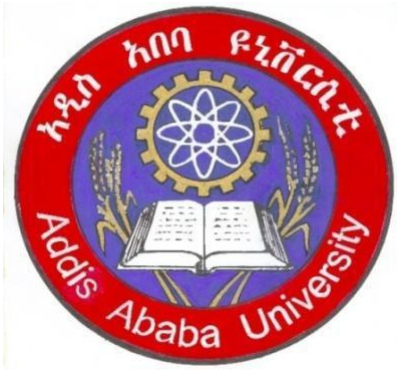


ADDIS ABABA UNIVERSITY

SCHOOL OF COMMERCE

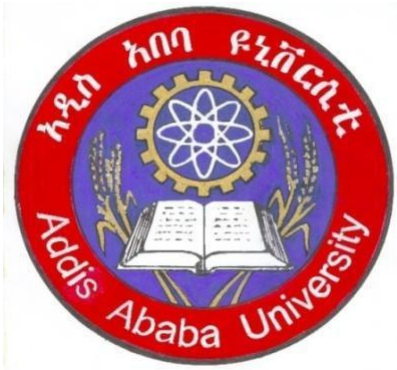
DEPARTMENT OF PROJECT MANAGEMENT

EFFECT OF AGILE SOFTWARE DEVELOPMENT PRACTICES ON SCOPE MANAGEMENT: THE CASE OF JSI (JOHN SNOW INC.)

By: Ehetemariam Moges

June, 2018

Addis Ababa, Ethiopia



EFFECT OF AGILE SOFTWARE DEVELOPMENT PRACTICES ON SCOPE MANAGEMENT: THE CASE OF JSI (JOHN SNOW INC.)

A PROJECT WORK SUBMITTED TO ADDIS ABABA UNIVERSITY SCHOOL OF COMMRCE IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF MASTER OF ARTS IN PROJECT MANAGEMENT

BY: EHETEMARIAM MOGES

ADVISOR: TEKLEGIORGIS ASSEFA (ASST.PROF.)

JUNE,2018

ADDIS ABABA, ETHIOPIA

ADDIS ABABA UNIVERSITY

SCHOOL OF COMMERCE

DEPARTMENT OF PROJECT MANAGEMENT

EFFECT OF AGILE SOFTWARE DEVELOPMENT PRACTICES ON SCOPE MANAGEMENT: THE CASE OF JSI (JOHN SNOW INC.)

BY

EHETEMARIAM MOGES

Approval Board Committee:

_____	_____
Examiner	**Signature**
_____	_____
Examiner	**Signature**
_____	_____
Advisor	**Signature**

Statement of Declaration

I, the undersigned, declare that the study entitled *"Effect of Agile Software Development Practices on Scope Management: The case of JSI (John Snow Inc.)"* is the outcome of my own effort and study. I have conducted this study independently with the guidance and comments of my advisor. All sources of materials used for the study have been duly acknowledged. This study has not been presented for a degree in any university.

Ehetemariam Moges

Signature: _____

Date: _____

Advisor: Teklegiorgis Assefa (Asst.Prof.)

Signature: _____

Date: _____

Acknowledgement

First and foremost, I would like to thank the Almighty God, for giving me the strength and ability to start and complete this class.

I would like to express my very great appreciation to Teklegiorgis Assefa (Asst.Prof.) for his valuable and constructive suggestions during the planning and development of this research work. His willingness to give his time so generously, and his availability at all times has been very much appreciated.

I would also like to thank my fellow colleagues at JSI, the project manager and all the MIS team members of the projects for dedicating their time in answering my questions presented regarding the topic of the study and also giving me their support in every other way I needed.

Last but not list am very grateful for my family and friends for their continual encouragement and support. My special thanks goes to My better half Mesele Teferi, without whose constant support I never could have gotten this far.

Contents

List of Tables

List of Figures

Acronyms

CMS	Contract Management System
FE	Facility Edition
FMHACA	Food, Medicine and Health Administration and Control Authority
HCMIS	Health Commodity Management Information System
HE	Hub Edition
JSI	John Snow Inc.
MIS	Management Information System
PFSA	Pharmaceutical Fund and Supply Agency
PMBOK	Project Management Book of Knowledge
PMI	Project Management Institute
USAID	United States Agency for International Development

Abstract

The Project 'Effect of Agile Software Development Practices on Scope Management: The case of JSI (John snow Inc.)' begins by raising the question how projects are managed at JSI. Due to cost and time constraints the scope of this study is delimited to JSI delivered projects only. The research method used after thorough literature review is open ended interview and participatory observation. The sampling method used is a purposive non-random sampling. The participants are selected from the management information system team including project manager, senior software developers, software developers, product managers, analysists, and hub it supports. The qualitative data analysis is done from the input from participants and literatures reviews. The main findings are that JSI delivered and is delivering software's currently in use by the health sector. It follows agile software development, and it helps for a flexible scope management purpose. The main factors for agile software development related to scope management are Development guide by test, Objective documentation, Small teams led by the facilitator, Features defined by the customer, Stand-up meetings, Frequent releases and Lead-programmer. Finally, further research is recommended not only in the area of scope management but also agile software development practice in Ethiopian context, as this study doesn't cover it all.

Key words: *project scope, scope management, agile software management, project management knowledge, software development.*

CHAPTER ONE
INTRODUCTION

1.1 Introduction

In project management, a project is a temporary endeavor undertaken to create a unique product, service or result. A project is temporary in that it has a defined beginning and end in time, and therefore defined scope and resources. The development of software for an improved business process is one example of Project.

Software project management is the art and science of planning and leading software projects. It is a sub-discipline of project management in which software projects are planned, implemented, monitored and controlled.

Project management is also the activity of organizing and managing project resources and constraints with the aim of producing a successful completion and achievement of specific project goals and objectives. Every project is constrained in different ways by its scope, time and cost. These constraints are known as the triple constraint.

The scope constraint refers to the activities that need to be done to ensure that the project deliverables have been reached. It is a statement that defines the boundaries of the project. There have been various names given to scope. It is referred to as functional specification in the systems industry, statement of work in the engineering profession amongst others.

A change in one constraint can require a change in another constraint in order to restore balance of the project (Robert K. Wysocki, 2014).

The scope triangle practically allows the project manager to ask the question ‘who owns what?’ The client and senior management own time budget and resources.

The PMBoK guide presents project management as a set of ten knowledge areas, being the scope management responsible for delimiting what will be done in the project, defining a group of processes responsible for ensuring that all the work needed, and only what is necessary, to complete the project successfully, is carried out (PMBoK, 2013). In this context, the objective of the scope management is to control product and project boundaries, which can be a complex

activity because the boundaries are not always clear and well defined and may involve political, social, technological, organizational and economic forces (Alexander, G.M., Wilcox, T., Woods, R., 2009). It is worth mentioning that small variations in scope can cause costly impacts in different areas, as time, cost and quality (PMBoK, 2013; Solemon, Badariah ; Sahibuddin, Shamsul ; Ghani, Abdul Azim Abd.,2009).

According to Smith (2002) one of the main problems in information technology projects is related to the system requirements. Errors in the requirements are costly and can lead to loss of time, revenue and reputation of the responsible organization. Furthermore, when considering the correction of these requirements when they have already been implemented, the cost associated with correcting errors could generate even greater impacts under the project budget (Badariah *et al*, 2009).

The elaboration of requirements involves several stakeholders that are directly or indirectly affected by the project. These stakeholders have different experiences and expectations with the project. Thus, the requirements analysis process must be performed completely, because these stakeholders may not be able to define exactly what they actually need. So, they can express their needs incompletely, which increases the probability of failures on project scope (Sommerville, 2011). Therefore, this is why this study is focused on the scope management of software development Projects.

1.2 Background of the Project

This research will try to study the practical scope management followed by JSI (John Snow Inc.) starting from the inception of HCMIS (Health Commodity Management Information System) to the current IImport and the future plans.

On any day, the Pharmaceuticals Fund and Supply Agency (PFSA) in Ethiopia manages thousands of transactions: orders, receipts, issues, and transfers health commodities, which are the lifeblood of the public sector healthcare supply chain. PFSA, a Federal Ministry of Health (FMOH) agency, ensures that all public-sector health facilities have the medicines and health commodities they need. Each transaction moves commodities one step closer to their end destination: -the patients.

To inform supply decisions, and to ensure the right quantities of the right commodities are going to the right locations, PFSA uses a locally developed warehouse and logistics information management system: The Health Commodity Management Information system (HCMIS).

Health Commodity Management Information System (HCMIS) is a warehouse and facility management system developed by the USAID | DELIVER PROJECT.

HCMIS is used by both the Ethiopian Ministry of Health and the Pharmaceutical Fund & Supply Agency (PFSA) to implement and support the country's pharmaceutical logistics management system**.** HCMIS a locally developed, user-friendly inventory and information management software has been implemented in almost 500 health centers and hospitals throughout Ethiopia. HCMIS is helping health facilities to improve commodity management, data visibility, and overall performance.

HCMIS is designed as an inventory management system that helps health facilities track their commodities. However, the application is more than just an inventory tracking system. It includes major components of the Logistics Management Information System in which it provides the user site ability to:

- Track consumption rates,
- Manage re-order levels,
- Enforce FEFO and

- Provide many advanced logistics reports that allow proper facility and warehouse management rules to be enforced by the facility

Since HCMIS JSI has widened its scope and went on developing more interrelated software's:

- Directory Services create a common repository for items that can be shared by a number of systems.
- Mbrana (mobile application for vaccine items and malaria from Hub to Woreda to Health Centers),
- iImport (mobile application to sync Purchase Orders from Center to FMHACA)
- fanos (a web application which is a dashboard to show the overall status of each hub for the donors and all other interested parties),
- Forecasting & Procurement (desktop App for PFSA to handle forecasting and procuring of new items (drugs)
- HCMIS Mobile (Mobile application to handle the process starting from port to transit to center)
- CMS (contract management system)

1.3 Statement of the Problem

For decades as information technology (IT) projects grew bigger and more complex, project failures seemed to become increasingly common, in spite of intense efforts to apply traditional project planning. Those traditional planning tools focused on balancing the triple constraints of cost, schedule, and scope to create a plan. Then those tools unsuccessfully focused on delivering the planned scope within the planned cost and schedule.

According to Smith (2002) one of the main problems in information technology projects is related to the system requirements.

Requirements continue to change in response to customer business needs, changes in the industry, changes in technology, and things that were learned during the development process. A project manager will receive requests for changes from end users, stakeholders, or client managers.

The elaboration of these requirement changes involves several stakeholders that are directly or indirectly affected by the project. These stakeholders have different experiences and expectations with the project. Thus, the requirements analysis process must be performed completely, because these stakeholders may not be able to define exactly what they actually need.

Errors in the requirements are costly and can lead to loss of time, revenue and reputation of the responsible organization. Furthermore, when considering the correction of these requirements when they have already been implemented, the cost associated with correcting errors could generate even greater impacts under the project budget (Badariah *et al*, 2009).

Each scope change can affect work that has been already performed. This means reworks costs for work that has already started or worse, been completed.

For product quality each change needs to be analyzed thoroughly and incorporated in product design. When not analyzed thoroughly, scope changes lead to quick fixes that can affect product quality.

Each scope change can lead to diversion of, precious project resources to activities that were not identified in the original project scope, leading to pressure on the project schedule. The project manager must also consider impact on the project's critical path.

Creating project team accountability is the other problem in handling change, as the project team members can have a lot of interaction with the client, they are the ones who field scope-change requests the most often. Therefore, the entire project team must understand the importance of scope-change management.

The motivation for this study is that agile methodologies are flexible enough to be adopted by software developers but they need tailoring to provide a one size fit-all approach. The main pillar of agile methodologies involves the interaction between developers and customers. The greatest problem agile teams face is too little involvement from the customer.

Thus this study helps The team and the client to understand the purpose of scope-change management and processes should be followed in order to deliver a successful Product.

Furthermore, regarding the topic of the study, no research has been done which focused primarily on agile practice on the management of scope related to HCMIS. Thus, this research is undertaken to contribute, by knowing the current practice in the study ‘s knowledge area and to explain the current practice JSI is following to develop software’s.

1.4 Basic Research Questions

This study regarding the Scope management of software's developed by JSI, tries to answer the following questions:

Main Question:

- What is the effect of agile software development practice on scope management?

Sub Questions

- What Scope Management Processes are followed in managing projects at JIS?
- What actions are taken to identify and manage scope creep at JIS?
- What actions are performed to manage scope planning at JIS?
- What are the relation between agile project management and Scope?

1.5 Objective of the Study

This study describes scope management in agile system for the software projects developed by JSI software Development team.

General Objective

- ✓ To find out what the effect of agile software development practice has on scope management.

Specific Objective

- ✓ To find out the current scope management process's followed in project management at JSI.
- ✓ To find out what scope creep is and how it is managed at JIS.
- ✓ To find out the actions performed to manage scope planning at JIS.
- ✓ To describe the relationship of agile project management and scope.

1.6 Significance of the Study

This study will have practical significant for JSI and other software developing companies specially located in Ethiopia; to learn how scope is managed successfully and what agile scope management means specifically how it will impact the Project.

According to The Federal Democratic Republic of Ethiopia Ministry of Communication and Information Technology, Information and Communication Technology is one of the strategic priorities in the country. In lieu of the country's stride for substantial growth, the Government of Ethiopia highly encourages and promotes the use of information and communication technology in all development sectors (Ethiopia, Ministry of Communication and Information Technology 2009:1).

Successful software development can be one of the attributes of Ethiopia. There are Many Ethiopians that have the technical skill for developing software's, This study will strength the awareness for how to manage the scope of the project by elaborating the requirements.

The Scope Management Plan details how the project scope will be defined, developed, and verified. It clearly defines who is responsible for managing the projects' scope and acts as a guide for managing and controlling the scope.

It will also include the traditional way of scope management and the agile way how they are different and why one is better than the other.

1.7 Delimitation and Limitation of the study

The scope of the research is delimited to Software's developed by JSI only and from the nine Project management knowledge areas to one of the knowledge management area, Scope Management. The extent of the study is identifying, describing and interpreting the result as found.

The critical limitation of this study is the time given. In the short time the study is conducted it is a bit hard to go to the depth and detail it could possibly be done.

1.8 Organization of the Study

The study consists of five chapters. Chapter one is an introductory part containing discussions on background of the study, research problems, objective of the study, significance of the research, delimitation and limitation of the research and organization of the research report. The next chapter, chapter two briefly discusses literature relevant to the study which includes theory and empirical evidence related to the research topic. The third chapter discusses about the research design and methodology which was applied in the study. Results/findings of the study is presented and the findings are interpreted in chapter four. The last chapter, chapter five is the summary of the findings, conclusions and recommendations.

CHAPTER TWO

LITERATURE REVIEW

2.1 Project

Project is a temporary endeavor undertaken to create a unique product, service or result. A project is temporary in that it has a defined beginning and end in time, and therefore defined scope and resources.

2.1.1 Project Management

Project management is the elaboration of requirements involves several stakeholders that are directly or indirectly affected by the project. These stakeholders have different experiences and expectations with the project. Thus, the requirements analysis process must be performed completely, because these stakeholders may not be able to define exactly what they actually need.

Project Management Body of Knowledge (PMBOK) identifies nine knowledge areas of project management: they are the skills a project manager must practice and master to manage a project efficiently.

1. Project Integration Management
2. Project Scope Management
3. Project Time Management
4. Project Cost Management
5. Project Quality Management
6. Project Human Resources Management
7. Project Communications Management
8. Project Risk Management
9. Project Procurement Management
10. Project Stakeholder Management

2.2 Project Scope, Project Scope Management, Scope Creep

Project Scope refers to all the work involved in creating the deliverables of the project and the processes used to create them.

There are two types of scope that are Product Scope and Project Scope

Product scope: The features & functions that characterize the product, service, or result documented usually by the Business Analyst in consultation with the stakeholders.

Project Scope: The work that needs to be accomplished to deliver a product, service, or result with the specified features and functions.

Project scope management includes the processes required to ensure that the project includes all the work required, and only the work required, to complete the project successfully.

Scope creep (also called requirement creep, function creep, or kitchen sink syndrome) in project management refers to changes, continuous or uncontrolled growth in a project's scope, at any point after the project begins. This can occur when the scope of a project is not properly defined, documented, or controlled.

Scope planning, scope definition, scope verification, and scope control are all processes that are defined in the *PMBOK® Guide* to prevent scope creep, and these areas earn great attention from project managers.

Those who use agile methods believe these deserve great attention as well, but their philosophy on managing scope is completely different. Plan-driven approaches work hard to prevent changes in scope, whereas agile approaches expect and embrace scope change. The agile strategy is to fix resources and schedule, and then work to implement the highest value features as defined by the customer. Thus, the scope remains flexible.

Scope creep doesn't exist in agile projects, because scope is expected to change. Scope management in agile is primarily a function of rolling wave planning and the management of the product backlog.

Scope is defined and redefined using five different levels of planning that take the team from the broad vision down to what team members plan to complete today.

Scope is verified by the customer, who is responsible for accepting or rejecting the features completed each iteration.

Scope is controlled through the use of the backlog, rolling wave planning, and the protection of the iteration. This might lead to schedule expansion and even total project failure.

Client satisfaction has shown to be the most essential factor when it comes to the perception of project success (Dov and Lechler, 2003). Clients seem to be satisfied when the quality of a product is better than the standard of that product. Further, Dov and Lechler (2003) argue that advantages of achieving a high level of client satisfaction are the following: Improving communication between parties, Enabling mutual agreement, Evaluating progress towards the goal , Monitoring accomplished results and changes, etc."

2.3 Software, Agile Software Development

Software is a general term for the various kinds of programs used to operate computers and related devices. Software that is being produced today typically falls into one of three categories:

1. Software as a component, such as the fuel management system embedded in your car.

2. Software as a tool, such as an 'Integrated Development Environment' (IDE) in which other software is developed and maintained.

3. Software as a process, no longer simply deployed as a tool to help organizations do the things they do, but a result of rethinking what an organization does and how it does it.

Studies have shown that traditional plan-driven software development methodologies are not used in practice. It has been argued that the traditional methodologies are too mechanistic to be used in detail. As a result, industrial software developers have become skeptical about "new" solutions that are difficult to grasp and thus remain unused. Agile software development methods, "officially" started with the publication of the agile manifesto, make an attempt to bring about a paradigm shift in the field of software engineering. Agile methods claim to place more emphasis on people, interaction, working software, customer collaboration, and change, rather than on processes, tools, contracts and plans.

2.4 Traditional vs Agile Project Scope Management Processes

Traditional project methodologies arose in the middle of 19th century during the cold war. The reason for the occurrence of traditional methodologies was due to the constant striving of being ahead of enemies. To achieve completing tasks on as short lead-time as possible, several activities were performed in parallel with each other (Gustavsson, 2013). However, by the time, project participants started to make counter reactions to this kind of project management. In reality, it was not possible to perform activities in parallel and tasks were instead performed sequentially. (Gustavsson, 2013).

Agile methodologies on the other hand could be described as "the ability to rapidly respond to change" (Oxford University Journals, 2013). The agile methodologies developed as a response to dissatisfactions about traditional methodologies, which are keeping time, cost and scope fixed in a project. This means that the traditional methodologies were seen as inflexible and therefore a group of software developers came up with the "agile" methodologies which are more flexible (Oxford University Journals, 2013).

Several scientific researchers use different terminology when comparing both types of project management. Traditional approaches are often referred to as plan-driven, task-driven, document-driven or marked as large ''heavy-weighted'' methodologies (Henderson-Sellers & Serour, 2005). Heavy-weighted because of the focus on the process; this is in contrast to the light-weighted agile approaches such as Scrum and XP (Henderson-Sellers & Serour, 2005). A plan-driven or document-driven approach refers to the documented process plans (tasks and milestones) and product plans (designs, architectures and requirements) (Boehm, Port, & Brown, 2002). Since agile approaches also include planning activities (only in shorter iterations), the term ''traditional project management'' is used when comparing to agile project management.

Traditional and agile project management styles consist of a set of specific characteristics and underlying philosophy. Agile is based on the values of the Agile Manifesto (Beck et al., 2001) and this philosophy provides guidance on how to apply agile project management (Owen, R., Koskela, L.J., Henrich, G., & Codinhoto, R 2006). Traditional project management pursues an ultimate goal: optimization and efficiency in following initial detailed project plan, or, having said in usual way, to finalize project within planned time, budget, and scope (Spundak, 2014). The basis of a traditional approach is predictability, while agile is characterized by the adaptability to changes.

2.4.1 Scope Planning

The first process in this knowledge area is to Plan Scope Management which is coming under planning process group. This is the process of creating a scope management plan that documents how the project scope will be defined, validated, and controlled. It provides guidance and directions on scope will be managed.

Table 2.1 Scope Planning

Traditional	**Agile**
Prepare a project scope management plan Document	Commit to following the framework as outlined in the chosen agile process

Note. Michele Sliger and Stacia Broderick(2008) The Software Project Manager's Bridge to Agility:Scoope Management

2.4.2 Scope Definition

Scope Definition comes under planning process group**,** the process of developing a detailed description of the project and product. The project scope serves as a reference for all future project decisions.

Table 2.2 Scope Definition

Traditional	**Agile**
Prepare a Project Scope Statement document that includes items such as the following: Project boundaries and objectives, product scope description…	Conduct a vision meeting to share the product vision; confirm and clarify the boundaries, objectives, and product scope description using exercises such as the elevator statement and design the box.
And major milestones and project deliverables…	Conduct a planning meeting to prepare the product roadmap, as well as release or quarterly planning meetings that also include milestones and deliverables at an iteration level.
And product specifications and acceptance criteria…	Conduct an iteration planning meeting that results in the detail around each feature, and the tasks needed to complete the feature according to the team's definition of "done" and the acceptance criteria defined by the customer.
And assumptions and constraints.	All planning meetings identify and/or review assumptions and constraints.

Note. Michele Sliger and Stacia Broderick(2008) The Software Project Manager's Bridge to Agility:Scoope Management

WBS Creation

Creating Work Breakdown Structure comes under planning process group, in this process subdividing the project deliverables and project work into smaller and more manageable components. The work breakdown structure is a deliverable-oriented hierarchical decomposition of project work.

Table 2.3 WBS Creation

Traditional	Agile
Create a work breakdown structure diagram.	Conduct planning meetings and give the team the responsibility for breaking down the work into smaller work packages (features and tasks), displayed as the release plan at the high level, and the iteration plan at the more detailed level.

Note. Michele Sliger and Stacia Broderick(2008) The Software Project Manager's Bridge to Agility:Scoope Management

2.4.3 Scope Verification

Validate Scope comes in monitoring and control process group, means formalizing the acceptance of the completed project deliverables by the customer. This includes reviewing deliverables with the client and obtaining formal acceptance of deliverables. Scope validation is concerned with acceptance of deliverables by the external customer while quality control is concerned with checking the correctness of the deliverables internally and meeting quality requirements.

Table 2.4 Scope Verification

Traditional	Agile
Document those completed deliverables that have been accepted and those that have not been accepted, along with the reason.	Documentation of accepted features may be done informally (by moving the sticky notes to the "done" pile) or formally.
Document change requests.	Customer updates the backlog.

Note. Michele Sliger and Stacia Broderick(2008) The Software Project Manager's Bridge to Agility:Scoope Management

2.4.4 Scope Control

The last process in this knowledge area which comes under monitoring and control process group is to control scope means monitoring the status of the project & products scope and managing changes to scope baseline. Ensure all requested changes and recommended corrective or preventive actions are processed through the "Perform Integrated Change Control" process. Control Scope process will control the scope creep.

Table 2.5 Scope Control

Traditional	Agile
Use a change control system to manage change.	The customer manages the product backlog; once the team commits to the work to be done in an iteration, the scope is protected for that duration.
Update all documents as appropriate with the approved changes.	The team revisits release plans and product roadmaps regularly, making changes as needed to better reflect the team's progress and changes requested by the customer.

Note. Michele Sliger and Stacia Broderick(2008) The Software Project Manager's Bridge to Agility:Scoope Management

Table 2.6 Agile Project Manager's Change List for Scope Management

I used to do this	Now I do this
Prepare a formal Project Scope Management plan.	Make sure the team understands the framework and process structure of the chosen agile approach.
Prepare a formal Project Scope Statement document.	Facilitate planning meetings—vision, release, iteration, daily stand-up—and arrange for the informally documented plans to be highly visible to all stakeholders.
Create the WBS.	Facilitate the release planning meeting so that the team can create the plan showing the breakdown of work across several iterations.
Manage the change control system and try to prevent scope creep.	Step away from the backlog; it is owned by the customer. If needed, remind the customer that during the iteration, the team is protected from scope changes.
Manage the delivery of tasks to prevent or correct scope creep at the task level.	Allow team members to manage their daily tasks and facilitate conversations with the customer to avoid unnecessary work or "gold plating."

Note. Michele Sliger and Stacia Broderick(2008) The Software Project Manager's Bridge to Agility:Scoope Management

2.5 Empirical Literature Review

The traditional project management approach employs a linear execution, attempting to get all activities done following detailed planning at once upfront (Augustine et al. 2005; Weinstein, 2009). And this is perceived as the heart of heart of the difference between agile and traditional project management (Hass, 2007). Though there are slight arguments that traditional project management approach also encompasses iterations, they are only within each stage and not between different stages due to rigid planning and control (Cadle & Yeates, 2008; Collyer & Warren, 2009).

The need for organizations to develop higher quality software products with reduced cost and flexible scope is increasing. Among the principles advocated in the Agile Manifesto are communication, objectivity, a greater focus on development and customer interaction, conceptual simplicity, high quality, technical excellence, lower costs, dynamism regarding changes to project requirements, flexibility, autonomy, efficiency of development and quick delivery of functional software.

In order to understand the factors that determine agility dimensions in a project, Lee and Xia (2010) suggest a trade-off relationship between response extensiveness and response efficiency of the team.

According to Sletholt, M, Hannay, J, Pfahl, D, Langtangen, H (2005) Projects that use agile practices work better with activities related to test and requirement analysis. These activities, if implemented well, can yield good results, achieving affordable cost and flexible scope.

For the scope criterion, multifunctional teams led by a facilitator reinforce the presence of a leadership representative on teams, helping in issues as customer requirements understanding, assisting in the scope's project definition. The facilitator usually has more experience than team members and can have a better understanding of selected tasks, and can define what should be prioritized and implemented before the deadline, with a more realistic understanding of time and budget.

These findings should be interpreted as recommendations for practitioners, meaning that it is possible to get better and satisfactory results in cost and scope using these practices to improve team management abilities, to organize management of requirements, to improve code quality and to deliver software on budget and on-time.

A team with high performance should be functional, write good and maintainable code, be creative and innovative enough to better understand their customer's needs and prioritize tasks in the project scope that will add value. These abilities are essential for a team to work better in projects with a flexible scope.

From the literature reviews the importance of agile scope management in software development is essential. Therefore, this research tries to show the practice of the project scope management and to know if the common process, techniques and tools are applied to manage scope at JSI software development projects.

2.6 Conceptual Framework

The Manifesto of agile development principle was launched and it was a breakthrough that had implicit goals of producing working software that values meeting the requirements of the customer while having little documentation. Therefore, the Manifesto inspired developers to produce software that was responsive to customers' needs and employ a light-weight development methodology. There is a shift from producing working software to improving the experience of the customer with the software and instead of responding to change, agile developments have to predict the change.

Areas that agile development focuses on are the communication among all stakeholders, iterations with continuous integration, and feedback. One of the main reasons for using agile methodologies is to satisfy the needs of the users. The informal communication among stakeholders and developers sometimes raises problems such as inability to cope with system complexity and rapidly changing requirements.

Agile methods are labeled as agile because of their ability to handle changing requirements. It is also expected that the agile methods themselves are flexible and can be tailored to the needs of the developers and the needs of the software project. Agility is attained when software development methodologies attain to external factors and by being flexible internally in software development. Developers practices and eliciting the requirements play a major role in making the software development successful.

Therefore, this conceptual framework created two customized practices for making agile methodologies handle different situations when the original practices do not account for these situations. There is a shift from producing working software to improving the experience of the customer with the software and instead of responding to change, agile developments have to predict the change.

CHAPTER THREE

RESEARCH DESIGN AND METHODOLOGY

3.1 Introduction

This chapter presents the description of the research process. It provides information concerning the method that was used in undertaking this research as well as a justification for the use of this method. The chapter also describes the various stages of the research, which includes the selection of participants, the data collection process and the process of data analysis. The chapter also discusses the role of the researcher in qualitative research in relation to reflexivity. The chapter ends with a discussion of Ethical consideration in qualitative research.

3.2 Research Design

The term research design 'refers to and encompasses decisions about how the research itself is conceptualized, the subsequent conduct of a specific research project, and ultimately the type of contribution the research is intended to make to the development of knowledge in a particular area. Importantly, the process of developing a research design combines three broadly connected and interdependent components: the theoretical, methodological, and ethical considerations relevant to the specific project' (Cheek in Given, 2008: 761).

The two main theoretical approaches to research include the quantitative and qualitative methods. Each method is independent in terms of theory and practice. The research design for this research is a qualitative one. Qualitative research is a systematic, subjective approach used to describe work experiences and give them meaning. The purpose is to gain insight; explore the depth, richness, and complexity inherent in the phenomenon. It is done in natural settings, without any experimentation. Basic elements of analysis are words rather than numbers.

The goal is to develop rich understanding of a phenomenon. Requires on going analysis of the data to formulate subsequent strategies to determine when fieldwork is done. It includes inductive data analysis Basic of knowing:

- meaning and discovery Data are collected in the form of words, rather than numbers.

- Researcher's personal experiences and insights are an important part of the inquiry.
- Direct data collection is also a key characteristic of qualitative research studies;
- Narrative data is collected over long periods of time from observations and interviews

3.3 Sample and Sampling Techniques

Sampling is the process whereby some elements (individuals) in the population are selected for a research study. The population on the other hand consists of all individuals with a particular characteristic that is of interest to the researchers.

Census or a complete enumeration requires large amounts of resource, time and energy. Many researches are thus done by taking a sample to study a phenomenon and infer conclusion to the larger population (Babbie 2010; Kothari 2004; Tayie 2005). In this case the population is all the MIS team at JSI. Which is composed of 1 MIS deputy country director, 1 MIS manager (project manager) , 4 senior software developers, 19 software developers, 9 Analysts, 20 hub it support, 3 center IT support, 4 product manager and 4 driver/it technician which is total of 65 from the overall total of 174.

In this study, the researcher employed the purposive sampling method for the selection of the participants. Purposive sampling, which is categorized under non-probability sampling, refers to intentionally choosing sample according to the needs of the study. This means the researcher selects participants because they have indicated their willingness to participate in the study.

In addition to the time and cost advantage, this strategy also enables the researcher to collect relevant and useful information for answering the research question. According to (Saunders, Thornhill and Lewis,2009), this method is appropriate for small inquiries and researches by individuals. It is appropriate if the research is aimed at explaining a phenomenon rather than making a generalization (Best and Kahn, 2006).

Therefore, based on the above technique, The Project Manager, 3 of the Senior software developers, 10 out of the 19 software developers, 5 from the 9 analysts, 2 from the center IT support, and all the 4 product managers and 6 of the hub it supports where chosen to participate in this study.

The population was segmented into sub groups so it is classified as quota sampling – Segmented into mutually-exclusive sub-groups, then judgement is used to (non-randomly) select sample from each segment From the segments using convenience sampling which is defined as individuals believed to be representative of the population from which they are selected, but chosen because they are close at hand and easy to get access to them rather than being randomly selected.

3.4 Research Methods

Research methodology is basically a process of how a research is being conducted. It encompasses tools and techniques to conduct a particular research or finding. According to (Walliman, N. S. R. ,2011) Research method is a range of tools that are used for different types of enquiry. Therefore, it is important to select an accurate method that suits the research objective. In this section, the researcher will break the components into two: data collection method and data analysis method.

3.4.1 Data Collection

The researcher applied two methods of data collection techniques. This was done in order to collect adequate and relevant data to address the research objectives of this study. Nonetheless, the researcher used qualitative research method. These are The Library Research and Field Research

(G. Ramesh Babu, 2008) described library research as a process dealing with the analysis of evidences such as historical records and documents. Similarly, it means gathering data from library materials which includes textbooks, both published and unpublished academic documents such as journals, conference proceedings, dissertations and theses. Library research also includes information gathered from internet search.

Data gathered via library research is categorized as the secondary data. Secondary data means the data is readily available and is used by anyone besides researchers. This means that secondary data is not originally collected but rather obtained from published or unpublished sources.

Field research on the other hand composes a number of research methods to solve the existing research problems such as case study, interview and observation. For this study interview and direct observation is applied.

Interview method is one of the ways to obtain primary data. Primary data is understood as data which was collected during the conduction of a particular research. The data will be collected in the execution of an experiment. Therefore, in order to obtain primary information, face-to-face interviews were employed. The researcher, with prepared written questions to the interviewees, used the open-end interview. In this context, the interview protocols ask specific objective questions in an arranged order. This process encouraged the participants to respond to the questions as accurately as flexible as possible.

Burns (2006) argues that reflexivity allows the researcher to arrive at an in-depth understanding of the meaning of the phenomenon under investigation. This implies that the researcher is able to draw on his or her own experiences during the research process to enable him or her to understand and identify with what is being said. However, despite the use of the researcher's own experiences and viewpoints the focus of the investigation or research remains on understanding the phenomenon from the participants‟ perspective (Babbie & Mouton, 2001). The researcher therefore has to put aside her own understanding of the subject of investigation and open her mind to understand and listen to what is told to her by the participants.

3.4.2 Data Analysis

Data analysis is a very important segment in the research. In qualitative studies, data are usually recorded in the form of words; descriptions, opinions and feelings rather than numbers.

Data analysis is a process whereby researchers make search and arrange it in order to enhance their knowledge of the data and to present what they learned to others. Similarly, James (2004) highlighted that data analysis is to arrange data, separating it into effective units according to topics and themes.

The current research adopted the fundamental approaches in analyzing the output from the interviewees and Direct Observation. The analysis includes data reduction, transcription of interviews, data display and conclusion drawing. The approach involves arranging the data in order to create explanations and create new finding.

The researcher then analyzed and interpreted the information provided by the informants in accordance with the objectives of the study.

3.5 Ethical Consideration

Several ethical considerations were taken into account to ensure that the study was conducted in an appropriate manner. To comply with ethical considerations in conducting research all participants provided verbal consent to be interviewed and to participate in the research. The participants therefore willingly participated in the study after they were approached by the researcher and the research purpose and process were explained to them.

While it is common practice to request written consent, Silverman (2010) states that highly formalized ways of securing consent should be avoided in favor of fostering relationships in which ongoing ethical regard for participants is sustained. In this study verbal consent was deemed appropriate. In support of this form of consent Fritz (2008) has argued that the strength of qualitative research often lies in the informality of the communication as well as the interactive nature of the research process.

The response has been used only for the purpose of the study without making any adjustment. The responses were also kept confidential.

3.6 Validity and Reliability

Purposive sampling does not intend to generalize the drawn conclusion; rather it emphasizes on gathering and analyzing quality data from willing and capable informants. It provides reliable and robust data and contributes to internal validity but does not guarantee external validity (Tongco 2007).

Measures taken to improve internal validity in this study include due attention to careful selection of appropriate respondents in order to acquire the right information from the right person and ensuring anonymity to enable research subjects to provide honest responses (Oates 2006:288).

In developing the research instrument, due attention was given to avoiding leading questions. Questions were designed in such a way that they were understood by respondents in the same way in order to increase reliability of the study.

CHAPTER FOUR

DATA ANALYSIS AND PRESENTATION

4.1 Introduction

In this chapter the data gathered from the different source will be presented and analyzed. The first step in the data gathering stage is to find out what the governance model is like and what scope management process where followed. And also the overall scope management of the software's developed by JSI.

Data analysis was performed in two parts, using two different perspectives. The first part, preprocessing, used descriptive data analysis while the second part, exploratory analysis, used multivariate data analysis. It will be followed by the Interviews held with the IT analysist's, the product managers, the software developers and senior software developers involved in the project.

The goal was to analyze which agile practices are used, according to practitioners' perceptions, to improve the scope performance criteria in software projects and focuses answering the research questions raised in chapter one of this study.

4.2 Data Result

The participant of these research where all cooperative and available. They also participated on discussion of literature besides their experience on the selected case HCMIS.

the key points elaborated by each interviewee on the project management followed by the company is similar to the results found on the literature review. Due to the specific nature that IT industry requires high responsiveness and the fact that customer requirement are intrinsically ambiguous and uncertain, IT companies rarely follow traditional project management strictly as the way traditional project management has been applied to industries such as engineering and construction (Erickson, J., Lyytinen, K., & Siau, K. 2005).

The major characteristics of agile project management mentioned by all the interviewees are its iterative feature and importance of communication (Owen et al., 2006). Customer involvement is also perceived as a main feature and other mentioned include empowerment, encouraging change and making decision only when necessary.

4.2.1 Key Stakeholders

The data analysis process starts by identifying the Key stakeholders that have a major role in developing HCMIS include: Ministry of Health (MOH) ,The Pharmaceutical Fund and Supply Agency (PFSA), Project Sponsor, Regional Health Bureaus , Health Facilities Project Team

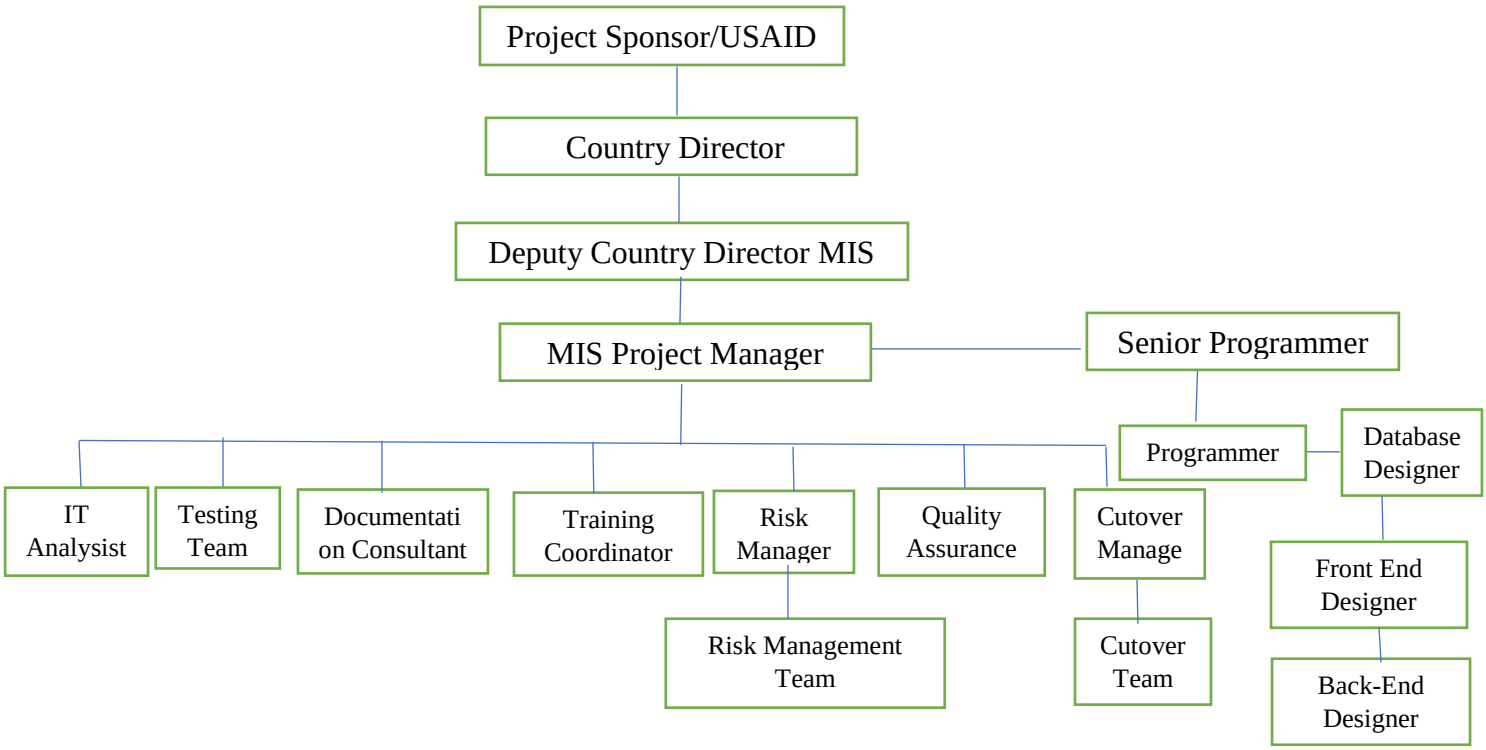
4.2.2 Governance Model

The Governance Model describes the governance rules which are used in the HCMIS and clarifies the Project Sponsor, Project Manager and Project Team relationship, and addresses the governance structure that will be put in place so as to make this project a success.

It, therefore, gives a paramount emphasis on the design, development, implementation, tests, and training regarding the HCMIS.

The clear ambition of the HCMIS governance model is to have a lightweight management structure that allows effective design and development of the system.

Fig 4.1: Project Organization of HCMIS Project



Note: *Strengthening High Impact Interventions for an AIDS-free Generation (AIDSFree) Project. 2017. AIDSFree Ethiopia Workplan FY 2017–2018. Arlington, VA: AIDSFree.*

4.2.3 Project Team and Responsibilities

Project Sponsor

- Supports the activities of the project team through the course of the project
- Ensures the achievement of the overall project objectives o Ensures that adequate resources are aligned to the project
- Approves changes to project scope

Project Team Leader

- Provides overall leadership and direction for all aspects of the project
- Defines/manages contractual obligations with software and other project related processes
- Manages project resources/teams and facilitates and maintains communication
- Defines and manages project scope
- Develops project work plan
- Conduct work product/deliverable reviews
- Defines and ensures project deliverables are met

Risk Manager and Risk management team

- Provide the overall project risk strategy and to coordinate the project
- Ensure the objective of risk management plan has met
- Reports different risk status reports to project
- To collect, capture and coordinate the necessary project risk information.
- Reports various risk status report to the project manager

Quality Assurance Manager

- Manages baseline audit processes
- Reports audit results to Project Management Team
- Ensures the integrity of the release baselines

Cutover Manager and Cutover Team

- Coordinate the activities to ensure successful completion of the cutover activities
- Reports to Project Management Team

- Collect Cutover process status and report to project manager o Involves in Cutover process activities
- Work with developers to ensure that Cutover process to be as expected
- Assists the end users

IT Analyst

- Coordinates and collects all the required processes to collect requirements
- Responsible for defining requirements
- Actively participates in the project team meetings to comprehend the system and facilitate the system development processes
- Closely works with the project team on test plan development and training related issues
- Actively participates in issue resolution, project plan development, scope development and management
- Works with the rest of the project team so as to ensure that project deliverables are met

System and Database Designers

- Responsible for organizing the collected requirements
- Responsible for analyzing and modeling the system requirements
- Responsible for designing and developing the database
- Responsible for designing and implementing the architecture of the system
- Ensures the system met the requirements
- Works with the rest of the project team to ensure the quality system and database design

Programmer

- Responsible and accountable for the quality and validity of the HCMIS
- Responsible for implementing the system and database design in to application module
- Responsible for unit and integral module testing
- Closely works with Test coordinator

Testing Team

- Responsible for developing a comprehensive testing plan
- Responsible for developing test criteria and plan with the development team

- Responsible in identifying any possible error and related issues
- Closely works with programmers

Documentation consultant

- Works on the system documentation o Responsible for producing the user manual guide

Training coordinator

- Responsible for developing training plan
- Coordinate and provide all the necessary training to End users
- Ensures availability/adequacy of training environment
- Responsible for the organization/delivery of user training
- Responsible for the development of training materials

End user

- Assists in the formulation of work flow and processes
- Serves as a project and system users advocate
- Actively participates in issues resolution, project plan development, scope development and management

4.2.4 Project Approach

This describe overall strategy for delivery of the HMICS project and will define the overall parameters of HCMIS project and establish the appropriate project management and quality environment required to complete HCMIS project.

4.2.4.1 Constraints and Assumptions

The HCMIS HE is a graphical application that requires users to have a computer with the right hardware and software components. Users are also required to have basic computer skills and familiarity with Graphical User Interfaces (GUI).

Besides, HCMIS HE demands Microsoft .NET Framework 4.0 and Microsoft SQL Server later than 2005 to be properly installed and configured. In addition to this, HCMIS HE will be designed so as to be user friendly, reliable, error-free and secure. Therefore, it can be used by users with various backgrounds ranging from limited computing experience to advanced users and experts.

Moreover, HCMIS' HE code base shall be comprehensively commented, conventions explained, and ambiguities noted to ease the task of maintenance for future developers.

4.2.4.2 Key Milestone and Deliverables

Table 4.1: Deliverables of HCMIS Project

Phase of Development	Deliverable/Milestone	Target Date
Planning	Project Charter	March 2007
	Quality Assurance plan	May 2007
	Configuration Management Plan and Procedures	April 2007
	Risk Management Plan	May 2007
	Cutover Plan and Schedule	May 2008
	System Test Plan Document	March 2008
	Acceptance Test Plan Document	May 2008
	Multi Unit Test Plan Document	February 2008
	Performance Test Plan Document	April 2008

Requirement definition	Draft Requirements Specification	June 2007
	Draft Design Specification	August 2007
	Final Requirement Specification	June 2007
	Final Design Specification	August 2007
Design	Draft System Design Specification	September 2007
	Draft Database Design Specification	November 2007
	Final System Design Specification	October 2007
	Final Database Design specification	December 2007
	System Design Document	January 2008
Implementation	HCMIS HE Application Module	January 2008
Installation	HCMIS System	May 2008
	Test Plan	March 2008
	System Test Plan	April 2008
	Performance Test Plan	May 2008
	User Acceptance plan	June 2008

Note.USAID | Deliver projet, (2010) http://documentation.hcmisonline.org

4.2.4.3 Design Principle and Ethiopian Context

The Health Commodities Management Information System, HCMIS, is designed with the aim to develop a system that was as 'simple as possible' application but yet meet the Ethiopian Government's requirement of proper logistics management information system. This vision required a system that can be implemented for the country that was best fit for the type of facilities, warehouses as well as the skill level of those who are expected to operate the system.

The entire work of HCMIS was developed in Ethiopia from inception, design, development, testing and deployment to ensure the project was as localized as possible.

4.2.4.4 Technology Selection

The technology used in HCMIS was picked after a careful deliberation of various factors which included:

1) Availability of skilled and qualified developers/programmers,
2) Adoption and acceptance of technology in other sectors and the local IT job market
3) Cost effectiveness of technology expenditure requirements

4.2.5 Project Progress

Due to its limitations and lack of prominent functionalities, the old health commodities management manual system fails short fulfilling the successful management of the warehouse and facilities. Hence, a new automated system (i.e. HCMIS) had to be put in place.

The functional as well as non-functional requirements of the NEW system are briefly described. Furthermore, the project team used system models such as use cases and class diagrams for illustration purposes.

4.2.5.1 Functional Requirements

The proposed system should support the effective management of both the warehouse (hub) and facilities. The system to be implemented at the hub level should carry on the warehouse management as well as inventory control. However, at the facility level, only inventory control will be put in place.

Taking this fact into account, the basic functionalities listed out in the next section are classified as those that apply to the warehouse (hub) and/or facilities.

Table 4.2: Functional requirements of HCMIS Project

No	Functional Requirements	Applies to	
		Warehouse/Hub	Facilities
1.	Managing user accounts	☒	☒
2.	Managing system settings	☒	☒
3.	Editing pipeline information	☒	☒
4.	Adding/editing/deleting supplies list	☒	☒
5.	Adding/editing/deleting drug's information	☒	☒
6.	Managing hub information	☒	☒
7.	Customizing drug list	☒	☒
8.	Setting up warehouse profile including number of racks, pallet locations dimensions etc.	☒	☐
9.	Maintaining manufacturer	☒	☐
10.	Maintaining dimensions of packages for items per manufacturer	☒	☐
11.	Maintaining different Storage types	☒	☐
12.	Maintain preferred storage type and storage location for items in the warehouse	☒	☐
13.	Maintain separate logical stores for different programs in the same warehouse	☒	☒
14.	Setting/selecting storage location for pallets and non palletized items	☒	☐
15.	Confirming if items are actually put on their planned locations	☒	☐
16.	Recording item requests from facilities	☒	☐
17.	Approving amount to be issued for facilities	☒	☐
18.	Generating pick list including the location, item name, batch number and expiry date of items to be issued	☒	☒
19.	Confirming if the issue items are actually picked up and issued, generating Stock Transfer Voucher	☒	☐
20.	Consolidate non full pallets to save space	☒	☐
21.	Palletizing items on receipt	☒	☐
22.	Managing internal movements of items	☒	☐
23.	Handling loss/adjustment	☒	☒
24.	Managing Pick Face replenishment	☒	☒
25.	Handling inventory control	☒	☒
26.	Controlling receive transaction activity log	☒	☒
27.	Controlling issue transaction activity log	☒	☒
28.	Controlling loss/adjustment log	☒	☒
29.	Controlling inventory log information	☒	☒
30.	Handling database backup and restore	☒	☒

Note.USAID | Deliver projet, (2010) http://documentation.hcmisonline.org

4.2.5.2 Non-Functional Requirements

User Interface (UI): HCMIS shall be designed flexible and user friendly. Besides, it shall be designed as a graphical user interface that is easy to understand and use. The system shall also enable users to perform their business in effective and efficient way.

Hardware Considerations: As far as the existing hardware infrastructure is concerned, [for the HCMIS] there is no need of acquiring a special purpose hardware device. This makes the HCMIS affordable and less costly. Hence, the new system shall be designed to support single machine installation at the facilities.

Backup and Recovery: HCMIS shall support data backup and recovery. Moreover, the backup shall be taken from the warehouse and facilities on regular basis. Taking backups on regular basis will help restore/recover the HCMIS at times of system failure.

Extensibility: The new system shall be designed in such a way that it supports the addition of new features and customizations at next major version upgrade.

Error Handling and Extreme Conditions: The HCMIS shall have an automatic error handling mechanism which shall leave users informed about the errors that they have committed and so that they can rectify the problems.

Platform Compatibility: The system shall be designed to run on Windows XP and/or later operating systems that integrate the .Net 3.4 runtime framework.

Reliability: One of the pertinent non-functional requirements of the system is reliability. Hence, the HCMIS shall be designed and developed in such a way that it produces accurate and timely outputs (such as reports).

Security Issues: HCMIS system shall require user name and password of users to grant them access to the system in general and to create, modify and delete information in particular. Moreover, the HCMIS shall require administrator ID and password from system administrators in order for them to undertake necessary system administration tasks. The HCMIS shall also be developed and maintained in compliance with internationally established guidelines and standards for protecting computer systems, networks and information.

Usability by Target User Community: User's acceptance of the system is one of the many significant things that the new system shall fulfill. Hence, the HCMIS shall be acceptable by the target user community. To this end, the system shall be designed and developed in such a way that it envisions all the requirements of users.

4.2.6 Cutover Plan and Schedule

The Cutover plan provides the sequence of events a hub/warehouse will take to handle the transition from its current manual operations to compliance with the HCMIS HE.

It is done after the requirements are met and tested to ensure that the new system is performing as specified based on the deliverables identified.

The approach used in determining the cutover strategy and plan was to discuss the conversion strategies with senior technical developers in the project, who are responsible for executing the system changes.

Numerous meetings occurred until an agreement was reached that the plan mitigated all the risks that could be foreseen.

The Prospective audience for the cutover plan includes all constituents (internal – Development Team; external – service providers/vendor, employee at hub/warehouse or end user).

It is the responsibility of the project management to inform the hub/warehouse at each site and associated projects of required downtime 30 days for the cutover to take place.

Once the data to be converted and the conversion method are identified from the Conversion Plan, the next step is to determine and document the timing and sequence of the conversion program.

Obviously this will bring different questions. Among the questions to be considered include:

- When can all existing system operations be completed?
- When can the legacy system be shut down to begin the data conversion effort, and when must the new system be operational?
- How long will the data conversion processes, data verification, and reconciliation processes require?
- When should data backups be performed, and how much time is required for database recovery?

- How much data will be converted? As the volume of data is critical in determining the processing
- Time of the conversion program or resources required for manual data entry
- What type of hardware is being used? Determine how the hardware will affect the speed of automated conversion processes and transaction processing for manual efforts
- At what time must the new system be operational?

From a preliminary survey, users in different sites at hubs/warehouses are ambitious to use the new HCMIS system in order to avoid routine paper based activity and enhance their day to day activities.

Even if, most employees at the health facilities have a working knowledge on computers, conducting HCMIS training is inevitable.

Once the cutover plan has been completed, the major activities can be transferred to the cutover team who will provide the necessary activities.

Trainings will be required before the cutover activity is completed and the hubs begin work with the new system. One week to learn about HCMIS including its functionalities should be sufficient.

4.2.7 Contingency Plans

- The purpose of this contingency plan is to address project implementation risks that may result in one of the following scenarios:
    - Decision to delay the system go-live
    - Decision to proceed with limited functionality or restricted scope
    - Decision to back out the system after, or in the process of going live, based on a catastrophic event

4.2.8 Installation

The step by step installations are provided considering Windows XP as the operating system. The installation steps are:

- Insert either the CD-ROM or the removable media (such as Flash Disk) containing the HCMIS HE setup

- From the root folder double click the setup.exe
- Next, the setup will check for the installation prerequisites. If .NET framework 4.0 is not installed, it will automatically start to install the .NET Framework 4.0 setup
- Just follow the simple installation wizard until the installation is completed
- Next it will detect the presence of the SQL Server Express edition installation. If the installation is not present, it will prompt the user to install it. Follow the instruction wizard until all dependencies are properly installed. And, finally, it will install the software
- Once the installation is successfully completed, double click the HCMIS Icon from the desktop or from the program's menu list.
- Then, enter the username and password on the login window and start using the system

To put the HCMIS HE work, at minimum, it requires a standalone desktop computer with 2.4 GHz Pentium IV Processor, 512 MB RAM, and 20GB Hard Disk.

In addition to the hardware requirements, the HCMIS HE demands the following software.

- **Operating System**: Windows XP Service Pack 3
- **Database**: SQL Server 2005 Express Edition
- **Framework**: .Net Framework 4.0

4.2.9 Output Devices

- For various reports including Stock Status, Expired Product, Near Expiry Product, Balance, RRF, Summary, Stock Expiry Status and Cost Summary, the hubs are required to setup printers.
- High quality black and white as well as color printers are required to produce hard copies of the aforementioned reports that will be presented to decision makers and other concerned bodies.

4.2.10 Communications and Changes Management Plan

The Project Manager will take a proactive role in ensuring effective communications on the project.

Different media forms will be used on the HCMIS HE for conveying information, and for maintaining communication between project team members and project stakeholders. Face to face communication is the most effective, but is not always feasible, due to stakeholders being separated geographically.

The various media to be employed by the HCMIS HE project, includes:

E-mail: - E-mail will be used for normal day-to-day communication and dissemination of information. E-mail will be the media of choice for distributing document among team members as well as stakeholders. It will also be used to schedule meeting and for maintaining project calendars.

Telephone: - Conference calls will be utilized to conduct meetings as needed

Hardcopy: - Project documents will be prepared using the Microsoft Office suits. This includes status reports, training manuals, presentations, project plans, and project documents as may be required

Meetings: - With the exception of informal meetings, agendas should be prepared and followed for all status, board and program review meetings.

The goals of the communications strategy are

- To create a better understanding among all team members as well as stakeholders about the role of the HCMIS HE
- To maintain productivity and smooth working environment
- To build confidence among all audiences about the new HCMIS HE
- To communicate an underlying sense of dignity and respect to all team members

The key strategies are

- □ All the listed media will be utilized on regular basis
- □ Communications will be properly managed as an essential business process throughout the HCMIS HE life cycle
- □ Top management of HCMIS HE will serve as the communications leader
- □ Communication with all team members will create an opportunities for feedback
- □ Information will be communicated as soon as it is available to all project team members

4.3. Data Analysis

From the results of the analysis, the factors presented below were identified, where certain practices are used based on the perceptions and experiences of research participants. Thus, factors related to scope in software project management based on software agile practices are:.

Development guide by test: this first factor had high ratings in the following practices: developing test cases, unit tests and refactoring. In this factor, the developers create tests for features that isn't complete yet, writes code and testing this new features until it becomes errorless and functional, improving the code quality through the refactoring practice. This perception denotes a high value to the project insofar as it avoids delays in the planned scope originating from poor code maintenance, which in turn helps to improve code quality and to get maturity on tests.

Objective documentation: the second factor represents the use of the following practices: vision document, Unified Modeling Language (UML) diagrams, screening errors and use cases. The context described in this factor includes the use of UML diagrams to represent the classes and use cases of the system, and modifying them frequently according to the feedback of acceptance tests made by the customer and registered in the vision document. The vision document is streamlined and simplified documentation written in common language according to the customer's needs and what the software functionalities should be. This factor contributes positively to the scope criterion, and provides consistency and flexibility according to practitioners' perceptions.

Small teams led by the facilitator: this third factor had high factorial weights in practices related to the formation of small teams led by the team's facilitator. The facilitator works to keep the team focused on the Sprint scope, helping to solve problems on the tasks and giving recommendations on best practices in tasks development. In case tasks had problems in Sprint, the facilitator allowed postponing it in Sprint backlog. This role functions as a mediator of the team's problems.

Features defined by the customer: this fourth factor represents the practitioner's appreciation for the customer by joining the team for the backlog development activities, activities such as choosing and prioritizing tasks which will add greater value to the product. The customer's on-site behavior contributes, as with the third factor mentioned above, to the team members remaining focused on the scope, responding better and negotiating deadlines in case of changing requirements.

Stand-up meetings: the fifth factor shows positive practitioners' perceptions of the stand-up meetings practice in an attempt to improve the flow of communication among team members and to update the team's knowledge of the Sprint's progress. For the scope criterion, this factor contributes to a more informal and frequent status presentation of the project. It allows members to understand others' responsibilities and what tasks are done or not, without consulting the project schedule document with unnecessary frequency.

Frequent releases: the sixth factor shows the practitioners' perceptions in the practice product potentially shippable, known as a functional release. The practitioners' believe that developing a functional release at the end of each Sprint and presenting it to the customer can contribute to a planned scope because the software must be delivered in functional parts. This factor also decreases the risk of errors and of having features poorly implemented throughout the entire project.

Lead-programmer: the seventh factor had the highest factorial weights in the practice: lead-programmer role on team. This factor explains practitioners' positive perceptions of the value of this role, which involves working as a developer but also being responsible for helping the project manager develop consistent and clear software architecture and solve problems. In the scope criterion, the role of the lead-programmer presents a solution to improve features development and helps to define it better. In addition, this role also involves being a leader of the development team and providing technical support to team members when necessary.

CHAPTER FIVE
SUMMARY CONCLUSION AND RECOMMENDATION

5.1 Summary

Health Commodity Management Information System development project is an IT system development project. The goal of the project is clear; to develop an organized record-keeping system that can help to efficiently manage daily transactions of health commodities at warehouses and health facilities. However, the solution is not clearly known at the beginning of the project. Therefore, HCMIS Project falls under quadrant II of Robert K. Wysocki's(2013) project landscape diagram.

Table 5.1: Project goal-solution landscape

SOLUTION			
GOAL		**Clear**	**Not Clear**
	Not Clear	Quadrant 4 **Emetrtxe projects**	Quadrant 3 **Extreme Projects**
	Clear	Quadrant 1 **Traditional Projects**	Quadrant 2 **Agile Projects** **E.g.- HCMIS Project**

Note. Robert K. Wysocki,(2013) Effective Project Management: Traditional, Agile, Extreme

The scope of the HCMIS project is determined by both the project team and stakeholders. The scope includes the core or high-priority components that the Project Team has determined to be part of the project. Then the project scope statement forms the basis for an agreement among project stakeholders by clarifying both the project objectives and the project deliverables.

The HCMIS system is designed and developed in order to alleviate problems that are experienced in the hub as well as facility levels. The approaches that are put in place to efficiently tackle the limitations of the current system are outlined.

The approaches include:

- Requirement analysis,
- Design,
- Implementation and all other related phases that are vital for the successful implementation of the HCMIS system.
- Issues like increasing inventory accuracy, batch processing, expiry date tracking (i.e. using FEFO method), keeping inventory from being out of balance, keeping track of important transactions, and fast and flexible item search can be effectively implemented.

The Health Commodities Management Information System, HCMIS, is an inventory management designed to operate both at the Facility Level (branded as 'Facility Edition') such as Hospitals and large Health Centers as well as the Hub Warehouses (branded as 'Hub Edition').

While HCMIS is presented to the end user either as a Facility or Hub Edition and each have their own separate User Manuals, the System Documentation is common. Architecturally, both Editions come from a single source package; share a single database, design elements as reflected in the common features and functionality.

Generally, the HCMIS covers the following general aspects:

- Properly locating items in the warehouse or facilities
- Handling the transaction of receiving and/or issuing pharmaceuticals
- Managing the movement of items with in the warehouse or facilities
- Efficiently managing expiry dates using the FEFO system
- Applying the batch and recall methods
- Efficient data management and backup
- Security of the system

The HCMIS project covers the following specific aspects:

- The HCMIS software will be designed for standalone personal computers and this software will be implemented on Enterprise Application software that would be running on Windows operating system.
- The HCMIS will be designed to be user friendly and easy to use.
- The HCMIS will provide a facility administration option that allows users to manage facility related settings. These settings will include facility general information, facility settings and customizing drug lists.
- The HCMIS will be designed to support item Receive, Issue, Loss/Adjustment, and Inventory transactions.
- The HCMIS will be designed to record all transactions that are made during item Receive, Issue, Loss/Adjustment, and Inventory transactions.
- The HCMIS will provide detail log information for each and every transaction.
- The HCMIS will provide timely reports for item Receive, Issue, Loss/Adjustment and Inventory transactions which are grouped based on their functionality. Besides, the system will be designed to provide graphical report summary for In Stock, Out of Stock, Over Stock, Near EOP, and Below EOP items.
- The HCMIS will handle database functionalities such as taking a backup, exporting, restoring, and importing the database for the purpose of recovering the system at the time of system failure.
- The HCMIS system will have integrated security mechanism that will allow authorized system access.
- A comprehensive project plan will be developed to document all the necessary resources and implementation requirements to successfully manage and implement the project.
- The project team structure will be developed to address related requirements for the project.
- Detailed work flow diagrams will be developed to align work flow processes with the system.

- A comprehensive testing plans and test scripts will be developed for testing all requirements of the system.
- Multi-Unit, system, acceptance, and performance testing will be conducted to ensure that all system requirements are fulfilled.

Finally, the HCMIS development team will have the role of the software provider, documentation, support, installation, and prepare a comprehensive training plan and training materials.

The development team is informally classified to web developers mainly focused on developing web applications, back end developers focus on developing API's and accessing database and, mobile developers to develop mobile applications, Database Designers to design the database, create jobs, and sync packages.

After HCMIS was developed and deployed then came additional requirement Procurement Management. As a result, Procurement was handled as a different independent Module with high cohesion with in itself but also related with the existing Project HCMIS.

Procurement Management handles all the steps from request for procurement from forecasting or Ministry of Health goes through the process of approving the request, tender process (Procurement Plan and Performance, Budget Evaluation and Analysis, Tender Status) , Award (partial winner list, official winner list and unsuccessful bidders) is all handled by this Module.

While this by itself is enough for the procurement process the scope is a further widened when the project CMS (contract management process) a web application is developed. This project handles all the status and letter generations PFSA used to do manually to communicate with FMHACA, banks, suppliers and all the others involved in procurement Process.

The mobile application developers are busy developing mBrana, A Vaccine Supply Chain strengthening on woreda level. To further rich woreda's and also HCMIS transit management system to handle the process from the airport up to the hub. Where the transistor accepts and gives PFSA the items.

iImport is another web application developed by the team that directly syncs the purchase order that was created at PFSA to be accessible for FMHACA for approval and also sync back with success or failure from FMHACA back to PFSA

For all these systems there is one central database pds.hcmisonline.org a web application that is holding all the lookup information shared by all such as Item List, Suppliers, Manufacturers' administrative units (woreda, zone , region)…., . It has a user interface for entering new data's and a common database from where all these system retract there desired data's.

Along with these projects the web application developers developed the Dashboard where all the information from the 18 hubs is synced to a central database and different reports are generated and live status of each hub is made available for the sponsors.

In agile project scope creep is expected. JSI MIS team handles this by widening the scope with in the project as possible but if not possible break it into another module or project and continue with another team developing that requirement.

5.2 Conclusion

With the factor analysis study, the study was able to identify small groups based on practitioners' perceptions and experiences, showing how JSI is applying agile practices in its environment. Thus, represents how these practices can be grouped into structures named factors making possible to organizations' improving teams' abilities to plan scopes successfully and optimize defined and managed processes.

This study's goal was to analyze the agile practices used to improve scope management in software projects, seeking to answer the following research question: which of the main agile practices can contribute to better efficiency scope in software development?

Agile is an important tool for software development because it provides many benefits to the teams and project owners. The management style employed in the development process motivates the team members, which enable them to work effectively and accomplish the project tasks. The motivation increases their creativity and innovativeness, which contribute new ways of addressing the project problems effectively in order to deliver high quality software. The method provides different iterations, during which the developers can make appropriate changes based on the customer needs. The teams collaborate with the stakeholders during the process, thus enabling them to determine the successes and failures of the project in order to respond to them in good time. The control modes used by the leaders during the process enable the developers to cope with

the changing needs of customers, thus leading to project success. Agile methodology enables the developers to detect and respond to the defects early enough.

For the scope criterion, multifunctional teams led by a facilitator reinforce the presence of a leadership representative on teams, helping in issues as customer requirements understanding, assisting in the scope's project definition. The facilitator usually has more experience than team members and can have a better understanding of selected tasks, and can define what should be prioritized and implemented before the deadline, with a more realistic understanding of time and budget.

These aspects summarize practitioners' perceptions about using agile practices in software projects to handle flexible scope. Another important conclusion was that results can be more widespread by analyzing a set of agile practices rather than a specific methodology. This measure could represent a better path to achieving mature results on projects using the agile development approach.

In summary, I expect that the study results encourage further quantitative research in software engineering field, so that organizations and the academic community can improve their investments, resources and efforts in software development using the agile approach to implement creative solutions to get affordable cost and flexible scope.

5.3 Recommendation

From the observation and interview and also the literature studied the following points are the recommendation of this research.

Individual tasks should be small, and in line with agile methodology. Small tasks mean delivering small project increments in a short period of time.

Regular communication is essential to keep everyone involved and the project moving forward. Clear incremental objectives, agreed-upon deadlines for these small steps (sprints), and progress reports maintain the productive engagement of all team members. It also helps the team member with the next task on a project to schedule the time to work on it. This in turn supports progress even in the face of competing time pressures.

Silence from a team member doesn't mean that person is necessarily deeply engaged in their portion and deserves follow-up communication.

Agile methods are not for every kind of project. For projects where scope will not change, more traditional methods are perhaps a better choice. However, where there are projects with high levels of change, Agile methods tend to be a better choice.

This research has limitations concerning the interpretations of results. First, the sample collected is limited, implying limited inferences, and so the results should be interpreted as initial recommendations regarding agile software development. Future work is suggested, work involving more in-depth statistical research and analysis. In addition, conducting qualitative and quantitative studies to further investigate factors affecting scope and also the pros and cons of using agile software development practice in Ethiopia is highly recommended.

JSI is one of a successful software developing company currently delivering usable software's in Ethiopia as a reason a valuable lesson can be learned from this organization by doing a wider study of this companies work experience.

References

Babbie, E. 2010. *The Practice of Social Research.* 12th edition. Wadsworth: Cengage Learning.

Babbie, E. & Mouton, J. (2001) The Practice of Social Research. South Africa Oxford University Press, Cape Town.

Best, J.W. & Kahn, J. (2006) Research in Education, New Delhi: Prentice Hall of. India Pvt. Ltd.

David, Silverman (2010) Doing Qualitative Research (3rd ed.) London: Sage

Erickson, J., Lyytinen, K., & Siau, K. (2005). Agile Modeling, Agile software development, and extreme programming: the state of research. *Journal of Database Management,* 16 (4), 88-100.

Ramesh Babu G. (2008) Research Methodology in Social Sciences, Concept Publishing Co.

Ian Sommerville, (2011) Software Engineering, International Computer Science Series, Addison Wesley

Jainendrakumar, T. D. (2015) Project Scope Management in PMBOK made easy,PM World Journal vol. IV, issue IV – April 2015

Jo Tacchi, Joann Fildes, Kirsty Martin, Kiran Mulenahalli, Emma Baulch and Andrew Skuse. 2007 http://ear.findingavoice.org/dealing/index.html

Katherine Fritz (2008) Ethical Issues in Qualitative Research, Johns Hopkins University

Kothari, CR, (2004) Research Methodology: Methods and Techniques (2nd ed.) New Delhi: New Age International (P) Ltd.

Lee, G, Xia, W (2010) Toward Agile: An Integrated Analysis of Quantitative and Qualitative Field Data On Software Development Agility, Journal of MIS Quarterly, Volume 34 Issue 1, Pages 87-114

Mariana de Azevedo Santos, Paulo Henrique de Souza Bermejo, Marcelo Silva de Oliveira, Adriano Olimpio Tonelli, Enio Junior Seidel, (2013) Improving the Management of Cost and

Scope in Software Projects using Agile Practices, International Journal of Computer Science & Information Technology (IJCSIT) Vol 5, No 1

Michele Sliger and Stacia Broderick (2008)*:* The Software Project Manager's Bridge to Agility: Scope Management, Addison-Wesley Professional

Owen, R., Koskela, L.J., Henrich, G., & Codinhoto, R. (2006). Is Agile Project Management Applicable To Construction?. In: Proceedings 14th Annual Conference of the International Group for Lean. Santiago, Chile,, 51-66.

Project Management Institute. 2013. A Guide to the Project Management Body of Knowledge (PMBOK Guide), 5th edition. Pennsylvania: Project Management Institute.

Robert K. Wysocki, (2014), Effective Project Management: Traditional, Agile, Extreme (7th ed.), John Wiley & Sons, Inc.

Santos, MA, Bermejo, PHS, Oliveira, MS, Tonelli, AO (2011) "Agile Practices: An Assessment of Perception of Value of Professionals on the Quality Criteria in Performance of Projects". Journal of Software Engineering and Applications, v. 04, p. 700-709.

Saunders, Thornhill and Lewis (2009) Research Methods for Business Students, 5th Edition, Prentice

Silverman, D. (2010). Interpreting qualitative data (3rd ed.), Sage

Sletholt, M, Hannay, J, Pfahl, D, Langtangen, H (2005) "What do we know about agile practices in scientific software development"? International Journal of Computer Science & Information Technology (IJCSIT) Vol 5

Smith A. D. (2002), The Problem of Perception (1st edition), Harvard University Press

Solemon, B., Sahibuddin, S., & Ghani, A. A. A. (2009). Re-defining the requirements engineering process improvement model.

Spundak, M. (2014). Mixed agile/traditional project management methodology–reality or illusion? *Procedia-Social and Behavioral Sciences, Volume: 119*, Page 939-948.

Tayie, S. 2005. Research Methods and Writing Research Proposals. Cairo: Center for Advancement of Postgraduate Studies and Research in Engineering Sciences, Faculty of Engineering - Cairo University (CAPSCU).

Tom Mochal, (2004) Poor scope-management practices could precipitate project failure

Tongco, MD. 2007. Purposive Sampling as a Tool for Informant Selection. *Ethnobotany Research & Applications* 5: 147-158.

USAID|DELIVER PROJECT, 2010.Health Commodity Management Information System(HCMIS) Documentation.

Walliman, N. S. R. (2011) Research methods, London: Sage