخیر	B

9. چند مرتبه قبلاً کمپیوتر را استفاده نموده اید؟

هیچگاه	چند مرتبه	چندین مرتبه	بیشترین اوقات
D	C	B	A

		پروگرام ها
A	B	C
A	B	C
A	B	C
A	B	C
A	B	C

11. کدامیک از پروگرام های ذیل را قبلاً استفاده نموده اید؟

12. مایکروسافت ویندوز

13. مایکروسافت ورد

14. مایکروسافت اکسل

15. مایکروسافت اکسس

16. انترنیت اکسپلورر

17. یک رنگ خوب برای پس زمینه (Background) مودل اولیه عبارت است از:

A	آبی
B	خاکی
C	زرد

18. دکمه ها باید :

A	کوچک
B	متوسط
C	بزرگ

19. کدام قسمت از سافت ویر برای شما آسان است و کدام قسمت از آن برای شما آسان نیست؟

.....

20. کدام سایز از حروف برای مودل اولیه مناسب است؟

.....

21. کدام رنگ برای حروف مودل اولیه مناسب است؟
.....
.....

22. چه نوع تغیرات دیگر را برای استفادهٔ آسان این سافت ویر پیشنهاد مینمایید؟
.....
.....
.....

23. کدام نوع معلومات را ضرورت دارید تا درسیستم اپدیت و یا پاک شود؟
.....
.....

24. آیا برای داخل شدن به سیستم یک نام کاربری و رمز کفایت میکند ویا برای هر استفاده کننده جدا باشد؟
.....
.....

25. کدام قسمت سافت ویر برای شما خوب نیست؟
.....
.....

26. . چه نوع معلومات را میخواهید که سیستم برای تان پیشکش نماید؟

A	معلومات در باره مریضان
B	معلومات در باره داکتران
C	معلومات در باره داکتران و مریضان

27. کدام نوع ابزار برای سیستم معلوماتی صحی ضرورت است؟

A	آسان برای استفاده
B	بسیار جاذب در شکل
C	هر دو A و B

28. آیا سیستم به لسان دری باشد و یا به انگلیسی؟

A	انگلیسی
B	دری

29. ایا خارج شدن از پروگرام آسان است؟

27. کدام نوع ابزار برای سیستم معلوماتی صحی ضرورت است؟

A	آسان برای استفاده
B	بسیار جاذب در شکل
C	هر دو A و B

28. آیا سیستم به لسان دری باشد و یا به انگلیسی؟

A	انگلیسی
B	دری

29. ایا خارج شدن از پروگرام آسان است؟

A	بلی
B	نخیر

30. آیا حرکت کردن بین صفحات آسان است؟

A	بلی
B	نخیر

31. در باره عملکرد های دیتابیس چی فکر میکنید؟

A	تمامی عملکرد ها آماده هستند
B	عملکرد ها کامل نیستند

32. ایا این سافت ویر برای استفاده کننده گان آسان و دوستانه است؟

A	بلی
B	نخیر

33. در باره فیلد های database (ID، نام ، نام پدر، عمر، وظیفه ، پوهنحی ، نمبر تیلیفون)چی فکر میکنید؟ اگر جواب B باشد لطفاً درقسمت خالیگاه پایین بنویسید.

A	کافی است
B	کافی نیست

38. آیا رفتن از یک صفحه سیستم به صفحه دیگر سیستم برای شما اسان است؟

A	بلی
B	نخیر

39. آیا رنگ صفحه اصلی (Homepage) سیستم (Interface) مورد پسند شما است؟

A	بلی
B	نخیر

40. اگرکدام یادداشت و یا تبصره دیگر برای بهبود مودل اولیه سیستم معلوماتی صحی ضرورت باشد، لطفاً در ذیل توضیح نمایید.

.....
.....
.....
.....
.....
.....
.....
.....
.....
.....