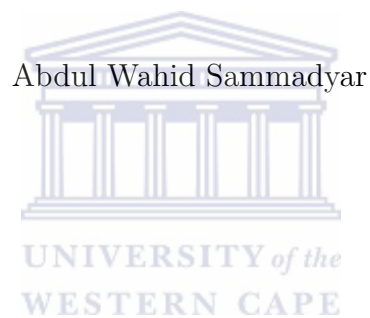


An Investigation into the Appropriateness of Using Agile Processes to Build an Educational Management Information System



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 2010

Declaration



I, ABDUL WAHID SAMMADYAR, declare that this thesis “*An Investigation into the Appropriateness of Using Agile Processes to Build an Educational Management Information System*” 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:

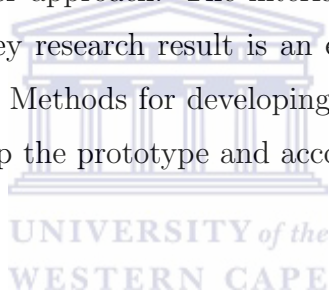
Date:

ABDUL WAHID SAMMADYAR.

Abstract

Since there was a need for an Educational Management Information System (EMIS) in the Ministry of Education in Afghanistan, we designed and implemented a prototype for use in the ministry and investigated the appropriateness of using Agile methods for producing the EMIS. The prototype consisted of a database containing data about schools and a Dari interface which was used by educational planners, managers and policy makers of the ministry for decision making and planning.

Agile methods are relatively new in software engineering and have an approach and development guidelines which strive for user satisfaction and early incremental delivery of software. We adapt them to local conditions due to their collaborative client-developer approach. The interface, a key component, is easy to use and efficient. The key research result is an evaluation concerning the appropriateness of using Agile Methods for developing the EMIS. Focus groups and surveys were used to develop the prototype and accomplish the study.



Key words

Visual programming

Agile Methods/Processes

Design

Education Management Information System (EMIS)

Software Engineering

Unified Modeling Language (UML)

Interface

Database

Ministry of Education

Schools

Afghanistan

District

Extreme Programming

Dari

Java Language

Use case

User Story





UNIVERSITY *of the*
WESTERN CAPE

Acknowledgment

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 for supporting and encouraging me during my Masters study. Without his help, this work would not have been possible. At this time I would like to extend a very special thanks to Prof. I.M. Venter and Verna Connan, 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

Declaration	iii
Abstract	v
Key words	vii
Acknowledgment	ix
List of Tables	xvii
List of Figures	xxi
Glossary	xxiii
1. Introduction	1
1.1 Background	1
1.2 The Role of an EMIS	2
1.3 The Software Engineering Process Model for Agile Methods	2
1.4 Main components of an EMIS	3
1.4.1 Data	3
1.4.2 Data processes	3
1.4.3 Data collection	3
1.4.4 Data analysis	3
1.5 EMIS in other Developing Countries	3
1.5.1 Zimbabwe	3
1.5.2 Namibia	4
1.5.3 Pakistan	4
1.5.4 India	5
1.6 Aim of the study	6
1.7 Research Question	6
1.8 Research Design	6
1.9 Chapter outline	7
2. Literature Review	9
2.1 Introduction	9
2.2 What is an information system?	10
2.3 Purpose of EMIS	10

2.4	EMIS Definition and Development	10
2.5	The data and information flow of EMIS	11
2.6	Agile Methods	11
2.7	What are Agile processes?	12
2.8	Different Agile Methods	12
2.9	Agile method techniques	12
2.10	Extreme Programming	12
2.11	Using Agile methods	13
2.12	The Agile method process	13
2.13	Adapting the Agile process to this project	14
2.14	Object oriented approaches in Agile methods	14
2.15	EMIS Project Development in Agile Methods	15
2.16	EMIS in Pakistan	15
2.16.1	Monitoring and Evaluation	15
2.16.2	District level of EMIS	16
2.17	Zimbabwe	16
2.18	India	16
2.19	Namibia	17
2.20	Unified Modeling Language (UML) and Use cases	17
2.21	Software process models	18
2.22	Rapid Prototyping and Joint Application Development	18
2.23	Extreme Programming Planning Game	19
2.24	XP Sample Design and Refactoring	19
2.25	XP Tests	19
2.26	Pair Programming	20
2.27	Extreme Programming	20
2.28	Data analysis	20
2.29	Conclusion	20
3.	Research design and methodology	23
3.1	Introduction	23
3.2	Research Hypothesis	23
3.2.1	Research Questions	23
3.3	Research Instruments	24
3.3.1	Sample design and sample techniques	24

3.3.2	Questionnaire	24
3.3.3	Interviews and Focus Groups	24
3.3.4	Data collection through Focus Groups	24
3.3.5	User story	25
3.4	Agile Methods	25
3.5	Development of the EMIS	25
3.5.1	Using Agile method	25
3.5.2	Using Unified Modeling Language diagram	26
3.5.3	Using Net beans software and Java language for design interface	27
3.5.4	Using Microsoft Access for development database	27
3.6	Research prototype	27
3.6.1	The first part of this prototype provides functionality	27
3.6.2	Interface	28
3.7	Prototype Development	30
3.7.1	The first version	30
3.7.2	The second version prototype	33
3.7.3	The third version of the prototype	34
3.8	EMIS Window in Dari	34
3.8.1	Differences between second and third version of prototype	35
3.9	Data used by the EMIS	36
3.10	Data collection	37
3.10.1	First survey	37
3.10.2	Second survey	37
3.10.3	Third survey	37
3.11	Data editing	38
3.11.1	The data coding	38
3.12	Rationale behind the selection of data analysis procedure	38
3.13	Conclusion	39
4.	Results: Presentation and Discussion	41
4.1	Report and Results	41
4.1.1	First focus group summary	41
4.1.2	Second focus group summary	41
4.1.3	Third focus group summary	42
4.1.4	The first focus groups code	42
4.2	Second focus group code	45
4.2.1	Part 1 Background information	45

4.2.2	Part 2—Some information about work field	51
4.2.3	Part 3—Some information about kind of EMIS 52	
4.3	The third focus groups code	52
4.3.1	Part 1—Some background information	52
4.4	Part 2	56
4.5	Summary of the main results	56
4.6	Comparing the Research Findings/Results with the Hypothesis . . .	60
4.7	Main trends and patterns	61
4.8	Summary	61
5.	Conclusion and recommendations	63
5.1	Introduction	63
5.2	Overview	63
5.3	Research findings	64
5.3.1	The data and information flow in the MoE	64
5.3.2	Data Collection	65
5.3.3	Storing the information	65
5.3.4	Extreme programming	65
5.4	Research questions	65
5.4.1	The first research question	65
5.4.2	The second research question	66
5.4.3	The third research question	66
5.5	Other findings	67
5.5.1	Computers	67
5.5.2	Internet Connection	67
5.6	Further research	67
5.7	Conclusion	68
A.	School Education in Afghanistan	69
A.1	Afghanistan Schools Survey 2007	69
A.2	Students—Formal Education	69
A.3	Students—Non-Formal Education	69
A.4	School Survey 2007, National Level Summaries Schools and Enrollment	70
A.5	Teachers Summary	70

B. The questionnaires	73
B.1 First survey	73
B.2 Second survey	74
B.2.1 Questionnaire 2	74
B.3 The third survey	76
C. The Java code for the EMIS	77





UNIVERSITY *of the*
WESTERN CAPE

List of Tables

3.1	Data coding of the first three survey questions	39
4.1	Survey 1 Part 1 Question 1: Name of department	42
4.2	Survey 1 Part 1 Question 2: Position in department	43
4.3	Survey 1 Part 1 Question 3: Working experience	43
4.4	Survey 1 Part 1 Question 4: Have used Microsoft Access	43
4.5	Survey 1 Part 2 Question 1: Know a programming language	44
4.6	Survey 1 Part 2 Question 2: Have used SQL	44
4.7	Survey 1 Part 2 Question 3: Have used Microsoft Office	44
4.8	Survey 1 Part 3 Question 1:	45
4.9	Survey 1 Part 3 Question 2: Type of interface needed for EMIS	45
4.10	Survey 1 Part 3 Question 3: Type of data for database	45
4.11	Survey 1 Part 3 Question 4: Can interface be in English	46
4.12	Survey 1 Part 3 Question 5: Current difficulties within education system	46
4.13	Survey 1 Part 3 Question 6: Data analysis to improve educational services	46
4.14	Survey 1 Part 3 Question 7: Does data flow from schools up or down .	46
4.15	Survey 1 Part 3 Question 8: Structure of the database	47
4.16	Survey 1 Part 3 Question 9: How should information be collected? . .	47
4.17	Survey 1 Part 3 Question 10: Collection of information	47
4.18	Survey 1 Part 3 Question 11: Schools have computers	47
4.19	Survey 1 Part 3 Question 12: Internet connection to provinces	47
4.20	Survey 1 Part 3 Question 13: How will data be collected	48
4.21	Survey 1 Part 3 Question 14: Software and hardware needed	48
4.22	Survey 1 Part 3 Question 15: Schools' data used in national Statistics	48
4.23	Survey 1 Part 3 Question 16: Who has used the data?	48
4.24	Survey 1 Part 3 Question 17: Who should analyze the data?	49

4.25	Survey 2 Part 1 Question 1: Name of department	49
4.26	Survey 2 Part 1 Question 2: Position in department	50
4.27	Survey 2 Part 1 Question 3: Average years of experience	50
4.28	Survey 2 Part 1 Question 4: Use Microsoft Access	50
4.29	Survey 2 Part 2 Question 1: Can program in Java	51
4.30	Survey 2 Part 2 Question 2: Use Access and SQL server	51
4.31	Survey 2 Part 2 Question 3: Knowledge of software engineering	51
4.32	Survey 2 Part 3 Question 1:	52
4.33	Survey 2 Part 3 Question 2: Sorting function in EMIS tool	52
4.34	Survey 2 Part 3 Question 3: User-friendliness of interface	52
4.35	Survey 2 Part 3 Question 4: Is English interface better?	53
4.36	Survey 2 Part 3 Question 5: Which data is useful?	53
4.37	Survey 2 Part 3 Question 6: The functionality of the Add button	53
4.38	Survey 2 Part 3 Question 7: The Delete window	53
4.39	Survey 2 Part 3 Question 8: The Delete button	53
4.40	Survey 2 Part 3 Question 9: Suitability of Search window	54
4.41	Survey 2 Part 3 Question 10: Connection between interface and database	54
4.42	Survey 2 Part 3 Question 11: Language of interface	54
4.43	Survey 2 Part 3 Question 12:	54
4.44	Survey 2 Part 3 Question 13:	55
4.45	Survey 3 Part 1 Question 1: Name of department	55
4.46	Survey 3 Part 1 Question 2: Position in department	56
4.47	Survey 3 Part 1 Question 3: Departmental experience	56
4.48	Survey 3 Part 2 Question 1: Used SQL server	56
4.49	Survey 3 Part 2 Question 1: Able to use Net Beans	57
4.50	Survey 3 Part 2 Question 2: Able to create a Use Case Diagram	57
4.51	Survey 3 Part 2 Question 3: Know about XP	57
4.52	Survey 3 Part 3 Question 1: Like database application	57

4.53	Survey 3 Part 3 Question 2: Like sort button	57
4.54	Survey 3 Part 3 Question 2: Interface is user friendly	58
4.55	Survey 3 Part 3 Question 4: Prefer English or Dari interface	58
4.56	Survey 3 Part 3 Question 5: Opinion of add window function	58
4.57	Survey 3 Part 3 Question 6: Opinion of sort window	58
4.58	Survey 3 Part 3 Question 7: Delete button okay	58
4.59	Survey 3 Part 3 Question 8: Show window okay	59
4.60	Survey 3 Part 3 Question 9: Opinion of sort window	59
4.61	Survey 3 Part 3 Question 10: Like sort button	59
4.62	Survey 3 Part 3 Question 11: Shortcomings of Dari interface	60
A.1	Schools and student enrollment	70
A.2	Teaching and other staff employed by the Ministry of Education	70





UNIVERSITY *of the*
WESTERN CAPE

List of Figures

2.1	The data and information flow in EMIS	11
3.1	Use case diagram 1	25
3.2	Use case diagram 2	26
3.3	Use case diagram 3	26
3.4	Prototype interface	29
3.5	Prototype database	29
3.6	Add Entry Form	30
3.7	Search Entry form	31
3.8	Report window	31
3.9	Delete Form	32
3.10	First version of sort window	32
3.11	Sort Window in second version prototype	33
3.12	Search Window	33
3.13	EMIS Window in Dari	34
3.14	Database	35
3.15	Add button window in Dari	35
3.16	Delete button window in Dari	36
3.17	Sort button window in Dari	36
3.18	Search button window in Dari	36
A.1	Ministry of Education structure	70
A.2	Formal vs. non-formal education in Afghanistan	71
A.3	Deployment of teachers in the Ministry of Education	71



UNIVERSITY *of the*
WESTERN CAPE

Glossary

EMIS	Education Management Information System
UML	Unified Modeling Language
XP	Extreme Programming
Dari	Local language in Afghanistan
Java	Java programming language





UNIVERSITY *of the*
WESTERN CAPE

Chapter 1

Introduction

1.1 Background

An Educational Management Information System (EMIS) is a system for the collection, integration, processing, maintenance and dissemination of data and information to support decision making, policy analysis and formulation, planning, monitoring and management at all levels of an education system (Ajayi and Omirin, 2007). Efforts to improve the quality of the data and information available to describe education systems and support decision-making have been under continuous improvement over the years. The development of computer-based data and information management systems began in the mid 1980s with projects in many countries to computerize the annual school census conducted in many countries (Wako, 2003). In addition to this were efforts to computerize many of the routine administrative and management functions in ministries of education and in related adjunct bodies, e.g., examination bodies. There are efforts to develop more comprehensive, integrated computer-based education management information systems to support the monitoring and evaluation of education system performance and the crafting of effective education policies. This was started in some countries as early as in the late 1980s. Efforts to strengthen and extend all of these systems continued throughout the 1990s and into the new millennium. The development of EMIS is an explicit objective in many of these projects, but many also continue to include objectives to build or strengthen computer-based systems to meet the routine administrative and management needs of a specific division or department, in regional, district and /or municipal offices. The Ministry of Education in Afghanistan (MoE) does not have an EMIS.

1.2 The Role of an EMIS

In Afghanistan since the MoE is not computerized all information and statistics about schools is recorded using a paper-based manual system. The aim of the EMIS is for managing information and statistics about schools. An important goal of this system is to build a central repository for the collection of schools data for the MoE. The database will be used administer all the primary and secondary schools of a given country. Data concerning teachers and learners at all public schools are the major source of information for the database. The MoE is responsible for the policy planning and decision making used for Educational development based on this information. Information provision and statistics is the main component of this system. Educational planners need information about the number of primary schools, secondary schools, high schools, school names, numbers of teachers, number of learners and schools district. Access to data needs to be comprehensive so the interface must provide all the necessary functions needed by the users.

1.3 The Software Engineering Process Model for Agile Methods

The primary purpose of software is that of information transformer. Software is used to produce, manage, acquire, modify, display, and transmit information anywhere in the world. The days of the lone programmer are gone because modern software applications are developed by teams of software specialists. Yet, the software developer's concerns have remained the same. Why does software take so long to complete? Why does it cost so much to produce? Why can't all errors be fixed before software is delivered to the customer? Software engineering is a disciplined approach using processes, methods, and tools to develop quality software (Cohen et al., 2004). Software engineering now provides a new method for development called the Agile method (Pressman, 2001). As a process model for the development of an EMIS, the Agile approach method was utilized. The Agile method outlines the steps for Extreme Programming (XP) (Boehm, 2002), which in itself is a set of practices, implemented using Java Studio Enterprise and UML to capture, model the requirements and build the application.

1.4 Main components of an EMIS

The main components of an EMIS are:

1.4.1 Data

For data, schools are the core source. The Ministry of Education collects information from schools. For this purpose, there is a need to design a database. Typical data to be stored are school names, numbers of teachers, number of learners, schools district, etc.

1.4.2 Data processes

Access to data needs to be easy for the system users, for this purpose, an interface must be designed. This interface must easily provide all the necessary functions needed by the users. This interface must be friendly to the end users and end user training should be provided (Carrizo et al., 2003).

1.4.3 Data collection

If the system of records management at school level is poor then we cannot expect to get the results we require from the data collection system. The school remains the core source of most of the required data items, which are not often recorded in the desired manner. It is the responsibility of the EMIS manager to see to it that the records at school level are properly maintained (Ellison, 2004).

1.4.4 Data analysis

Until we have processed our data, we have little or nothing to analyze. Data analysis involves looking more closely at the data and in various ways, in order to extract information useful for planning and decision making. The analysis is done to provide information to planners, decision makers, researchers, policy makers and other users so that they can establish whether their actions are appropriate or whether the action need to be modified (Nijkamp, 2009).

1.5 EMIS in other Developing Countries

1.5.1 Zimbabwe

In Zimbabwe, the Ministry of Education created an EMIS responsible for the collection and use of information for policy planning/implementation and decision

making (Lloyd and Hewett, 2003). Zimbabwe's EMIS centre is located at the Ministry of Education in Harare, Zimbabwe. This centre is divided into district and provincial levels. The district level liaises with schools and gives quick feedback. Other roles of the district include information relay, guidelines, and collation of reports from the centre to schools/receiving information from the schools. The user of the information is central to the function and management of the EMIS. There are internal and external users of educational information. Internal users are planners and decision makers, while external users are other government and non-government institutions. Planners need information about the number of existing primary schools in urban and rural areas, the distribution of enrolment in each school by sex, and the number of children per class (Santiago Cueto, 2005).

1.5.2 Namibia

A small group of Namibian staff members from the EMIS department of the Ministry of Education Development have implemented an educational information system (Brunette, 2006). They also developed the Namibian EMIS database. This database contains information about schools. A user that needs information about students, accesses information from this centre. The data processing system comprises of various components: such as computer hardware and software, data entry and verification systems, data processing systems generating desired outputs, and systems, giving users access to information. After the implementation of the design elements mentioned above, data is collected, and information is then accessible to users. The Namibian EMIS produced the following results (Voigts, 2005):

The creation of an education statistical database meeting a significant number of the information needs of the users; the consistent collection of education statistics in two annual surveys; the existence of will maintained database of educational data and the capacity to respond to ad-hoc requests for statistical information.

1.5.3 Pakistan

In Pakistan, the Ministry of Education has created an Educational Management Information System (Thomas et al., 1993). In the Ministry of Education, the system is used for managing sector policy and planning. The Ministry of Education has overall responsibility for formulation of education sector policy, planning and development of curriculum and delivery of education and implementation of plans

devolved to provincial and district levels. The main role of provincial education authorities is the coordination, support and provision of educational services and implementation of education plans at the district level. The provincial Education Departments are headed by their respective Provincial Education Ministers. The provinces are further divided into districts for the purpose of administration. The Educational Management Information System is divided over several levels viz. Federal Level, Provincial Level, District Level and Institutional (School) Level. The Federal level is part of the National Educational Management Information System (NEMIS) project. The Academy of Educational Planning and Management carries out the tasks of collection, processing and production of statistics for the basic education pre-primary, primary and secondary education at the national level. District level process the data and send the data file to the provincial EMIS center to update the provincial EMIS database (Ministry of Education Pakistan, 2005).

1.5.4 India

India has designed a system for collecting information about schools (Kaul and Neelam, 2009). This system is the basis for all sectors of Indian school education. The National Institute of Educational Planning and Administration (NIEPA) designed a program for the Indian education system. The planning in this program is of two types; micro and macro level planning. In the micro level planning, education plans are prepared at sub national level, such as institution, villages and districts. Whereas macro plans are developed at the level which is just above the sub national level state. At the district, village and education institutions level are the unit of micro planning, but at the state level, district is a central unit of micro planning. In India, education planning is carried out at state level that does not ensure adequate participation of functionaries working at the grassroots.

Once the population has access to educational facilities, the next important variable on which information is required is the number of institutions. Within institutions, the first important variable is availability of infrastructural facilities in schools and their utilization. Information needs to be collected relating to buildings, playgrounds and other ancillary facilities, such as, drinking water, toilets and electricity. Similarly, information relating to number of classrooms and their utilization, class size, number of schools distributed according to class sizes and number of sections is also required which can be used in institutional planning and

related exercises (Mehta, 1998a,b).

1.6 Aim of the study

The aim of this thesis is to model, design and implement a database consisting of data about schools with an interface in Dari (the local language) for the Ministry of Education (MoE) in Afghanistan using a new Software Engineering (SE) methodology called Agile methods. It has been suggested in a previous study that a successful Educational Management Information Systems should have a user-focused approach (Lindvall et al., 2002) to ensure that the user expectations are satisfied in the final EMIS produced. Agile methods encourage a user-focused approach by incorporating the concept of user stories to capture the features and functions needed by the user.

1.7 Research Question

Are Agile methods an appropriate / viable model which can be used to build an EMIS for the MoE of Afghanistan?



1.8 Research Design

1. Study and become familiar with the Agile Methodology
2. Study the EMIS in Pakistan in much more detail since it is a neighbouring country and thus easily accessible. The primary goal would be to determine what features and functions are currently available
3. Determine the Agile process steps that will be used to build the EMIS in terms of ISO 9126 Quality factors, e.g. understandability, reliability, efficiency, modifiability, functionality, portability.
4. Obtain requirements and data from the Ministry of Education.
5. Proceed with the following broad steps.
 - (a) Create the requirements model (UML).
 - (b) Create classes to provide functions as identified in the requirements model.

- (c) Generate code.
- (d) Test the code and remove errors.
- (e) Deploy the prototype with users for feedback using a focus group survey.
- (f) Iterate the above processes for next increment / prototype based on user feedback.

We will design a database and design an interface. We will use Net Beans, the Extreme programming method and Unified Modeling Language (UML) for designing the user interface.

1.9 Chapter outline

In the first chapter we state the purpose of the study and explain the study's position. This is followed in Chapter 2 by a literature review. The purpose of the study and the theoretical framework is given. We describe the research methodology in Chapter 3 where we discuss and justify the data gathering methods used. Our results are presented and addressed in Chapter 4 where we overview the results of the data acquired in the questionnaires. In Chapter 5 we present our conclusions and make some future recommendations and reiterate and interpret the meaning of our findings.



UNIVERSITY *of the*
WESTERN CAPE

Chapter 2

Literature Review

2.1 Introduction

Pakistan, India, Namibia, Zimbabwe already have EMISs in operation for at their ministries of education (Khan and Sethi, 2009; Lloyd and Hewett, 2003). These systems are designed to collect information about schools, and other related information required for educational planning. These systems form the basis for all sectors of school educational planning.

In a recent project undertaken by the Afghanistan Basic Education Consortium (ABEC) (Zhao and McNerney, 2006) to improve aspects of basic education (especially for the youth), accurate data was needed for planning and development. Since there was no EMIS available it was necessary to collect data before the project could proceed with further planning and implementation. This demonstrates that readily available data is necessary for planning and development. This data could be entered into an EMIS system for management so that similar projects could be implemented. One particular finding was that proper documentation and sharing of information is important for knowledge gathering and development. Studying an article on the “Assessing the impact of a multi-agency project on Afghan basic education” (Zhao and McNerney, 2006) observed that an EMIS system is needed to support educational development projects like these. The quality of data is important within any decision making process. In (Chapman, 1991), a paper on “Education data quality”, the authors emphasize the need for quality within such data and recommend quantitative—numeric—data as a way to improve quality as well as ensuring the accurate recording of data at schools. If users or decision makers question the data quality, their confidence levels in that data decreases. An EMIS must therefore ensure that only quality data items are captured in order to promote confidence in the EMIS and to encourage its usage for management and planning.

2.2 What is an information system?

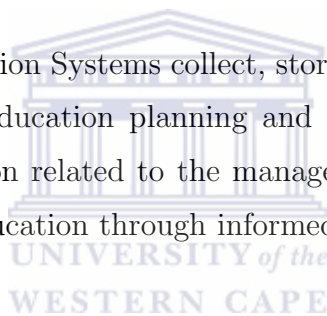
An information system accepts information resources as input, and processes them into data products as output. Information systems have four component parts; input, processes, output and feedback (Moses, 2001).

In an educational setting, inputs would be raw data from questionnaires, surveys, output reports, statistical manifestos and monitoring reports.

Processes include data verification and data analysis. Feedback always passes the finished product back to those who filled in the questionnaire. All information systems use people, hardware and software to perform input, processing output storage and control activities to transform data resources into information products (Carrizo et al., 2003).

2.3 Purpose of EMIS

Educational Management Information Systems collect, store, process, analyze and display or print information for education planning and management (Cassidy, 2005). EMISs integrate information related to the management of education activities and allow for managing education through informed decision making.



2.4 EMIS Definition and Development

An Educational Management Information System forms a unit in the department of education and has the responsibility to develop, maintain and integrate information for the management of education (Cambridge Education, 2006). Wako in (Wako, 2003) gives a general overview of an EMIS used in Zimbabwe.

A clear definition of an EMIS is given which states that it is a system designed systematically to organize information related to the management of educational development. Critical success factors for EMISs are political commitment, good governance and strong management. The development or engineering of an EMIS requires strong management. No details are provided about engineering the EMIS other than that a user-focused approach is a good way of developing it. A strong emphasis is necessary to ensure that the user's expectations and needs are satisfied. The first phase in the software engineering process, viz. establishing the requirements to address the user needs is done with user stories in an Agile process. The school teachers are identified as the key source of information for an EMIS and

should be adequately resourced to provide good EMIS data.

2.5 The data and information flow of EMIS

The EMIS at the district level is the most critical component in this EMIS development plan.

2.6 Agile Methods

In the case of pre-primary, primary and adult and non-formal education, data and information forms are filled in annually at the school level and sent to the ward education coordinators for verification. In Tajikistan ward education coordinators then send the verified forms to the district office. Statistics and logistic officers in district head quarters input the data into computer systems and establish district databases. By using the district database, statistics officers and other education officers compile statistics notebooks, and district and ward school reports according to the needs of stakeholders in the district. Figure 2.1 illustrates the data and information flow in an EMIS (Villanueva, 2003).

Agile methods are a new software engineering approach for designing software. Agile software development is a conceptual framework for software engineering that promotes development in rapid iterations throughout the life cycle of the project. The Agile method has efficient processes for software development. One of these Agile methods is Extreme Programming (XP). XP uses an iterative and incremental software process executed in relatively short cycles. Koskela reports that there are more experimental reports available for XP than for other types of software projects (Koskela et al., 2004). The basic concept of XP methodol-

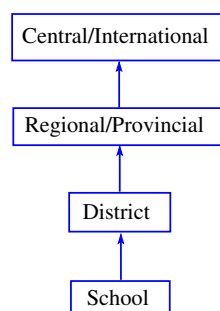


Figure 2.1: The data and information flow in EMIS

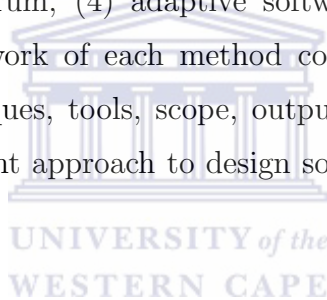
ogy is open communication between the customer and the project team. Extreme programming is the best known Agile method (Strode, 2005).

2.7 What are Agile processes?

Agile processes are a family of software development methodologies that produce software in short iterations and allow for rapid changes in design (Bose et al., 2008). Agile methods aim to support the software team to achieve effective delivery of software and focuses on communication and other mechanisms for managing iterative and incremental development (Schwaber and Beedle, 2002).

2.8 Different Agile Methods

The Agile method has five sub methods; (1) dynamic systems development method, (2) extreme programming, (3) scrum, (4) adaptive software development, and (5) crystal methods. The framework of each method consists of the following parts: philosophy, models, techniques, tools, scope, outputs and practice. Each of these sub methods has a different approach to design software (Lindvall et al., 2002).



2.9 Agile method techniques

Specific project characteristics will determine the prudence of using an Agile or a plan driven methodology or a hybrid of the two. In order to achieve their aim, software development teams need to understand and choose the right models and techniques to support their projects. They must study key questions such as: “What is the best life-cycle model to choose for a particular project?” (Boehm, 2002).

2.10 Extreme Programming

Extreme programming (XP) belongs to the family of Agile software methodology (Beck and Andres, 2002). It focuses on communication, simple design and group work, working on the theory that increasing communication minimizes the amount of documentation produced before implementation can begin. XP is a widely used Agile method, it expresses requirements as user stories, which are implemented directly as a series of tasks. Programmers should preferably work in pairs and

develop tests for each task before writing the code. Tests must be successfully executed before new code is integrated into the system. There are short time gaps between releases.

2.11 Using Agile methods

Agile software development is a framework for software engineering that promotes development repeatedly throughout the life cycle of the project. There are many Agile development methods. Most methods minimize risk by developing software in short periods. Software developed during one time unit is referred to as an *iteration*. The iteration covers the full cycle of software development including planning, requirements analysis, design, coding testing, and documentation. One iteration may not add enough functionality to warrant releasing the product to the market but the goal is to have a release available at the end of each iteration. At the end of each iteration the team re-evaluates project priorities and reworks the iteration if the goals have not been met (DeMarco and Boehm, 2002).

2.12 The Agile method process

For designing a project which uses the XP method, we need some values and principles. XP provides a set of each day's practices that are combined in a day's work. The XP practices are: whole team, planning game, customer tests, small releases, simple design, pair programming, test driven development, design improvement, continuous integration, collective ownership, coding standard, sustainable pace, and metaphor (Hayes and Andrews, 2003).

The details of the XP process are planning, design, coding, test and release:

1. The planning step includes collecting the user stories, developing an acceptance test, doing an iteration test and making an iteration plan.
2. The design step is to simply design solution prototypes,
3. The coding step is by pair programming and unit testing with continuous integration.
4. The test steps include unit testing, continuous integration and acceptance testing.

5. The release step includes the software increment project velocity computation.

2.13 Adapting the Agile process to this project

While most Agile methodologies assume that change is inevitable, current approaches have adopted the strategy of defining a static set of practices and activities that are general enough to be adapted to many project settings (Mellor, 2005). These methodologies have the ability to address variance and adaptability within the processes, but lack the ability to adopt different methodologies to meet the evolving needs of projects as they progress through their life cycles, or change methodologies to meet new business or user conditions. For example, a project may begin with a scrum-based process, but require some XP processes or even heavyweight processes later in the life cycle. Agile methodologies should be able to adapt to these changes with appropriate practices and processes that fit project needs at any stage. As software designers we are expected to work with our the customers to ensure that we build the right product. For designing a project we need to plan. The plan tells us how to extract information and what we are going to do next (Mellor, 2005).

2.14 Object oriented approaches in Agile methods

The Agile and Extreme Programming process steps are as follows. The first step of the Agile methods is *planning*. Planning begins with the creation of user stories by teams. So teams determine their own story. After the first step, design follows. For difficult design problems, a design prototype is created during the design step, followed by coding and testing. Customers list all the stories they expect to include in the application. Developers put estimates against each story. When a story is too large or too vague to be estimated, the customers and developers break it down into smaller stories, and developers re-estimate new stories. The team decides on an iteration size, developers guess the initial velocity, the team decides how much iteration to have in each release, and customers group stories into releases. Each release should have an overall purpose and produce a reasonable, working application. Stories may need to be broken down across different releases. In the next iteration, these steps are repeated. For a project, the most important step

is planning with user stories, then design, coding, test and release (Melnik and Maurer, 2005).

2.15 EMIS Project Development in Agile Methods

The development of an EMIS needs strong management (Strode, 2006). In an XP project, all the features required of the application are recorded as stories. A story is a short description of a feature but big enough for the developers to estimate the work required, but certainly not enough information to complete the implementation. Stories are similar to use cases, but the stories have a lot less detail than use cases (Carrizo et al., 2003). XP methodology encourages open communication between the customer and the project team. For developing an EMIS using Agile methods, the steps outlined in the section above are followed.

2.16 EMIS in Pakistan

2.16.1 Monitoring and Evaluation

Monitoring is a major aspect of EMISs and its activities. Monitoring allows us to review our original objectives. In the Pakistan EMIS the provincial EMIS units usually collect data from the district EMIS cells and the district education offices in their turn collect and receive data from schools, and then the district level sends to the provincial EMIS unit. Where the district level has data processing facilities it would process the data and before sending the data to the provincial EMIS unit to update the provincial EMIS database. The district offices are responsible for annual statistics for development planning and they monitor the local development of education in the district. They are also required to create an education budget for their districts. The district level is becoming more important for providing and managing education services. The provincial EMIS units are responsible for education data collection, process, and analysis. The provincial EMIS units provide technical assistance to the district EMIS. The provincial EMIS staff members require planning skills, an understanding of education planning, monitoring and evaluating, and analyzing and projecting education resources (Ministry of Education Pakistan, 2005).

2.16.2 District level of EMIS

The provincial EMIS in Pakistan receives information through their district EMIS. The main tasks of the EMIS center in the district education offices are to collect and compile the data from schools, and then, send such data to the provincial EMIS unit if there are data processing facilities at the district level. The data is processed and the data files are sent to the provincial EMIS unit to update the provincial EMIS database in order to produce the annual statistical year book. The main function of the EMIS system at the district level is to process the annual school statistics forms. The production of summary statistical tables for various reports using Excel duplicates the statistical tasks done at the district level (Ministry of Education Pakistan, 2005).

2.17 Zimbabwe

For proper data collection reliable school level records management systems are needed. Without good records the management system cannot produce correct results. Questionnaires are used annually to collect information from schools. The school is the best point of data collection for the EMIS. The school managers, teachers, or record officers complete the questionnaire with the necessary information. Most schools have their data on paper. The central EMIS can receive information from provinces and the provinces receive information from districts. Each provincial EMIS unit designs school statistic forms for gathering data. The school manager can enter the data into the database. This tool must be friendly to all EMIS managers. The data records the name, father name, class number, grade, and student numbers and the addresses of students. Information about teachers is also captured. Good systems are supported by comprehensive databases (Wako, 2003).

2.18 India

Data integration is one of the most important EMIS development strategies. It involves gathering data from multiple sources: e.g. payroll, achievement, school census; gathering data over multiple years and multiple levels: e.g. student, teacher, school level; and can be linked, integrated, and merged together. Data integration adds value to the data that has been collected and made available separate places

within the same system. Data integration is done by an educational policy analyst. It is common to see multiple units within a ministry of education collect and manage large databases and not share them with each other. These different sets of data are collected to explain elements of the system, for example in a ministry of education, the data on student achievements are often collected, managed and made available at an examination unit, data on teacher qualifications and salary are at a payroll office, data on enrollment and school inputs are collated at a statistics unit and data on supplies of textbooks, classroom hardware and other teaching resources are often at a supply office. These offices often have separate databases for their private task planning and management and they share them with other offices. These multiple sets of data are often designed in different applications and organized in different interfaces (Mehta, 1998a).

2.19 Namibia

A small group of Namibian staff members of the ministry of education developed an EMIS. If the users require information, they retrieve it from the system. Whenever users need information about students they retrieve it from this center. The data processing system comprises of computer hardware and software for data entry and verification, data processing systems generating the desired outputs of information, and systems giving users access to information. Data collection and data entry are an important part of an EMIS system. In Namibia the development of the EMIS produced several results such as: (1) The creation of an educational statistical data set meeting most of the information needs of the users. (2) The consistent collection of education statistics in two annual surveys. (3) The creation of a well-maintained database of educational data. (4) The capacity to respond to ad-hoc requests for statistical information (Voigts, 2005).

2.20 Unified Modeling Language (UML) and Use cases

UML is an object-oriented modeling language that provides the elements and relationships to model software requirements and UML enables system developers to specify, visualize and document models in a manner that supports scalability and security. UML is designed specifically to represent object oriented systems (Hekmat, 2005). Object oriented development techniques describe software as a set of

cooperating classes consisting of data and behavior.

Use cases have been used mainly to capture requirements, but can describe interactions between the software system and anything external such as a user of the system (Pender, 2002, Page 4). Each use case captures a way of using the system that is a business function. Because use cases focus on the functionality that the system will provide rather than the business processes that it should support, the resulting system will be function centric. Use cases separate the system into actors and use cases. Actors represent roles that can be played by users of the system. Use cases describe the behavior of the system when one of these actors sends one particular stimulus.

2.21 Software process models

In contrast to software life cycle models, software process models often represent a networked sequence of activities, objects, transformations, and events that embody strategies for accomplishing software evolution. Such models can be used to develop more precise and more formalized descriptions of software life cycle activities. Their power emerges from their utilization of a sufficiently rich notation, syntax, or semantics, often suitable for computational processing (Highsmith, 2002).

2.22 Rapid Prototyping and Joint Application Development

Prototyping is a technique for providing a reduced functionality or a limited performance version of a software system early in its development. In contrast to the classic system life cycle, prototyping is an approach whereby more emphasis, activity, and processing are directed to the early stages of software development. In turn, prototyping can more directly accommodate early user participation in determining, shaping, or evaluating emerging system functionality (Bevier et al., 1989). Therefore, these up-front concentrations of effort, together with the use of prototyping technologies, seek to trade off or otherwise reduce downstream software design activities and iterations, as well as simplify the software implementation effort. Software prototypes come in different forms including throwaway prototypes, mock-ups, demonstration systems, quick-and-dirty prototypes, and incremental evolutionary prototypes.

2.23 Extreme Programming Planning Game

Extreme programming teams capture work packages as hand written descriptions of user functionality often written on an index card. These index cards are called story cards to remind project participants that the contents of each card should describe a story that a user might tell about functionality instead of the technical activities that software developers instinctively describe. These story cards, or work packages, are used to facilitate estimates, build schedules, authorize work, drive testing and report status (Blower, 2004).

Work authorizations are made at every iteration, typically every two weeks. At the beginning each iteration the developers estimate each story card as a team. Any story card that is estimated to exceed the length of the iteration is then decomposed into two or more new story cards that as a set represent the same functionality as the original (Blower, 2004). These new story cards or work packages are then estimated. Estimated stories are then grouped into releases of useful functionality that could add some measure of value to the user, or at least help reduce the risk of the project. Each release is then broken down into two-week development iterations and stories are again assigned to the iterations.

2.24 XP Sample Design and Refactoring

Extreme Programming focuses on building reliable and simple implementations to satisfy the requirements of the work packages that have already been authorized. As new features are added or stories are authorized, the design is reworked and modified to accommodate the most recently scheduled features (Schneider and Johnston, 2003). This reworking of the design is done one step at a time, or what Extreme programmers call refactoring. After each incremental step of refactoring is completed, the entire application is controlled for quality.

2.25 XP Tests

Each and every public method or function must have a functional specification captured in the form of an automated test suite. These automated unit tests are considered to be such an important part of the design process, that they are required to be written before the code they are intended to test is written (Holcombe et al., 2001). The power of these low level specifications expressed in executable

code is that the computer can be used to verify conformance. It would not be unusual for a member of an Extreme programming team to run the entire suite of all unit tests repeatedly in one hour.

2.26 Pair Programming

Pair programming is the practice of coding the system with pairs of programmers, seated at the same machine with only one mouse and one keyboard. All software produced with the XP methodology follows this guideline. One person can be coding whilst the other can be focusing on whether or not it is correct, efficient, and simple enough. During this process the people in the pairs swap around, allowing everybody to learn each others strengths and weaknesses (Williams and Kessler, 2001).

2.27 Extreme Programming

Extreme programming accepts that humans are fallible and builds a process that not only accepts progressive elaboration, but makes this reality a central theme to all of its other practices. There is also the recognition that following the proscribed practices in the real world, every day can be very challenging. To overcome this difficulty the practices interlock and compliment each other (James et al., 2001).

2.28 Data analysis

The analysis of processed data, intended for use by planners and decision-makers of ministries of education, must take into account urban and rural disparities as well. The current practice of producing annual statistical abstracts, indicator reports and basic facts and figures must also begin including analytical tables on urban and rural disparity. Moreover, those countries that have started using a planning and projection model should also take the urban/rural variables into account (Hua and Herstein, 2002).

2.29 Conclusion

In this chapter we gave some definitions and overviewed EMIS in some developing countries. An information system is the basis of management, planning and

evaluation of an education system. The data could be entered into an EMIS system for management and planning so that similar projects could be implemented. A clear definition of an EMIS is that it is a system which provides information to support educational management. Agile software development is a conceptual framework for software engineering that promotes development iteration throughout the life cycle of the project. For developing this project we used Agile methods. Agile methods have several approaches. We specifically use the XP method. We described UML and use cases in this chapter.





UNIVERSITY *of the*
WESTERN CAPE

Chapter 3

Research design and methodology

3.1 Introduction

The MoE in Afghanistan does not have an Education Management Information System (EMIS) and all information and statistics about schools are recorded using a paper-based manual system.

Information is released one year after collection. There is no assurance of the quality of the information which consequently impairs effective planning. EMIS is a system for organizing data in a systemic way for the management of educational development and planning. These statistics are an important part of the Education Management Information System. For the Education Management Information System in Afghanistan, the EMIS must be centrally created within the MoE to collect, process, analyze, publish, store and provide high quality educational information for users. The provincial offices and district offices receive information from the schools and processes it so that the district level gives information to provincial level and the provincial level gives the information to the central level. The schools are the best source of EMIS information. The MoE is responsible for managing and delivering both formal and non-formal education.

3.2 Research Hypothesis

If we use Agile methods as the software engineering approach/process, we will be able to develop an EMIS for Afghanistan.

3.2.1 Research Questions

Are Agile methods appropriate for developing and an EMIS in Afghanistan?

Can we use UML for developing a model for an EMIS?

Should extreme programming (XP) be the method within the Agile approach to develop an EMIS?

3.3 Research Instruments

3.3.1 Sample design and sample techniques

The common method for selecting participants for focus groups is called convenience sampling. A sample of convenience is the terminology used to describe a sample in which elements have been selected from the target population on the basis of their accessibility or convenience to the researcher (Dawson and Manderson, 1993). Types of convenience samples also include accidental samples and available samples.

3.3.2 Questionnaire

Questionnaires are usually paper-and-pencil instruments that the respondent completes. The purpose of this questionnaire is to determine what system is best for an educational management system. We have interviewed the policy maker and other role players in the Ministry of Education in Afghanistan. One way to collect core regular data is with a survey. We have used questionnaires as part of our research instruments. We collected core regular data using questionnaires at the Ministry of Education perhaps because this is a simple and relatively inexpensive method (Burgers, 2001).

3.3.3 Interviews and Focus Groups

Interviews provide a question framework but add the benefit of allowing the researcher to ask follow up questions. For the purposes of analysis, responses can be grouped into corresponding themes opinions etc. results can also be presented as case studies to illustrate trends. Due to time constraints, sample groups will tend to be smaller than with questionnaires. Focus groups also provide further qualitative data. They can be very useful as a starting point as they are likely to increase the researcher's knowledge of the subject under investigation and introduce possible lines of enquiry to be followed up (Krueger and Casey, 2000).

3.3.4 Data collection through Focus Groups

Data collection through focus groups operates by bringing together the target people in one place to exchange views and discuss specific question issues. During a focus group meeting, deliberate efforts can be made to focus the discussion on specific issues, to try to reach consensus, and systematically record information

from the exchanges. This method can be a good way for collecting a wide variety of views in a short period. We also use focus groups for data collection (Krueger, 2002).

3.3.5 User story

The first step of extreme programming is to create a user story. From the user stories we received information about the functions and features needed at the MoE. We used a series of questions to extract information. These questions are described in Chapter 4. The most important aspect of the story was that the school is the main focus of EMIS (Koskela et al., 2004). Figure 3.1, Figure 3.2 and Figure 3.3 show the user story in the form of UML Use case diagrams.

3.4 Agile Methods

3.5 Development of the EMIS

3.5.1 Using Agile method

Software engineering is a disciplined approach using processes, methods, and tools (Williams, 2007). Agile software development has good methods and has good frameworks for software engineering that promote development iteration throughout the life cycle of the project. The Agile method has some different approaches like the extreme programming method. We use extreme programming (XP), which is the best known and most widely used and has a series of characteristics (Williams,

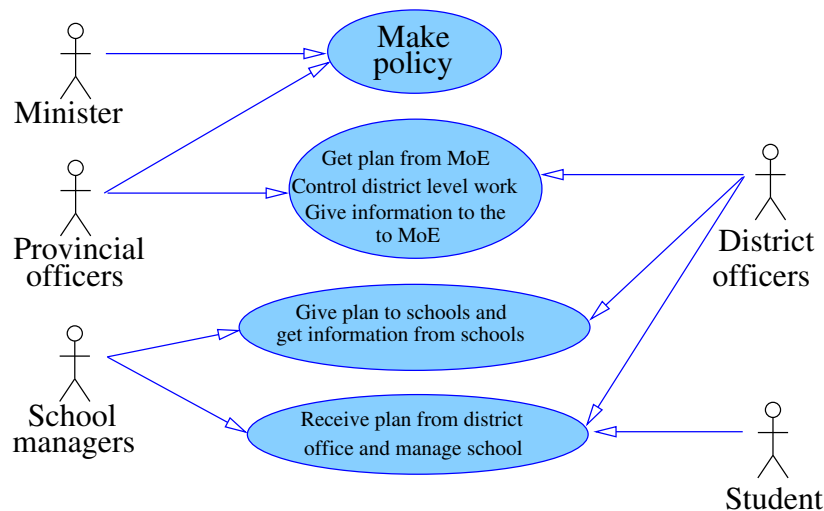


Figure 3.1: Use case diagram 1

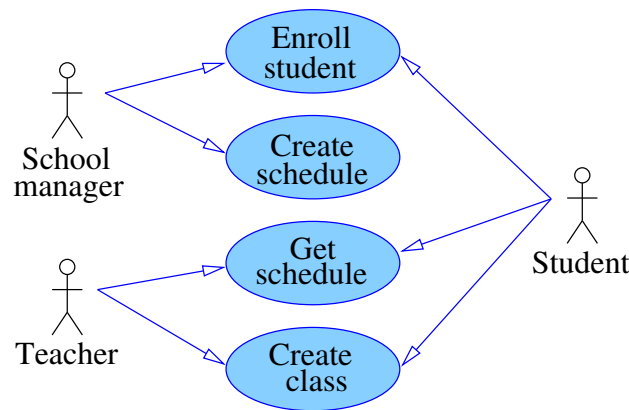


Figure 3.2: Use case diagram 2

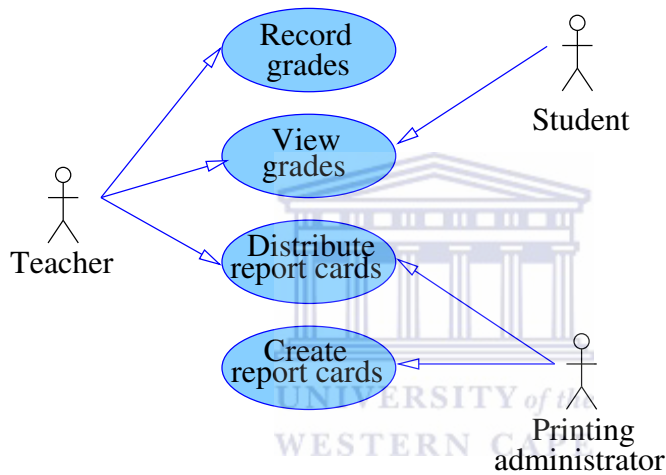


Figure 3.3: Use case diagram 3

2007). The user story is very important part of extreme programming. User stories contain brief requirement descriptions (Wahono et al., 2002).

3.5.2 Using Unified Modeling Language diagram

Unified Modeling Language (UML) provides a set of standard diagrams for graphically depicting an object-oriented system. Object-oriented development methods explain software as a set of data and behaviors (Pender, 2002). We use UML in Net Beans v6 to capture, specify, visualize, construct and document all aspects of the EMIS application development. The UML has different diagrams, see Figures 3.1, 3.2 and 3.3, on Pages 25 and 26 for the Use case diagrams. These Use case diagrams show the various activities the users can perform on the system. Use cases are used during the analysis phases of a project to identify and partition the

system functionally. They separate the system into actors and Use cases. Actors represent roles that can be played by users of the system. Those users can be humans, other computers, pieces of hardware, or even other software systems. We used Java programming language for the design interface and we used Net Beans software for designing the UML and interface. The database is the main part of EMIS system. We used Java database connectivity utilities for connecting the interface and database.

3.5.3 Using Net beans software and Java language for design interface

We used the NetBeans IDE for designing the UML diagrams and interfaces. These interfaces must be friendly to the end user.

3.5.4 Using Microsoft Access for development database

For storing the data, we need database and the database must have some fields such as a *name field*, *surname field*, *class field*, *province field*, *district field*, *mark field*, etc. The best element of the EMIS is the data, database design and database key elements. We needed to design the database and obtain data from MoE to populate the database We used Microsoft Access for the database to store the data. A good system for an educational management information system needs a good database system.

3.6 Research prototype

A research prototype is a framework that we can use to demonstrate the concepts and find out more about the problem and its possible solutions. Prototypes allow the user to see how well the system supports their work. To design a good system the designer needs to design a prototype (Strode, 2005). This prototype has two parts, the first part is an interface and the second part is a database.

3.6.1 The first part of this prototype provides functionality

In this step we explain the functions of the prototype.

- Component for adding data to database

If the user presses the Add button, the user can input information about a student. This function elicits information about the person, for example the person's ID, name, and surname.

- Searching information about users in the database

Searching and reporting is an important component of the EMIS. The program can search information needed by the user and the user gets the data that they need. The policy maker can search data from different cities or provinces and individual schools. The button has been provided for searching information in the database. If the information is found on the database a complete report is provided to the user.

- Reporting is an important part of our work

This part of the EMIS reports what has been planned and implemented and what remains to be done. Reporting allows us to see the problem about the EMIS system and also allows us to see how to solve these problems. The important function of that button is to display information about learners.

- Deleting information from the database

The object of this function is to delete information from the database. If the user selects the Delete button on EMIS window, the user can delete information from database.

- Sorting information in database

Sorting is an important part of the EMIS system. Sorting is the process of arranging a list of items into a well defined order (Dale et al., 2006). We develop a general sorting interface that can be used to define a sort for any list of objects. If the data is not sorted in the database we cannot access the information speedily on the EMIS system. We can sort the data in the database on any attribute, for example we can sort the data by name or by ID. The function of this object is sorting data in database and displaying it. Data can be sorted in the database in ascending or descending order (Lafore, 2003).

3.6.2 Interface

This prototype has tools created in Java and a database in MS Access. The tools have a window populated with components. Each component has attributes.

- A window has a title bar and a height and width.
- The window has in Figure 3.4 has five buttons to execute various actions.

If we press each button on the main window of the EMIS interface, for example if we want to add new information into database select the Add button (Zadel and Scavone, 2006). This tool connects with the database.

The second part of this prototype is a database for storing information for the Ministry of Education in Afghanistan. The database has a table with six fields, *ID* field, *name*, *surname*, *class*, *roll number*, and *city*. This database is connected with the interface illustrated in in Figure 3.4.

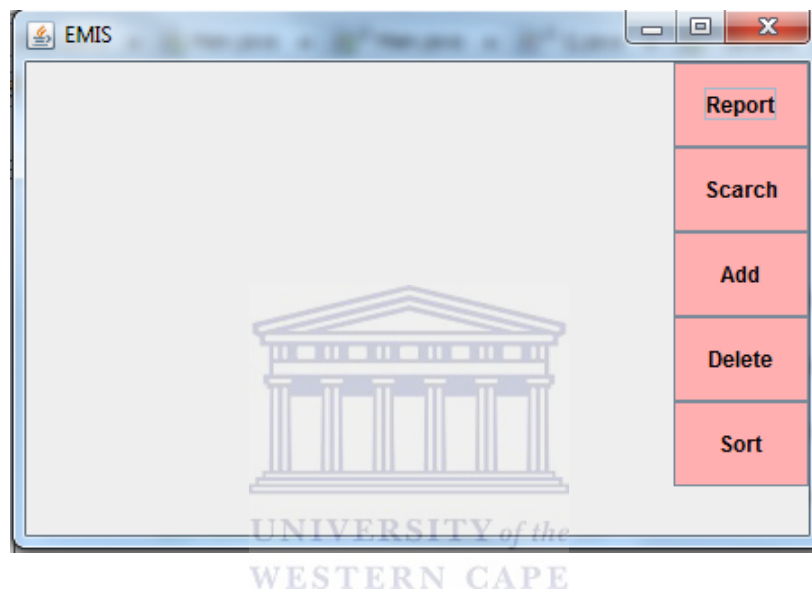


Figure 3.4: Prototype interface

 A screenshot of Microsoft Access showing a table named "tblStudents". The table has six columns: ID, Name, Father Name, Class, Roll Number, and City. The data is as follows:

ID	Name	Father Name	Class	Roll Number	City
2	adfd	dfa	12	34	parwan
5	Asad	Nabi	5	345	Mazar
6	Samad	Kabair	45	57	Jalalabad
7	Ajmal	Asad	12	1230	Parwan
8	Jamal	Asad	12	1111	kabul
9	Ahmad	Asad	11	123	Jalalabad
10	Tawab	wahid	2	12	maidan
*	(AutoNumber)			0	

Figure 3.5: Prototype database

3.7 Prototype Development

For designing the EMIS we use extreme programming (XP). This method is a sub-method of an Agile method, extreme programming has several steps. *Planning* is the first step in developing an EMIS system. Our planning was based on user stories derived from a survey and user questionnaires. The user stories formed the beginning of the Use case diagrams are illustrated in Figures 3.1, 3.2, and 3.3.


The other steps are design, coding, and test. The design used UML and the code was written using Net Beans and the database was done with MS Access. In the first part of this prototype we designed an interface connected to the database with additional buttons to invoke various commands.

3.7.1 The first version

The prototype was used to test and uncover problems. One problem was that the database interface and the database was not connected. Figure 3.4 shows the prototype that has five buttons each button has a specific function and features.

Selecting a button on the prototype window executes the action that belongs to that button. The interface is illustrated Figure 3.4. The most important button on the main window is the Add button. The school is the main focuses of the EMIS. For student registration we need data entry software. For this purpose the main window prototype was provided with an Add button which is used to enter data in the database.

The Add button function inserts new information into the database. Pressing it invokes the Add entry form illustrated in Figure 3.6. This form has fields for



The image shows a screenshot of a software window titled "Add Entry Form". The window has a standard Windows-style title bar with minimize, maximize, and close buttons. The main content area of the window is white and contains the following elements:

- A large blue heading "Add Entry Form" centered at the top.
- Five input fields arranged vertically, each preceded by a label:
 - "Enter ID:" followed by a single-line text box.
 - "Enter Name:" followed by a single-line text box.
 - "Enter Fater Name:" followed by a single-line text box.
 - "Class:" followed by a single-line text box.
 - "Roll Number:" followed by a single-line text box.
- A "Save" button located at the bottom left of the form area.

Figure 3.6: Add Entry Form

entering the student ID, student name, student surname, and class and roll number. This form has a Save button to store information into the database.

The Search button in the main window shown in Figure 3.4 activates the Search Entry Form which is used to retrieve information from the database. The Search Entry Form in Figure 3.7 has four radio buttons and one text field for entering the ID number, entering student name, student score and student grade in text a field by selecting the corresponding radio button. If that ID exists in the data base the program displays it on the monitor.

Another button on the main window is the Report button. The Report button displays all information on the screen using the Report Window illustrated in Figure 3.8

The Delete button deletes information from the database. Pressing the Delete button raises the Delete Form shown in Figure 3.9. It has two radio buttons and

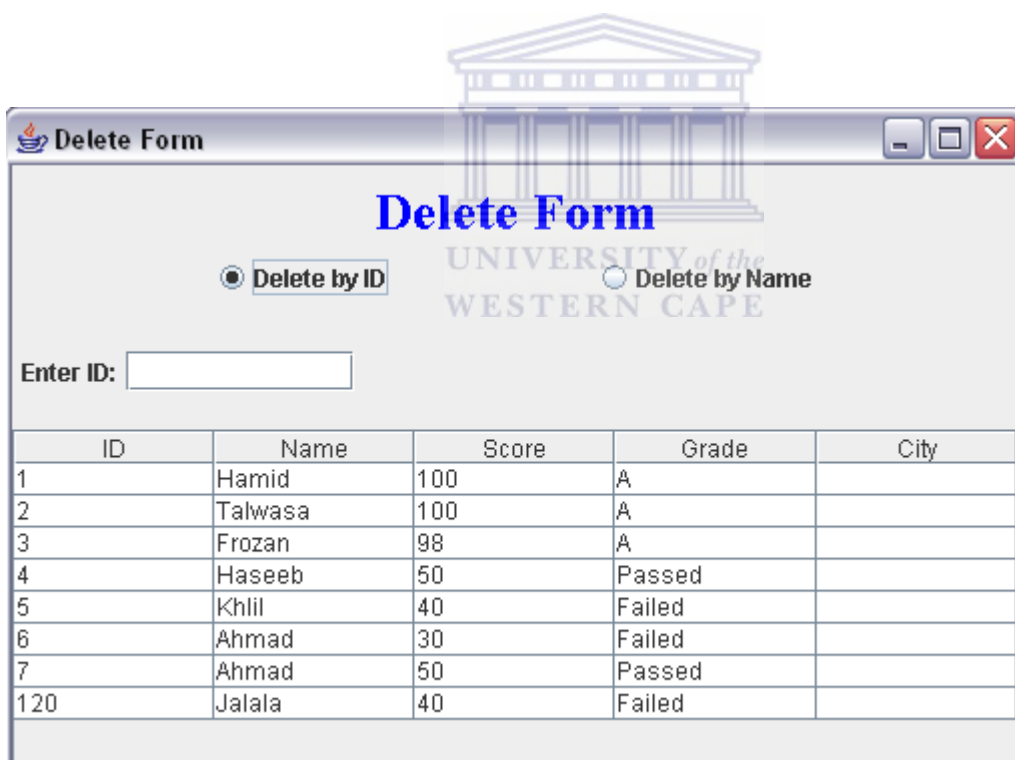
Figure 3.7: Search Entry form

ID	Name	Father...	Class	Roll_N...	City
3	Tawab	Sama...	2	122	Wardak
4	Ahad	samad	12	123	Kabul
1	Wahid	Anwar	5	345	Parwan
5	Ahmad	Asad	11	112	Mazar

Figure 3.8: Report window

their associated text boxes.

The Sort button raises the Sort window shown in Figure 3.10. It has a combo box and a radio button, used to select the sort field and order of the sort.



ID	Name	Score	Grade	City
1	Hamid	100	A	
2	Talwasa	100	A	
3	Frozan	98	A	
4	Haseeb	50	Passed	
5	Khilil	40	Failed	
6	Ahmad	30	Failed	
7	Ahmad	50	Passed	
120	Jalala	40	Failed	

Figure 3.9: Delete Form



Figure 3.10: First version of sort window

3.7.2 The second version prototype

We tested the first version of the prototype on the users and received their feedback which leads to the improved version. We now discuss the second version prototype.

In the first version of the prototype the sort window had a radio button to sort in ascending order and a combo box to select the sort field. See Figure 3.10. In the second prototype we added another radio button for descending sorting. Figure 3.11 shows the sort window after adding the new button. For designing this project we used Agile methods to develop the software incrementally. In the first step we recorded user stories and designed the prototype from the stores. The prototype was constructed and shown to the users for feedback. After considering the user feedback we redesigned the prototype to accommodate their comments.

In the first version prototype the Search button window had four radio buttons: Search for ID, Search for Name, and Search for Score and Search Grade. We changed the Search button form. The second version Search button window has one combo box, one text field and one button. Figure 3.12 shows the Search button window of the second version. We presented this version of the prototype to other users for three weeks. They answered another set of questionnaires which in time led to the third version.

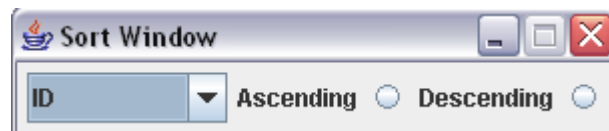


Figure 3.11: Sort Window in second version prototype

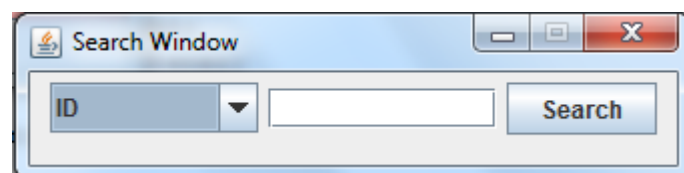


Figure 3.12: Search Window

3.7.3 The third version of the prototype

We replaced the second version prototype which had an English interface with one that has a Dari interface.¹ We also exposed the users to the English interface again for one week. The users gave the some feedback. We adopted their suggestions leading to the third version prototype in Dari. The main window of the Dari version is illustrated in Figure 3.13. Comparing the Dari version of the main EMIS window with the English version in Figure 3.4 on Page 29 the only difference is in the language. The functions of the third version prototype are otherwise exactly the same.

3.8 EMIS Window in Dari

The first version was also presented to a different group of people for three weeks and we created a second set of questionnaires. That questionnaire appears in Appendix B.

The third version of the prototype is the final version. This prototype has all the features and functions explained above. We exposed the final prototype to more users for 3 weeks and received further feedback. In Figure 3.13 and Figure 3.14 the third version prototype of the software and the database were also redesigned into Dari. The users in the provinces and districts do not know enough English.

¹Dari and Pashto are the two official languages of Afghanistan. Dari is the *lingua franca* of Kabul and is predominant in Afghani government circles.

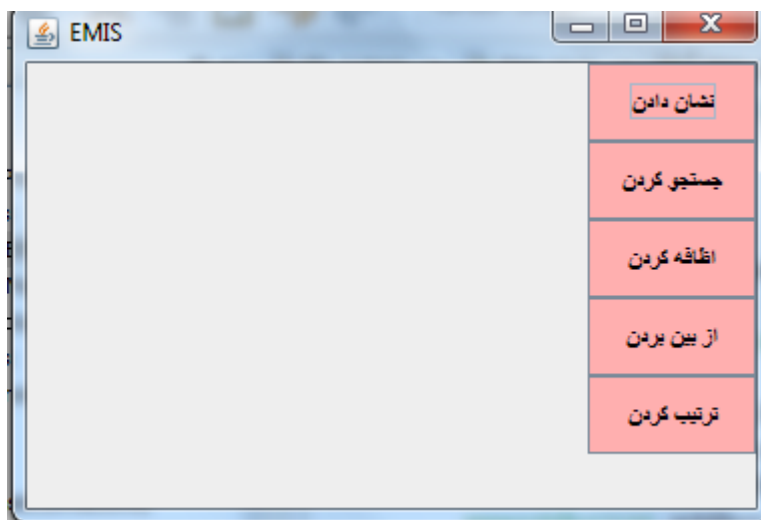


Figure 3.13: EMIS Window in Dari

Doing the questionnaire in Dari was essential to get any feedback from people in the provinces who do not speak English. We also changed the Add button, Delete button and Search button into their respective Dari versions.

3.8.1 Differences between second and third version of prototype

Figure 3.14 shows the Dari Add button window.

- In the third version the Add button window form differs from the second version.
- The third version Delete button window form also differs from the second version delete button window.
- The third version Sort button window is in Dari.
- The third version Search button window is in Dari.

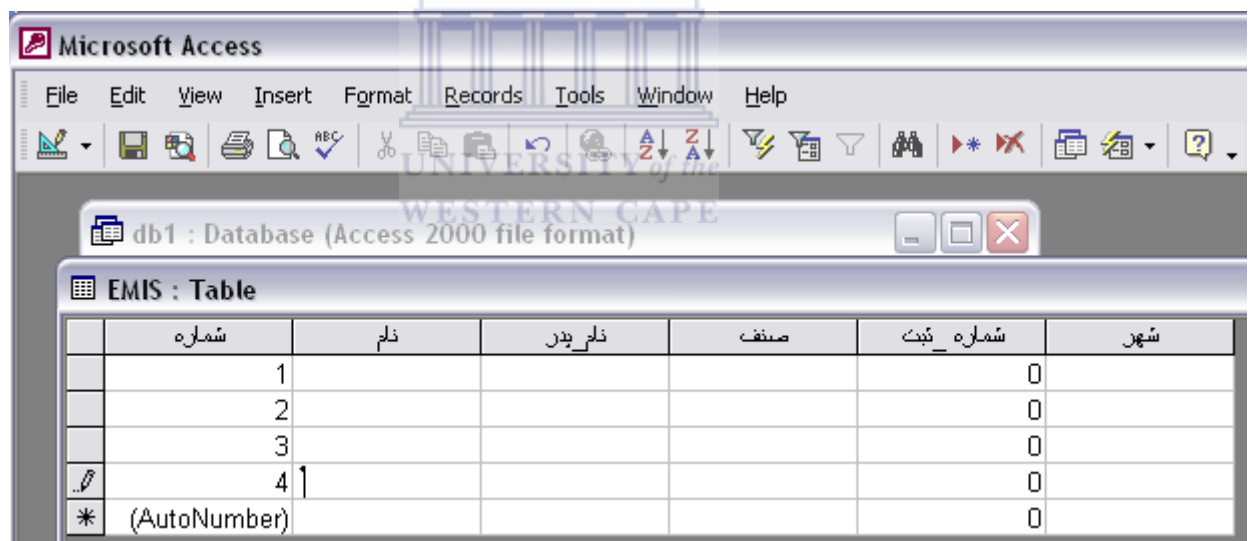


Figure 3.14: Database

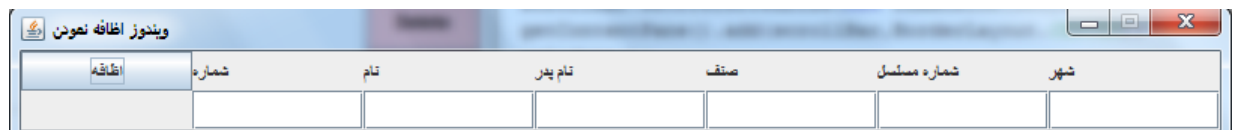


Figure 3.15: Add button window in Dari

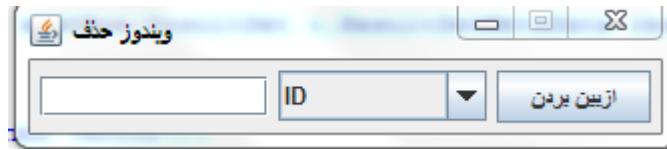


Figure 3.16: Delete button window in Dari

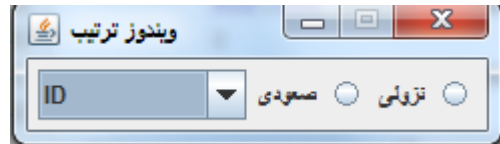


Figure 3.17: Sort button window in Dari

3.9 Data used by the EMIS

We needed to collect information from the Ministry of Education to for use in the EMIS. The system had to be very simple for collecting information about education so that policy decisions could be made at provincial and district level. The Ministry of education in Afghanistan supported the idea of an EMIS. In 2005 The Ministry of Education did not have the ability to collect information properly (Hodges and Shahram, 2006). For data collection questionnaires were necessary at the school level. This data included: location of schools, number of students by gender and age, number of repeaters, number of teachers by gender and qualification. We collect data from Ministry of Education (MoE) for this purpose we designed questionnaires on paper about name of students, surname, class number, school name, student nationality, gender of student, age of students, id number student, teachers name and other information of student and teacher appeared on the questionnaires paper (Kruchten, 2004).

Each provincial office receives information from the districts who in turn receive data from the schools (Rauch and Edwards, 2002) [15]. School managers must send information about students and teachers twice annually.



Figure 3.18: Search button window in Dari

3.10 Data collection

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

3.10.1 First survey

The purpose of this survey was to collect the requirements for designing a prototype for an EMIS for storing information for the Ministry of Education.

Initially data was collected in a survey from 5–10 June 2010 with 34 people. Some respondents interviewed were in the Ministry of Education and 14 were teachers. They are all involved in educational management and had the background to help us. 12 respondents work in the policy and planning department in the Ministry of Education. They all worked in the Information Technology (IT) field. We received good feedback about the management system. Eight of the respondents worked in the provincial education office in Jalalabad. Four people worked in database and three people worked in students department and one of the respondents worked in an Information Technology (IT) management field. In Afghanistan each city has a provincial education office. We used Questionnaire 1 that is discussed in detail in Chapter 4.

3.10.2 Second survey

For one week the users tried out the first version prototype from 12–22 July 2010. Following this we did a second survey and collected data from 25 July 2010 to 1 August 2010. The purpose of the second survey was getting new ideas about the first version prototype. We did the second survey with the 40 persons. They knew the EMIS system and we used their comments to improve the software. For the second survey we designed Questionnaire 2—see Chapter 4. After we received that feedback we could upgrade the prototype.

3.10.3 Third survey

The purpose of the third survey was to capture new ideas about the second version. The respondents used the second version for three weeks from 23–30 August 2010. We did the third survey from 5–25 September 2010. After three weeks we received those users' feedback. The third survey was done with 34 participants, who are all education administrators or teachers. We used Questionnaire 3 to develop the third version of the prototype—see Chapter 4.

3.11 Data editing

We used focus groups for collecting another type of data. Focus groups gather rich data with more detail. The important data editing task was reading and using interviewer comments. Comments were typed by interviewers. The same questionnaires were used for male and female respondents.

The data editing procedures for the EMIS consists of four main tasks:

1. Managing and resolving problem cases,
2. Reading and using feedback to make data updates,
3. Coding questions with text strings and other specific responses and
4. Verifying data editing updates.

For editing purposes the questionnaires were divided into three parts. The first part collects some background information; the second part concerns the use of computers in education. The third part consists of specific questions about the EMIS system. For better understanding we designed the questionnaires in Dari and then translated the responses into English. After the coding process, we inserted the data in Excel sheets to extract some simple statistics

3.11.1 The data coding

We illustrate the format of the first three questions used in the questionnaires in Table 3.1. The statistics are shown in in Chapter 4.

3.12 Rationale behind the selection of data analysis procedure

In focus groups the data analysis has the following steps:

Review of observer notes and transcripts. Good analysis depends on understanding the data. For qualitative analysis, this means we read and re-read the text.

Focus the analysis:

Review the purpose of the evaluation and what we want to find out. Identify a few key questions that we want our analysis to answer.

Categorize information:

Some people refer to categorizing information as coding the data or indexing the data.

Table 3.1: Data coding of the first three survey questions

Question 1: What is your department name?	
Code	Meaning
1	School
2	Provincial education office in Jalalabad
3	Policy and planning department in the IT field Kabul
Question 2: what is your position in that department?	
Code	Meaning
T	Teacher
DBM	Database Field
ITM	Information Technology
DB	Work in database
IT	Work in IT field
Question 3: How long do you have been in that department?	
Code	Meaning
1	Average of 15 years experience
2	Average of 10 years experience
3	Average of 5 years experience
4	Average of 20 years experience

Identify pattern and connection within and between categories:

As we organize the data into categories either by question or by case we will begin to see pattern and connections both within and between the categories.

Interpretation bringing it all together:

Use our themes connections to explain out findings. It is often easy to get side tracked by the details and the rich description in the data (Wong, 2008).

3.13 Conclusion

This chapter discussed our research design and methodology. In the first part of this chapter we discussed the introduction, research hypotheses, research instrument, research survey, research prototype, user story, data collection and data editing and code procedures. Also in this chapter we used focus group steps for data collection and data analysis.



UNIVERSITY *of the*
WESTERN CAPE

Chapter 4

Results: Presentation and Discussion

4.1 Report and Results

We conducted three surveys to determine the user requirements and discuss them here.

4.1.1 First focus group summary

The purpose of the first survey was to gather information to design a prototype for an EMIS, which could be designed for managing information for the Ministry of Education. We received responses from 34 individuals representing different educational fields and types of organizations. In the first survey 41% (14 persons) were teachers. They were teachers but now work in the Ministry of Education. 24% (8 respondents) have basic programming skills and can use Microsoft Access. 24% (8 respondents) work in the Information Technology (IT) department. They understand databases and programming. 35% (12 respondents) are in the policy and planning department in the IT field, and are from Kabul. They include the manager of the IT department, the database manager and the policy and planning manager. They all have programming knowledge but play the role of end users for our project.

4.1.2 Second focus group summary

The purpose of the second survey was to receive feedback on the appropriateness of our first prototype.

In the second survey we received responses from 40 people. Here 40% (16) were teachers working in the Ministry of Education. Most teachers work in the office as users. Some teachers are able to program and they use Access and program in Java. 35% (14) of the respondents in the Jalalabad Provincial Education Office know Access and Java fundamentals and can use SQL server. 25% (10) of

respondents work in the student management department in Kabul. They have skills in Microsoft Access and programming. Some respondents have knowledge of Java, some respondents have knowledge of SQL server, Access and in the fundamentals of databases. 24% (8 respondents) have basic programming skills and use Microsoft Access in the IT department of the Jalalabad Provincial Education Office. The report is summarized in Table 4.1.

4.1.3 Third focus group summary

The purpose of the third survey was to capture feedback on the appropriateness of our second version prototype. The third survey had 45 respondents, where 38% (17) of respondents work in the Provincial Education office in Kabul in the IT field and have skills in Java and databases, 44% (20) of respondents work in the Department Provincial Education in the Wardak province in the IT field. They have SQL server and Access knowledge. 18% (8) of the respondents were teachers now working in the Office of the Ministry of Education. Most of the users can do some programming and know Microsoft Access. The survey results follow.

4.1.4 The first focus groups code

Part 1—Background information

Table 4.1: Survey 1 Part 1 Question 1: Name of department

1—What is the name of your department?		
Response	N	%
School	14	41%
Provincial education office in Jalalabad	8	24%
Policy and planning department in the IT field Kabul	12	35%
Total	34	100%

Table 4.2: Survey 1 Part 1 Question 2: Position in department

2—What is your position in that department?		
Response	N	%
Teacher	14	41%
Database field	3	9%
Information technology	4	21%
Work in database	6	18%
Work in IT field	7	21
Total	34	100%

Table 4.3: Survey 1 Part 1 Question 3: Working experience

3—How long do you have been in the department?		
Response	N	%
Average of 15 years experience	11	32%
Average of 10 years experience	9	26%
Average of 5 years experience	7	21%
Average of 20 years experience	7	21%
Total	34	100%

Table 4.4: Survey 1 Part 1 Question 4: Have used Microsoft Access

4—Have you used Microsoft Access?		
Response	N	%
Yes	22	65%
No	12	35%
Total	34	100%

Part 2—Some information about field of work

Table 4.5: Survey 1 Part 2 Question 1: Know a programming language

1—Do you know any programming languages?		
Response	N	%
Yes	20	59%
No	14	41%
Total	34	100%

Table 4.6: Survey 1 Part 2 Question 2: Have used SQL

2—Have you used SQL?		
Response	N	%
Yes, we used SQL	25	74%
No, we did not work in SQL	9	26%
Total	34	100%

Table 4.7: Survey 1 Part 2 Question 3: Have used Microsoft Office

3—Have you used Microsoft Office?		
Response	N	%
Yes	27	79%
No	7	21%
Total	34	100%

Part 3—Some information about kind of EMIS

Table 4.8: Survey 1 Part 3 Question 1:

1—What type of information and statistics would you like an EMIS to provide for you?		
Response	N	%
Information about schools	6	18%
Statistics about schools	6	18%
Both information and statistics about schools	22	65%
Total	34	100%

Table 4.9: Survey 1 Part 3 Question 2: Type of interface needed for EMIS

2—What type of interface is needed for the EMIS?		
Response	N	%
Easy	10	29%
Easy and friendly	24	71%
Total	34	100%

UNIVERSITY of the
WESTERN CAPE

Table 4.10: Survey 1 Part 3 Question 3: Type of data for database

3—What type of data do you want to put in the database?		
Response	N	%
Students	12	35%
Students and teachers	22	65%
Total	34	100%

Question 5 in Part 3 of the first survey asked what the current difficulties are within the educational system. The biggest problem is that very few records have as yet been captured directly onto computers. A secondary problem was seen to be the lack of computers at schools making it virtually impossible to record data onto a database.

4.2 Second focus group code

4.2.1 Part 1 Background information

Table 4.11: Survey 1 Part 3 Question 4: Can interface be in English

3—Can the interface be in the English language only or do you prefer a local language?		
Response	N	%
In local language	22	65%
In English	12	35%
Total	34	100%

Table 4.12: Survey 1 Part 3 Question 5: Current difficulties within education system

5—What are the current difficulties within the education system?		
Response	N	%
Data is recorded on paper	28	82%
Schools do not have computers	6	18%
Total	34	100%

Table 4.13: Survey 1 Part 3 Question 6: Data analysis to improve educational services

6—How can the data be used to help educational authorities improve education services?		
Response	N	%
Data analysis can help education authorities to improve services	30	88%
Data analysis cannot help to improve education services	4	12%
Total	34	100%

Table 4.14: Survey 1 Part 3 Question 7: Does data flow from schools up or down

6—Does the data flow down from the central ministry or up from schools?		
Response	N	%
Data flows from the local level up to the central level	32	94%
Data flows from the central level down to the local level	2	6%
Total	34	100%

Table 4.15: Survey 1 Part 3 Question 8: Structure of the database

8—What structure will the database have?		
Response	N	%
Simple structure	32	94%
Complex structure	2	6%
Total	34	100%

Table 4.16: Survey 1 Part 3 Question 9: How should information be collected?

9—What technique should be used to collect necessary information?		
Response	N	%
Best technique is internet connection	18	53%
Collect on paper	16	47%
Total	34	100%

Table 4.17: Survey 1 Part 3 Question 10: Collection of information

10—What technique has been used to collect necessary information?		
Response	N	%
Questionnaires answered by principals needed	33	97%
Questionnaires answered by principals not needed	1	3%
Total	34	100%

Table 4.18: Survey 1 Part 3 Question 11: Schools have computers

11—Does each school have computers?		
Response	N	%
The schools have not computers	24	71%
The schools have computers	10	29%
Total	34	100%

Table 4.19: Survey 1 Part 3 Question 12: Internet connection to provinces

12—Does each province have an internet connection?		
Response	N	%
Each province has an internet connection	31	91%
Each province have not internet connection	3	9%
Total	34	100%

Table 4.20: Survey 1 Part 3 Question 13: How will data be collected

13—How will the data be collected or estimated?		
Response	N	%
Collect data on papers	9	28%
Collect data via internet	25	74%
Total	34	100%

Table 4.21: Survey 1 Part 3 Question 14: Software and hardware needed

14—What software and hardware is needed in order to meet the needs for data collection and analysis?		
Response	N	%
Computer, internet connection, storage memory, database, interface for database, Microsoft Office	18	53%
Computer, storage memory, database, interface for database, Microsoft Office	10	29%
Computer, storage memory, database, interface for database	6	18%
Total	34	100%

Table 4.22: Survey 1 Part 3 Question 15: Schools' data used in national Statistics

15—Are these schools' data included in national statistics?		
Response	N	%
Data included in national statistics	31	91%
Data was not included in national statistics	3	9%
Total	34	100%

Table 4.23: Survey 1 Part 3 Question 16: Who has used the data?

16—Who has used the data?		
Response	N	%
The Ministry people can use the data	32	94%
Students use the data	2	6%
Total	34	100%

Table 4.24: Survey 1 Part 3 Question 17: Who should analyze the data?

17—Who should analyze the data?		
Response	N	%
The EMIS center can analyze the data	26	76%
The schools can analyze the data	8	24%
Total	34	100%

**Table 4.25:** Survey 2 Part 1 Question 1: Name of department

1—What is your department name?		
Response	N	%
School	16	40%
Provincial Education Office, Jalalabad	14	35%
Student Management Department, Kabul	10	25%
Total	40	100%

In Table 4.26 the response ‘Teacher’ covers 16 teachers are deployed as office staff in the Ministry of Education. Four of these people are school managers, eight work in the student management office and four work in the teachers management office. The students’ managers manage all students information and are responsible for maintaining the scoring sheets of students, class timetables and exam timetables,

Table 4.26: Survey 2 Part 1 Question 2: Position in department

2—What is your position in that department?		
Response	N	%
Teacher	16	40%
Database user in Jalalabad	1	3%
Information Technology in Jalalabad	2	5%
Enrolment student information	9	23%
Work in IT field	11	28%
Students manager	1	3%
Total	40	100%

Table 4.27: Survey 2 Part 1 Question 3: Average years of experience

3—How long do you have been in that department?		
Response	N	%
15 years experience	8	20%
10 years experience	12	30%
5 years experience	9	22%
20 years experience	11	28%
Total	40	100%

Table 4.28: Survey 2 Part 1 Question 4: Use Microsoft Access

4—Have you created tables and queries in Microsoft Access?		
Response	N	%
Yes	28	70%
No	12	30%
Total	40	100%

4.2.2 Part 2—Some information about work field

Table 4.29: Survey 2 Part 2 Question 1: Can program in Java

2.1—Do you program in Java?		
Response	N	%
Work in Java	27	68%
Do not work in Java	13	32%
Total	40	100%

Table 4.30: Survey 2 Part 2 Question 2: Use Access and SQL server

2.2—Can you work in Microsoft Access and in SQL server?		
Response	N	%
Work in Access and SQL server	33	82%
Do not work in Access and SQL server	7	18%
Total	40	100%

Table 4.31: Survey 2 Part 2 Question 3: Knowledge of software engineering

2.3—Do you know about software engineering?		
Response	N	%
Know about software engineering	11	28%
Do not know about software engineering	29	72%
Total	40	100%

4.2.3 Part 3—Some information about kind of EMIS

Table 4.32: Survey 2 Part 3 Question 1:

1—What is do you think about the database functionality		
Response	N	%
It is Okay	36	90%
It is not Okay	4	10%
Total	40	100%

Table 4.33: Survey 2 Part 3 Question 2: Sorting function in EMIS tool

2—What do you think about the sorting function in the EMIS tool?		
Response	N	%
Sort window should have buttons for ascending and descending	38	95%
Sort window should have only an ascending button	2	5%
Total	40	100%

UNIVERSITY of the
WESTERN CAPE

Table 4.34: Survey 2 Part 3 Question 3: User-friendliness of interface

3—Is that interface friendly to users?		
Response	N	%
The interface is friendly to users	37	93%
The interface is not friendly to users	3	7%
Total	40	100%

4.3 The third focus groups code

4.3.1 Part 1—Some background information

Table 4.35: Survey 2 Part 3 Question 4: Is English interface better?

4—Is the English interface better?		
Response	N	%
Yes	21	53%
No	19	47%
Total	40	100%

Table 4.36: Survey 2 Part 3 Question 5: Which data is useful?

5—Which data is useful for data entry?		
Response	N	%
Data about students and teachers	31	77%
Data about students	9	23%
Total	40	100%

Table 4.37: Survey 2 Part 3 Question 6: The functionality of the Add button

6—What do you think about the function of the add button?		
Response	N	%
It is Okay	38	95%
It is not Okay	2	5%
Total	40	100%

Table 4.38: Survey 2 Part 3 Question 7: The Delete window

7—What do you think about the function of Delete window?		
Response	N	%
The delete window is Okay	39	98%
The delete window is not Okay	1	2%
Total	40	100%

Table 4.39: Survey 2 Part 3 Question 8: The Delete button

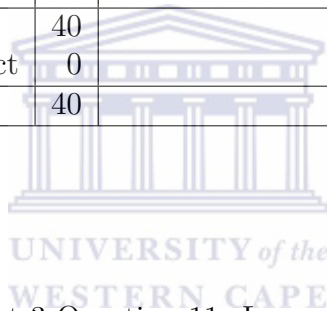
7—What do you think about the function of the Delete button?		
Response	N	%
It is Okay	40	100%
It is not Okay	0	0%
Total	40	100%

Table 4.40: Survey 2 Part 3 Question 9: Suitability of Search window

9—What do you think about the function of the Search window?		
Response	N	%
It is Okay	36	90%
It is not Okay	4	10%
Total	40	100%

Table 4.41: Survey 2 Part 3 Question 10: Connection between interface and database

10—How good is the connection between interface and database?		
Response	N	%
The connection is correct	40	100%
The connection is not correct	0	0%
Total	40	100%

**Table 4.42:** Survey 2 Part 3 Question 11: Language of interface

11—What drawbacks are there in the EMIS interface designed in English for Afghanistan?		
Response	N	%
EMIS must be designed in local language	32	80%
EMIS must be designed in English	8	20%
Total	40	100%

Table 4.43: Survey 2 Part 3 Question 12:

12—Are additional types of information necessary?		
Response	N	%
No	36	90 %
Yes	4	10 %
Total	40	100%

Table 4.44: Survey 2 Part 3 Question 13:

13—What do you think about the database's six fields? (ID, name, father name, Class, Roll number, City)?		
Response	N	%
Okay	40	100%
Not Okay	0	0%
Total	40	100%

**Table 4.45:** Survey 3 Part 1 Question 1: Name of department

1—What is your department name?		
Response	N	%
Work in provincial education office in Kabul in IT	17	38%
Work in provincial education in Wardak School	20	44%
School	8	18%
Total	45	100%

In Table 4.46 the teachers currently working as office staff in the Ministry of Education consist of 3 working in the Kabul provincial office, two are teachers and three are principals.

Table 4.46: Survey 3 Part 1 Question 2: Position in department

2-What is your position in that department?		
Response	N	%
Teachers working as office staff	8	18%
Database user in Wardak province	9	20%
Information Technology in Wardak	8	18%
Enrolment the students information in database	7	16%
Work in IT field	7	16%
Schools manager	6	12%
Total	45	100%

Table 4.47: Survey 3 Part 1 Question 3: Departmental experience

3- How long do you have been in that department?		
Response	N	%
About 15 years experience	10	22%
About 10 years experience	14	31%
About 5 years experience	8	18%
About 20 years experience	13	29%
Total	45	100%

4.4 Part 2

4.5 Summary of the main results

In Table 4.32, responding to the question “What do you think about the database functionality? Is it okay or not okay”, the majority of the respondents 36/40, i.e.

Table 4.48: Survey 3 Part 2 Question 1: Used SQL server

4- Did you use SQL server?		
Response	N	%
Yes	36	80%
No	9	20%
Total	45	100%

Table 4.49: Survey 3 Part 2 Question 1: Able to use Net Beans

1—Can you use Net Beans IDE?		
Response	N	%
Yes	27	60%
No	18	40%
Total	45	100%

Table 4.50: Survey 3 Part 2 Question 2: Able to create a Use Case Diagram

2—Can you create a Use Case diagram?		
Response	N	%
Yes	32	71%
No	13	29%
Total	45	100%

Table 4.51: Survey 3 Part 2 Question 3: Know about XP

3—Do you know about eXtreme Programming?		
Response	N	%
Yes	28	62%
No	17	38%
Total	45	100%

Table 4.52: Survey 3 Part 3 Question 1: Like database application

1—What do you think about the database application?		
Response	N	%
Okay	45	100%
Not Okay	0	0%
Total	45	100%

Table 4.53: Survey 3 Part 3 Question 2: Like sort button

2—What do you think of the sorting button in the EMIS interface?		
Response	N	%
Okay	45	100%
Not Okay	0	0%
Total	45	100%

Table 4.54: Survey 3 Part 3 Question 2: Interface is user friendly

3—Is that interface friendly to users?		
Response	N	%
Yes, it is friendly	44	98%
No, it is not friendly	1	2%
Total	45	100%

Table 4.55: Survey 3 Part 3 Question 4: Prefer English or Dari interface

4—Is the English interface better?		
Response	N	%
English interface is better	12	27%
Dari interface is better	33	73%
Total	45	100%

Table 4.56: Survey 3 Part 3 Question 5: Opinion of add window function

5—What do you think about add window function?		
Response	N	%
The add window function is Okay	45	100%
The add window function is not Okay	0	0%
Total	45	100%

Table 4.57: Survey 3 Part 3 Question 6: Opinion of sort window

6—What do you think of the sort window?		
Response	N	%
Sort window is right	43	96%
Sort window is not right	2	4%
Total	45	100%

Table 4.58: Survey 3 Part 3 Question 7: Delete button okay

7—What do you think about the function of the delete button?		
Response	N	%
Okay	45	100%
Not Okay	0	0%
Total	45	100%

Table 4.59: Survey 3 Part 3 Question 8: Show window okay

8—What do you think about the function of the show window?		
Response	N	%
Okay	45	100%
Not Okay	0	0%
Total	45	100%

Table 4.60: Survey 3 Part 3 Question 9: Opinion of sort window

9—What do you think about the function and feature of search window?		
Response	N	%
It is right	45	100%
It is not right	0	0%
Total	45	100%

96% felt that the functionality of the database is very good. In a similar question asked Survey 3 in Table 4.52, namely, “What do you think about the database application?”, 45/45 i.e. 100% responded that “It is Okay”.

In the first, second and third survey the respondents were asked “What do you think about the functions of the sort button in the EMIS tool?” The option of the button in the prototype were to sort the data in ascending or descending order. Most respondents opted for the two button approach, i.e. in surveys one, two and three the two button option received $(38/40)^2$, 100% 45/45³ and 96% (43/45).⁴ positive response.

So we conclude that the users were highly satisfied with the sorting feature.

²See Table 4.33.

³Table 4.53.

⁴Table 4.57.

Table 4.61: Survey 3 Part 3 Question 10: Like sort button

10—How good is the connection between interface and database?		
Response	N	%
Okay	45	100%
Not Okay	0	0%
Total	45	100%

Table 4.62: Survey 3 Part 3 Question 11: Shortcomings of Dari interface

11—Are there gaps in the Dari language interface for Afghan users?		
Response	N	%
There were no gaps in Dari language interface	45	100%
There were gaps in Dari interface	0	0%
Total	45	100%

Responding in the second survey in Table 4.40 about the suitability of the search window, 36/40, i.e. 90% of the respondents were satisfied. In a similar question in Survey 3, see Table 4.61, all 45 respondents replied that the search function is ‘right’.

The respondents found in the Table 4.42 and Table 4.55 that the interface was in Dari was preferable and in Tables 4.9 and 4.54 that the interface user-friendly. In the first survey, second survey and third survey the respondents replied in the Table 4.42 and 4.55 which the interface in Dari—the local language—is preferable.

The respondents felt in Table 4.37 and Table 4.56 that the Add window function is very good for adding the new information to the database. In Table 4.39 and Table 4.59 about the function of the Delete button, respondents replied that Delete button is very good for deleting information from the EMIS database.

In the second and the third surveys where the interface was in in Dari the respondents said Dari interface had no problems. See Table 4.62.

We designed a Show button window for displaying information about learners at school. The respondents said in Table 4.38 and Table 4.59 felt the show button function is good for displaying information concerning learners at school. We concluded that the EMIS needs this feature.

4.6 Comparing the Research Findings/Results with the Hypothesis

If we use Agile Methods as the software engineering approach/process, we will be able to develop an EMIS for Afghanistan.

We applied extreme programming in the design of the prototype and removed the problems in prototype. We designed the third version prototype in Dari. We did three focus group, the respondents said the interface and the database work

were appropriate for EMIS for the Ministry of Education in Afghanistan. The interface is used for data entry, searching data in the database, and displaying data on screen. Data can be deleted from the database and can sort information in the database. The main result of the project is an EMIS for the Ministry of Education in Afghanistan in Dari.

4.7 Main trends and patterns

All respondents said in their comments that the interface must be in Dari. Respondent commented that they were satisfied about the function and form of the database, which behaved correctly according to their expectations.

All respondents replied in their comments about the functions of the Add window, Delete window, Search window, Report window were correct. They said that using Agile methods and XP involved them as users and that it was therefore a good way of engineering the EMIS. Respondents said the Java is good programming language for developing this system.

4.8 Summary

The positive points in this chapter are that we successfully designed an EMIS for the Ministry of Education in Afghanistan. This system has two main parts, the first part of the EMIS is an interface and the second part of that system is a database. The other positive point is the interface is in local language Dari making the interface easy to use and friendly to users.



UNIVERSITY *of the*
WESTERN CAPE

Chapter 5

Conclusion and recommendations

5.1 Introduction

In this chapter we summarize our research findings and conclusions from the data provided by the respondents to the focus group survey statements in three questionnaires and give recommendations for future research.

5.2 Overview

The Ministry of Education in Afghanistan does not have an EMIS. The focus of this research project was to investigate the appropriateness of using Agile processes to build an EMIS for the Ministry of Education in Afghanistan. This was done using three focus group surveys for collecting data.

Chapter 1 dealt with the aims of the research, introducing the proposed study by stating and describing the functions of an EMIS and clarifying some of its concepts.

The Ministry of Education cannot access information conveniently and since the data was accessible on paper it was not always accurate and complete. The enrolment of students is very difficult and the Ministry of Education does not have a database system for storing the information.

In Chapter 1 we briefly overviewed EMISs in Pakistan, Zimbabwe, Namibia, India and other countries. We gave an overview of the EMIS for the Ministry of Education in Afghanistan. In order to design an EMIS we needed to understand the role of an EMIS and discussed our research problem. This chapter focused on the research question and discussed the research design. We also gave a chapter-by-chapter thesis outline.

Chapter 2 focused on a literature review of Education Management Information Systems, Agile methods, Extreme Programming and Unified Modeling Language. We also discussed definitions of the Education Management Information

Systems and overviewed them. We discussed the process in terms of the data flow, data analysis, monitoring and data integration, and introduced Agile methods and the Extreme programming method.

Chapter 3 described the methodology and design of the research project. This chapter discussed the research hypothesis and the key concepts and variables. To design an EMIS, we needed to create a requirements model in UML. For designing the interface we used the Agile approach and Java to generate the code for the interface. The database of the EMIS system was done using Microsoft Access.

We explained the research design and methods and discussed the research instruments. We did focus group surveys using three questionnaires as our main instruments and derived three prototypes from the surveys. We released a final version with a Dari interface.

Chapter 4 discussed the results and the actual questions covered in the surveys. We discussed the focus group summary and analysis and main results. This analysis gave us a very clear idea of the role of the EMIS needed for the Ministry of Education. It also became clear that interface needed to be in Dari.

5.3 Research findings

In light of the literature review and responses from the focus group interviews, our research findings are presented in two categories namely, findings from Chapter 2 literature reviews and the focus group interviews.

The following are findings obtained from the literature review:

5.3.1 The data and information flow in the MoE

We found from the literature that the data and information flow is from the school level to the central level:

School ⇒ District ⇒ Regional / Provincial ⇒ Ministry of Education.

The data flow in the EMIS is from the school level to the central EMIS. The information about a student's enrollment, number of classes, etc. is initially entered into the student's school record form and is then passed up to the district office, where it is summarized and sent up to the regional offices where it is finally processed before passing up to the ministry.

5.3.2 Data Collection

Data collection is the main task of the Ministry of Education, for every province, district and school. A form is used for data collection. The form has fields for the student name, parent name, village, and the birth date and is filled in at the school attended by the student. This information is then sent to the district office where the information from the questionnaires is gleaned and aggregated on paper and entered into a database.

5.3.3 Storing the information

For storing the information the Education Management Information System needs a proper database which is the most essential part of the EMIS so that users can retrieve information about students more efficiently. The information in the database system is secured from the other users by password access. The data entry interface connects the users with the database.

5.3.4 Extreme programming

The software must be designed rapidly and still be of the best quality. Extreme programming is a good method for achieving this. Extreme programming enables us to develop software in short iterations. The first step of Extreme programming is to extract the user's story. For designing software the developer must collect the necessary information from the customer about the software needs. For this we elicited user stories from the officials in the ministry responsible for its statistical data.

5.4 Research questions

An analysis of qualitative data collected through focus group interviews revealed the following findings: We had three research questions.

5.4.1 The first research question

Are Agile methods appropriate for developing and an EMIS in Afghanistan?

Agile modeling is a practice based methodology for effective modeling and documentation of software based systems. Agile modeling is a set of values, principles and practices for modeling software that can be applied on a software development project in an effective and simple manner. We used Agile methods for designing

the EMIS. Agile methods are an efficient way to develop software. It was easy to determine from the user stories what had to be done after each of the three focus group meetings since Agile methods have short iterations for developing software. Both the users and the developers are aware of the most recent user stories. The users demanded a Dari version of the interface making it clear that the Agile method yielded a sensible and appropriate outcome for the EMIS.

5.4.2 The second research question

Can we use the UML for developing a model for an EMIS?

Unified Modeling Language (UML) is a software modeling language that has become a *de facto* standard. UML provides a variety of useful capabilities to the software designer, including “*multiple, interrelated design views, a semiformal semantics expressed as a UML meta model, and an associated language for expressing formal logic constraints on design elements*” (Medvidovic et al., 2002). The UML models of a software system consist of several partial models, each of which addresses a certain set of issues at a certain level of fidelity. UML models address a number of design issues through a variety of diagrams, classes and their declared attributes, operations and relationships, the possible states and behavior of individual classes, packages of classes and their dependencies (Medvidovic et al., 2002).

We used the Unified Modeling Language (UML) for design objects for the Education Management Information System. UML is a good way to achieve the design the objectives of the Education Management Information System. User stories serve the same purpose as use cases but are not the same. We created time estimates for the release planning meetings. User stories are written by the customers as things that the system needs to do for them. The user stories are similar to usage scenarios. The use case diagrams describe system functionality as a set of tasks that the system must carry out and actors who interact with the system to complete the tasks.

5.4.3 The third research question

Should extreme programming (XP) be the method within the Agile approach to develop and EMIS?

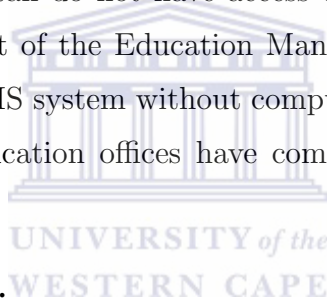
Extreme programming projects start with a “release planning phase”, followed by

several rounds of “iteration”, each of which ends with user acceptance testing. When the software has enough features to satisfy users, the user-developer team closes the round and releases the software. Users write “user stories” to describe their software needs. User stories assist the team to understand the time and resources necessary to build the release and to define user acceptance tests. A user or a representative is part of the XP team, so the user can add detail to requirements while the software is being built. Extreme programming is a vehicle for designing good quality software rapidly. We used extreme programming to collect, design, code and test our EMIS system and found it to be beneficial.

5.5 Other findings

5.5.1 Computers

All the schools in Afghanistan do not have access to computers. Computers are the most critical component of the Education Management Information System. We cannot have a good EMIS system without computers and computer end users. However, all provincial education offices have computers, but it is essential for schools to have computers.



5.5.2 Internet Connection

Each province must have an Internet connection to every district office and with the central Education Management Information System. Without an Internet connection, it is impossible to send information to each education level. Currently only four provinces have Internet connections to the Ministry of Education. It is essential that provinces have Internet connections with their districts and with the central ministry.

5.6 Further research

The following recommendations are suggested for further research: We recommended that the Education Management Information System be developed in Pashto. Afghanistan has some provinces where Pashto is the prevalent language. We recommend that provinces develop a database for teachers’ and parent information. Further research is needed into determine the best language to use for creating the EMIS interface and implementing the database.

We recommend that the EMIS interface should be designed in three languages, English, Dari, and Pashto and that the database must be extended to accommodate more fields. We recommend that further research develops five sub-EMIS systems for the five big provinces. The central EMIS is connected to five the big provinces sub-EMIS systems.

5.7 Conclusion

In this chapter a summary of the research project has been given. The management function of the EMIS includes collecting, storing, integrating, processing, organizing, outputting, and marketing educational data and statistics in a timely and reliable manner. These specific tasks serve the needs of educational management, resource allocation, and policy formulation, such as planning and budgeting, policy research and analysis, monitoring and evaluation, allocation of school supplies, and domestic and global communication and collaboration. An EMIS is also a set of formalized and integrated operational processes, procedures and cooperative agreements by which data and information about schools and schooling, such as facilities, teachers, students, learning activities and evaluative outputs are regularly shared, integrated, analyzed, and disseminated for educational decision use at each level of educational hierarchy.

An Education Management Information System is a very important instrument for the to Ministry of Education. Key components are data analysis, storing data, collecting data, data entry, and output data and Education Management Information System can be built using Agile methods. Agile method is a good method for development of EMIS.

Appendix A

School Education in Afghanistan

A.1 Afghanistan Schools Survey 2007

The Ministry of Education in Afghanistan conducted a school survey in mid-2007. The Ministry of Education (MoE) is responsible for managing and delivering both formal and non formal education. The responsibility for formal education is limited to: General Education, Islamic Education, Teacher Education, Technical and Vocational Education, Community Based Education. Non-formal education is confined to literacy training for out of school children and adults. There are 34 provinces in Afghanistan. The Ministry of Education of Afghanistan has established one provincial education office for each of the 34 provinces and one provincial education office for Kabul city due to the large number of students in Kabul city.

A.2 Students—Formal Education

There are 5,950,455 students registered in MoE and Community Based Education (CBE) schools delivering primary and secondary education. Of these, 5,675,951 (95.39%) are General Education, 91,362, (1.54 %) are Islamic Education, 14,294 (0.24%) are Teacher Training and 10,366 (0.17) Technical/Vocational students. There are 158,482 (2.66%) students registered in CBE classes. Table A.1 shows the student statistics in more detail.

A.3 Students—Non-Formal Education

There were 280,000 students enrolled in literacy courses across the country in nearly 5000 classes at the time of the survey.

Table A.1: Schools and student enrollment

	Schools	Schools %	Males	Females	M/F ratio	Males %	Females %	Totals	% of Total
General Education	9,062	95.63%	3,667,862	2,008,089	0.55	64.62%	35.38%	5,675,951	95.39%
Islamic Education	336	3.55%	84,446	6,916	0.08	92.43%	7.57%	91,362	1.54%
Teacher Training*	34	0.36%	9,156	5,138	0.56	64.05%	35.95%	14,294	0.24%
TVET	44	0.46%	9,321	1,045	0.11	89.92%	10.08%	10,366	0.17%
CBE			63,436	95,046	1.50	40.03%	59.97%	158,482	2.66%
Total	9,476	100.00%	3,834,221	2,116,234	0.55	64.44%	35.56%	5,950,455	1

A.4 School Survey 2007, National Level Summaries

Schools and Enrollment

There are 280,000 students enrolled in literacy courses. The literacy courses and Community Based Education classes were not surveyed.

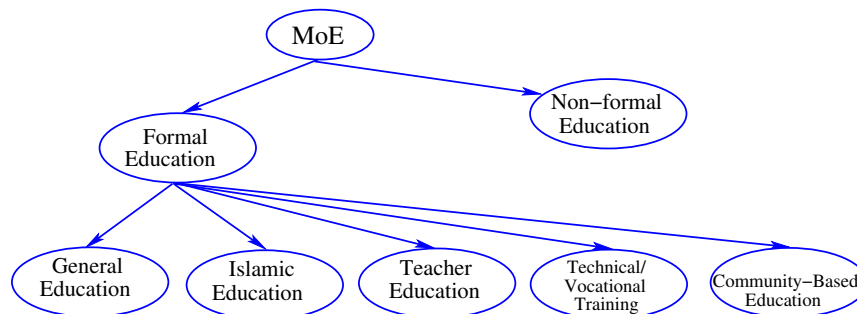
A.5 Teachers Summary

Table A.2 shows the numbers of teachers deployed over the branches of education in Afghanistan in 2007.

1,860 Officials are employed in the MoE Head Office and in the Provincial and District offices and 2,153 teachers are also deployed in these offices. Figure A.1

Table A.2: Teaching and other staff employed by the Ministry of Education

	Teachers Male	Teachers Female	M/F ratio	% Male	% Female	Teachers Total	%	% Teachers
Admin. General	1,860	293	0.16	86.39%	13.61%	2,153	1.44%	1.18%
Education	103,047	39,461	0.38	72.31%	27.69%	142,508	95.38%	78.15%
Islamic Education	3,420	54	0.02	98.45%	1.55%	3,474	2.33%	1.91%
Teacher Training	358	62	0.17	85.24%	14.76%	420	0.28%	0.23%
Technical/Vocational	670	192	0.29	77.73%	22.27%	862	0.58%	0.47%
Total	109,355	40,062	0.37	73.19%	26.81%	149,417	100.00%	81.94%

**Figure A.1:** Ministry of Education structure

shows the structure of the main activities of the MoE. The Ministry of Education is responsible for formal education and non-formal education. Formal education includes general education, Islamic education, teacher education, technical and vocational education, and community based education. Figure A.2 shows the relationship between form and non-formal education in the MoE.

The deployment of teachers by the MoE is illustrated in Figure A.3.

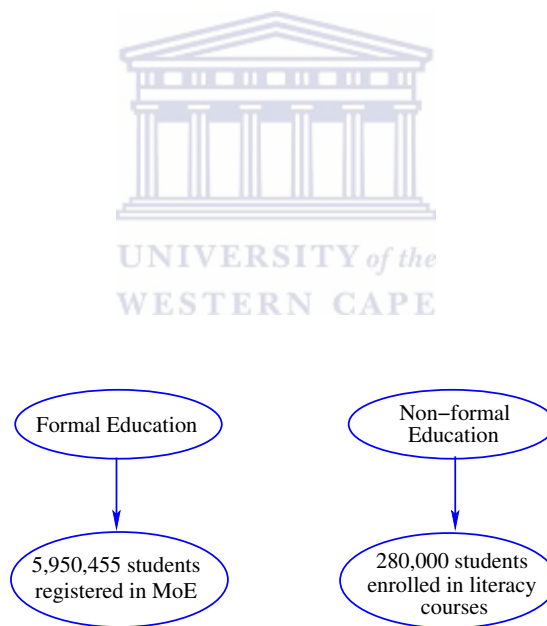


Figure A.2: Formal vs. non-formal education in Afghanistan

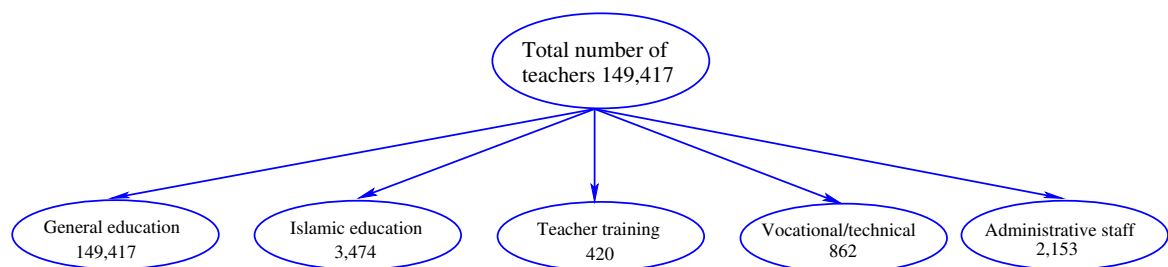


Figure A.3: Deployment of teachers in the Ministry of Education



Appendix B

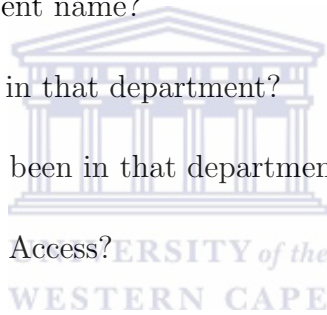
The questionnaires

B.1 First survey

The purpose of this questionnaire is to determine the need for an Education Management Information System (EMIS) and what form the EMIS will have?

Part 1 Background information

- 1.1 What is your department name?
- 1.2 What is your position in that department?
- 1.3 How long do you have been in that department?
- 1.4 Did you use Microsoft Access?



Part 2 Technical information

- 2.1 Do you know some programming language?
- 2.2 Did you use SQL?
- 2.3 Did you do computer Microsoft office?

Part 3 Some information about kind of EMIS

- 3.1 What type of information and statistics would you like an EMIS to provide for you?
- 3.2 What type of Interface is need for EMIS?
- 3.3 What type of data do you want to put in the database?
- 3.4 Can the interface is in the English language only or do you prefer a local language?

- 3.5 What are the current difficulties within the education system?
- 3.6 How can the data be used to help educational authorities improve education services?
- 3.7 Do data only flow from the local level up to the central level, or are data feedback to the service providers at the local level?
- 3.8 What structure will the database have?
- 3.9 What technique will be used to collect necessary information?
- 3.10 What technique has been used to collect necessary information?
- 3.11 Does each school have a computers?
- 3.12 Does each province have an internet connection?
- 3.13 How will the data be collected or estimated?
- 3.14 What software and hardware do they need in order to meet the needs for data collection and analysis?
- 3.15 Are these data (for example data about schools) included in national statistics?
- 3.16 Who has used the data?
- 3.17 who should analyze the data?

B.2 Second survey

B.2.1 Questionnaire 2

The purpose of the second survey was receiving feedback about the function my first version prototype.

Part 1 Background information

- 1.1 What is your department name?
- 1.2 What is your position in that department?
- 1.3 How long do you have been in that department?

1.4 did you create table and queries in Microsoft Access?

Part 2 Technical information

2.1 Can you work in Java programming language?

2.2 Can you work in Microsoft Access and in SQL server?

2.3 Do you know about software engineering?

Part 3 Some information about kind of EMIS

3.1 What is your think about the database function?

3.2 What do you think about the function of sorting object in EMIS tool?

3.3 Is that interface friendly to users?

3.4 Is the English interface better?

3.5 Which data is useful for data entry?

3.6 What do you think about the function of add button?

3.7 What do you think about the function of delete button?

3.8 What do you think about the function of search window?

3.9 What is the connection between interface and database?

3.10 What gaps are there in EMIS interface which design in English for Afghanistan?

3.11 Are additional types of information necessary?

3.12 What is your kink about database; the database has six field (ID, name, father name, class, roll number, city)?

B.3 The third survey

The purpose of the third survey was to capture feedback about the second version

Part 1 Background information

- 1 What is your department name?
- 2 What is your position in that department?
- 3 How long do you have been in that department?
- 4 Did you use SQL server?

Part 2 Technical information

- 1 Can you use Net beans IDE?
- 2 Can you create Use Case diagram?
- 3 Do you know about software engineering XP method?

Part 3 Some information about kind of EMIS

- 1 What is your think about the database function?
- 2 What do you think about the function of sorting button in EMIS interface?
- 3 Is that interface friendly to users?
- 4 Is the English interface better?
- 5 What do you think about add window function?
- 6 What do you think about sort window function?
- 7 What do you think about the function of delete button?
- 8 What do you think about the function of show window?
- 9 What do you think about the function and feature of search window?
- 10 What is the connection between interface and database?
- 11 What Gaps are there in this EMIS interface in Dari language for Afghanistan?

Appendix C

The Java code for the EMIS

```
package EMIS;

import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.sql.*;
import java.util.Vector;
import javax.swing.*;
import java.awt.*;

public class EMIS extends JFrame{
    private JButton Query_by_city;
    private JPanel querybycity;
    private JComboBox QueryActionComboBox;
    private JButton QueryActionButton;
    private JTextField QueryActionTextField;
    private String [] QueryActionComboBoxsItems={"City"};

    private JButton displayButton;
    private JButton searchButton;
    private JPanel searchPanel;
    private JButton searchActionButton;
    private JTextField searchActionTextField;
    private JComboBox searchActionComboBox;
    private String[] searchActionComboBoxsItems={" ",""," ",""," ",""," "};

    private JButton addButton;
    private JLabel freeLabel;
    private JLabel addActionLabel_ID,addActionLabel_Name,addActionLabel_FatherName,
        addActionLabel_Class,addActionLabel_RollNumber,addActionLabel_City;
    private JPanel addPanel;
    private JButton addActionButton;
    private JTextField addActionTextField_ID1,addActionTextField_ID,
        addActionTextField_Name,addActionTextField_FatherName,
        addActionTextField_Class,addActionTextField_RollNumber,
        addActionTextField_City ;
    private JButton deleteButton;
    private JPanel deletePanel;
    private JTextField deleteActionTextField;
    private JButton deleteActionButton;
    private JComboBox deleteActionComboBox;
    private String[]
        deleteActionComboBoxsItems={"ID","Name","Father_Name","Class",
            "Roll_Number","City"};

    private JButton sortButton;
    private JRadioButton asscendingRadio,desscendingRadio;
    private JLabel asscendingLabel,desscendingLabel;
    private ButtonGroup group;
    private JPanel sortPanel;
    private JComboBox sortActionComboBox;
    private JRadioButton radioButtonOfAsscending,radioButtonOfDesscending;
    private String[] sortActionComboBoxsItems={"ID","Name","Father_Name",
        "Class","Roll_Number","City"};

    public JTable tableOfStudents;
```

```

private JLabel IntrocuptionLabel;

private JPanel panelForButtonsOnTheTop;
private JPanel panelForOurIntroduction;

private Connection connection;
// private JTextField addActionTextField_RollNumber;
// private JTextField addActionTextField_City;

/** Creates a new instance of JavaDatabaseConn */
public EMIS() {
super("EMIS");
setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);

buildpanelForButtonsOnTheTop();
buildpanelForOurIntroduction();

theConnection();
setLayout(new BorderLayout());

add(panelForButtonsOnTheTop, BorderLayout.EAST);
panelForButtonsOnTheTop.setLayout(new GridLayout(5,1));

add(panelForOurIntroduction, BorderLayout.SOUTH);

setSize(450,300);
setVisible(true);
}

private void buildpanelForButtonsOnTheTop()
{
panelForButtonsOnTheTop=new JPanel();

displayButton=new JButton("Report");
displayButton.addActionListener(new ActionsOfButtons());
searchButton=new JButton("Scarch");
searchButton.addActionListener(new ActionsOfButtons());
addButton=new JButton("Add");
addButton.addActionListener(new ActionsOfButtons());
deleteButton=new JButton("Delete");
deleteButton.addActionListener(new ActionsOfButtons());
sortButton=new JButton("Sort");
sortButton.addActionListener(new ActionsOfButtons());
Query_by_city=new JButton("Query_By_City");
Query_by_city.addActionListener(new ActionsOfButtons());

displayButton.setForeground(Color.BLACK);
displayButton.setBackground(Color.PINK);
searchButton.setForeground(Color.BLACK);
searchButton.setBackground(Color.PINK);
addButton.setForeground(Color.BLACK);
addButton.setBackground(Color.PINK);
deleteButton.setForeground(Color.BLACK);
deleteButton.setBackground(Color.PINK);
sortButton.setForeground(Color.BLACK);
sortButton.setBackground(Color.PINK);
Query_by_city.setForeground(Color.BLACK);
Query_by_city.setBackground(Color.PINK);
panelForButtonsOnTheTop.add(displayButton);
panelForButtonsOnTheTop.add(searchButton);
panelForButtonsOnTheTop.add(addButton);
panelForButtonsOnTheTop.add(deleteButton);
panelForButtonsOnTheTop.add(sortButton);
// panelForButtonsOnTheTop.add(Query_by_city);

}

private void Image() {

```

```

    JButton b1=new JButton("Click Me");
    ImageIcon icon = new ImageIcon("ahmadsha.JPG");
    JLabel rockLabel = new JLabel("See the ");
    JLabel textLabel = new JLabel("ohhhh, a pretty label");

    Container canvas = getContentPane();
    canvas.setBackground(Color.white);
    b1.setBackground(Color.cyan);
    b1.setForeground(Color.white);
    b1.setFont(new Font("Ariel",Font.BOLD,22));
    b1.setToolTipText("Click the button, see what happens!");
    rockLabel.setToolTipText("This is a label with an icon.");
    textLabel.setToolTipText("This is a label in Times Roman font.");
    textLabel.setFont(new Font("TimesRoman",Font.ITALIC,26));

    textLabel.setForeground(Color.magenta);
    canvas.setLayout(new FlowLayout());
    canvas.add(rockLabel);
    canvas.add(textLabel);

    pack();
    setVisible(true);

}
public void actionPerformed(ActionEvent e)
{
    JOptionPane.showMessageDialog(this,"Thank you");
}

private void buildpanelForOurIntroduction()
{
    panelForOurIntroduction=new JPanel();
    IntrocuptionLabel=new JLabel("");
    panelForOurIntroduction.add(IntrocuptionLabel);
}

public void theConnection()
{
    try
    {
        Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
        connection=DriverManager.getConnection("jdbc:odbc:EMIS","Group A","a");
    }

    catch(ClassNotFoundException classNotFoundException)
    {
        JOptionPane.showMessageDialog(null,"class does not exist");
    }
    catch(SQLException except)
    {
        JOptionPane.showMessageDialog(null,"sql exception 1");
    }
}

public void buildResult(ResultSet r) throws SQLException
{
    Vector NamesOfFields;
    Vector theRows;
    ResultSetMetaData resultMetaData;
    boolean records=r.next();

    if(!records)

```

```

{
    JOptionPane.showMessageDialog(null,"no more records");
    return;
}

NamesOfFields=new Vector();
theRows=new Vector();

try
{
    resultMetaData = r.getMetaData();

    for(int i=1;i<=resultMetaData.getColumnCount();++i)
    {
        NamesOfFields.addElement(resultMetaData.getColumnName(i));
    }

    do
    {
        theRows.addElement(nextRow(r,resultMetaData));
    }while(r.next());

    tableOfStudents=new JTable(theRows,NamesOfFields);

    JScrollPane scrollBar=new JScrollPane(tableOfStudents);
    scrollBar.setEnabled(false);
    scrollBar.setPreferredSize(new Dimension(300,200));
    getContentPane().add(scrollBar,BorderLayout.CENTER);
    validate();
    pack();
}
catch(Exception e)
{
    JOptionPane.showMessageDialog(null,"Getting data from table has Failed");
}
}

private Vector nextRow(ResultSet r,ResultSetMetaData resultMetaData) throws
SQLException
{
    Vector row=new Vector();

    for(int i=1;i<=resultMetaData.getColumnCount();++i)
    {
        switch(resultMetaData.getColumnType(i))
        {
            case Types.VARCHAR:
                row.addElement(r.getString(i));
                break;
            case Types.INTEGER:
                row.addElement(new Long(r.getLong(i)));
                break;
            default:
                System.out.println("unsupported data type");
        }
    }
    return row;
}

private void searchButtonsWindow()
{
    JFrame searchFrame=new JFrame("");
    searchFrame.setDefaultCloseOperation(JFrame.HIDE_ON_CLOSE);
    searchPanel=new JPanel();
    searchActionComboBox=new JComboBox(searchActionComboBoxItems);
    searchActionComboBox.addActionListener(new ActionsOfButtons());
}

```



```

searchActionButton=new JButton("");
searchActionButton.addActionListener(new ActionsOfButtons());
searchActionTextField=new JTextField(10);
searchPanel.add(searchActionComboBox);
searchPanel.add(searchActionTextField);
searchPanel.add(searchActionButton);
searchFrame.add(searchPanel);
searchFrame.setVisible(true);
searchFrame.setResizable(false);
searchFrame.pack();
}
private void Query_by_cityButtonWindow()
{
    JFrame QueryFrame= new JFrame("Query_By_City");
    QueryFrame.setDefaultCloseOperation(JFrame.HIDE_ON_CLOSE);
    querybycity= new JPanel();
    QueryActionComboBox= new JComboBox(QueryActionComboBoxsItems);
    QueryActionComboBox.addActionListener(new ActionsOfButtons());
    QueryActionButton= new JButton("Query_By_City");
    QueryActionTextField= new JTextField(10);
    querybycity.add(QueryActionComboBox);
    querybycity.add(QueryActionTextField);
    querybycity.add(QueryActionButton);
    QueryFrame.add(querybycity);
    QueryFrame.setVisible(true);
    QueryFrame.setResizable(false);
    QueryFrame.pack();
}

private void deleteButtonsWindow()
{
    JFrame deleteFrame=new JFrame(" ");
    deleteFrame.setDefaultCloseOperation(JFrame.HIDE_ON_CLOSE);
    deletePanel=new JPanel();
    deleteActionComboBox=new JComboBox(deleteActionComboBoxsItems);
    deleteActionComboBox.addActionListener(new ActionsOfButtons());
    deleteActionTextField=new JTextField(10);
    deleteActionButton=new JButton(" ");
    deleteActionButton.addActionListener(new ActionsOfButtons());
    deletePanel.add(deleteActionTextField);
    deletePanel.add(deleteActionComboBox);
    deletePanel.add(deleteActionButton);
    deleteFrame.add(deletePanel);
    deleteFrame.setResizable(false);
    deleteFrame.setVisible(true);
    deleteFrame.pack();
}
private void addButtonWindow()
{
    JFrame addFrame=new JFrame(" ");
    addFrame.setDefaultCloseOperation(JFrame.HIDE_ON_CLOSE);
    addPanel=new JPanel();
    freeLabel=new JLabel(" ");
    // addActionLabel_ID=new JLabel(" ID1");
    addActionLabel_ID=new JLabel(" ");

    addActionLabel_Name=new JLabel("");
    addActionLabel_FatherName =new JLabel(" ");
    addActionLabel_Class=new JLabel("");
    addActionLabel_RollNumber=new JLabel(" ");
    addActionLabel_City=new JLabel("");
    //addActionTextField_ID1=new JTextField(10);

```

```

addActionTextField_ID=new JTextField(10);
addActionTextField_Name=new JTextField(10);
addActionTextField_FatherName=new JTextField(10);
addActionTextField_Class=new JTextField(10);
addActionTextField_RollNumber=new JTextField(10);
    addActionTextField_City=new JTextField(10);
addActionButton=new JButton("");

addActionButton.addActionListener(new ActionsOfButtons());
addActionButton.addActionListener(new ActionsOfButtons());
addPanel.add(addActionButton);
//addPanel.add(addActionLabel_ID1);

addPanel.add(addActionLabel_ID);
addPanel.add(addActionLabel_Name);
addPanel.add(addActionLabel_FatherName);
addPanel.add(addActionLabel_Class);
addPanel.add(addActionLabel_RollNumber);
addPanel.add(addActionLabel_City);

addPanel.add(freeLabel);
// addPanel.add(addActionTextField_ID1);

addPanel.add(addActionTextField_ID);
addPanel.add(addActionTextField_Name);
addPanel.add(addActionTextField_FatherName);
addPanel.add(addActionTextField_Class);
addPanel.add(addActionTextField_RollNumber);
addPanel.add(addActionTextField_City);
addPanel.setLayout(new GridLayout(2,7));
addFrame.add(addPanel);
addFrame.setResizable(false);
addFrame.setVisible(true);
addFrame.pack();
}
private void sortButtonsWindow()
{
    JFrame sortFrame=new JFrame(" ");
    sortFrame.setDefaultCloseOperation(JFrame.HIDE_ON_CLOSE);

    sortPanel=new JPanel();

    sortActionComboBox=new JComboBox(sortActionComboItems);
    sortActionComboBox.addActionListener(new ActionsOfButtons());
    sortPanel.add(sortActionComboBox);
    group=new ButtonGroup();
    asscendingRadio=new JRadioButton();
    asscendingRadio.addActionListener(new ActionsOfButtons());
    desscendingRadio=new JRadioButton();
    desscendingRadio.addActionListener(new ActionsOfButtons());
    asscendingLabel=new JLabel("");
    desscendingLabel=new JLabel("");
    group.add(asscendingRadio);
    group.add(desscendingRadio);
    sortPanel.add(asscendingLabel);
    sortPanel.add(asscendingRadio);
    sortPanel.add(desscendingLabel);
    sortPanel.add(desscendingRadio);
    sortFrame.add(sortPanel);
    sortFrame.setResizable(false);
    sortFrame.setVisible(true);
    sortFrame.pack();
}

private class ActionsOfButtons implements ActionListener

```

```

{
public void actionPerformed(ActionEvent e)
{
    if(e.getSource()==searchButton)
    {
        searchButtonsWindow();
    }
    else if(e.getSource()==addButton)
    {
        addButtonsWindow();
    }
    else if(e.getSource()==sortButton)
    {
        sortButtonsWindow();
    }

    else if(e.getSource()==deleteButton)
    {
        deleteButtonsWindow();
    }
    else if(e.getSource()==Query_by_city)
    {
        Query_by_cityButtonWindow();
    }

    if(e.getSource()==searchActionButton)
    {
        if(searchActionComboBox.getSelectedIndex()==0)
        {
            Statement statement;
            ResultSet result;

            try
            {
                statement=connection.createStatement();
                result = statement.executeQuery("SELECT * FROM tblStudents
                    WHERE ID=" + searchActionTextField.getText() + "");
                buildResult(result);
                statement.close();
            }
            catch(Exception ex)
            {
                JOptionPane.showMessageDialog(null,"Executing the query has Failed");
            }
        }
        else if(searchActionComboBox.getSelectedIndex()==1)
        {
            Statement statement;
            ResultSet result;

            try
            {
                statement=connection.createStatement();
                result = statement.executeQuery("SELECT * FROM tblStudents
                    WHERE Name='" + searchActionTextField.getText() + "'");
                buildResult(result);
                statement.close();
            }
            catch(Exception ex)
            {
                JOptionPane.showMessageDialog(null,"Executing the query has Failed");
            }
        }
        else if(searchActionComboBox.getSelectedIndex()==2)
        {
            Statement statement;

```

```

ResultSet result;

try
{
    statement = connection.createStatement();
    result = statement.executeQuery("SELECT * FROM tblStudents
        WHERE Father_Name='" + searchActionTextField.getText() + "'");
    buildResult(result);
    statement.close();
}
catch(Exception ex)
{
    JOptionPane.showMessageDialog(null,"Executing the query has Failed");
}
}

if(searchActionComboBox.getSelectedIndex()==3)
{
    Statement statement;
    ResultSet result;

    try
    {
        statement=connection.createStatement();
        result = statement.executeQuery("SELECT * FROM tblStudent
            WHERE Class='" + searchActionTextField.getText() + "'");
        buildResult(result);
        statement.close();
    }
    catch(Exception ex)
    {
        JOptionPane.showMessageDialog(null,"Executing the query has Failed");
    }
}

else if(searchActionComboBox.getSelectedIndex()==4)
{
    Statement statement;
    ResultSet result;

    try
    {
        statement=connection.createStatement();
        result = statement.executeQuery("SELECT * FROM tblStudents
            WHERE Roll_Namber=" + searchActionTextField.getText() + "");
        buildResult(result);
        statement.close();
    }
    catch(Exception ex)
    {
        JOptionPane.showMessageDialog(null,"Executing the query haFailed");
    }
}

}

else if(e.getSource()==QueryActionButton)
{
    //if(QueryActionComboBox.getSelectedIndex()==0){
    Statement statement;
    ResultSet result;
    try
    {
        statement=connection.createStatement();
        result = statement.executeQuery("SELECT * FROM tblStudents
            WHERE City='Kabul'");
        buildResult(result);
        statement.close();
    }
}
}

```

```

catch(Exception ex)
{
    JOptionPane.showMessageDialog(null,"Executing the query his Failed");
}
// }

}
else if(e.getSource()==deleteActionButton)
{
    if(deleteActionComboBox.getSelectedIndex()==0)
    {
        Statement statement;
        ResultSet result;

        try
        {
            statement=connection.createStatement();
            statement.executeUpdate("DELETE * FROM tblStudents
                WHERE ID=" + deleteActionTextField.getText() + "");
            result=statement.executeQuery("SELECT * FROM tblStudents");
            buildResult(result);
            statement.close();
        }
        catch(Exception ex)
        {
            JOptionPane.showMessageDialog(null,"Executing the query has Failed");
        }
    }
    else if(deleteActionComboBox.getSelectedIndex()==1)
    {
        Statement statement;
        ResultSet result;

        try
        {
            statement=connection.createStatement();
            statement.executeUpdate("DELETE * FROM tblStudents
                WHERE Name='" + deleteActionTextField.getText() + "'");
            result=statement.executeQuery("SELECT * FROM tblStudents");
            buildResult(result);
            statement.close();
        }
        catch(Exception ex)
        {
            JOptionPane.showMessageDialog(null,"Executing the query has Failed");
        }
    }
    else if(deleteActionComboBox.getSelectedIndex()==2)
    {
        Statement statement;
        ResultSet result;

        try
        {
            statement=connection.createStatement();
            statement.executeUpdate("DELETE * FROM tblStudents
                WHERE Father_Name='" + deleteActionTextField.getText() + "'");
            result=statement.executeQuery("SELECT * FROM tblStudents");
            buildResult(result);
            statement.close();
        }
        catch(Exception excep)
        {
            JOptionPane.showMessageDialog(null,"Executing the query ha Failed");
        }
    }
    if(deleteActionComboBox.getSelectedIndex()==3)

```

```

{
Statement statement;
ResultSet result;

try
{
statement=connection.createStatement();
statement.executeUpdate("DELETE * FROM tblStudents
WHERE Class='" + deleteActionTextField.getText() + "'");
result=statement.executeQuery("SELECT * FROM tblStudents");
buildResult(result);
statement.close();
}
catch(Exception ex)
{
JOptionPane.showMessageDialog(null,"Executing the query has Failed");
}
}
else if(deleteActionComboBox.getSelectedIndex()==4)
{
Statement statement;
ResultSet result;

try
{
statement=connection.createStatement();
statement.executeUpdate("DELETE * FROM tblStudents Roll_Number="
+ deleteActionTextField.getText() + "");
result = statement.executeQuery("SELECT * FROM tblStudents");
buildResult(result);
statement.close();
}
catch(Exception ex)
{
JOptionPane.showMessageDialog(null,"Executing the query has Failed");
}
}
else if(deleteActionComboBox.getSelectedIndex()==5)
{
Statement statement;
ResultSet result;
try
{
statement=connection.createStatement();
statement.executeUpdate("DELETE * FROM tblStudents City='"
+ deleteActionTextField.getText() + "'");
result=statement.executeQuery("SELECT * FROM tblStudents");
buildResult(result);
statement.close();
}
catch(Exception ex)
{
JOptionPane.showMessageDialog(null,"Executing the query has Failed");
}

}

}
else if(e.getSource()==addActionButton)
{
Statement statement;
ResultSet result;
try
{
statement=connection.createStatement();
statement.executeUpdate("INSERT INTO tblStudents

```

```

VALUES (" + addActionTextField_ID.getText() + ",'"
+ addActionTextField_Name.getText() + "','"
+ addActionTextField_FatherName.getText() + "','"
+ addActionTextField_Class.getText() + "','"
+ addActionTextField_RollNumber.getText() + "','"
+ addActionTextField_City.getText() + "')");
JOptionPane.showMessageDialog
    (null,"You have succsesfully added to the database");
result = statement.executeQuery("SELECT * FROM tblStudents");
buildResult(result);
statement.close();
}
catch(Exception exc)
{
    JOptionPane.showMessageDialog(null,"Executing the query has Failed");
}
}
else if(e.getSource()==displayButton)
{
    Statement statement;
    ResultSet result;
    try
    {
        statement=connection.createStatement();
        result=statement.executeQuery("Select * From tblStudents");
        buildResult(result);
        statement.close();
    }
    catch(Exception exce)
    {
        JOptionPane.showMessageDialog(null,"Executing the query has failed");
    }
}
else if(sortActionComboBox.getSelectedIndex()==0)
{
    if(e.getSource()==ascendingRadio)
    {
        Statement statement;
        ResultSet result;

        try
        {
            statement=connection.createStatement();
            result=statement.executeQuery("SELECT * FROM tblStudents ORDER BY ID ASC");
            buildResult(result);
            statement.close();
        }
        catch(Exception ex)
        {
            JOptionPane.showMessageDialog(null,"Executing the query ha Failed");
        }
    }
    else if(e.getSource()==desscendingRadio)
    {
        Statement statement;
        ResultSet result;

        try
        {
            statement = connection.createStatement();
            result = statement.executeQuery("SELECT *
            FROM tblStudents ORDER BY ID DESC");
            buildResult(result);
            statement.close();
        }
        catch(Exception ex)
    }
}

```

```

    {
        JOptionPane.showMessageDialog(null,"Executing the query has Failed");
    }
}
else if(sortActionComboBox.getSelectedIndex()==1)
{
    if(e.getSource()==ascendingRadio)
    {
        Statement statement;
        ResultSet result;

        try
        {
            statement = connection.createStatement();
            result = statement.executeQuery
                ("SELECT * FROM tblStudents ORDER BY Name ASC");
            buildResult(result);
            statement.close();
        }
        catch(Exception ex)
        {
            JOptionPane.showMessageDialog(null,"Executing the query has Failed");
        }
    }
    else if(e.getSource()==dassingRadio)
    {
        Statement statement;
        ResultSet result;

        try
        {
            statement = connection.createStatement();
            result = statement.executeQuery("SELECT * FROM tblStudents
                ORDER BY Name DESC");
            buildResult(result);
            statement.close();
        }
        catch(Exception ex)
        {
            JOptionPane.showMessageDialog(null,"Executing the Failed");
        }
    }
}

else if(sortActionComboBox.getSelectedIndex()==2)
{
    if(e.getSource()==ascendingRadio)
    {
        Statement statement;
        ResultSet result;

        try
        {
            statement = connection.createStatement();
            result = statement.executeQuery("SELECT * FROM tblStudents
                ORDER BY Father_Name ASC");
            buildResult(result);
            statement.close();
        }
        catch(Exception ex)
        {
            JOptionPane.showMessageDialog(null,"Executing the query has Failed");
        }
    }
    else if(e.getSource()==dassingRadio)
    {

```



```

Statement statement;
ResultSet result;

try
{
    statement = connection.createStatement();
    result = statement.executeQuery("SELECT * FROM tblStudents
        ORDER BY Father_Name DESC");
    buildResult(result);
    statement.close();
}
catch(Exception ex)
{
    JOptionPane.showMessageDialog(null,"Executing the query has Failed");
}
}
}
else if(sortActionComboBox.getSelectedIndex()==3)
{
    if(e.getSource()==ascendingRadio)
    {
        Statement statement;
        ResultSet result;

        try
        {
            statement = connection.createStatement();
            result = statement.executeQuery("SELECT * FROM tblStudents
                ORDER BY Class ASC");
            buildResult(result);
            statement.close();
        }
        catch(Exception ex)
        {
            JOptionPane.showMessageDialog(null,"Executing the query ha Failed");
        }
    }
    else if(e.getSource()==desscendingRadio)
    {
        Statement statement;
        ResultSet result;

        try
        {
            statement = connection.createStatement();
            result = statement.executeQuery("SELECT * FROM tblStudents
                ORDER BY Class DESC");
            buildResult(result);
            statement.close();
        }
        catch(Exception ex)
        {
            JOptionPane.showMessageDialog(null,"Executing the query has Failed");
        }
    }
}
}
else if(sortActionComboBox.getSelectedIndex()==4)
{
    if(e.getSource()==ascendingRadio)
    {
        Statement statement;
        ResultSet result;

        try
        {
            statement = connection.createStatement();
            result = statement.executeQuery("SELECT * FROM tblStudents
                ORDER BY Roll_Number ASC");
        }
    }
}
}

```

```

        buildResult(result);
        statement.close();
    }
    catch(Exception ex)
    {
        JOptionPane.showMessageDialog(null,"Executing the query ha Failed");
    }
}
else if(e.getSource()==desscendingRadio)
{
    Statement statement;
    ResultSet result;

    try
    {
        statement = connection.createStatement();
        result = statement.executeQuery("SELECT * FROM tblStudents
            ORDER BY Roll_Number DESC");
        buildResult(result);
        statement.close();
    }
    catch(Exception ex)
    {
        JOptionPane.showMessageDialog(null,"Executing the query has Failed");
    }
}
}
else if(sortActionComboBox.getSelectedIndex()==5)
{
    if(e.getSource()==ascendingRadio)
    {
        Statement statement;
        ResultSet result;

        try
        {
            statement = connection.createStatement();
            result = statement.executeQuery("SELECT * FROM tblStudents
                ORDER BY City ASC");
            buildResult(result);
            statement.close();
        }
        catch(Exception ex)
        {
            JOptionPane.showMessageDialog(null,"Executing the query has Failed");
        }
    }
    else if(e.getSource()==desscendingRadio)
    {
        Statement statement;
        ResultSet result;

        try
        {
            statement = connection.createStatement();
            result = statement.executeQuery("SELECT * FROM tblStudents
                ORDER BY City DESC");
            buildResult(result);
            statement.close();
        }
        catch(Exception ex)
        {
            JOptionPane.showMessageDialog(null,"Executing the Failed");
        }
    }
}
}
}

```

```
    }  
    public static void main(String[] args) {  
        EMIS j=new EMIS();  
    }  
}
```





UNIVERSITY *of the*
WESTERN CAPE

Bibliography

- Ajayi, I. and Omirin, F. F. (2007). The use of Management Information Systems (MIS) in decision making in the South West Nigerian universities.
- Beck, K. and Andres, C. (2002). *eXtreme Programming explained: embrace change*. Addison-Wesley, Reading, MA, 2nd edition.
- Bevier, W. R., Warren A. Hunt, J., Moore, J. S., and Young, W. D. (1989). An approach to systems verification. *Journal of Automated Reasoning*, 5(4).
- Blower, N. G. (2004). Investigation into XP Techniques applied to the Software Hut. Honours thesis in Computer Science. University of Sheffield, UK.
- Boehm, B. (2002). Get ready for Agile methods, with care. *Computer*, 35:64–69.
- Bose, S., Kurhekar, M., and Ghoshal, J. (2008). Agile methodology in requirements engineering. SetLabs Briefings Online. URL <http://www.infosys.com/research/publications/Documents/agile-requirements-engineering.pdf>.
- Brunette, H. C. (2006). *Technical Education in Namibia: Past trends, present circumstances and future prospects*. PhD thesis, University of the Orange Free State, Bloemfontein, South Africa.
- Burgers, J. (2001). *Appraisal of Guidelines for Research and Evaluation Instruments (AGREE)*. The AGREE collaboration.
- Cambridge Education (2006). *Education management Information system: A short case study of Mozambique*. Infodev. Michael Trucan (Ed.). Prepared by Cambridge Education.
- Carrizo, L., Sauvageot, C., and Bella, N. (2003). Information tools for the preparation and monitoring of education plans. *UNESCO: Education policies and strategies*, 5:1–121.
- Cassidy, T. (2005). *Education Management Information System (EMIS) Development in Latin America and the Caribbean: Lessons and Challenges*. IDB,

- Washington, DC. A report prepared for the IADB Caribbean Sub-regional Education Policy Dialogue Meeting.
- Chapman, D. (1991). Education data quality in the Third World: A five country study. *International Review of Education*, 37(3):365–379.
- Cohen, D., Lindvall, M., and Costa, P. (2004). An introduction to Agile methods.
- Dale, N., Joyce, D. T., and Weems, C. (2006). *Object oriented data structures using Java*. Jones and Bartlett Pub., 2nd edition.
- Dawson, S. and Manderson, L. (1993). A manual for the use of focus groups. Technical report, International Nutrition Foundation for Developing countries (INFDC), Boston, MA.
- DeMarco, T. and Boehm, B. (2002). The Agile methods fray. *Computer*, 35:90–92.
- Ellison, R. (2004). A practical guide to working with Education Management Information Systems. Technical report, UK: Department for International Development, London. Statistical consultant under contract to DFID (First draft).
- Hayes, S. and Andrews, M. (2003). An introduction to Agile Methods. URL <http://www.scribd.com/doc/6736392/Agile-Methods>.
- Hekmat, S. (2005). UML process. PragSoft Corporation. URL <http://www.progsoft.com>.
- Highsmith, J. (2002). *Agile Software Development Ecosystems*. Addison-Wesley, Boston, MA.
- Hodges, S. and Shahram, I. (2006). *wasp: a platform for prototyping ubiquitous computing devices*. Microsoft Research, Cambridge.
- Holcombe, M., Gheorghe, M., and Macias, F. (2001). Teaching xp for real: Some initial observations and plans. In *Proc. XP 2001*, pages 14–17, Sardinia, Italy.
- Hua, H. and Herstein, J. (2002). Education management information system (EMIS): Integrated data and information systems and their implications in education management. In *2003 Annual Conference of Comparative and International Education Society*, pages 1–26, New Orleans, LA.

- James, D. J., Johnson, D. H., and Caristi, J. (2001). Extreme programming and the software design course. In *Proceedings of XP Universe*, pages 273–285. Addison-Wesley.
- Kaul, A. and Neelam, R. (2009). *Elementary Education in India*. National University of Educational Planning and Administration, New Delhi, India.
- Khan, M. M. S. and Sethi, A. (2009). Management education and corporation governance: A case of India and Pakistan. In *2009 Oxford Business and Economics Conference Program*. Oxford, UK.
- Koskela, J., Myllyaho, M., Kääriäinen, J., Bendas, D., Hyysalo, J., and Virta, A. (2004). Experiences of using extreme programming to support a legacy information system migration project. VTT Technical Research Center of Finland.
- Kruchten, P. (2004). *The Rational Unified Process: An Introduction*. Addison-Wesley, Boston, MA.
- Krueger, R. A. (2002). Designing and conducting focus group interviews. Technical report, University of Minnesota, St. Paul, MN.
- Krueger, R. A. and Casey, M. A. (2000). *Focus groups: A Practical Guide for Applied Research*. Sage Publications, Thousand Oaks, CA, 3rd edition.
- Lafore, R. (2003). *Data structures and Algorithms in Java*. SAMS Publishing, 2nd edition.
- Lindvall, M., Basili, V., Boehm, B., Costa, P., Dangle, K., Shull, F., Tesoriero, R., Williams, L., and Zelkowitz, M. (2002). Empirical findings in Agile Methods. In *Proc. Extreme Programming and Agile Methods—XP/Agile Universe 2002*, pages 197–207. Springer-Verlag.
- Lloyd, C. B. and Hewett, P. C. (2003). *Primary schooling in sub-Saharan Africa: Recent trends and current challenges*. Policy Research Division Working Paper no. 176. Population Council, New York, NY.
- Medvidovic, N., Rosenblum, D. S., Redmiles, D. F., and Robbins, J. E. (2002). Modeling software architectures in the unified modeling language. *ACM Transactions on Software Engineering and Methodology*, 11(1):2–57.

- Mehta, A. C. (1998a). Information requirements for educational planning: Limitations and gaps in the existing system. I. *Journal of Education and Social Change*, XI(4).
- Mehta, A. C. (1998b). Information requirements for educational planning: Limitations and gaps in the existing system. II. *Journal of Education and Social Change*, XII(1).
- Mellor, J. S. (2005). Adapting Agile approaches to your project needs. *IEEE Software*, 22(3):17–20.
- Melnik, G. and Maurer, F. (2005). A cross-program investigation of student’s perceptions of Agile methods. In *Proc. 27th International Conference on Software Engineering New York*. ACM Press.
- Ministry of Education Pakistan (2005). Educational statistics capacity-building programmed to support the Islamic Republic of Pakistan. Jointly prepared by the UNESCO Institute for Statistics and The National Technical Committee on Educational Statistics Capacity-Building. Funded by the European Commission.
- Moses, K. D. (2001). Education management information system: What is it and why do we not have more of it? *TechknowLogia*. URL http://www.TechKnowLogia.org/TKL_Articles/PDF/221.pdf.
- Nijkamp, B. (2009). Using functional design patterns to improve Exports Management Information Systems. *IEEE Transactions on Professional Communication*, 40(4):310–312.
- Pender, T. (2002). *UML 2*. Wiley Publishing Inc, USA.
- Pressman, R. S. (2001). *Software engineering a practitioner’s approach*. R.S. Pressman and Associates, Boston, MA.
- Rauch, J. and Edwards, S. (2002). Data processing procedures. California Health Interview survey.
- Santiago Cueto (2005). *Empirical information and the development of educational policies in Latin America*. IDB, Washington, DC. A report prepared for the IADB Caribbean Sub-regional Education Policy Dialogue Meeting.

- Schneider, J.-G. and Johnston, L. (2003). eXtreme programming at universities—an educational perspective. In *Proceedings of 25th International Conference on Software Engineering*, pages 594–599, Portland, OR. IEEE Computer Science.
- Schwaber, K. and Beedle, M. (2002). *Agile Software Development with SCRUM*. Prentice-Hall, Upper Saddle River, NJ.
- Strode, D. E. (2005). The agile methods: An analytical comparison of five agile methods and an investigation of their target environment. Technical report, Massey University, Palmerston North, New Zealand.
- Strode, D. E. (2006). Agile methods: a comparative analysis. Proc. 19th Annual Conference of the National Advisory Committee on Computing Qualifications (NACCQ 2006). Wellington, New Zealand.
- Thomas, F. C., Khan, A., and Noonan, R. D. (1993). National Education Management Information System, prepared for UNDP, UNESCO and the Government of Pakistan. Prepared for UNDP, UNESCO and the Government of Pakistan.
- Villanueva, C. C. (2003). Education Management Information System (EMIS) and the formulation of Education for All (EFA) plan of action, 2002–2015. In Cooperation with UNESCO Almaty Cluster Office and the Ministry of Education of Tajikistan.
- Voigts, F. G. G. (2005). *Development of an Education Management Information System (EMIS) in Namibia*. Ministry of Basic Education and Culture, Namibia.
- Wahono, R. S., Far, B. H., and Cheng, J. (2002). A framework of object identification and refinement process in object-oriented analysis and design. In *Proceedings of the 1st IEEE International Conference on Cognitive Informatics, ICCI 2002*, Calgary, Canada.
- Wako, T. N. (2003). *Education Management Information Systems*. NE-SIS/UNESCO, Harare, Zimbabwe.
- Williams, L. (2007). A survey of agile development methodologies. *UNKNOWN*, pages 209–227.

- Williams, L. A. and Kessler, R. (2001). Experimenting with industry's "Pair-Programming" model in the computer science classroom. *Computer Science Education*, 2001, 11(1):7–20.
- Wong, L. P. (2008). Focus group discussion: a tool for health and medical research. *Singapore Med J.*, 49(3):256–261.
- Zadel, M. and Scavone, G. (2006). Different strokes: a prototype software system for laptop performance and improvisation. In *Proceedings of the 2006 conference on New interfaces for musical expression*, NIME '06, pages 168–171. IRCAM & Centre Pompidou.
- Zhao, Y. and McNerney, F. (2006). Assessing the impact of a multi-agency project on Afghan basic education. In *American Educational Research Association Annual Conference April 7–11, 2006*, San Francisco, CA.

