

**ADDIS ABABA UNIVERSITY
COLLEGE OF BUSINESS AND ECONOMICS
SCHOOL OF COMMERCE**



**Agile project management practice and
challenges in the case of CoreIT project at
Starcom Network Solutions PLC**

By: Wondwossen Assefa

A Project work submitted to the Addis Ababa University College of Business and Economics School of Commerce in partial fulfillment of the Requirements for award of Master of Arts Degree in Project Management

Advisor: Solomon Markos (PhD)

July 2022
Addis Ababa, Ethiopia

Agile project management practice and challenges in the case of CoreIT project at Starcom Network Solutions PLC

By: Wondwossen Assefa

A Project work submitted to the Addis Ababa University College of Business and Economics School of Commerce in partial fulfillment of the Requirements for award of Master of Arts Degree in Project Management

Advisor: Solomon Markos (PhD)

July 2022
Addis Ababa, Ethiopia

Declaration

I, the undersigned, declare that this project paper is my original work, prepared under the guidance of my advisor, Dr Solomon Markos. All sources of material used while working on this research have been duly acknowledged. I further confirm that the research paper has not been submitted either in part or in full to any other higher learning institution for the purpose of earning any type of degree.

Name

Signature

Certification

This is to certify that Wondwossen Assefa Kassaye has worked on the title “Agile project management practices and challenges in the case of CoreIT project at Starcom Network Solutions PLC” under my supervision. This work is original in nature and is suitable for submission in the partial fulfillment of the requirement for the Degree of Master of Arts in project management.

Research Advisor: Solomon Markos (PhD)

Signature: _____

Approval

**Addis Ababa University
School of Graduate Studies**

This is to Certify that this research paper prepared by *Wondwossen Assefa*, entitled: *Agile project management practices and challenges in the case of CoreIT project at Starcom Network Solutions PLC* submitted in partial fulfillment of the requirements for the degree of Master of Arts in Project Management complies with the regulations of the University and meets the accepted standards with respect to originality and quality.

Signed by the Examining Committee:

_____	_____	_____
External examiner	Signature	Date
_____	_____	_____
Internal examiner	Signature	Date
_____	_____	_____
Advisor	Signature	Date

Chair of Department or Graduate Program Coordinator

Acknowledgement

I would like to thank first and foremost God for his wonderful gift of life.

I would like to thank my advisor, Dr Solomon Markos, for his guidelines, his comments and most of all, his patience.

This research paper is dedicated to my wife Yabi, who was there from the very beginning till the end, with support and encouragement. Yabi, I wouldn't have done it without you.

Finally, my thanks to my entire supporting family, the project team who volunteered to participate in the data gathering process, Starcom Network Solutions, and everybody else without whom this research wouldn't have been possible.

Abstract

This research is a study based on CoreIT project. Its main focus of study is the identification of challenges that arise in agile project management approach. In parallel, the study assesses the agile practice at Starcom. A brief literature review is presented to define the context of agile. It contains agile definition, its concepts, values and known problems. On top of that, an empirical review was conducted to review what research has been done on the subject. Since the study is a descriptive research, it follows a descriptive approach in its questionnaire design and data analysis tools. Data gathering methodology was a classical one: primary data from the project team and secondary data from the developed software and the product log were gathered. A questionnaire consisting of Likert scale was distributed to the project team members (48 out of 53 responded). 34 questions divided in three categories were presented in a 7 option Likert scale. Gathered data was later analyzed using descriptive statistics measurements like data frequency, central tendency, and dispersion. Data analysis suggests there is good agile practice at Starcom, but challenges lie in CoreIT software development, and they have affected the outcome of the project, time wise, cost wise, and scope wise. The recommendations are directed toward a hybrid system of waterfall and agile methodology, as it brings together the strength of each methodology: control and flexibility.

Keywords: agile practice, challenges, hybrid, project management

Table of Contents

- Title..... i
- Declaration.....
- Certification.....
- Approval.....
- Acknowledgement.....
- Abstract..... ii
- Table of Contents..... ii
- List of Acronyms..... v
- List of Figures..... vi
- List of Tables..... vii

- Chapter I - Introduction..... 1**
 - 1.1 Background of the study..... 1
 - 1.2 Background of the organization..... 4
 - 1.3 Statement of the problem..... 5
 - 1.4 Research Objectives..... 6
 - 1.4.1 General Objectives..... 6
 - 1.4.2 Specific Objectives..... 7
 - 1.5 Significance of the study..... 7
 - 1.6 Delimitation / Scope of the study..... 8
 - 1.7 Limitation of the study..... 8
 - 1.8 Definitions of Terms..... 8
 - 1.9 Organization of the research report..... 10

- Chapter II - Literature Review..... 11**
 - 2.1 Introduction..... 11
 - 2.2 Theoretical review..... 12
 - 2.2.1 What Exactly Is Agile?..... 12
 - 2.2.2 Agile Concepts..... 14
 - 2.2.3 Agile Values..... 16

2.2.4	<i>Declaration of Interdependence</i>	18
2.2.5	<i>Why use agile project management?</i>	19
2.2.6	<i>How agile methods work</i>	21
2.2.7	<i>Agile Project Management Process</i>	26
2.2.8	<i>Agile problems</i>	27
2.3	Empirical review.....	30
2.4	Conceptual Framework	37

Chapter III - Research Methodology..... 39

3.1	Introduction	39
3.2	Research approach and design	39
3.2.1	<i>Research approach</i>	39
3.2.2	<i>Research design</i>	39
3.3	Data types, sources and method of data collection	40
3.3.1	<i>Data types</i>	40
3.3.2	<i>Data sources</i>	40
3.3.3	<i>Data collection</i>	41
3.4	Target Population.....	41
3.5	Data analysis	41
3.6	Data validity and reliability	42
3.7	Ethical consideration.....	42

Chapter IV - Data presentation, analysis and interpretation 43

4.1	Introduction	43
4.2	Response rate	43
4.3	Agile project management practices	43
4.3.1	<i>Relevant single variables</i>	44
4.3.2	<i>Aggregate agile practice</i>	45
4.4	Agile project management challenges.....	46
4.4.1	<i>Relevant single variables</i>	47
4.4.2	<i>Aggregate agile challenges</i>	50

Chapter V - Summary, conclusion and recommendations.....	53
5.1 Summary	53
5.2 Conclusion	53
5.3 Recommendations	54
References	56
Annex 1 - Questionnaire	59

List of Acronyms

GDAD – Geographically Distributed Agile Development

ICT – Information Communication Technology

IT – Information Technology

ITSM – Information Technology Services Management

PM – Project Manager

PO – Product Owner

SD – Standard Deviation

SLR – Systematic Literature Review

SM – Scrum Master

List of Figures

Figure 2.2-1 Agile Value Statements 14

Figure 2.2-2 Agile vs Traditional stages 15

Figure 2.2-3 Traditional and Agile method difference 22

Figure 2.2-4 Traditional, Agile and Extreme difference 26

Figure 2.4-1 Conceptual Framework 38

Figure 4.4-1 - Agile phases 51

Figure 4.4-2 Aggregate variables mean bar analysis 51

List of Tables

Table 3.6-1 - Cronbach's alpha score 42

Table 4.3-1 Agile project management practices 44

Table 4.4-1 Agile project management challenges 47

Table 4.4-2 Time and cost overruns 49

Chapter I - Introduction

1.1 Background of the study

Projects, when simply defined as temporary activities in order to produce unique results, well they have been around for thousands of years. Great examples would be the Pyramids, the Great Wall of China, the Parthenon and the Coliseum. One way or the other, these projects were managed using some sort of project management (Seymou & Hussein, 2014).

Modern Project Management is a field of study finding its roots in the 19th century. Kwak identified actually four eras: prior to 1958, 1959 to 1979, 1980 to 1994 and 1995 to present time (Seymou & Hussein, 2014). It's a discipline that teaches the art and the science to bring a specific project from its conception to its realization within acceptable constraints of time, budget, quality and scope.

Today, traditional project management that is still using tools that were developed in the 1950s suffer from what is called scope bloat. This concept, even though there is radical development in technology, affects software developers still using project management methodologies derived from manufacturing processes. This is clearly illustrated in the example of the research of the Standish group where, from the entire functionality of the software Microsoft word, a normal user would always use 7% of the entire functionalities, often the 13%, sometimes the 16%, rarely the 19% and never the 45%. This actually represents a lot of waste of functionalities, hence a lot of waste of software development followed by waste of billions of dollars. (Layton & Ostermiller, 2017)

This scope bloat is what traditional project management practice couldn't address. The need for another project management approach was clear and necessary, and that gave

birth to agile project management.

Agile project management is a model that is actually designed in a way that it minimizes scope creep; it maximizes project success rates, and accepts change. Agile simply, through its adaptation methods, succeeds in minimizing the number of outright failing projects to 9%, a percentage held high up to 29% by traditional project management. (Layton & Ostermiller, 2017)

Yet, agile project management is not the magical solution for every project. First of all, even though it's low, it has a failure percentage. Secondly, high project success rate doesn't necessarily mean the conception, implementation or management of that project is free of challenges. Agile project management has its own challenges that will be discussed later in chapter 2. According to Canty, Agile mitigates risk because it is flexible with adding / changing user requirements at any time in the project, it has regular feedback methods that reduce risk related expectations and it also has reduced investment related risk due to its iterative nature (Canty, 2015). Having this in mind, let's have a brief introduction of the project on which this research is based on.

CoreIT is an IT management system, a platform Starcom developed where IT administrators use to manage their systems. It gives a complete control for the IT administrator with the following modules: asset management module, device administration module, self-help service, incident management module and project management module.

This software developer team consists of 5 sub-teams for each module. Each team

comprises a scrum master and a product owner. There is one project manager, more of a coordinator of the product owners. The developers are grouped in the following manner:

Asset management module - 1PO, 1SM, 5 developers, 2 testers

Device administration module - 1PO, 1SM, 9 developers, 2 testers

Self-help service module - 1PO, 1SM, 7 developers, 2 testers

Incident management module - 1PO, 1SM, 4 developers, 2 testers

Project management module - 1PO, 1SM, 7 developers, 2 testers

1 project manager

This software development is initiated because the client that needed an ITSM for its IT department. It's the software development process that will try to analyze, an agile approach in this case.

This research focuses on this particular project because there are observed problems in its result while using an agile approach. The triple constraints were simply not respected. Indeed the agile values may be different from traditional ones, but project results and success are still bound to these triple constraints. In fact, assessing these problems, in this case the challenges, is also indirectly assessing the agile practice of the company.

Now, let's bring back the focus from CoreIT project to project management.

As in every study field, project management has its own branches and sub branches to address the different requirements of projects. Today, besides extreme, two major approaches of project management are identified, namely Traditional and Agile. The Traditional approach is the one that has been used widely, especially in the construction field where change request in the final product is rare. It's an approach that creates a plan and does not deviate from it. It consists of linear models like waterfall and incremental

ones, whereas the agile approach uses iterative and adaptive models. These models leave room for change at any time for the agile approach. Born officially in 2001 with the coming together of 16 software engineers, the original intents behind its conception were the flexibility, rapid response and change accommodation it promised for software development. It didn't take long before its benefits were realized and the approach started to be applied in other fields than software development. The birth of the agile approach was carried out with what is called the Agile Manifesto, a set of principles laid down by the creators themselves (PMI, 2017).

The Project Management Institute states clearly the four agile values on PMI Agile book. The agile authors, through developing software, have uncovered four fundamental values. These are the four directives that guide an agile approach. The first value is to consider that individuals and interactions are more important than processes and tools. This emphasizes that agile is not an approach that has its eye fixed on process. The second value is that the fact that software is working is more important than one that has a complete and detailed documentation. The third value involves the customer. The agile approach, once again, prioritizes customer collaboration over contract negotiation. In a traditional approach, the latter is prioritized. Finally, the fourth value says it all: agile is not an approach that is designed to follow a predetermined plan, it adapts instead to change along the way. The authors did though specify that when making comparison in values, taking the first value as an example (individuals and interactions over process and tools), they did recognize the latter has their values, but the former has greater value. (PMI, 2017)

1.2 Background of the organization

Starcom network solutions plc is originally a network installation company founded in

2005, Addis Abeba. It carries a vision of becoming a leader in ICT and telecommunications. Since 2011, the company started developing software with the help of freelance software engineers. Three years later, with 8 software developers, it became rapidly one of the top companies for software outsourcing jobs. Today, the company has become one the major software companies in Ethiopia with 67 software developers. CoreIT services management software, PinG network analysis software and VGate ticketing software are some of its successful projects. Besides network installations and software development, the company provides consulting services for organization.

1.3 Statement of the problem

When faced up with a project, the very first step should be deciding how to will tackle it. Certain types of project are done so many times over and over that they are no longer considered as projects but processes (PMI, 2017). Every aspect of the process is known and the risk or uncertainty level of the work done is virtually null. On the other hand, there are projects that begin with clear vision with the final product but the way to get there is not known. «Extensive testimonial data suggests that more than 70 percent of all projects should have used some type of Agile Project Management (APM) model but didn't » (Wysocki, 2011). This indicates that except the cases where every aspect of the project is clearly known, it is better to use an agile approach to get better results. Yet, it is doubtful that this agile approach solves everything and like every approach has its benefits and it problems. The current known fact of the project is that it was implemented using agile approach, it had challenges, and the outcome was a working software outside the triple constraint. The desired implementation of the project is, while still using an agile approach, to address the challenges in order to obtain a working software within the triple constraint. The problem statement then lies on how to do that.

The study begins with the assumption that, in a software development project, it is better to use an agile approach despite the implementation challenges. This assumption finds its roots from existing literature.

This research therefore tries to study the agile implementation in a real project and measure its success. Having this in mind, the researcher will try to answer these research questions that complement each other and try to answer the problem statement:

1. What does the agile implementation at Starcom look like?

This research direction will study the agile practice at hand

2. What are the challenges encountered?

This research question will assess the challenges

3. How to address the challenges encountered?

The answers to this question are basically suggestions Starcom should use for further projects.

1.4 Research Objectives

This research is a study of the case of CoreIT software development project. Its objectives are to assess current practice along with the challenges and indicate directions on how to address these problems in order to improve project outcome with regard to the triple constraint.

1.4.1 General Objectives

This study primarily tries to examine the challenges of agile project management. In parallel, it assesses the practice of agile project management at Starcom Network Solutions

PLC. Finally, the researcher will suggest possible solutions on how to address these issues.

1.4.2 Specific Objectives

1. Assess Starcom's agile practice.
2. Assess existing challenges.
3. Suggest ways to tackle these problems.

1.5 Significance of the study

This study intends to outline the challenges a specific software company faced while implementing an agile project management approach. More precisely, it's a research trying to critically describe how the agile approach was implemented in the case of Core IT software and the results obtained as a consequence. It has therefore an aspect of "what to do's and "what not to do's while implementing the agile approach.

Moreover, it can be said that the software development area in Ethiopia is only at its beginning and there is so much to do. This agile approach, should it prove to be effective and efficient in the Ethiopian context, other software companies can learn from the approach used in the case project. It can be one of the foundations for future Ethiopian software development culture.

The company itself, while using this software development approach and learning how it is successfully implemented, it will adapt and improve its software development process.

The agile approach is also supposed to welcome feedback, and thus in this case, it should enable the software company build an application that fits and suits well the need of its customers, in this case organization that have IT departments and evidently, the Ethiopian

IT professionals.

1.6 Delimitation / Scope of the study

This study will be a descriptive review of the CoreIT software development based on the management approach used in the project. It tries to weigh the challenges of the approach used against the success of the project. This research is not trying to develop new theories or substitutes to the agile approach. It strictly examines the practice of agile approach at Starcom Network Solutions, and concerns only the developer team that worked on CoreIT project.

This research has no need for budget, since all data collection will be through internet. Participants are willing to contribute freely, and the contents in this research will help them improve their practice. This research is limited to Starcom Network Solutions PLC which is located in Addis Ababa. It will be no more of a descriptive research, and the data gathering bound to a questionnaire and product backlog consultation.

1.7 Limitation of the study

The time available for this study does not allow data validity and reliability through triangulation. The reader should take this research as only part 1. In order for the findings to be valid and reliable, the same research with the same parameters and under the same conditions should be undertaken on another software development occurrence.

1.8 Definitions of Terms

All the definitions below are taken from the PMI Agile Book of Project Management

Institute (PMI, 2017).

Adaptive model: a model that emphasizes on decisions being based on previous process results

Agile project management: an approach that is fast and iterative in order to produce intermediate results for directing the next step to undertake

Competitive advantage: market conditions that allow a company to win the market by producing the same quality with competitors but at a lower price

Deliverables: resulting elements that are supposed to be obtained from the implementation of the project

Incremental model: a model of the traditional approach. A proceeding by incrementing and adding a feature to the product.

Iterative model: a model of the agile approach. A cycling through many individual iteration (each with different output) until the desired result is obtained

Lean: project management concept meaning eliminating waste and maximizing efficiency

Project: A project is a temporary endeavor undertaken to create a unique product, service, or result within the constraints of time and cost

Project management: Project management is the set of activities aimed at organizing the smooth running of a project and achieving its targeted objectives within the time and cost set initially

Project manager: Member of the team that manages all the organizational, logistical and practical aspects of a given project

Scope bloat: See scope creep

Scope creep: Scope creep in project management refers to uncontrolled changes in the scope of a project. This phenomenon generally appears when the objectives of a project are not properly defined, documented or controlled

Scrum: project management methodology consisting of short work cycles called sprints

Scrum master: obstacle facilitator in a team that follows scrum methodology in the project

Time to market: time taken to market a product

Traditional project management: an approach that sees events sequentially in a logical order one after the other

Triangulation: verifying data validity through multiple sources

Waterfall model: linear model of the traditional approach. Everything is planned in the beginning and every action sticks to the plan during the implementation phase

1.9 Organization of the research report

The research report is in a standard format. It consists of five major parts. The first chapter which is the introduction contains most of the proposals items. It contains the background of the study, statement of the problem, basic research questions, objectives of the study, definition of terms, significance of the study and scope. The second chapter, which is Literature review, consists of an introduction, a theoretical and empirical literature review that is relevant to the research. The fourth item in this chapter is the conceptual framework. The third chapter, namely research methodology, is basically the explanation of data collection methods, the instruments used, procedures and analysis methods in place. The fourth chapter, data presentation, analysis and interpretation, is the stage where results are shared. Links between the literature review and the results found are the topic of the analysis phase. Finally, the fifth chapter that presents the summary, conclusion and recommendation, is where the problem statement is answered and recommendations are given.

Chapter II - Literature Review

2.1 Introduction

A project is a temporary endeavor undertaken to create a unique product, service, or result (PMI, 2017, p. 4).

A project is the set of actions to be undertaken in order to meet a defined need within fixed deadlines. Thus, a project being a temporary action with a beginning and an end, mobilizing identified resources (human and material) during its implementation, it also has a cost and is therefore the subject of a budgeting of means and a balance sheet independent of that of the company. The expected results of the project are called “deliverables” (PMI, 2017). The difficulty in carrying out the project lies largely in the multiplicity of actors it mobilizes. Indeed, unlike personal projects or small-scale internal projects for which the need and the response to this need can be carried out by the same person or by a limited number of contributors, in a project, the expression of the need and the satisfaction of this need are carried out by generally distinct actors. In this way, it is necessary to ensure throughout the project, that the product in progress clearly corresponds to the expectations of the “customer” (San Cristóbal, et al., 2018). As opposed to the traditional trader model ("seller / buyer") where a customer buys a product already made in order to satisfy a need, the project aims to produce an original creation responding to a specific need that should be rigorously expressed.

Traditional and agile are two methods of carrying out the project management exercise. In our world, especially in the IT, software development projects rarely keep their original requirements. IT project requirement change is not a caprice of the user, but rather a sequence of expression of the client’s needs.

Following the Standish group on success and failure from a sample of 10 000 software development projects that used a traditional approach, 29% failed outright, 60% were challenged (didn't finish either on time, or within budget). The actual percentage that was a success was the remaining 11% (Layton & Ostermiller, 2017). This research actually tries to identify the reasons that lie behind these numbers in the Ethiopian context.

In this literature review, the researcher thoughts are organized in the following manner:

First of all, the definition of what an agile approach is will be given. This includes the philosophies that lie behind agile, like concepts and values. Secondly, the importance of agile project management will be stated. Third, an investigation on how agile works will take place. In the fourth part, the agile management process will be examined. Finally, in the fifth part, some agile problems will be identified. Indeed, parts one to three of this literature review correspond sequentially to the “what”, the “why” and the “how” of an agile management approach. So, here is the “what”...

2.2 Theoretical review

2.2.1 What Exactly Is Agile?

Agile project management, in comparison to traditional project management, is an approach that delivers more quickly. The traditional approach comprises 47 grouped process and 5 process groups, while the agile approach consists of 4 values, 12 principles and a Declaration of Independence. This comparatively speedy approach is what the “agile” adjective brings to the project management (Canty, 2015). Agile changes are inherent, meaning that the approach and change are inseparable. Agile approach is one that

accommodates change very well, unlike traditional projects where change is costly, especially in advanced stages. Other important aspects of the agile approach are the properties “value focused” / “value driven” and risk focused. (Canty, 2015)

The project management methodologies known as agile methods all have one thing in common: they are directly inspired by the Agile Manifesto published in 2001 by software developers who are determined to improve their process and reduce their failure rate. From there were, therefore, gradually born various methods united by a new way of approaching product development by placing in particular the customer at the heart of the action and by advocating the adaptation of creative processes over the course of the evolution of the project. They have since established themselves as standards in the field of application development (Wysocki, 2019).

An agile methodology, whatever it is, provides for the splitting of software development stages. Unlike the traditional method which provides for the total planning of the project even before its development, the Agile Manifesto rather advocates the setting of short-term objectives. The project is thus fragmented into several sub-parts that the team in charge of it must gradually achieve by readjusting the objectives if necessary to meet the client's expectations as much as possible. Agile methods make it a point of honor to strengthen the relationships between the members of the project team, but also between the team and the client. It is for this reason that flexibility and agility in the organization are two fundamental pillars of agile methods (Cho, 2008).

2.2.2 Agile Concepts

The agile concept takes its birth from the “lean” concept. Charles Cobb says “A systematic approach to identifying and eliminating waste through continuous improvement by flowing the product at the demand of the customer.” (Cobb, 2011)

When comparing traditional project manager roles to agile ones, Cobb says the following: “On small-scale projects, the Scrum Master may perform the tasks that a project manager might normally perform; however, the Scrum Master role is very different from most typical project management roles. A number of books have compared the role of the Scrum Master to a “sheep dog” that guards and shepherds a flock of sheep. He/she keeps the sheep from straying too far out of the group and protects the flock from unwanted interference and obstacles. It is much more of a facilitative role than it is a directive role” (Cobb, 2011).

The Agile approach was born in 2001 with its manifesto. Its four values are summarized as follows in a comparative table with traditional approach

Agile Value Statements

Agile Value	<i>Rather Than</i>	Traditional Value
Individuals and interactions	<i>Over</i>	Processes and tools
Working software	<i>Over</i>	Comprehensive documentation
Customer collaboration	<i>Over</i>	Contract negotiation
Responding to change	<i>Over</i>	Following a plan

Figure 2.2-1 Agile Value Statements

(Canty, 2015)

The above table shows us clearly the different perspectives of the two approaches over their concerning their values. Below is another table comparing their stages:

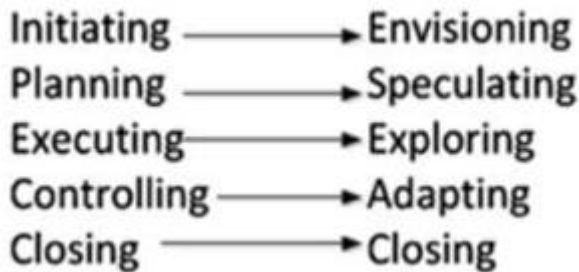


Figure 2.2-2 Agile vs Traditional stages

(Canty, 2015)

Although agile is thought to be a good alternative to traditional projects, yet it is not always the case. It depends highly of the project in question, and it is especially recommended to IT projects.

Here are the twelve guiding principles (PMI, 2017)

1. Our highest priority is to satisfy the customer through early and continuous delivery of valuable software.
2. Welcome changing requirements, even late in development. Agile processes harness change for the customer's competitive advantage.
3. Deliver working software frequently, from a couple of weeks to a couple of months, with a preference for the shorter timescale
4. Business people and developers must work together daily throughout the project.
5. Build projects around motivated individuals. Give them the environment and support they need, and trust them to get the job done.

6. The most efficient and effective method of conveying information to and within a development team is face-to-face conversation.
7. Working software is the primary measure of progress.
8. Agile processes promote sustainable development. The sponsors, developers, and users should be able to maintain a constant pace indefinitely.
9. Continuous attention to technical excellence and good design enhance agility.
10. Simplicity - the art of maximizing the amount of work not done is essential.
11. The best architectures, requirements, and design emerge from self-organizing teams.
12. At regular intervals, the team reflects on how to become more effective, then tunes and adjusts its behavior accordingly.

These principles are pretty much self-explanatory. They are “rules” to follow, more of guidelines, to help us reach our goals effectively using the agile approach.

In fact, the agile approach is one that is based on intermediate results instead of what was planned ahead in the project planning phase. “Agile favors empirical and value-based measurements instead of predictive measurements. Agile measures what the team delivers, not what the team predicts it will deliver” (PMI, 2017, p. 61)

2.2.3 Agile Values

The four agile values are the ones indicated in the introduction of this research:

❖ People and their interactions more than processes and tools

Processes are the backbone of most businesses. They seek to structure things, to frame

everything, which can be understood until a certain point. Focusing on processes and tools ends up holding teams back, preventing them from doing things (PMI, 2017). When you come to have to ask 3 people to put a signature on an order form to order 1 pen, you tell yourself that there is a problem. The first agile value makes it possible to refocus on employees and their interactions, in other words, their collaboration (Scotcher, 2015). This is what creates value. Teams involved, committed, free to take initiatives and organize themselves as they wish to deliver quality work. An agile company is a company that has understood that employees and collaborative work must come before the implementation of tools and processes.

❖ Functional software more than exhaustive documentation

Documentation is essential because otherwise, after several updates of your software you end up with an incomprehensible code, each one adding its own little layer on top of the rest, which is difficult to understand. In the end, there are thousands of lines of code that are kept "just in case" and which are probably not important, and the final result is a super slow code, super heavy to maintain, full of bugs of all kinds... What must be understood here is that although documentation is essential, it is preferable to have a tool that works rather than having exhaustive documentation (Scotcher, 2015). It can always be done later.

❖ Collaboration with customers more than contract negotiation

Very often, with the classic methods of project management, the slightest change, the slightest evolution can call into question the whole project, and will lead to negotiations, on the budget, on the deadlines of the project, etc... This often creates frustration on the client side, and friction between the project team and end users. The agile approach wants

to put end users back at the center of attention (PMI, 2017). Rather than developing software for months underwater and then finally presenting it to the client to get their opinion, the agile approach is smarter: it works by successive iteration. In other words, a piece of software is delivered by piece of software, this allows end users to start testing the product, to make their feedback. And this allows the team to carry out the corrections and adaptations necessary to deliver the end of beautiful software, which corresponds exactly to what the customer wanted. “We collaborate all the time, at all levels and thanks to successive iterations and feedback, we build value” (Scotcher, 2015).

❖ Adapting to change more than executing a plan

Between theory and practice, there is always a gap, more or less important depending on the level of preparation of the project.

Faced with big changes, in classic project management the situation can sometimes feel desperate, without knowing how to integrate these changes. This can lead to the project being stopped or called into question. Agile wants change to be the norm: it's predictable and teams should expect it. The changes are therefore integrated into the developments in progress, which makes it possible to increase the reactivity of the teams in the face of change, and to stabilize the project while controlling the costs (Scotcher, 2015). Changes are no longer considered as problems, but as opportunities. Any change, any evolution, any modification thus becomes an opportunity to bring additional value in the development of the software.

2.2.4 Declaration of Interdependence

The declaration of interdependence was created two years after the creation of the agile

manifesto. It consists of six principles:

1. Increase in return on investment is obtained by making the continuous flow of value our focus.
2. Reliable results are delivered by engaging customers in frequent interactions and shared ownership.
3. Uncertainty is expected and managed through iterations, anticipation, and adaptation
4. Creativity and innovation are unleashed by recognizing that individuals are the ultimate sources of value, and creating an environment where they can make a difference
5. Performance is boosted through group accountability for results and shared responsibility for team effectiveness.
6. Effectiveness and reliability are improved through situational specific strategies, processes, and practices.

(Scotcher, 2015)

These declarations are a kind of an “annex” to the manifesto itself. They are elements that help us achieve our goals successfully (Scotcher, 2015)

2.2.5 Why use agile project management?

Having seen what agile is, now it's time to see the why it is used. Agile project management is commonly used for application or software development because software is continuously evolving and so are the associated needs. In this context, linear project management methods like the waterfall model are less effective. Other reasons also push teams to use agile methods:

Agile methods are adaptable

It is not for nothing that this methodology is called “agile”. One of the main advantages of agile processes in software development is the ability of teams to quickly revise their strategy without disrupting the flow of the project. In the waterfall model, the phases of the project follow each other (Wysocki, 2019). So any change in strategy can disrupt the rest of the project roadmap. Since software development is a much more adaptable field, rapid changes can be complex to manage using traditional project management methods; this is partly why agile project management is preferred for software development.

The Agile approach promotes teamwork

One of the agile principles specifies that to communicate effectively in a team, it is preferable to favor face to face. Encourage teams to break down simultaneously project information, and you have the magic recipe for ideal collaborative teamwork. Even though technologies have evolved since 2001 and telecommuting has gained momentum, the effectiveness of face-to- face work has not changed (Wysocki, 2019).

Agile methods focus on customer needs

Uniquely, software development teams focus much more on customer needs than in other industries. With the rise of cloud software, teams can quickly gather feedback from their customers. Since customer satisfaction is one of the key elements of software development, it is easy to understand why it is part of the agile process. By collaborating with customers, agile teams can prioritize features that meet their needs. When these evolve, the teams only

have to follow an agile approach to move on to another project.

2.2.6 *How agile methods work*

Agile methods start from the principle that specifying and planning in detail the entirety of a product before developing it (predictive approach) is counterproductive (Layton & Ostermiller, 2017). This can be mirrored to planning in detail a road trip From Addis Ababa to Dire Dawa. Specifying each town and village crossed, the associated time of driving, each off-road trip, amount of gas consumed, kilometers traveled, etc. Unforeseen events are expected to happen: traffic jams, closed roads, flooded routes, even car breakdowns, etc. making the planning and specifications irrelevant.

The idea is to set a first short-term objective (i.e. arrival at Kality) and get on the road without delay. Once this first objective has been achieved, you take a short break and adapt your itinerary according to the current situation. And so on until you reach the final destination. This is therefore an empirical approach. As part of a software development project, the client develops his vision of the product to be produced and lists the functionalities or requirements of the latter. He submits this list to the development team, communicates directly with them (rather than by paper) who estimates the cost of each element on the list.

It's therefore possible to get an approximate idea of the overall budget. The team then selects a portion of the requirements to complete in a short portion of time called iteration. Each iteration includes work design, functional and technical specification when necessary, development and testing. At the end of each of these iterations, the partial but usable product is shown to the customer (Layton & Ostermiller, 2017). The latter can then realize for himself very early on the work carried out, the alignment with the need. The end user can

project himself into the use of the product and provide valuable feedback for future iterations. The offered visibility is the key.

This transparency can also bring more trust and collaboration in the customer/supplier relationship. As for the risks, they are lifted very early.

If the customer has carefully prioritized his need, he can seize the opportunity to accelerate the "time to market" if he considers that the product in its (partial) state can go into production. Thus saving its budget and obtaining a first return on investment. He also has the ability to change the priority of features that have not yet been developed (planned for future iterations) along the way. In order to delay a feature whose need is not that important; add a new crucial feature in exchange of it (thus respecting budget and deadlines), etc.

Traditional methods, also called waterfall model, is visually compared in Fig 2.7-1 with agile way of functioning on <https://kissflow.com>, an online project management application:

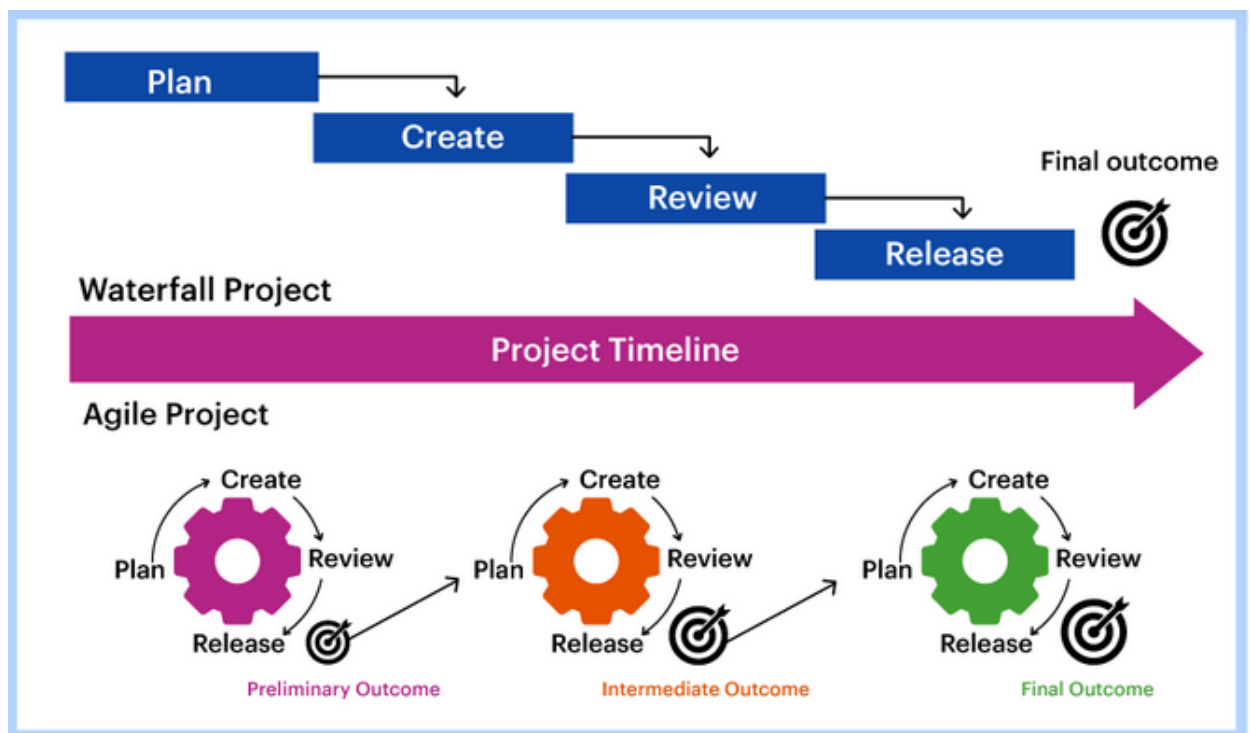


Figure 2.2-3 Traditional and Agile method difference

The Project Management Institute lists in the PMI agile book (PMI, 2017) the following most popular agile methods.

The Agile framework encompasses various approaches, some of the most common of which are:

❖ Kanban

The Kanban method is a visual approach to the agile methodology. Teams use online Kanban boards to indicate what stage of the development process certain tasks are at. The tasks are represented by cards on a board, and the different stages in the form of columns. When teams are working on a task, the cards are moved from the backlog column to the column that corresponds to the stage of the task. This method helps teams identify obstacles and visualize the work accomplished.

❖ Scrum

SCRUM consists of three principles, namely transparency, inspection and adaptation. In Scrum, the process is iterative and incremental, requirements are permitted to change over a period of time, and the end users are participating actively throughout the project. Yet having these advantages, this method is prone to a member leaving the team and that has major impact on the project. Experience is of great importance, and the method is not complicated at all. The Scrum methodology is a common agile approach for small teams. The team is led by a Scrum Master, whose main mission is to remove all obstacles faced by those who perform routine tasks. Work is executed in the form of sprints, and Scrum teams meet on a daily basis to discuss ongoing tasks, roadblocks, and anything else that may be hindering the development team.

❖ Adaptive Project Framework (APF)

The Adaptive Project Framework method was born from the idea that unknown factors can arise at any time during the project. This technique is mainly used for IT projects, for which more traditional project management techniques are not suitable. This framework is based on the idea that project resources can change at any time. For example, the budget may change, as may the deadlines or the availability of team members. The APF method focuses on the resources the project has, rather than those it needs.

❖ Extreme Project Management (XPM)

The XPM methodology is often used for complex projects subject to a high level of uncertainty. It requires constantly adapting the processes, until they give the desired results. This kind of project involves a lot of spontaneous changes, and teams can adapt their strategy from week to week. The XPM methodology requires a lot of flexibility, which is why each sprint lasts no more than a few weeks. This methodology leaves room for frequent change, a trial-and-error approach to problem management, and many iterations of self-correction. XP has its advantages, like being very responsive to change, high priority features are developed first and pair programming enhances creativity. On the other hand, it has large overhead, the order of feature importance can be subjective and it should not be used on large projects with large teams.

❖ Adaptive Software Development (ASD)

This Agile methodology allows teams to adapt quickly to changing requirements. This process is mainly based on continuous adaptation. The phases of this type of project (speculation, collaboration and learning) promote continuous learning as the project

progresses. It is not uncommon for teams following this methodology to be in all three phases at once. As the structure of ASD is not linear, its different phases often overlap. This type of management is very fluid, so the constant repetition of the three phases is likely to help team members identify and solve problems faster than following traditional project management methods.

❖ Dynamic Systems Development Method (DSDM)

The DSDM method focuses on the complete life cycle of the project, which is why it is based on a rigorous structure, unlike other agile methods. The DSDM method is divided into four phases:

- Feasibility study and business study (functional analysis)
- Iterative functional mode
- Iterative design and production
- Implementation

❖ Feature Driven Development (FDD)

Feature Driven Development brings together several agile best practices. Although also based on iterations, this model focuses more on the software features developed by the team. It is largely dependent on customer input, as the team prioritizes the features the customer needs. This model allows teams to update their projects frequently and easily implement solutions in the event of an error, as the phases of the Feature Driven Development framework are constantly evolving.

Here is a summary figure comparing traditional, agile and extreme approaches:

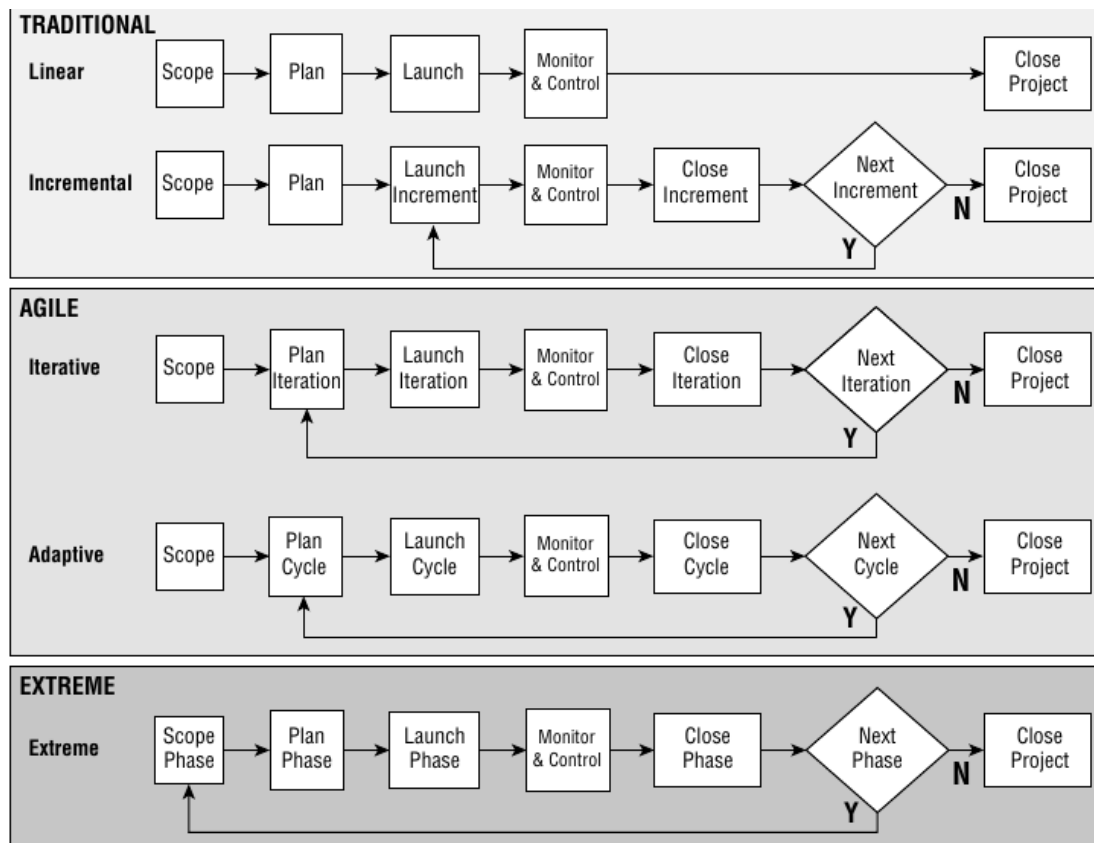


Figure 2.2-4 Traditional, Agile and Extreme difference

2.2.7 Agile Project Management Process

Agile lifecycle differ from traditional ones as they are emergent, incremental, iterative and evolutionary (Goodpasture, 2010).

Goodpasture has identified the following agile merits and demerits (Goodpasture, 2010). Agile can deliver numerous benefits. First of all, it provides rapid and frequent supplies to production (software). Business milestones are held in high regard. Further, relying on the agile fundamentals of change, the project remains current and relevant by effectively adapting to changing customer priorities and requirements. For teams of 25 or less developers, this approach is a very cost-effective solution. Another important aspect to keep in mind is that customers are given a place at the table to help shape the project's

value proposition as it develops.

From the developer team side, each successful iteration and release provides a sense of accomplishment as well as a reason for celebration and reinforcement. (Goodpasture, 2010).

Generally speaking, the project's goal is to focus on the customer, rather than following an outdated plan.

Yet, the agile approach is not a problem free walk in the park. Agile approach is always subject to change in requirements; there is therefore a lack of commitment to the total cost and scope. Also, the team staffing is vulnerable to turnover. Besides these obvious ones, scaling small-team dynamics to an enterprise-scale project is difficult. Same goes for scaling without a commitment to documentation is difficult.

Furthermore, contracting the work team is difficult because the requirements and scope are not known with sufficient precision.

Finally, it depends a lot on good logistics for team co-location, face-to-face communications, a pool of competent multi-disciplined employees, and fast access to knowledgeable and empowered clients or end users. (Goodpasture, 2010).

2.2.8 Agile problems

Deepa Kurup in his research paper for ISM6316.001 Project Management on summer 2015 at University of South Florida, citing the last edition of State of Agile Development survey (2014), indicated the following challenges of Agile Project Management:

Communication in Distributed Environments

Outsourcing is becoming a big part of every organization. Likewise, globalization is also another factor that pushes us to work online from every corner of the planet through internet. It is very common these days for a programmer in India work for a company located in the US. Therefore, it is clear that various parts of the team which are separated geographically have communication problems.

Developer's Skill and Motivation levels

Agile management suggests periodic reporting and tracking progress through scrum meetings. On the other hand, these meetings might highlight errors made. Sometimes, the developers are not comfortable with these as their errors are judged or may unveil their lack of skill.

Stress on Social Skills of the team

Agile process, beside the fact that the developers need technical expertise, they are required to have social skills in communication, since they are expected to communicate efficiently with clients as well as with other members of the team in order to succeed.

Lack of business knowledge in the team

When the size of an agile team or the complexity of the product grows, keeping each developer up to speed on business information, which is mostly received from the business team or the customer directly on a daily basis, becomes a difficulty. Because each team member is regarded a complete expert in his subject area, this influence is greater for agile.

Hybrid and waterfall and agile processes

Since upper management prefers to see a strategy in place, most businesses now try to combine Agile and Waterfall approaches. This strategy involves creating a plan based on the traditional waterfall paradigm, but allowing the development team to work in an agile style. In the majority of situations, there have been disagreements between the development team and management over the true nature of the process and reporting.

Inefficient Customer Feedback Process

Agile technique relies on a feedback process that allows you to see the results of your work and offer changes as needed. This essential procedure necessitates the team's knowledge of the genuine consumer. (Sometimes it's the Product Manager, other times it's the Business/User Groups.) Any ambiguity in this job and duty definition can lead to scope creep, disagreements, overlooked inputs, wishful programming, and other problems.

(Kurup & Sidhardhan, 2015)

Scaling Agile to larger project teams:

For the sake of simplicity, the most widely used approach of agile methodology which is scrum is selected. Remember, agile teams are not supposed to be large. In fact, the ideal number would be 7, even though team members ranging from three to nine are possible (Layton & Ostermiller, 2017). So, how is it possible to deliver big software in very limited amount of time?

When there is a considerable amount of software to develop in a short time, usually the software is decomposed into modules. Each modules will then have a team that will develop it. Indeed, a project may have multiple agile teams, with each one of them having their own scrum master and product owner. Yet, there should be a system or person that should coordinate the product owners as there may arise prioritization problems. Only one product backlog should exist in such cases in order to centralize documentation of product modification (Layton & Ostermiller, 2017).

Agile problems are not limited to the above mentioned ones. These are just some examples as it is not possible to give an exhaustive list.

2.3 Empirical review

There have been many studies carried out to identify success factors of agile methodology. While this research is not trying to identify the success factors, it uses findings from these studies to articulate recommendations for CoreIT development project. Multiple articles have undertaken research concerning agile challenges, but not many are focused on Ethiopian software companies.

Agile challenges are broad. They can happen at any of the five major success factors, namely organizational factors, people factors, process factors, technical factors and project factors (Chow and Cao). It is not possible to give an exhaustive list of agile challenges, hence the researcher rather focuses on a limited number of already identified ones on real cases.

Here are 11 case studies relevant to this research where agile practice has been checked, and agile challenges have been identified. In this section, the word « problem » is used to refer to challenges in an agile practice.

Case 1 - A Survey of Key Challenges of adopting agile in global software development: a case study with Malaysia perspective

Agile methodologies are beneficial for use in regional software development projects. Global software deployment that is agile, however, requires care. Agile global software development faces a number of major obstacles that must be overcome in order to improve the processes. A quantitative approach was intended to show a real-world case scenario of these specific major issues experienced by one of the software companies, BestWeb Malaysia. To determine whether the major difficulties identified in the systematic review still exist, the results of quantitative methods and systematic literature are compared. By implementing the best practices in their international software development projects, small and medium-sized businesses can avoid these difficulties with the help of the proposed search and its exploratory results. According to this survey (Batool, 2019), three major problems were identified:

- lack of motivation
- lack of trained staff in agile
- lack of knowledge in management

Case 2 - Critical Success Factors for Agile Project Management in Non-Software Related Product development Teams

Jeff Totten in his research tried to see whether agile methods (like scrum) that were already popular and successful in the software world had the same profile in non-software projects. In his study (Totten, 2017), the results unveiled the following problems :

- Team members on multiple projects
- Project interruptions
- Lack of clear company vision
- Unclear project scope

- Product owner on multiple projects
- Lack of upper management support
- Lack of product owner involvement
- Team distributed in diff. Locations

Case 3 - Identifying Benefits and Challenges in the Agile methodologies in software development

The outcome of Alistair Streek's empirical investigation to identify challenges on agile software development show that improved communication between development team members is the main advantage. The calculation of the time and effort required to accomplish Agile projects and the fact that Agile only functions with competent, motivated individuals were significant challenges. The survey (Streek, 2015) has found the following problems and indicates that projects fail when:

- Project teams are not motivated.
- Teams do not have the support from management.
- Change control is locked in a traditional project management loop.
- Project progress is still managed in terms of milestones and features and not working software.

Case 4 - Agile Software Development: A survey of early adopters

Case studies have revealed evidence of primarily favorable experiences with the application of agile development approaches. In order to learn more about the factors influencing the adoption and use of agile practices, as well as their advantages and challenges as perceived by early adopters of this software development methodology, Vijayasathya and Turk used a

survey strategy, which enables data collection from a larger sample of organizations. According to their poll, individual motivation drives the adoption of agile practices, but managerial passivity and organizational opposition serve as major roadblocks. The following are the major identified problems (Vijayasathy & Turk, 2008) :

- Organisational Resistance
- Management Apathy
- Inadequate Training

Case 5 - An empirical study of agile software methodologies: a Sri Lankan Perspective

This study conducts an empirical investigation on Sri Lankan software professionals' perceptions of the use of Agile techniques. It includes the Agile approaches that are believed to be most beneficial. The authors want to educate the nation's Agile methodology practitioners and make them more aware of their practices. The early adopters who plan to employ agile methodologies in the future might use this document as a resource (Safwan, et al., 2013). In this research, the following problems are identified :

- Costly
- Insufficient team support

Case 6 – Agile Methods in Ethiopia : An empirical study

This study was intended to find out whether Ethiopian companies have adopted the agile methodology in their software development process and it offered empirical proof of the adoption of agile methodology in Ethiopia's smaller businesses. The significance of agile concepts, values, and behaviors was noted by research participants. Agile methodologies are applied to resolve requirements-related problems and promote user involvement. However, it was found that the study's participants were working on software projects for the government,

which required extensive documentation and complex stages approval processes. They also used fixed price contracts with predetermined delivery dates Agile Methods in Ethiopia: An Empirical Study (Regassa, et al., 2017). The following are the most important problems encountered :

- Contract challenges
- User involvement challenges

Case 7 - Agile Software Development Vulnerabilities and Challenges: An Empirical Study

A significant change from conventional, plan-based approaches to software engineering is represented by agile software development. Up until and including 2005, empirical research on agile software development was systematically reviewed. 36 of the 1996 studies that were found using the search approach were empirical studies. Introduction and adoption, human and social variables, views of agile methods, and comparative studies were the four categories used to organize the studies. The review looks into what is currently known about the advantages and drawbacks of agile methodologies as well as the reliability of the supporting data. (Wei, et al., 2022). Here is one major problem identified in the research :

- Open source vulnerability for hackers

Case 8 - Empirical studies of geographically distributed agile development communication challenges: A systematic review

Studying and using regionally distributed agile development is becoming more popular (GDAD). There has been a lot written on GDAD communication. The body of research on the communication difficulties associated with GDAD needs to be thoroughly reviewed and synthesized. 21 pertinent empirical articles were found and reviewed in this paper using the

SLR approach and specific search criteria generated from the research topics. In order to identify communication difficulties and the methods employed to address them, data from these studies was extracted. When deciding on future research priorities and directions, GDAD practitioners and researchers can refer to the results of this study as a resource (Alzoubi, et al., 2016). The major studied problem in this research is :

- Geographical challenges

Case 9 - The Impact of Agile Development Practices on Project Outcomes

Challenges within the Team

The study demonstrates that the majority of the Agile approaches under investigation have an impact on project success conditions that is both favorable and negative. Only a small number of the frequently mentioned Agile techniques are demonstrated to have an entirely positive impact. The study identifies discrepancies in the quantity of research on the results of the various Agile methods that is currently available. This study identifies Agile practices and impacts that require additional study. This study demonstrates that Agile practices should be applied in practice while keeping in mind the organizational and contextual environment, as well as the desired outcomes (Ghimire & Charters, 2022). Here are the major identified problems :

- Communication between team members
- Project priority disagreement with customer

Case 10 - Empirical Findings in Agile Methods

There hasn't been any empirical data collecting, analysis, or classification of this effectiveness or of the proper conditions for Agile initiatives. An online workshop was conducted by researchers from four different institutions to allow for simultaneous and virtual discussion and to collect experiences and information from 18 Agile experts located all over the world. These professionals described Agile Methods and shared their experiences applying them from small to very big teams. They talked about how crucial it is to have talented developers on Agile teams. They discussed universal success elements and recognized early indicators of issues in Agile initiatives. Researchers and practitioners who are building an experience base for better decision making can make use of these and other discoveries and heuristics obtained through this beneficial interaction (Lindvall, et al., s.d.). Here are the problems identified :

- Team gets behind planned sprints
- Useless documentation
- Daily meetings should not be overlooked
- Low morale

Case 11 - Empirical studies of agile software development: A systematic review

Software requirement analysis is a crucial stage in software development. In reality, stakeholders and system engineers talk about software needs during meetings. Software requirements' quality is influenced by the caliber of meetings. Poor meetings will result in poor software. However, it can be challenging to gauge the effectiveness of meetings because oral interactions between stakeholders and system engineers typically take place during meetings for software needs. As a result, the research suggests a new metric for determining how well software requirements are met. Measuring simply who and when speaks is one of the measures' features. The metrics can be easily modified to different software domains

because they don't rely on natural language approaches to analyze the substance of talks (Dingsoyr & Dyba, 2008). Here is one identified problem :

- Meeting quality metric

2.4 Conceptual Framework

In this part of the research, the following variables are declared :

1. Agile practice at Starcom

This variable studies the agile practice at Starcom. It will be compared against literature standards in order to find its standing point

2. Agile challenges at Starcom

This variable studies the challenges encountered during the projet development. A search for solution in literature and past agile project implementations will provide a solution.

The study of these two variables will help the researcher suggest for Starcom solutions for having successful project outome withing the triple constraint.

As this study tries to describe the agile culture at Starcom along with the challenges it carries and suggest solution , the following conceptual model is developed:

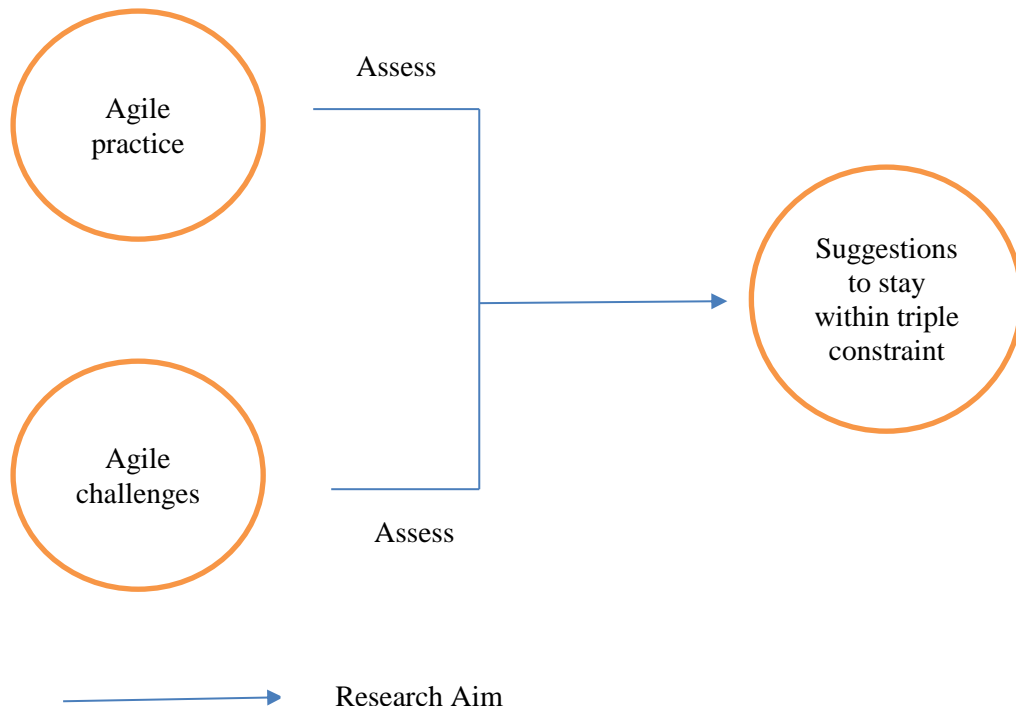


Figure 2.4-1 Conceptual Framework

Chapter III - Research Methodology

3.1 Introduction

In this part of the research paper, the researcher will discuss in the first place the research approach, methods and design that are used with justification. Later, data types, data collection methods and sources will be presented. Once our tools specified, it's time to identify the target population. Finally, after data collection, data analysis will be presented to the reader. To conclude, ethical consideration and organization of the study will be specified.

3.2 Research approach and design

3.2.1 Research approach

The study tries to describe the use of agile at Starcom PLC through Likert scale questionnaire. It would then make sense to do a quantitative analysis of the data that will be gathered. Therefore, the focus will be on key indicators that

- will assess the practice of agile
- will assess the challenges of agile

These key indicators of agile challenges will be the why the project is or is not within the famous triple constraints of time, cost and scope.

3.2.2 Research design

This study uses a descriptive approach. It assesses in detail the agile approach used

and examine the agile challenges associated with it in the case of CoreIT software development through collected data. This process is executed in a univariate descriptive statistics approach.

3.3 Data types, sources and method of data collection

3.3.1 Data types

The data type for this research is a quantitative one. The questionnaire consists of 34 variables. These variables are summarized in two different categories (part 1 and 2) and form the following aggregate variables:

- Agile project management practice
- Agile project management challenges

This questionnaire is adapted from Amna Sowaidan's research paper entitled "The impact of agile project management on the effective delivery of innovative products" (Sowaidan, 2016) and Alistair Streek's "Identifying benefits and challenges in the application of agile methodology in software development" research paper (Streek, 2015).

3.3.2 Data sources

Primary data is obtained from the developing team itself through the questionnaire responses. Secondary data will be taken from the developed software and the product backlog.

3.3.3 Data collection

Data collection method from the project team was conducted through questionnaires (through Google forms). A Likert scale with a 7 option was used to capture project participants' level of agreeing with prepared statements. These statements are in the majority affirmative ones. Negative ones are also included to test responses validity (whether responses of an individual are consistent with each other). This method was used to detect careless questionnaire fillings.

3.4 Target Population

The target population consisted of 53 individuals, total number of the development team of the CoreIT software project. It included the software developers, the scrum masters, the product owners and the project manager, whose role was also to coordinate the product owners.

The target population was not big in size. Since the total number was below 100, census survey was used. Complete enumeration survey method was implemented where every individual was included in the research.

3.5 Data analysis

The fact that this research followed a descriptive approach dictated the data analysis phase. Indeed, the data analysis was a descriptive one.

Descriptive analysis methods such as frequency, mean and standard deviation were employed to analyze the data. The choice for these specific methods was made based

on the interval nature of the gathered data. SPSS data analysis software was the tool chosen for the above indicated calculations.

3.6 Data validity and reliability

- a. In order to ensure data reliability, Cronbach's alpha may be used (Pallant, 2001).

Using SPSS,

The coefficient alpha is calculated for every identified aggregate variable and the following results are obtained:

	Reliability Statistics	
Variable	Cronbach's Alpha	N of Items
Agile practice	.811	16
Agile challenges	.717	18

Table 3.6-1 - Cronbach's alpha score

According to Pallant, a Cronbach's alpha that is between 0.7 and 0.9 is considered to be reliable.

- b. Questionnaire data validity was obtained from advisor and 3 software development experts: one consultant from Starcom company itself, and two others from Leapfrog Software Technology Africa plc.

3.7 Ethical consideration

This research was based solely on voluntary participants. The data gathering details were communicated to the participants and they responded within their own schedules. Every response was anonymous.

Chapter IV - Data presentation, analysis and interpretation

4.1 Introduction

The questionnaire was created online using google form multiple grid format.

Part 1 and 3 of the questionnaire were presented in a positive manner, and part 2 in a negative manner to ensure data validity. In order to be able to analyze the questionnaire responses, numbers or scores are attributed to the responses available:

“Strongly agree” was attributed the highest score, i.e. 7, “Agree” was attributed score 6, “Somewhat agree” score 5, “Neutral” score 4, “Somewhat disagree” score 3, “Disagree” score 2, and finally “Strongly disagree” the least score 1.

It is important to note that the data was not analyzed while respecting the questionnaire’s theme categorization. This was due to the fact that questions were found to be relevant to more than one category. They were therefore exploited in categories where they were found to be the most pertinent. In this section, the word variable is used to describe a question.

4.2 Response rate

Out of 53 questionnaires form 48 were collected. This represents a response rate of 92.46 %.

4.3 Agile project management practices

The variables that assess agile project management practices are summarized below:

No	Variable	Freq & %	Response							Total	Mean	SD
			Strongly Agree	Agree	Slightly Agree	Neutral	Slightly Disagree	Disagree	Strongly Disagree			
1	Starcom uses agile method for all software development projects	Freq.	13	25	9	1	0	0	0	48	6.04	0.74
		%	27.08%	52.08%	18.75%	2.08%	0.00%	0.00%	0.00%	100.00%		
3	Team members have good coordination with scrum master	Freq.	10	25	11	2	0	0	0	48	5.9	0.78
		%	20.83%	52.08%	22.92%	4.17%	0.00%	0.00%	0.00%	100.00%		
4	Scrum masters have good communication with Product Owners	Freq.	6	32	8	2	0	0	0	48	5.88	0.67
		%	12.50%	66.67%	16.67%	4.17%	0.00%	0.00%	0.00%	100.00%		
7	Product backlog is continuously maintained	Freq.	14	26	7	1	0	0	0	48	6.1	0.72
		%	29.17%	54.17%	14.58%	2.08%	0.00%	0.00%	0.00%	100.00%		
9	The project manager has good communication with the client	Freq.	12	23	11	2	0	0	0	48	5.94	0.81
		%	25.00%	47.92%	22.92%	4.17%	0.00%	0.00%	0.00%	100.00%		
10	The software requirements were clearly stated	Freq.	13	20	13	1	0	1	0	48	5.88	0.98
		%	27.08%	41.67%	27.08%	2.08%	0.00%	2.08%	0.00%	100.00%		
11	You have achieved your goal	Freq.	21	18	8	0	1	0	0	48	6.21	0.87
		%	43.75%	37.50%	16.67%	0.00%	2.08%	0.00%	0.00%	100.00%		
12	The teams achieved their goal	Freq.	14	32	1	1	0	0	0	48	6.23	0.59
		%	29.17%	66.67%	2.08%	2.08%	0.00%	0.00%	0.00%	100.00%		
13	The product was delivered according to specifications	Freq.	11	30	5	2	0	0	0	48	6.04	0.71
		%	22.92%	62.50%	10.42%	4.17%	0.00%	0.00%	0.00%	100.00%		
16	Agile is better than traditional for software development	Freq.	10	30	8	0	0	0	0	48	6.04	0.62
		%	20.83%	62.50%	16.67%	0.00%	0.00%	0.00%	0.00%	100.00%		
28	The software is intended to solve a problem observed	Freq.	5	22	15	6	0	0	0	48	5.54	0.85
		%	10.42%	45.83%	31.25%	12.50%	0.00%	0.00%	0.00%	100.00%		
29	The client initiated the project	Freq.	2	10	17	14	3	2	0	48	4.75	1.12
		%	4.17%	20.83%	35.42%	29.17%	6.25%	4.17%	0.00%	100.00%		
30	Software development process uses agile approach	Freq.	17	25	6	0	0	0	0	48	6.23	0.66
		%	35.42%	52.08%	12.50%	0.00%	0.00%	0.00%	0.00%	100.00%		
31	The software development process has 6 stages	Freq.	21	17	9	1	0	0	0	48	6.21	0.82
		%	43.75%	35.42%	18.75%	2.08%	0.00%	0.00%	0.00%	100.00%		
32	The iteration phase includes feedback	Freq.	15	22	8	2	0	0	1	48	5.96	1.09
		%	31.25%	45.83%	16.67%	4.17%	0.00%	0.00%	2.08%	100.00%		
Aggregate mean and SD of agile practices										5.93	0.37	

Table 4.3-1 Agile project management practices

4.3.1 Relevant single variables

The above table contains 15 variables that describe the agile practice at Starcom.

According to this table, only two variables, 29 and 32, with means of 4.75 and 5.96 respectively, have a standard deviation greater than 1. These are therefore the variables that obtained the most dispersed responses, and hence show greater team members disagreement than of the rest of the variables. Yet, the mean of 4.75 of variable 29 is closer to 5 (somewhat agree) and 5.97 of variable 32 to 6 (agree). Concerning the rest, all 13 variables have a mean closer to 6 and a standard deviation less than 1. This indicates that most the team members endorse the agile practices.

4.3.2 Aggregate agile practice

The means of the questions in the table 4.3-1 are used as input data to compute the mean of the aggregate variable agile practice. Likewise, the standard deviation is computed on the means of the questions. The values obtained are 5.93 and 0.37 respectively. This can be interpreted as follows. When the team members are presented with questions suggesting good agile practice at Starcom, they tend to agree with a conclusive majority. On top of that, 13 variables means fall within one standard deviation (5.93 ± 0.37). That represents 86.5% of the data. This percentage is higher than the one of a Gaussian / normal distribution (68%). In another words, 86.5% of the data is concentrated very closely to the mean. To finalize, 94.25% of the data lies within 2 Sd, and 100% within 4 Sd. This tendency of having uniform responses somehow indicates that Starcom has some good agile practices. According to table 4.3-1, it show the existence of good communication between the client, project manager, scrum masters and team members (variable 3,4,9), that feedbacks are included in iteration loops (variable 32), that product backlog is maintained (variable 7), and software requirements are clearly stated (variable 10) are indicators of a good agile practice in place (PMI, 2017).

4.4 Agile project management challenges

The variables that assess agile project management challenges are summarized below:

No	Variable	Freq & %	Response							Total	Mean	SD
			Strongly Agree	Agree	Slightly Agree	Neutral	Slightly Disagree	Disagree	Strongly Disagree			
5	Product owners have good communication with project manager	Freq.	1	2	4	13	16	6	6	48	3.27	1.38
		%	2.08%	4.17%	8.33%	27.08%	33.33%	12.50%	12.50%	100.00%		
14	The project was finished on time	Freq.	0	1	2	18	16	10	1	48	3.27	0.98
		%	0.00%	2.08%	4.17%	37.50%	33.33%	20.83%	2.08%	100.00%		
15	The project managed to stay within budget	Freq.	1	0	3	17	16	9	2	48	3.29	1.11
		%	2.08%	0.00%	6.25%	35.42%	33.33%	18.75%	4.17%	100.00%		
17	The developers have good programming skill	Freq.	12	17	11	4	1	2	1	48	5.52	1.41
		%	25.00%	35.42%	22.92%	8.33%	2.08%	4.17%	2.08%	100.00%		
19	Team members have good communication between them	Freq.	9	20	11	4	2	1	1	48	5.48	1.32
		%	18.75%	41.67%	22.92%	8.33%	4.17%	2.08%	2.08%	100.00%		
20	Team members have good coordination with scrum master	Freq.	10	15	13	6	1	3	0	48	5.38	1.35
		%	20.83%	31.25%	27.08%	12.50%	2.08%	6.25%	0.00%	100.00%		
22	The project manager managed well Pos in relation with project priorities	Freq.	0	1	3	5	15	21	3	48	2.73	1.09
		%	0.00%	2.08%	6.25%	10.42%	31.25%	43.75%	6.25%	100.00%		
23	The team motivation level is high	Freq.	1	1	1	14	18	10	3	48	3.15	1.17
		%	2.08%	2.08%	2.08%	29.17%	37.50%	20.83%	6.25%	100.00%		
26	Upper management approved the use of agile	Freq.	12	18	13	4	1	0	0	48	5.75	1.00
		%	25.00%	37.50%	27.08%	8.33%	2.08%	0.00%	0.00%	100.00%		
27	Scope change was managed successfully	Freq.	1	1	3	8	22	13	0	48	3,17	1,10
		%	2.08%	2.08%	6.25%	16.67%	45.83%	27.08%	0.00%	100.00%		

Table continued

No	Variable	Freq & %	Response							Total	Mean	SD
			Strongly Agree	Agree	Slightly Agree	Neutral	Slightly Disagree	Disagree	Strongly Disagree			
2	The team received training on agile methodology	Freq.	0	0	7	14	14	8	5	48	3.21	1.20
		%	0.00%	0.00%	14.58%	29.17%	29.17%	16.67%	10.42%	100.00%		
6	Every team uses the same product backlog	Freq.	28	17	2	1	0	0	0	48	6.50	0.68
		%	58.33%	35.42%	4.17%	2.08%	0.00%	0.00%	0.00%	100.00%		
8	The sprints of the different teams were synchronized	Freq.	2	0	1	4	14	15	12	48	2.48	1.38
		%	4.17%	0.00%	2.08%	8.33%	29.17%	31.25%	25.00%	100.00%		
18	Team roles were clearly defined	Freq.	13	20	8	3	3	1	0	48	5.71	1.24
		%	27.08%	41.67%	16.67%	6.25%	6.25%	2.08%	0.00%	100.00%		
21	Having multiple product owners had a positive impact	Freq.	0	0	1	5	15	24	3	48	2.52	0.85
		%	0.00%	0.00%	2.08%	10.42%	31.25%	50.00%	6.25%	100.00%		
24	Efficient customer feedback exists	Freq.	4	21	13	5	4	1	0	48	5.27	1.16
		%	8.33%	43.75%	27.08%	10.42%	8.33%	2.08%	0.00%	100.00%		
25	The fact that the development team was physically in the same location while developing helped	Freq.	6	17	20	3	1	0	1	48	5.42	1.09
		%	12.50%	35.42%	41.67%	6.25%	2.08%	0.00%	2.08%	100.00%		
33	The testing phase is automated	Freq.	0	1	9	17	7	8	6	48	3.38	1.36
		%	0.00%	2.08%	18.75%	35.42%	14.58%	16.67%	12.50%	100.00%		
34	Testing helped produce reliable software	Freq.	9	29	10	0	0	0	0	48	5.98	0.64
		%	18.75%	60.42%	20.83%	0.00%	0.00%	0.00%	0.00%	100.00%		
Aggregate mean and SD of agile challenges										4.57	1.10	

Table 4.4-1 Agile project management challenges

4.4.1 Relevant single variables

The above table gathers variables that identify challenges in the CoreIT project.

According to table 4.4-1, variables that have the lowest means in this manner are:

- variable 5 indicates that Product Owners have poor communication with the project manager

- Variable 14 indicates that the project was not finished on time
- Variable 15 indicates that the project has cost overrun
- Variable 22 indicates that the way the project manager dealt with the Product Owners when it comes to project priorities was poor
- Variable 23 indicates that the project team members have low motivation level
- Variable 27 indicates that scope change was managed poorly

The following is also observed:

- Some challenges are successfully dealt with like variable 6 - usage of a single product backlog (mean is 6.5 with an SD of 0.68) and variable 34 – testing helped produce reliable software (mean of 5.98 and SD of 0.64). These are the variables that had a low SD. The team members have a uniform response to these questions. On another hand, variable 18 – clearly defined team roles (mean of 5.71 and SD of 1.24), variable 24 – efficient customer feedback exists (mean of 5.27 and SD of 1.16), variable 25 – being in the same physical location helped (mean of 5.42 and SD of 1.09) have a higher SD comparatively.
- Some challenges seriously threaten the project: variable 2 indicates there's a lack of training (the mean is 3.20, SD is 1.20), variable 8 shows unsynchronized sprints (mean is 2.48, SD is 1.38). These can cause delay and incoordination. Furthermore variable 21 suggests that the multiplicity of product owners created problem (mean is 2.52, SD is 0.85), and variable 33 unveils there were automation problems in testing (mean is 3.38, SD is 1.36).

This multiplicity of product owners might not be a problem by itself in theory. What's important when multiple product owners exist is that a coordinator (or project manager)

makes sure all of them have the same project priorities (Layton & Ostermiller, 2017). Another challenge to look out that Layton and Ostermiller insist on is the testing phase automation, which is considered by the team members as not addressed with a mean of 3.38.

These are challenges the project may have to overcome in order to meet the time and cost constraints (Conboy, et al., 2011). In fact, the project experienced cost overruns and delays in its schedule. The following table summarizes data found from the project manager:

	Expected value	Actual value	Difference
Cost (in Birrs)	215 000	273 000	58 000
Time (in months)	4	5.5	1.5

Table 4.4-2 Time and cost overruns

As the table clearly shows, the project was 45 days late and incurred a cost overrun of 58 000 birrs. What is known at this stage is that these overruns and delays are not due to client scope change request according to the product backlog. The time and cost overrun may be explained by the above variable or challenges. Further study is indeed necessary to find any existing relationship between these challenges and the project delays. According to table 4.4-1, variable 14 and 15 that surveyed time and cost overruns have respectively means of 3.27 and 3.29, which can be considered as low in this data set. This means that the majority did not agree with the questions. This can be interpreted as the team members can identify challenges that CoreIT project faces. In fact, the challenge that variable 5 presents joins the communication challenge (Cho, 2008).

4.4.2 Aggregate agile challenges

When considering the aggregate “agile project management challenges” variable, its mean is less than the previous one, but this time its standard deviation is 1.10 which is much greater. This indicates that variables are much scattered in this aggregate variable than the previous study which is on agile practices.

Table 4.4-1 shows how data is scattered across the possible values. Even though the mean is 4.10, the table clearly shows that there is a significant amount of scattered data with lower scores. This explains the larger amount of 1.10 SD. In fact, 50% of the variables are found within one SD (4.57 ± 1.25), and the other 100% within 2 SD. When compared with a normal distribution, this data is more scattered. The items with low score in the table represent identified challenges by the team members. The ones with high scores represent successfully deal with challenges.

According to Layton and Ostermiller, team members training is one important challenge to look out for (Layton & Ostermiller, 2017). Other common challenges can be ineffective product owner, lack of automated testing, lack of transition support, inappropriate physical environment, poor team selection, discipline slips, lack of support for learning, use of a multiple product backlogs and “diluting until dead (watering down agile processes with old waterfall habits erodes the benefits of agile processes until those benefits no longer exist)” (Layton & Ostermiller, 2017)

Let's recall agile phases with the following figure:

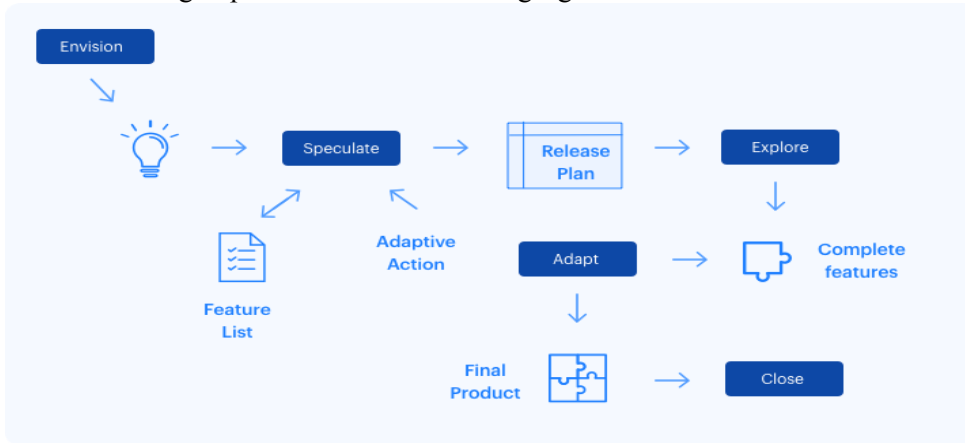


Figure 4.4-1 - Agile phases

It can therefore be said that the low score of variable 33 shows that the testing automation pitfall is threatening the exploration phase. Likewise, problems experienced in sprint synchronization (low score in variable 8) tell difficulties in the exploration phase.

On the other hand, high scores in team roles definition (variable 18) show good agile practice in the speculating phase. Also, variable 24 (efficient customer feedback exists) shows that the adapting phase is implemented in a good way.

Finally, to summarize the three aggregate variables dispersion, mean bars are used.

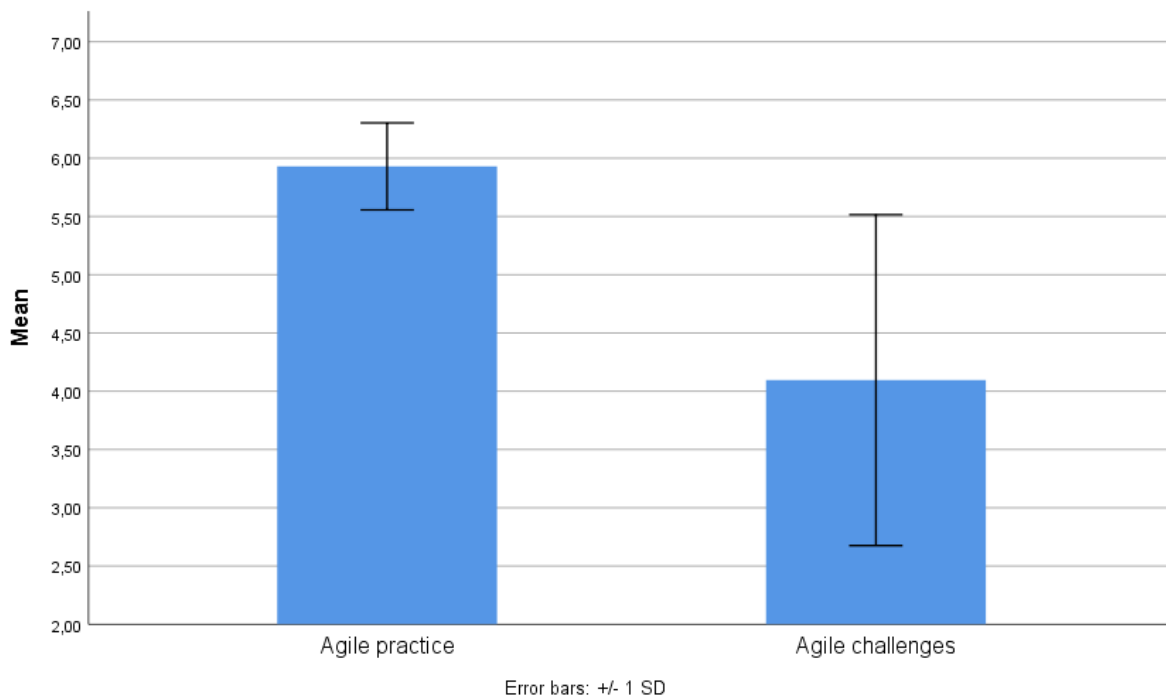


Figure 4.4-2 Aggregate variables mean bar analysis

a. Agile practice

The agile practice variable has a mean of approximately 6 and a SD of 0.37. This information can be interpreted in two ways. The first is that it indicates that all team members have responded generally the same way for all items. It therefore seems that all team members have the same level of information and understanding of the agile practice in place regarding this project. The second interpretation concerns the high score of 6. This value indicates generally that the project followed a good agile practice. Note that there was a lack of training for the team members (see section 4.5.1). It is interesting to know how such uniform results are obtained. Is this due to excellent management skills of the project manager? Is it due to chance? Or is it due to good implementation of agile standard guidelines? These are interesting questions that can be answered using inferential statistics.

b. Agile challenges

The aggregate agile challenges has bigger error bars. The mean is 4.57 the SD is 1.10. The fact that this data is more scattered may be due to different understandings of team members concerning challenges encountered in agile. In fact, the difficulty of challenges begins with identifying them. Other plenty of explanations may exist like that this due to lack of management skills, not respecting the agile values, gap in understanding in agile implementation methods.... (Layton & Ostermiller, 2017). Further studies are needed to find out the real causes in this situation.

Chapter V - Summary, conclusion and recommendations

5.1 Summary

A questionnaire designed to obtain data concerning the study objectives was distributed to the project team. Once the data gathering process terminated, the obtained information was classified under two categories: agile practice at Starcom and agile challenges. The gathered data was then analyzed using univariate descriptive statistics. Mean for central tendency and Standard deviation for data dispersion were the primary analysis tools.

5.2 Conclusion

This research paper studied the agile practice and challenges encountered in the implementation of CoreIT project. Dealing with the challenges in an effective way is imperative in order for the project product to stay within the triple constraints.

Here below are the major findings

- Good agile practices at Starcom
 - Communication between the client, PM, SM and team members
 - Feedbacks are included in iteration loops
 - Product backlog is maintained
 - Software requirements are clearly stated
- Agile challenges found
 - Communication between PO and PM
 - Software delivery delay
 - Cost overrun

- Scope change management
- PO priority management
- Lack of training
- Unsynchronised sprints
- PO multiplicity
- Automated testing

CoreIT project had faced multiple challenges. Despite that, it still has some agile practices in place (Abrahamsson, et al., 2004). After all, it produced a working software, which is one of the four agile values.

Agile methodology is a fantastic way for developing software. It is client-inclusive in its development, and when combined with an iterative nature, it provides the methodology powerful features that enables it to accommodate scope change. That doesn't make it challenge free. If these challenges are not dealt with, the project will still yield a result, but it will not be within the triple constraint (Cohn, 2009).

5.3 Recommendations

The identified challenges can be linked to many reasons discussed in chapter 4. Yet, due to scope limitation, this research described them in a univariate way. In order to have a deeper understanding of these elements, multivariate analysis is recommended for finding out causality and existence of relationship.

According to the analyzed data, Starcom should consider setting up trainings on agile for its developers (Agarwal & Rathod, 2006). More POs coordination should come from the project manager, and the POs should work on their communication with each other

(Cohn, 2009).

Starcom should follow standard agile guidelines in order to address the issues of sprint synchronization, automated testing and having multiple POs without hurting the flexibility offered by agile (Boehm & Turner, 2004). These are challenges that can be avoided by setting and following rules in spite of the use of agile.

Moreover, Starcom can use the features of traditional approaches to guide its work and control agile's flexibility in order to avoid problems such as scope creep resulting from poor scope management (Conforto & Amaral, 2016). Starcom can therefore consider hybrid (traditional and agile) methodology that can compensate the gaps encountered in implementation phase like unsynchronized sprints and communication problems between POs (Boehm & Turner, 2004).

References

- Abrahamsson, P., Salo, O., Ronkainen, J. & Warsta, J., 2004. *Agile software development methods: review and analysis*. Espoo: VTT publications.
- Agarwal, N. & Rathod, U., 2006. Defining "success" for software projects: an exploratory revelation. *International Journal of Project Management*, 24(4), pp. 358-370.
- Alzoubi, Y., Gill, A. & Al-Ani, A., 2016. Empirical studies of geographically distributed agile development communication challenges: A systematic review. *Information & Management*, 53(1), pp. 22-37.
- Batool, A., 2019. A Survey of Key Challenges of adopting agile in global software development: a case study with Malaysia perspective. *International Journal of Industrial and Manufacturing Engineering*, 13(10), pp. 1374-1377.
- Boehm, B. & Turner, R., 2004. *Balancing Agility and Discipline*. Boston: Addison-Wesley.
- Canty, D., 2015. *Agile for Project Managers*. New York: CRC Press LLC.
- Cho, J., 2008. Issues and challenges of agile software development with SCRUM. *Issues in information systems*, pp. 188-195.
- Cobb, C. G., 2011. *Making Sense of Agile Project Management : Balancing Control and Agility*. New Jersey: John Wiley & Sons, Inc.
- Cohn, M., 2009. *Succeeding with Agile Software development using scrum*. s.l.:Addison-Wesley.
- Conboy, K., Coyle, S., Wang, X. & Pikkarainen, M., 2011. People over process: Key challenges in agile development. *IEEE Software*, Volume 28, pp. 48-57.
- Conforto, E. & Amaral, D., 2016. Agile project management and state-gate model: a hybrid framework for technology basec companies. *Journal of Engineering and Technology management*.
- Dingsoyr, T. & Dyba, T., 2008. Empirical studies of agile software development: A systematic review. *Information and Software Technology*, Volume 50, pp. 833-859.
- Ghimire, D. & Charters, s., 2022. The impact of Agile Development Practices on Project Outcomes. *Software*, Volume 1, pp. 265-275.
- Goodpasture, J. C., 2010. *Project management the agile way making it work in the enterprise*. Florida: J. Ross Publishing, Inc..
- Kurup, D. & Sidhardhan, S. K., 2015. *Agile Project Management – Benefits and Challenges*, Florida: Unversity of South Florida.

- Layton, M. C. & Ostermiller, S. J., 2017. *Agile Project Management For Dummies*. New Jersey: Wiley.
- Lindvall, M., Basili, V., Boehm, B. & Costa, P., s.d. *Empirical Findings in Agile Methods*. North Carolina: s.n.
- Pallant, J., 2001. *SPSS survival manual: A step by step guide to data analysis using SPSS for Windows version 10*. Buckingham: Open University Press.
- PMI, 2017. *A guide to the project management body of knowledge (PMBOK guide)*. Pennsylvania: Project Management Institute, Inc..
- PMI, 2017. *Agile practice guide*. Pennsylvania: Project Management Institute, Inc..
- Regassa, Z., Bass, J. & Midekso, D., 2017. Agile Methods in Ethiopia: An Empirical Study. *Information and Communication Technologies for Development*, pp. 367-378.
- Safwan, M., Thavarajah, G., Vijayarajah, N. & Senduran, K., 2013. An empirical study of agile software methodologies: a Sri Lankan Perspective. *Scientific Research Journal SCIRJ*, 1(5), pp. 7-13.
- San Cristóbal, J. R. et al., 2018. Complexity and Project Management: A General Overview. *Complexity*, Volume 2018, p. 10.
- Scotcher, E., 2015. *Brilliant Agile Project Management A Practical Guide to Using Agile, Scrum and Kanban*. London: Pearson.
- Seymou, T. & Hussein, S., 2014. The History of Project Management. *International Journal of Management & Information Systems (IJMIS)*, 18(4), pp. 233 - 240.
- Sowaidan, A., 2016. *The impact of Agile project management on the effective delivery of innovative products*. Dubai: s.n.
- Streek, A., 2015. *Identifying Benefits and Challenges in the Agile methodologies in software development*. Bredfordshire: s.n.
- Totten, J., 2017. *Critical Success Factors for Agile Project Management in Non-Software Related Product development Teams*. Michigan: Dissertations.
- Vijayasathya, L. & Turk, D., 2008. Agile Software Development: A survey of early adopters. *Journal of Information Technology Management*, 19(2), pp. 1-8.
- Wei, X., Lakshmi, I. & Scott, S., 2022. Agile Software Development Vulnerabilities and Challenges: An Empirical Study. *AMCIS 2022 Proceedings*, Volume 15.
- Wysocki, R. K., 2011. *Effective Project Management: Traditional, Agile, Extreme*. 6th éd. Indianapolis: John Wiley & Sons, Inc.
- Wysocki, R. K., 2019. *Effective Project Management Traditional, Agile, Extreme, Hybrid*. 8th éd. Indianapolis: John Wiley & Sons, Inc..

Wysocki, R. K., 2019. *Effective Project Management Traditional, Agile, Extreme, Hybrid*. Indianapolis: John Wiley & Sons, Inc..

Annex 1 - Questionnaire

The designed questionnaire is intended to answer the research questions mentioned in the beginning of this study. Therefore, it will consist of two major themes: questions from 1 to 16 try to assess the company's agile practice, questions from 17 to 34 address the challenges encountered at Starcom. The format of questionnaire selected is one consisting of Likert scale so that it is easy to fill, quantifiable answers can be obtained and finally the respondents are presented with better choices to express themselves instead of a simple yes or no.

Here is a link to complete the questionnaire online:

<https://docs.google.com/forms/d/17y54JOoD->

[k9yJIGJ9VZfRvXfrVYLFoOlTQemX9ZOkv8/viewform](https://docs.google.com/forms/d/17y54JOoD-k9yJIGJ9VZfRvXfrVYLFoOlTQemX9ZOkv8/viewform)

Part 1 - Agile practice assessment at Starcom								
		Strongly agree	agree	somewhat agree	neutral	somewhat disagree	disagree	strongly disagree
1	Starcom uses agile method for all software development projects							
2	The team received training on agile methodology							
3	Team members have good coordination with scrum master							
4	Scrum masters have good communication							

	with Product Owners							
5	Product owners have good communication with project manager							
6	Every team uses the same product backlog							
7	Project backlog is continuously maintained							
8	The sprints of the different teams were synchronized							
9	The project manager has good communication with the client							
10	The software requirements were clearly stated							
11	You have achieved your goal							
12	The teams achieved their goal							
13	The product was delivered according to specifications							
14	The project was finished on time							
15	The project managed to stay within budget							
16	Agile is better than traditional for software development							
Part 2 – Encountered agile challenges								
17	The developers have poor programming skill							

18	Team roles were not clearly defined							
19	Team members have poor communication between them							
20	Team members have poor coordination with scrum master							
21	Having multiple product owners had a negative impact							
22	The project manager did not manage well product owners relation with project priorities							
23	The team motivation level is low							
24	Efficient customer feedback does not exist							
25	The fact that the development team was physically in the same location while developing did not help							
26	Upper management did not approve the use of agile							
27	Scope change was not managed successfully							
28	The software is intended to solve a problem observed							
29	The software company initiated the project							
30	Software development process uses agile approach							
31	The software development process has 6 stages: Concept - Inception -							

	Iteration / Construction - Release - Production - Retirement							
32	The iteration phase includes feedback							
33	The testing phase is automated							
34	Testing helped produce reliable software							