UNIVERSITY OF KWAZULU-NATAL

AN INTEGRATION OF TRADITIONAL PROJECT MANAGEMENT PRINCIPLES INTO AGILE SOFTWARE DEVELOPMENT METHODOLOGIES

By
Yonga Mapongwana
209501068

A dissertation submitted in fulfilment of the requirements for the degree of Master of Commerce in Information Systems & Technology

School of Management, IT and Governance
College of Law and Management Studies

Supervisor: Mr. Sanjay Ranjeeth

2016

DECLARATION

I , Yonga Mapongwana declare that

(i) The research reported in this dissertation/thesis, except where otherwise indicated, is my original research.

(ii) This dissertation/thesis has not been submitted for any degree or examination at any other university.

(iii) This dissertation/thesis does not contain other persons' data, pictures, graphs or other information, unless specifically acknowledged as being sourced from other persons.

(iv) This dissertation/thesis does not contain other persons' writing, unless specifically acknowledged as being sourced from other researchers. Where other written sources have been quoted, then:

a) their words have been re-written but the general information attributed to them has been referenced;
b) where their exact words have been used, their writing has been placed inside quotation marks, and referenced.

(v) Where I have reproduced a publication of which I am an author, co-author or editor, I have indicated in detail which part of the publication was actually written by myself alone and have fully referenced such publications.

(vi) This dissertation/thesis does not contain text, graphics or tables copied and pasted from the Internet, unless specifically acknowledged, and the source being detailed in the dissertation/thesis and in the References sections.

Signature: Y. Map

ACKNOWLEDGEMENTS

I am eternally grateful to my God and Saviour, Jesus Christ, for giving me the intellect, motivation, passion and ability to strive for better things.

My sincere thanks and appreciation goes to the following people who encouraged me to study further and/or who provided instructive and emotional support during the course of my research:

- My parents Vusumzi and Bulelwa Mapongwana for their special love, pride and confidence they had in me and for always inspiring me to reach for the stars.
- Mr. Sanjay Ranjeeth, my supervisor, for his guidance, availability and valuable feedback during this project.
- My fiancé, Thabang Qumza, for his ever-present love, encouragement, patience and support during my research.
- My friend, Abongile Bantom for the encouragement and constant support.
- The rest of my family and friends – thank you for your love and friendship

ABSTRACT

A large amount of money and effort has been invested by companies into establishing their project management (PM) environment and processes which follow the classical phased approach where requirements are defined upfront and fixed. However organisations also desire to react more quickly to new global challenges and to the changing business environment. These business requirements then result in the failure of these classical approaches to PM. There is therefore a need to enhance the current PM environment so that it is more adoptive to changes in the business environment. As a result of these changes in the business landscape agile software development methodologies (ASDM) have acquired a lot of popularity in the software development community. This popularity is being driven by their dynamic nature and the notion that user requirements do not have to be fully specified in the initial phases of the development process. This has resulted in the improvement in success levels of information systems (IS) projects that have made use of an ASDM. A shift to ASDM can increase the success rate of IS projects and mitigate some issues that are typical for heavy weight methods. Good examples can be found in the case studies (Balada, 2013; Raithatha, 2007), where agile methods were successfully used in software development projects of all sizes and complexity. However introducing ASDM for large and complex projects particularly in large enterprises can introduce a number of challenges (Thamhain, 2014). While agile principles foster great flexibility and agility in changing environments, they are very difficult to realize in larger projects that require more execution formality and discipline to deal with the specific complexities (Waardenburg & Vliet, 2013). In order to address these problems, the current study investigates the problem of integrating Traditional Project Management (TPM) techniques into the development of large scale IS projects in large enterprises with complex IT landscapes that make use of AM. This study followed a hybrid approach combining both quantitative and qualitative research methodologies. Data collection entailed semi-structured interviews and questionnaires. The sampling strategy that was used was purposive sampling. A phenomenographic approach was used to obtain an insight into the experience of software development (SD) by software practitioners who made use of ASDM. The qualitative data elicited from this phase of the study was analysed thematically to identify aspects of AM that had a pivotal influence on software practitioners' perspective on ASDM. A substantive component of this phenomenographic incursion was to establish whether there was some form of resonance between ASDM and PM or whether these methodologies were diametrically opposite to one another. The objective of the qualitative component of the study was to obtain sufficient information to enable the development of a model for SD that integrated the principles of PM into ASDM. This phase of the study was followed-up by a quantitative phase that was underpinned by the Unified Theory of Acceptance and Use of Technology (UTAUT) in order to ascertain software practitioners' acceptance of the proposed model (referred to as the Agile-Project Management Model (APMM))

The results of the UTAUT-based acceptance test indicate that the proposed APMM received a high acceptance rate by the software practitioners who constituted the main subjects of the current study.

Table of Contents

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

AD	Agile Development
AM	Agile Methodologies
APM	Agile Project Management
ASD	Agile Software Development
ASDM	Agile Software Development Methodologies
CMMI	Capability Maturity Model Integration
EV	Earned Value
IS	Information Systems
ISO	International Organization for Standardization
IT	Information Technology
NASA	National Aeronautics and Space Administration
PERT	Program Evaluation Review Technique
PM	Project Management
QA	Quality Assurance
RM	Risk Management
SD	Software Development
SDM	Software Development Methodologies
SP	Software Practitioners

TAM	Technology Acceptance Model
TPM	Traditional Project Management
TSDM	Traditional Software Development Methodology
UTAUT	Unified Theory of Acceptance and Use of Technology
WBS	Work Breakdown Structure
XP	Extreme Programming

1.0 INTRODUCTION

1.1 Introduction

Agile software development methodologies (ASDM) have acquired a lot of popularity in the software development (SD) community. This phenomenon may be attributed to the failure of traditional software development methodologies (TSDM) to deliver quality software that meets user requirements that are volatile and difficult to predict. The quest to find an alternative to TSDM has resulted in the propagation of the idea that software development methodologies (SDM) need to be dynamic and commensurate with the notion that user requirements do not have to be fully specified in the initial phases of the development process. The methodology needs to be reactive so that it may be reconfigured to accommodate changing requirements. These characteristics do however, tend to impart an amorphous underpinning to the methodology thereby making it difficult to define in a succinct and accurate manner. This dilemma is somewhat resolved by the suggestion from Kennaley (2010, p. 34) that agile software development (ASD) is an:

> *"…iterative and incremental (evolutionary) approach to software development which is performed in a highly collaborative manner by self-organizing teams within an effective governance framework with "just enough" ceremony that produces high quality software in a cost effective and timely manner which meets the changing needs of its stakeholders."*

Aligned to the Kennaley (2010) perspective on ASD, GatherSpaceTeam (2015, p. 11) provide a historical and a functional perspective by suggesting that:

> *"Agile Software Development is a concept, a philosophy, and a methodology which evolved in the 90's as an answer to the long growing frustrations of the waterfall SDLC concepts. The term promotes an iterative approach to software development using shorter and lightweight development cycles and some different deliverables."*

These perspectives were supplemented by Karlstro¨m & Runeson (2006) who added that ASDM offers a practical approach to SD, that is flexible and has a strong focus on the people involved rather than the process that needs to be followed. The priority attached to the people involved in the SD process is highlighted in the agile values (see Table 1 below). This was established and endorsed as the pioneering document that guided the implementation of ASDM.

Table 1 : Agile Values (Beck *et al.*, 2001)

1. Individuals and interactions over processes and tools
2. Working software over comprehensive documentation
3. Customer collaboration over contract negotiation
4. Responding to change over following a plan

This realignment by the software engineering community to embrace a more dynamic approach to SD as espoused by ASDM has resulted in reports of discernible software project success (documented in the CHAOS[1] report). The

[1] A study from the Standish Group International which was inspired by the large amounts of money lost in failed software development projects. The focus of the research was to

empirical evidence attesting to the success of ASDM has also been accompanied by a positive, analytical response towards the methodology from the academic community. According to Khalid *et al.* (2014) some of the benefits that are attained by making use of ASDM include:

- Improvements in project delivery
- ASDM enables a minimum viable product to be built and delivered in the least possible time. This increases customer satisfaction because the first delivery reaches the customer with the highest prioritised features in a short duration of time
- Responsiveness towards changing market needs
- The change in technology advancements and enhancements is very fast paced. As a result customer needs are also changing in line with these changes in the external environment. Thus a very important need in most businesses is to be able to cope with these changes efficiently. ASDM are customer-oriented and focused on embracing changing customer requirements.
- Risks are identified at an earlier stage
- The adaptive approach that ASDM take enables the team to tackle problems as they arise. Although problems are not foreseen at the inception of the project, the team is prepared to address any risks that may arise. Due to involvement of customers and the feedback in each of the stages, risks can be identified at earlier stages. In TSDM the team may realise prior to the final release that there is a major defect with the product so ASDM overcomes all these risks during product development.

identify the scope of software projects, causes of project failure and key factors to reducing project failures

Although ASDM has achieved a lot of success, there are “traditionalists” that advocate against the use of ASDM. They still believe in extensive planning, organized processes and laborious reuse to make development a proficient and conventional activity that progressively develops into perfection (Papatheocharous & Andreou, 2013). According to Murphy *et al.* (2013) the reason these “traditionalists” are hesitant with adopting Agile Methodologies (AM) is the lack of scalability of ASDM to large projects. They believe the lack of scalability is due to the weak architecture, lack of upfront design and documentation, and lack of risk management (RM) which are necessary for large projects and these are delivered by traditional project management (TPM). However good examples can be found in the case studies (Balada, 2013; Raithatha, 2007), where agile methods were successfully used in large complex software development projects.

Large projects are not exempt from the problems that are encountered by small and medium projects. However, the complexity intrinsic to large projects by virtue of the amount of effort, time and cost resources consumed by such projects make the risk of failure greater (Lee & Yong, 2013). This risk is somewhat mitigated by AM in the manner in which it addresses issues such as a changing operational environment, ambiguous user requirements, and time pressure (Lee & Yong, 2013). The dictates of proper PM practice are however not explicitly included in AM, thereby compromising and severely restricting the scalability of AM to handle the complexities intrinsic to large scale SD (Turk *et al.*, 2014). Thamhain (2014) opposes this view based on the results of a three-year field study into the practices of agile project management (APM) at 37 technology-intensive companies. The study revealed that agile principles are applicable to most projects independent of their nature, size, or IT-orientation, improving resource effectiveness, project execution time and overall project success. The study also shows that for large and highly complex projects, the agile methodology must be modified to fit the organisational processes and cultures of the enterprise. Other good examples that support this view can be found in studies (Bass, 2013; Dikert *et al.*, 2016).

One of the issues that impedes the acquisition of knowledge in the software engineering field is the lack of access to sources of empirical data. This dilemma has been somewhat resolved in the context of the current study because of an opportunity that was presented to the researcher by one of the big four banks in South Africa where there has been an intervention based on AM to improve the development of software. The bank did however make a request of the researcher not to reveal the name of the banking institution in the study. In order to abide by this request, the bank will be referred to as Bank A in the current study.

It should be noted that Bank A is one of the organizations in South Africa that have embraced ASDM as the de facto methodology to underpin all SD efforts. They have recently begun the transition from TSDM to ASDM. However the size and complexity of the projects that are developed at Bank A require extensive PM. This is to enable them to manage the consumption of resources as well as to ensure that the development of these complex systems is always on schedule. This is achieved by ensuring that there is extensive documentation of the system development effort in order to ensure accountability by the development team. However, the accountability that is usually guaranteed by virtue of strong PM may be compromised as part of the transition to ASDM. In order to manage this situation, Bank A has begun to adapt ASDM so that aspects of PM are included into the SD process. This is however done on an ad hoc basis with no formal integration of PM into AM. Therefore this study entails an exploration of the possibility of adapting AM accordingly so that it integrates well with PM principles. A proposal will be made on how to integrate AM with PM techniques in order to mitigate the risks associated with developing software in a completely agile demeanour. One of these risks imposed by ASDM is the tactic of using self-managed teams. These teams work with considerable freedom and autonomy and in their attempts to self-manage and develop their own work routine, they select and use their own PM and testing tools. According to Elbanna (2014) in many organizations this has led to the use of several different ways and tools to manage agile projects. He believes that IT managers find it difficult to merge teams and manage workloads under these conditions. A possible reason for this phenomenon is that in large companies, the project approval process requires up-front requirements to be reviewed by a board or committee. The board or committee conducts a strategic review for new IT initiatives and approves these initiatives if they are in alignment with the business strategy. The complication that arises out of this corporate practice is that in an agile environment, upfront documentation is either very sparse or not present at all, thereby making it very difficult for management to provide financial support for a project that has a kind of amorphous presence at the initial stages of development. The third risk highlighted by Abdelaziz *et al.* (2015) is the constant emergence and evolution of requirements. Flexibility to change course as needed and to ensure delivery of the right product is often seen as the definition of agile. However this presents the potential for scope creep, which can create the risk of projects that do not converge to an agreed date of completion. Abdelaziz *et al.* (2015) also highlights the risk of the lack of predictability in purely agile approaches at the start of the project and during the project. This makes it difficult to define a business case for the project and even harder to negotiate fixed-price projects.

1.2 Background to the study

As mentioned above, Bank A is one of the big four banks in South Africa. It operates in multiple countries around the world. It has been in existence for more than 100 years and even with that rich heritage there is still a need to respond to industry trends in order to remain relevant to its customers. Bank A has realised this and has since reviewed its strategy to being a customer-centric and innovative institution (BusinessTech, 2015). Innovation is critical in the banking landscape which is characterised by rapid change, an area that has been prioritised by Bank

A. Bank A also regards technology as one of the key enablers of innovation. According to PWC's 2013 banking survey Bank A was listed amongst the banks that expected to invest significantly in Information Technology (IT), projecting R3-5 billion. Ntuthuko (2013) makes reference to a survey conducted by the Innovation Agency on Innovation in the South African Banking Industry. According to this survey Bank A was rated the third most innovative bank, voted for by less than 8% of the customers surveyed despite the large amounts of money invested in IT. Some of the factors that were cited as reasons for this outcome were that most individuals associate innovation with technological advancement, industry leadership, customer centricity and ease of banking.

Industry leadership has been a challenge for Bank A. Using a bit of deductive logic, a possible cause for Banks A's inability to achieve competitive advantage in the banking sector is that Bank A is not achieving optimal value from its investment in the development of software systems. The adherence to a traditional, prescriptive approach to SD has compromised Bank A's ability to react in a dynamic way to new developments in the banking sector. This lack of competitiveness has been quite frustrating for many of the SD professionals in the organization. Many of these SD professionals expressed unhappiness with an approach to SD that was too bureaucratic and slowed down time to market, leading customers to perceive Bank A as a follower as opposed to being a leader in the industry. This is commensurate with the beliefs held by Mohammed *et al.* (2010) that traditional software processes were too prescriptive, thereby impeding the prospect of obtaining competitive advantage even if there was an investment in new technology. Traditional software process models such as the waterfall model for SD where the pedantic subservience to a sequential approach makes it difficult to provide early releases of working software. By engaging in such a methodology, the software that is produced may not be in line with the requirements of the customers at the time of release. It also provides very little flexibility in terms of adjusting scope and makes it difficult to respond to changes. Boehm (2002) suggests that the ineptitude of traditional software process methodology has been a catalyst for a “revolution” in SD where there has been a preference for software process models that are radically different. These process models have been engineered around a philosophy that espouses flexibility and a sensitivity to the people involved in the development process rather than the process of development. This philosophy is embedded into a guiding document named the Agile Manifesto (refer to Appendix E). Similarly Bank A responded to these challenges that plague the SD process and have embraced the implementation of an agile approach to SD. ASDM is centred on an iterative development approach that enables end users to interact with the evolving system and contribute in a dynamic manner to the system’s evolution. The benefits of this iterative, adaptive approach to SD as opposed to the rigid and rather prescriptive Waterfall approach are illustrated in Figure 1 below.

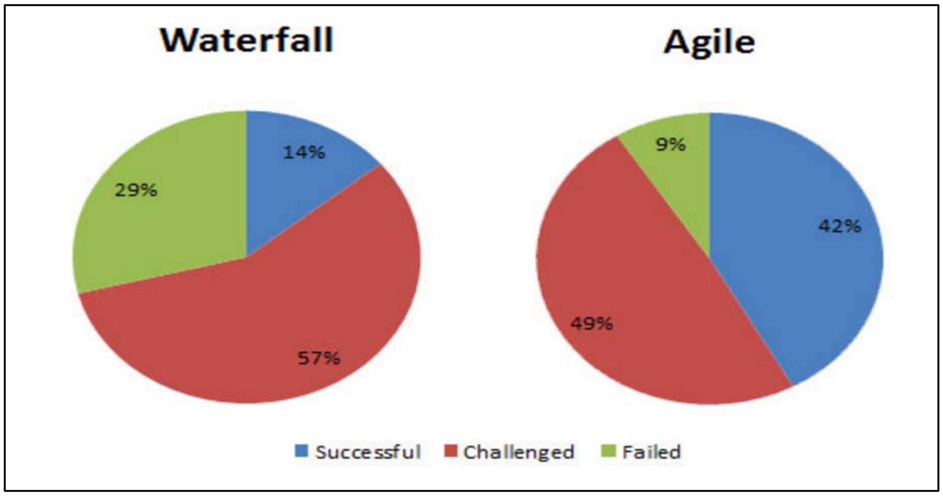


Figure 1: IS Project Success: Agile vs Waterfall Software Development Methodology(Manifesto, 2012)

The 2013 CHAOS qualifies the illustration in Figure 1 by claiming that the success of ASDM is confined to small scale IS projects. The empirical evidence attesting to the success of ASDM for small scale IS projects is commensurate with the prediction by Turk *et al.* (2014) who recognised that the operational aspects of

ASDM are aligned to the needs of small scale IS projects. However, ASDM may not scale up to the requirements of large scale IS projects. The rationale behind this claim is that the complexity of large projects and their stringent schedules and budgets do not resonate well with the dictates of ASDM. This assertion is corroborated in the claim by Hass (2010) that the complexities presented by large projects demand that particular attention be dedicated to planning the project, developing and delivering the solution, selecting the team members and maintaining a high-performing team over the long run. All of these issues fall under the ambit of PM. Conversely AM is reputed to have minimum regard for PM principles which are required in large projects .This is evident in their lack of adequate documentation, weak architecture and lack of RM (Qureshi, 2012). These factors impede the scalability of ASDM to handle the requirements of large scale IS projects (Lan *et al.*, 2004). The challenges of scaling AM to large enterprise development programmes have received attention from numerous practitioners (Ambler & Lines, 2012; Larman & Vodde, 2008; Leffingwell, 2007). Kruchten (2013) believes that AM can be stretched with variable success outside of the context in which they have been created. This view is supported by the systematic literature review done by Dikert *et al.* (2016) which identified 52 papers presenting 42 industry cases describing successful large-scale agile transformations. Additionally, Yahoo a large enterprise with a $32 billion market cap and is the largest enterprise dealing mainly with large projects successfully adopted AM and demonstrated good results (Salo & Abrahamsson, 2008). This indicates that while introducing agile in large and complex projects presents numerous challenges, it is not impossible.

The complexity and size of projects that Bank A undertakes creates a situation where Bank A becomes quite prone to the challenges presented by ASDM in large scale IS projects. According to CIO (2014) a large project is one that requires collaboration of two or more departments, the work effort in hours is greater than 250 hours, the impact is medium to large, costs more than R200 000 and the project sponsor is typically a Director/CIO/Committee. As a recourse to the shortcomings of ASDM, Bank A has adapted ASDM in order to incorporate a significant presence of PM principles into ASDM. However there has been no formal framework/ideology underpinning the integration of PM into ASDM at Bank A. The current study undertakes to provide a formal underpinning to the integration of PM principles into ASDM. As such, it is predicted that the outcome of this study will make a contribution not only to Bank A but also to the wider practitioner and academic communities of software engineering.

1.3 Problem Statement

1.3.1 Background to the Problem Statement

In today's dynamic business environments, large projects are also faced with the same challenges that are typically imposed on small and medium projects. Some of these challenges are ambiguous user requirements, a constantly changing environment and stringent project timelines (Lee & Yong, 2013). In this manner large projects would likewise benefit from making use of ASDM. It appears that a number of organizations are cautious when it comes to the application of ASDM in large and complex projects (Gandomani *et al.*, 2013). According to Iamandi *et al.* (2015) this is as a result of the approach that these methodologies take in addressing PM. APM is based on a philosophy whereby traditional elements of systems development such as upfront design and documentation, a robust architecture and RM are not given much priority (Iamandi *et al.*, 2015). However, these activities are all necessary for the success of large and complex projects. Therefore attributes such as RM cannot be omitted when dealing with projects of this nature (Dybå *et al.*, 2014). The CHAOS study 2013 states that large projects have twice the chance of being late, over budget, and missing critical features than their smaller project counterparts. The time constraint is especially of concern because in order to mitigate the risk of being late, other crucial aspects such as

RM and a structured PM approach may be sacrificed. Furthermore, SD organizations are required to deliver software rapidly while ensuring high levels of quality assurance (QA) and this is catered for in the TPM approach (Highsmith, 2013). Moreover, in large and complex projects, efficiency is required to meet the demands and to prevent time and resource consumption that could lead to project failure. Joslin & Müller (2015) suggest that efficiency requires the presence of PM activities to assist with appropriate implementation of SD tasks. PM is thus a support function that provides the backbone for efficient SD. There are however, detractors to this viewpoint, such as O' Sheedy (2012), who suggests that complete subservience to the principles of TPM does not resonate well with the dynamic, fast–paced and low budget projects that are becoming prominent in the current domain of SD projects. The underlying message that is being transmitted in this discourse is that TPM is required in a rather satisficing manner than an optimal one.

The preceding discussion has elucidated the shortcomings of AM with respect to PM, thereby magnifying the risk of failure for large complex projects. Nonetheless AM is still needed to address the problem of changing environments, unclear user requirements and time constraints. It is therefore imperative for AM to be adapted to incorporate the PM principles required for the management of large and complex projects. In so doing Bank A will be exploiting the benefits of the agility delivered by AM which will assist them in the fast delivery of software while accommodating changes and also reduces the risk of poor management of the software process.

1.3.2 Main and Sub Research Questions

Main Research Question:

How can the principles of PM be integrated into ASDM, at Bank A?

Research Question 1:

What are the PM related challenges of implementing ASDM at Bank A?

Research Question 2:

How can the PM related challenges of implementing ASDM at Bank A, be mitigated?

Research Question 3:

What is the acceptance by software practitioners (SP) at Bank A, of a framework that guides the integration of PM principles into ASDM?

1.4 Rationale of the study

According to Cooper & Sommer (2016); Rubin (2012) Extreme Programming (XP) and Scrum are the two most popular AMs. However, recently Scrum's popularity has exceeded all the other AMs by some degree (Cooper & Sommer, 2016). The main reason for this is that Scrum has elements of PM embedded within it. There is a Sprint review meeting, a product backlog meeting and daily sprint meetings where there is a potential for significant PM intervention (Cooper & Sommer, 2016; Schwalbe, 2015). However, there has been no formal integration of TPM into Scrum as yet. Fitzgerald *et al.* (2006) argue that XP and Scrum complement each other well, owing to the fact that XP provides support for the technical aspects in the project whereas Scrum provides support for PM aspects such as project planning and tracking. However not much is known about the tailoring and integration of these two aspects within the South African banking industry. The development of a framework that integrates ASDM with PM could herald a starting point for a critical analysis that will reveal the challenges that such an integration will create, thereby obviating a current gap in the body of knowledge with regards to the compatibility of ASDM and PM.

1.5 Objectives of the study

The primary objective of this study is to explore the possibility of coming up with a framework/set of recommendations to inform the integration of PM into AM. The research objectives are as follows:

- To ascertain the PM related challenge(s) of implementing ASDM.
- To establish how the PM challenges of implementing ASDM can be mitigated.
- To develop a framework that integrates PM related principles into ASDM.
- To determine the acceptance of the proposed framework by SP at Bank A

1.6 Dissertation Structure

This study is arranged into 6 chapters. Figure 2 illustrates the structure of this dissertation. At the beginning of each chapter, a dissertation map will indicate stages in the dissertation and what will discussed in the chapter.

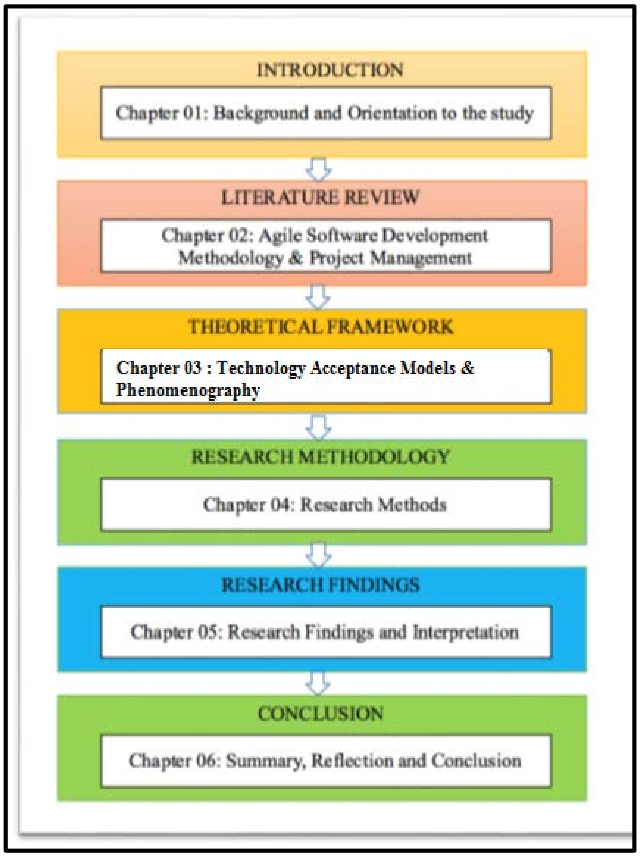

INTRODUCTION
Chapter 01: Background and Orientation to the study
LITERATURE REVIEW
Chapter 02: Agile Software Development Methodology & Project Management
THEORETICAL FRAMEWORK
Chapter 03 : Technology Acceptance Models & Phenomenography
RESEARCH METHODOLOGY
Chapter 04: Research Methods
RESEARCH FINDINGS
Chapter 05: Research Findings and Interpretation
CONCLUSION
Chapter 06: Summary, Reflection and Conclusion

Figure 2: Dissertation Layout

- Chapter One: Introduction and overview of the study. This chapter focuses on introducing the research topic and providing an overview of the study. The chapter also provides an outline of the sections that are covered in the rest of the paper as indicated below.
- Chapter Two: Literature Review. This chapter consists of the literature review detailing the main challenges that have been experienced in implementing ASDM for large projects, how PM is currently being done for large projects and what PM challenges exist in IS projects making use of AM.

- Chapter Three: Theoretical Framework. This chapter discusses the theoretical frameworks used for this study. The Unified Theory of Acceptance and Use of Technology (UTAUT) and Phenomenography will be discussed in detail.
- Chapter Four: Research Methodology. This chapter discusses the research design adopted for the study.
- Chapter Five: Analysis and Findings. This chapter includes analysis and findings of the study
- Chapter Six: Conclusion and Recommendations. This chapter includes concluding discussions and recommendations for further research.

1.7 Summary of the chapter

The current chapter introduced the challenges presented by ASDM to large projects. This is mainly the lack of upfront design and documentation, a robust architecture and RM. These activities are all necessary for the success of large and complex projects. Projects are an important part of a lot of organisations today as they make up the strategic management of businesses in building and maintaining competitive advantages. The high likelihood of large software projects failing to be delivered on time, within scope, within the allocated budget and meeting user requirements calls for more research to be done on what can be done to improve the situation. In an economic age where businesses are required to do more with less, a PM method that can help with successful implementation could prove beneficial to many organisations. In the current study the benefits and challenges of using ASDM will be explored with the intention of developing a framework that integrates principles of both these SD strategies. There will be a significant focus on areas of compatibility between ASDM and PM so that it may be used as a catalyst for the effort on integration. The background information, problem statement, objectives and overview of the study were presented in this chapter. The next chapter will present a detailed review of the literature pertaining to ASDM and PM.

2.0 LITERATURE REVIEW

"Agile Methodologies act in response to the limitations of traditional software development methodologies"

2.1 Introduction

Agile development (AD) has had a huge impact on the development of software universally (Dybå & Dingsøyr, 2008). The development and adoption of Agile Methodologies (AM) can be attributed to addressing the challenges that come with the unpredictable Software Development (SD) domain, emphasising the significance of skilful people and their interactions to SD (Dybå & Dingsøyr, 2008). By embracing this principle, AM may be seen as a response to the limitations of the Traditional Software Development Methodologies (TSDM). TSDM have been criticised for failing to accommodate changing user requirements, imposed by the dynamic and progressive nature of today's business environment (Ferreira & Cohen, 2008). While some organizations have succeeded in acquiring benefits from AM, many others face difficulties in transitioning from TSDM to AM (Dybå & Dingsøyr, 2008). This chapter will explore in greater detail some of the challenges that make this transition particularly difficult. Furthermore this chapter will provide a detailed overview of Project Management (PM) as a discipline, Agile Software Development Methodologies (ASDM), Challenges of adopting ASDM as well as PM in ASDM.

2.2 Traditional Project Management

The PMBOK standard (PMI, 2008, p. 22) describes a project primarily as :

> *"A temporary endeavor undertaken to create a unique product, service, or result. The temporary nature of projects indicates a definite beginning and end. The end is reached when the*

> *project's objectives have been achieved or when the project is terminated because its objectives will not or cannot be met, or when the need for the project no longer exists..."*

The PMBOK standard (PMI, 2008, p. 22) further explains PM as:

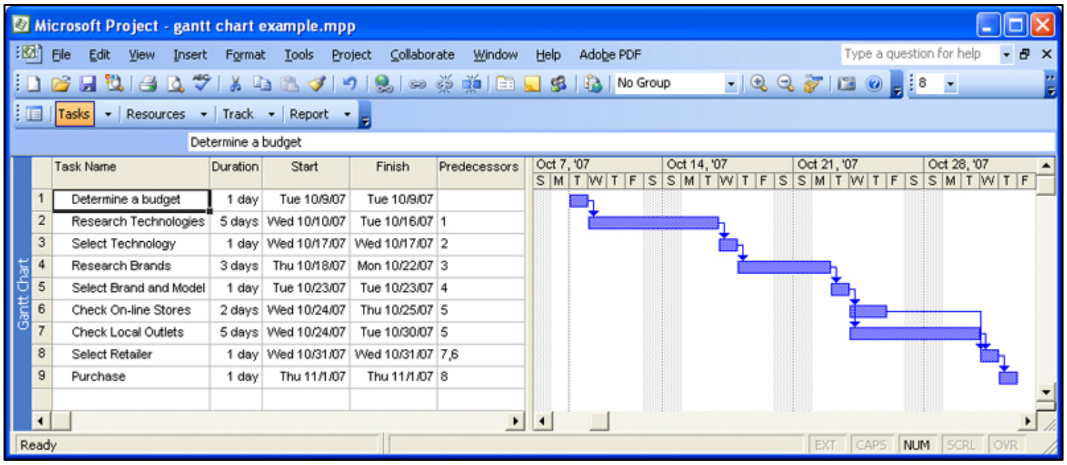
> *"The application of knowledge, skills, tools, and techniques to project activities to meet the project requirements."*

PM is not an idea that has been established recently. In the past, there have been a number of widespread construction projects, as well as undertakings such as the mounting of the pyramids, the construction of Stonehenge, and the establishment of highways and bridges by the Romans (Kwak & Anbari, 2008). So projects have been in existence for a considerable amount of time and people have been participating in them even before the formalisation of PM. From around the 15th century, major construction projects were established to engineering features that warranted successful completion of these projects. Engineering was being recognised as a science and a number of the advances in this discipline were as a result of the continuous research taking place (Morris, 1994).

This subsection will examine PM as an independent variable in sufficient detail. A brief discussion of the crucial aspects of PM that have been established over the years will be done. The first step in this chapter is to look at the history of PM and how it came into being and then a review of the pertinent literature, establishing a foundation upon which the research will be built.

2.2.1 Theory on Project Management

One of the tools that was used for the management of engineering jobs that was originally introduced is the Gantt chart. This tool was introduced by Henry Gantt and this event was inspired by the idea that a large number of business people were consumed by the generation of profits, instead of the generation of required products. He emphasised that enterprise resources may be better spent, increasing the revenue of factories and industries to improve supply to the community. So by presenting this chart method, Gantt's fundamental aim was to encourage employees and managers by empowering them to identify where production resources were being under-utilised (Clark, 1923). Up to today the Gantt chart is being used for project scheduling. It displays the time necessary for specific tasks and the order in which these tasks need to occur. It also shows the dependencies between the individual tasks and the teams performing the tasks (Clark, 1923). Unfortunately the Gantt chart hasn't retained its popularity. Some critics argue that they pull a project managers focus away from the project and onto perfecting the graphs (Seymour & Hussein, 2014). These critics believe that projects don't fail from of a lack of charts, graphs, reports or statistics, they fail from a lack of communication. They also argue that real-world projects don't run like an organized, Gantt-charted project plan (Seymour & Hussein, 2014). On the other hand Gupta *et al.* (2016) argue that while it's true that actual project milestones don't always happen exactly when they've been scheduled on Gantt charts, the argument that the charts are therefore useless is short-sighted. They believe the chart itself is not responsible for driving the project schedule – it's just a window into a series of tasks, dependencies and deadlines created by the project manager. If the Gantt chart is consistently out of alignment with the reality of a project, the project schedule itself needs to be re-examined or the chart is not being built with reliable data. Figure 3 below illustrates a basic example of a Gantt chart.



Determine a budget

	Task Name	Duration	Start	Finish	Predecessors
1	Determine a budget	1 day	Tue 10/9/07	Tue 10/9/07	
2	Research Technologies	5 days	Wed 10/10/07	Tue 10/16/07	1
3	Select Technology	1 day	Wed 10/17/07	Wed 10/17/07	2
4	Research Brands	3 days	Thu 10/18/07	Mon 10/22/07	3
5	Select Brand and Model	1 day	Tue 10/23/07	Tue 10/23/07	4
6	Check On-line Stores	2 days	Wed 10/24/07	Thu 10/25/07	5
7	Check Local Outlets	5 days	Wed 10/24/07	Tue 10/30/07	5
8	Select Retailer	1 day	Wed 10/31/07	Wed 10/31/07	7,6
9	Purchase	1 day	Thu 11/1/07	Thu 11/1/07	8

Figure 3: Example of a basic Gantt chart (PENNSTATE, 2008)

The second PM tool that materialised was the Program Evaluation Review Technique (PERT). This was introduced because of the cold war between the United States (US) and the Soviet Union with both countries going to war for nuclear control (Oberlender, 1993). This battle heightened military research and development and it was during this time that PM tools were enhanced. PERT was originally established for the management of missile production (Blanchard *et al.*, 1990). When using PERT, the project manager has to give estimates of the most optimistic timeline, the most pessimistic timeline, and the most probable timeline. A calculation of the mean of the estimations that were supplied is then given together with the calculated variance of every task (Klastorin, 2003). This tool allows the project manager to make an educated guess about the time it will take to complete each phase of the project and simplifies the process of tracking the project, offering a valuable tool for sequencing tasks. The tool also makes it possible to identify the critical path in the project, enabling project managers to identify the series of tasks that will require the greatest amount of time to achieve (Morris 1994). The chance for project managers to be able to include uncertainty whilst approximating the timelines for tasks in a project is one of the fundamental characteristics of this tool. It was a valuable shift in managing projects that contain uncertainty. Figure 4 below depicts the classic PERT model. The next sections discusses the role of the project manager in detail.

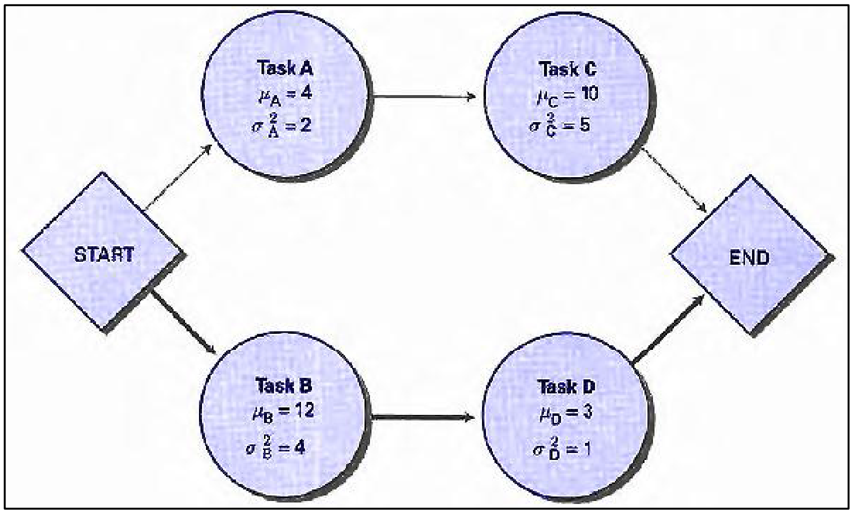


Figure 4: An example of the PERT model (O'Sheedy, 2012)

2.2.2 Project Manager

In the late 1950s organisations began appreciating the need for a person that would assume the responsibility for governing projects. Project managers were assigned the role of ensuring that a project was completed successfully (Stretton, 2007). The resulting responsibilities of a project manager included forming the project requirements; establishing attainable and unambiguous goals; and adjusting the requirements, plans, and approach to accommodate the different interests and expectations of the various participants. Fundamentally, the project manager is liable for providing direction for the project from its contributions, through its effort, to delivery of its outputs (Bredillet *et al.*, 2015). According to Bredillet *et al.* (2015) in order to accomplish these intricate duties, the project manager has to perform several roles including that of a manager, a front-runner, enabler, financier, arbitrator, planner and liaison. The project manager is expected to direct teams to operate cross-functionally for a common purpose. This has to be done while making sure that there is continuity and cohesiveness as the project progresses through the different processes and phases. PMI (2004, p. 41) states that a project manager *"also functions as a facilitator to inspire effective communication and coordination between design, procurement and construction activities."*

With the current evolution of the SD landscape and the introduction of ASDM, more and more people are wondering if these new methodologies have made the role of a project manager redundant (Coram & Bohner, 2005). According to Landry & McDaniel (2015) Scrum which is often classified agile's PM method defines only three roles: Scrum master, development team and Product owner. Many of the responsibilities of the traditional Project Manager are covered by these other roles. So, they pose the question that given that everything is covered by these roles, is there any value in assigning a project manager in the team? They believe that the answer to this question varies from project to project depending on the scale and complexity of the project and the wider environment in which it exists. Thamhain (2013) supports this as he states that small, co-located Scrum team delivering a software product with manageable risks and a very simple project environment do not need a project manager assigned. In such cases, PM responsibilities can be managed within the roles Scrum provides. He suggests that in this case it's best to follow a core agile principle: where responsibility can be devolved into the team then it should be. There is no value in having a Project Manager on the team just for the sake of it. However, the projects which are on a higher order of complexity across a number of factors, place different demands on the team. As each of these factors increase in relevance, the argument for the addition of a Project Manager becomes much stronger in such cases (Thamhain, 2013). On the other hand Thamhain (2013) believes that the role of project manager isn't going to vanish with the increase in popularity of AM. What is happening instead, is a gradual yet decisive whittling away of functions traditionally associated with project managers. The end-result of this process will be a leaner, more agile role that can fit in agile environments and help execute important tasks that complement those of the scrum master and are directed on a higher, team-wide and department-wide level.

The following development to the PM Toolset came about in 1961, after Russian Yuri Gagarin completed circling the globe in Vostok 1. Steered by this event, the US government assigned the National Aeronautics and Space Administration (NASA) the task of landing a man safely on the Moon by the end of the decade (Chertok, 2010). These undertakings were very expensive, and even though the aim of the project was clear, project managers were confronted by uncertainty with respect to the technical chunks of the project (Chertok, 2010). To moderate this uncertainty the US Department of Defence and NASA together distributed new task and cost control processes. These encompassed such tools as Earned Value (EV), Work Breakdown Structures (WBS), improved procurement processes as well as schedule managing (Morris 1994). The PERT technique was also adapted from being an entirely time-based estimating tool to an enhanced cost/benefit analysis technique. At this time, management science research advanced significantly, with the establishment of tools such as Value Engineering, Material Requirements Planning, Quality Assurance, and Cost Analysis to name but a few. A number of these tools have been developed over the years and are still being used today, and have since advanced to becoming standard practices in PM (Morris, 1994). The next subsection examines the processes that make up the PM practice.

2.2.3 Project Management Processes

PM is made up of process groups (PMI, 2004). The PM process groups are illustrated in Figure 5 and these are initiating, planning, executing, monitoring and controlling, and closing.

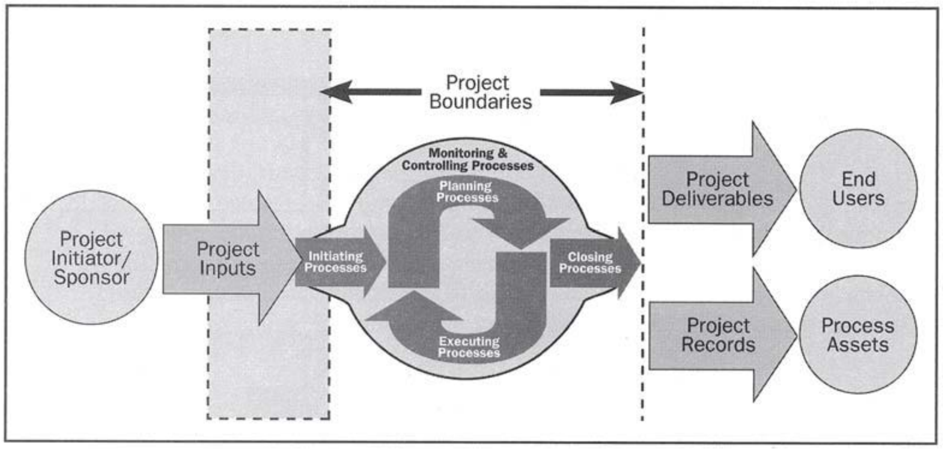


Figure 5: Project Management process groups (PMI, 2004)

Table 2 below, provides a description of each of these process groups.

Table 2: Project Management process groups (Schwalbe, 2015)

Process Group	Description
Initiating	Shapes and endorses the project
Planning	Defines the objectives of the project and sets up the order of events that are crucial in achieving the goals and scope that the project was intended to tackle
Executing	Brings together all resources required to accomplish the project's PM plan including people
Monitoring and Controlling	Evaluates and examines the progress of the project to

	identify inconsistencies from what is drawn in the PM plan to allow remedial actions to be taken when necessary to achieve project objectives
Closing	Makes the approval of the end product official and closes the project or a project phase in an organised manner

The process groups are dependent on each other and are interrelated as depicted on Figure 6.

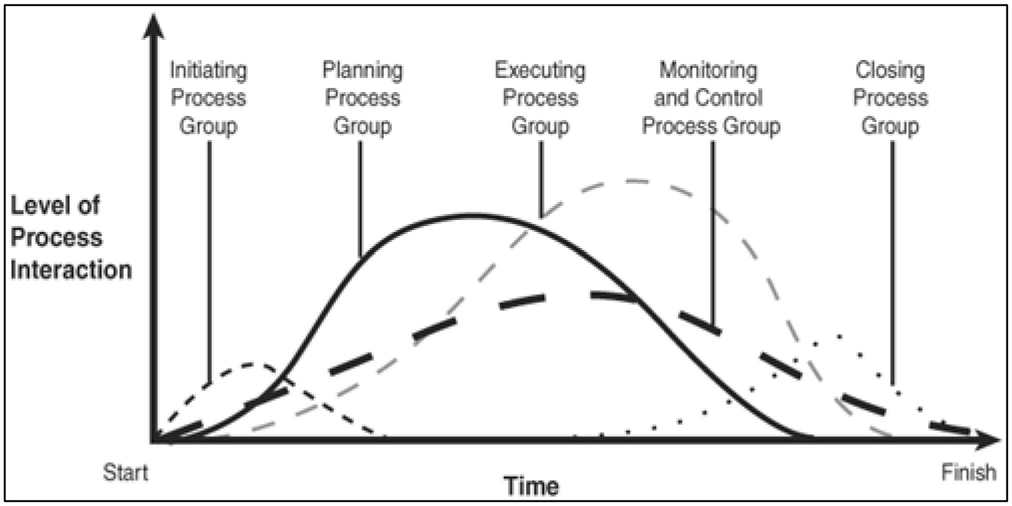


Figure 6: Levels of interaction for project management process groups(Schwalbe, 2015)

Figure 6 portrays the comparative extent and complexity of the interrelationship between these process groups (Schwalbe, 2015). From Figure 6 it can be ascertained that the scope of PM is extensive as it begins with initiating and advances through to closing, these steps correspond with the start and end of a particular project (Schwalbe, 2015). Monitoring and controlling occur during the course of the project and have a comparatively similar scope to that of executing. This illustrates the fact that a project is an impermanent effort and the consequence of scheduling the deliverable. Closing follows shortly after the conclusion of initiating. Planning and monitoring and controlling have a joint depth similar to that of executing, proving that these actions need some level of determination and have values similar to that of providing the service, developing the product, or delivering the outcome (Schwalbe, 2015).

Similarly Phillips (2013) suggests that the degree of interrelatedness of these five processes indicates a strong interactive dependence not limited to one another. He highlights the fact that a process does not just end and the next one begins. The rationale behind the interrelatedness and scope is that of on-going expansion. In line with this Rose (2013) states that projects are executed incrementally and some features of the project are discovered and implemented as time progresses—project goals/objectives are determined, discoveries are made, studies, investigations, and surveys are done, analysis is done, restrictions are improved, resources are reviewed, prospects are adopted, administration of changes is done, risk mitigation is performed, and unforeseen or unavoidable circumstances take place. He further states that management of the scope of a project calls for dynamic and practical PM during the course of the project. Initiating the project and/or planning it and not executing the plan is not adequate. Therefore constant planning, monitoring and control are essential(Rose, 2013).

When constant planning, monitoring and control have been done there must be a way of measuring whether a project was successful or not. For this reason a benchmark by which project success can be described and later assessed, needs to be instituted. Primarily, project success is described by Larson & Gray (2011, p. 55) as *"the delivery of the required product or service on time and within budget"*

To meet these objectives there are a number of limitations to be mindful of. In PM this is represented as a triangle known as the Iron triangle of PM constraints (Ebbesen & Hope, 2013). The iron triangle of PM constraints is made up of three constraints namely time, cost and scope. This triangle is depicted in Figure 7.

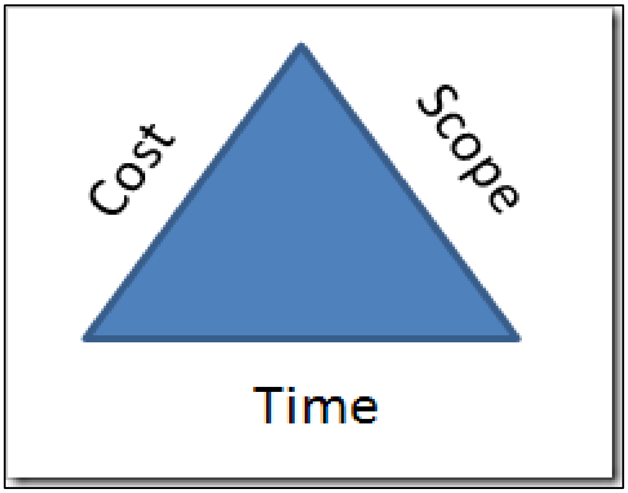


Figure 7: The iron triangle of project management (Ebbesen & Hope, 2013)

Table 3 below describes each of the constraints presented on the Iron Triangle of PM.

Table 3 : Iron triangle of PM constraints(Ebbesen & Hope, 2013)

Constraint	Description
Time	The time constraint speaks about the time that is required to successfully complete a project and is commonly

	presented as the schedule of the project.
Cost	The cost constraint, commonly referred to as the budget relates to how much it will cost to complete the project successfully.
Scope	The scope constraint speaks of the projected end result of the project.

This composition is comparable to a geometric triangle, where a modification on one of the sides of the triangle impacts on the other two sides (Ebbesen & Hope, 2013). Altering one of the constraints will alter the other two constraints (PMI, 2008).

Besides balancing the triple constraint, successful delivery of the project is thought to be dependent on the presence of the PM plan as well. Marchewka (2014) reports that the PM plan is a pivotal instrument in assisting the project manager to implement the project successfully. He defines the PM plan as a strategic and ceremonial plan on how to achieve the goals of the project. It does this by illustrating how a project will be executed and how it will be monitored and controlled. This involves generating a WBS for the project, describing and preparing how risk will be alleviated, identifying means of communicating well with stakeholders and other project team members, and creating a change management plan (Marchewka, 2014). Kerzner (2013) describes this document as a guide for implementing the project, and a method of getting stakeholders and the sponsor to buy-in and to grant approval before initiation of the project. He asserts that the document is kept up to date during the project lifecycle at prearranged milestones or significant instances to make room for the progressive, elaborative nature of the project. The PM plan will be different based on the scope, complexity, risk and sensitivity of the project. In order to successfully execute the PM plan, expertise in all the PM knowledge areas is necessary and is vital to the accomplishing of the project objectives (Kerzner, 2013).

Recently, debates have risen on what constitutes project success. This is the case due to the fact that the PMBOK's success is based on the iron triangle (project scope, time and cost) but current developments show that these are not enough for measuring success (Papke-Shields *et al.*, 2010), there also needs to be considerations of dimensions such as business results and preparation for the future (Saladis & Kerzner, 2011).

Burke (2013) asserts that creating a project is a distinctive action, implying that it cannot be standardized. However he also believes that the practice may be standardized by making use of specific well-proved and predefined models for designing, planning and implementing the project. He terms these models as project management methodologies. The two methodologies that are of interest to the current study are described below:

Table 4: PM methodologies(Burke, 2013)

PM Methodology	Description
Waterfall	Describes a series of phases to be accomplished, which bears a resemblance to a waterfall. This methodology splits the PM process into 7 successive phases: Requirements Specification, Design, Implementation, Testing, Deployment and Maintenance. The project can only progress to the next phase once the

	former one has been completed and verified.
Agile	This is an incremental and iterative PM methodology. Its key characteristic is that the end product's features and project lifecycle are not plainly outlined at the beginning. Instead of that, the work is done over a few iterative stages called "sprints". Agile project management enables project managers to continually gather feedback and improve the requirements between two iterations

Charvat (2003) states that in the waterfall methodology cost and schedule that are required to deliver the project are estimated based on the requirements. A plan is outlined and the team works to align to that plan. In essence, the development team attempts to estimate the future by approximating and restricting changes from occurring during the project life cycle. According to Sylvester (2013) this method is flawed because human beings are not perfect particularly at estimating. If they were perfect, delivering 100% of the requirements with the exact cost and schedule that was estimated would not be a problem. In addition to that no change would occur during the development of the project. However he believes that humans are not, and will never be, perfect. So when a project is late, the cost and/or schedule (or quality) will have to be adjusted to ensure that the project is delivered with all the functionality. So he basically believes that this is not a good methodology to use in an attempt to comply with the triple constraint.

The advantages that are put forward for the waterfall model take account of its straightforwardness and ease of scheduling in laying out steps for development

(Hass, 2007). Additionally the waterfall model is admired for its capability to improve quality management through its verification and validation processes (Cadle & Yeates, 2004). It is some these advantages that have allowed the waterfall model to become the backbone of PM. In contrast, Thomsett (2002) argues that the waterfall model is not suitable for the chaotic and client-driven business environment of the 21st century because of its tendency to be rigid. This is further supported by Kerzner (2013)' s claim that there is need to observe particular requirements when making use of the waterfall model, however, this falls short because reality shows that projects are not sequential in nature (Collyer *et al.*, 2010) and most importantly, in most cases, customers are not able to express all the project requirements during the initial phases of the project life cycle (Cicmil *et al.*, 2006).

Another disadvantage of TPM noted by (Conforto & Amaral (2016)) is that any design changes taken on during the testing and development phases of a project have the potential to cause disorder because of the waterfall model's requirement to complete the preceding tasks first. This may lead to project failure on the basis of time delay and quality; hence there is a need for methods that can handle this chaos. Furthermore according to (PMI, 2015), (Svejvig & Andersen (2015)) as well as Cui & Olsson (2009) late project changes in TPM are more costly and have insignificant beneficial effect on the resulting project delivery.

The agile methodology takes a very different approach to that taken by Waterfall; it flips the constraints upside down. Instead of confining requirements, it confines the cost and the schedule. The stipulated cost and schedule, govern the features that can be delivered. This methodology supports a process that makes an effort at adapting to change. By doing this, it warrants that projects are delivered within the defined cost and schedule. In addition the backlog is ordered according to value, the highest priority features are delivered for the cost/schedule. Commensurate to this view, Serrador & Pinto (2015) believe that this approach accepts change, which allows for changes in the priority of features as you learn. So when a project is behind schedule, the lowest priority features are taken out to make sure that the most important features are delivered within the given cost/schedule. This is evident when a project is delivered on time with some of the less important features not included in that release. They believe that this approach empowers the stakeholders to decide whether they would like to spend the remaining budget and time on the lower value features, or they would rather spend it on other projects and/or features. This proves that there has been changes that have taken place with time in the field of PM.

2.2.4 Success of Traditional Project Management in software development projects

Although there have been tools and techniques for control that PM has focused on creating with the goal of improving project success rate, research shows that only 28% of projects are successful (PMI, 2015). Maylor (2001) believes that the key emphasis of the traditional approach to PM is on standardizing and justifying the management of projects. This is done by aligning to a determined set of rules and in so doing warranting a better likelihood of success. However, he debates that to effectively use such methods and tools the situation is required to be predictable and unchanging, and that such situations no longer exist in today's business environment, highly characterised by unpredictability and instability. Therefore, projects are influenced by the dynamics of the environment, as well as technology and markets. He asserts that the traditional approach to PM presents challenges due to the fact that it fails to match the environment in which project managers function today. To validate this assertion he highlights the fact that the models of PM, notably PERT are highly deterministic and focused on procedures. Although there have been major improvements over the years these models are still not deemed as useful by a considerable number of world-class organisations. As an example, the method adopted by the Japanese automotive companies in their new product development projects does not make use of the TPM methodology. Maylor

(2001) stresses that while widespread endorsement of Japanese working practices is not being supported, the methods of Toyota in developing new vehicles in half the time of their western equals are undoubtedly worthy of analysis, predominantly because they vary so much from TPM. Given that many of the current business needs are parallel to that of Toyota, functioning in saturated, highly-competitive fast paced global markets rather than that of the cost-plus defence contractors of the past, it is fitting that PM be reviewed.

An additional weakness of PM that Maylor (2001) addresses is the fact that all of the project systems are fixated on making sure that there is conformance to budget, scope and time constraints. Higher level interests such as the need for constant development, excellence and accomplishing customer satisfaction are said to be beyond the scope of the project manager.

Marques (2012) infers that the several evaluations of PM have caused the assessment of new procedures to adapt management to the uncertainty and complexity conditions associated with the environment, goals and the activities in which projects are undertaken. The major point highlighted in the new methods is a contingency management of projects. The adaptive methodologies to PM focus on the following ideas:

- Dissimilarity and classifications of projects.
- Processes adaptive to different forms of projects and measures of success.

Shenhar & Dvir (2007) offered an adaptive approach that makes use of a model of variation and grouping of projects focusing on four aspects: innovation, technology, complexity and rhythm. When joined, these factors assess the intricacy and uncertainty of the situation, objectives and activities in which the project is launched.

Innovation is defined by considering the degree of innovation of the product or the result of the project for the prospective users. The degree of innovation signifies the level of knowledge of the product, its use and benefits (Shenhar & Dvir, 2007). It also indicates the uncertainty of the aim of the project, that is, the simplicity with which the requirements and wishes of users can de defined beforehand. Table 5 tabulates the degrees of innovation:

Table 5: Degrees of innovation (Shenhar & Dvir, 2007)

Degree of Innovation	Explanation
Derivatives	products are developments and improvements of products currently existing
Platform	products are new versions of products currently existing
Rupture	products are new discoveries that users have never seen or experienced before, arising from a new notion or theory

The technology piece focuses on the uncertainty of the project events and there are four degrees of uncertainty in technology as tabulated in Table 6:

Table 6: Degrees of uncertainty (Shenhar & Dvir, 2007)

Degree of Uncertainty	Explanation
Low	Projects that are governed by technologies that are well-established

Average	Projects that use of primarily well-established technologies but include new technologies that were not in existence in previous projects
High	Projects that use currently existing technologies that belong to the company
Super	Projects using technologies that were not in existence at the time of their commencement

The complexity aspect is associated with the extent of the project scope. There are three levels of scope complexity (Shenhar & Dvir, 2007). Table 7 tabulates levels of scope complexity

Table 7: Levels of scope complexity (Shenhar & Dvir, 2007)

Level of scope complexity	Explanation
Assembling	These projects have an amalgamated group of collective elements and subsystems that function differently together to accomplish a specific operational need.
System	These projects hold a combination of collaborative elements and subsystems that carry out many functions together, which accomplish a particular operational need.

Large System	These are projects that are made up of large and isolated groups of systems that work collectively to accomplish a common goal

The rhythm element addresses the urgency of the time allocated to deliver the project results. There are four levels of complexity of scope (Shenhar & Dvir, 2007):

Table 8: Complexity of scope (Shenhar & Dvir, 2007)

Complexity of scope	Explanation
Regular	These projects are not governed by time as a crucial factor of the immediate success of the organization.
Fast/Competitive	Time is a significant aspect concerning to organizational success in these projects.
Critical	Likewise these projects think of time as a serious component to the success of the organization
Blitz	These are projects whose main focus is dealing with crisis

Shenhar & Dvir (2007) also put forward five dimensions that could be used to evaluate project success and these are listed below:

- **Efficiency:** managing time and cost intentions.

- **Impact on consumers:** meeting user requirements.
- **Impact on staff:** satisfaction, staff retention and personal growth.
- **Results for business:** return on investment, market share and growth.
- **Preparing for the future:** new technologies, new markets and new skills

Marques (2012) reasons that the model offered by Shenhar & Dvir (2007) is useful in amending the examination of advantages and threats associated with intricate and indefinite projects. However, he argues that a point to be raised in the grouping of the elements of innovation, technology, complexity and rhythm, is exactly how to take into account the strength of each aspect within the same company. For example, assume two projects in the same company have been grouped in their separate areas as being of high innovation type. In trying to select which projects to incorporate in the company's portfolio of projects, the Shenhar & Dvir (2007) model does not provide a contrast between the different degrees of innovation.

The next section examines the ASDM which are said to enable organisations to react more quickly to new global challenges and to the changing business environment due to their dynamic nature and the notion that user requirements do not have to be fully specified in the initial phases of the development process.

2.3 Agile Software Development Methodologies

The background of AM can be traced back to the conference held by a group of developers in 2001, to deliberate the prospects of an innovative group of SD methods (O'Sheedy, 2012). The word 'agile' originated from this conference and turned out to be an umbrella term for all the development techniques that endorsed immediate response to software project requirement changes (O'Sheedy, 2012). The key outcome from this conference was the creation of the agile manifesto, in which the fundamental principles of agile development were made clear (Beck *et al.*, 2001). These are tabulated below in Table 9*.*

> *"We are uncovering better ways of developing software by doing it and helping others do it, through this work we have come to value"*(Beck *et al.*, 2001)*.*

Table 9: Agile values as listed on the agile manifesto

Left	Right
Individuals and interactions over	processes and tools
Working software over	comprehensive documentation
Customer collaboration over	contract negotiation
Responding to change over	following a plan

> *That is, while there is value in the items on the right, we value the items on the left more" (Beck et al., 2001).*

This subsection serves the purpose of providing the literary context for agile software development (ASD) as a dependent variable of the study. This is an imperative step in the process as it provides a frame of reference in relation to the entire study.

2.3.1 Theory on Agile Software Development Methodologies

The AM for SD is influenced by five attributes. These are: Iterative Development (ID), Continuous Integration (CI), Collective Ownership (CO), Test-Driven Design (TDD) and Feedback(Ferreira & Cohen, 2008). A short explanation of each of these attributes is given in Table 10 below:

Table 10: Agile Characteristics (Ferreira & Cohen, 2008)

Characteristic	Explanation
Iterative Development	Fast release of small functional and tested software, released at fixed times
Continuous Integration	New code is constantly integrated into the production base code, where possible after each activity has been finalised
Collective Ownership	Developers have equivalent privileges to make changes to the code, in the form of add-ons and code support, at any given time and anywhere in the system
Test-Driven Design	Tests are created by the developers before coding commences and this method is followed to inspire developers to reflect before coding begins
Feedback	A constant and regular feedback loop with the end user allows developers to validate the accuracy of the development process

- In a study done by Ferreira & Cohen (2008) to ascertain how these five significant features of AM influence client satisfaction, they discovered that each of these characteristics has a positive impact on client satisfaction. Their findings were reinforced by several factors including the following: ID has a major influence on client satisfaction. This is because it facilitates reprioritization of features which ensures that changes can be made when new information arises and when it is deemed necessary. This enables early and continuous demonstrations of the system value which is often exciting for the clients and encourages them to communicate more ideas. Furthermore, when the iterations are provided it is easy for the client to decide whether *"this is what they really want thus reducing the chances of the developers going down the wrong path"*.

According to Flora & Chande (2014) although this model addresses most of the problems brought about by the waterfall methodology it does however present new challenges. One of these challenges he believes is that the user community needs to be actively involved throughout the project. While this involvement can be helpful for the project, it is time consuming for the user community and can add to project delay. Fitsilis (2008) highlights the potential for scope creep introduced by this model of development since user feedback following each phase may lead to increased customer demands. As users see the system develop they may realise the potential of other system capabilities which would enhance their work. CI is imperative because it improves the likelihood of early detection of errors and integration issues. It also enables deployment to be automated. Gandomani *et al.* (2013) highlight the fact that CI tools maintenance and their administration have associated costs to it. Williams (2010) attests to this as he believes initial installations, configurations, and team changes are not without cost. He explains that these elements cost real time and money and, in the beginning, can create disruption. The business must invest the time required to initialize CI, making sure the customizations for business objectives and infrastructure operations are in place and operational. So from the evidence it can be concluded that CI is beneficial if the required tools can be put in place. CO is thought to be significant in supporting client satisfaction when understood correctly as it inspires team members to self-organize and deliver quick responses to different situations that may surface. It also reduces control of knowledge by specific individuals through variation of team members and further diminishes the risk stemming from the departure of a key team member. However there are strong arguments against CO. Williams (2010) shares some of the problems that stem from CO. The first problem he highlights is the prevalence of bugs. This is caused by people refactoring the code, majority of the time these individuals don't really understand the code and this results in them breaking something subtle in the original code. Contrary to the belief that CO assists team members to deliver quick responses Munassar & Govardhan (2010) believe that it slows down delivery because nobody has any expertise in any given domain, so people are spending more time trying to understand other people's code and less time writing new code. A TDD approach was found to improve client satisfaction due to the fact that it allows for an earlier and hence less expensive detection of defects. Regular feedback was discovered to help organisations in identifying changing requirements by giving the client sufficient time to express their desired changes.

This study exposes some of the reasons why AM have infiltrated SD practices universally and still continue to. It is apparent from the 2005 survey of the US and Europe cited by Dybå & Dingsøyr (2008) which asserts that 14% of companies were already making use of AM and 49% of the companies aware of AM had an interest in embracing them. Vijayasarathy & Turk (2008) give more reasons for the on-going adoption of the agile approach to SD. The factors that they discovered were personal curiosity, peer encouragement, expedition for productivity, significance and success. In addition to these factors other benefits that were highlighted by the study comprise of the capability to be flexible and provide quality software that satisfies customer needs faster.

Rosser *et al.* (2013) state that by accepting changes in requirements and upholding a culture of rich user engagement the AM produce increased productivity. They believe that this can be attributed to the importance placed on functioning software as the principal measure of development. This suggests that you can get the product to the market faster, make enhancements more proficiently and intensely grow your return on investment. An improvement in team morale is also attained and this is done through agile development's emphasis on strong collaboration and self-organization. Being able to effect decisions that concern you and interactions with users fosters job satisfaction (Dybå & Dingsøyr, 2008). Superior predictability of schedule/costs/quality accomplished through agile development's application of time-boxed, pre-set iteration schedules, the cost of each iteration is predetermined and restrained by the volume of work that can be completed by the team in the fixed-schedule time box. Knowledge transfer and sharing which is attained through rigorous communication between the developers and the client and rotation of team membership makes sure that knowledge is not controlled by a few individuals. These benefits were also acknowledged by Vijayasarathy & Turk (2008)as drivers of the adoption of ASDM.

Keil *et al.* (2003) suggested that software development projects are more often than not likely to go out of scope and effective PM is one of the means in which the consequences of such scope creep can be controlled and managed. Like traditional methods of SD, ASD applies PM; however APM is founded on different practices and philosophies. In their support of APM, Karlesky & Voord (2008) summarise a number practices associated with APM which they suppose makes it the most suitable method to manage a project based on the reasons outlined below:

Testing

Testing is thought to be imperative in defining the status of the features and creates the foundation for many of the other principles.

Iterations

This procedure acts in response to the shortfall in the technique applied by TPM which presents work in phases which take many weeks or months. Karlesky & Voord (2008) consider these phases to be too lengthy to effectively observe progress, react to changes and arising information, and act with sufficient time as to prevent disruption in the schedule. APM makes use of development iterations to split lengthy projects. With these short, well-defined, frequent periods of time (on the order of one or two weeks), metrics can be collected and used to forecast and control schedule changes after implementation of only a few iterations.

Feature-Driven Development

In an agile setting the aim of SD is to deliver working software that brings customer satisfaction. Software architecture and subsystems are there only to support features. Hence all PM is focused on providing features (Karlesky & Voord, 2008). According to Karlesky & Voord (2008) APM does exactly that. In APM, the customer prioritizes features in each iteration. This allows for priorities to be altered in each iteration as required in response to conditions. Recurrent and early release of features enables the end users to work with and test software long before the final test phase of TPM. This practice makes it easier for users to offer feedback that can be grouped into iteration planning and feature prioritization. In discovering things iteratively as the project advances, APM assists in identifying functionality that is not of significant value. This saves time and money while exposing and permitting for the implementation of the most valuable features. This exercise can significantly change the original set of requirements. However, by regularly re-examining feature priorities, the agile team can ensure that what is most important to a customer is always delivered first. Should schedule or budget changes cut the project short, the most important and valuable features are those that have already been delivered. Sharma *et al.* (2012) believe the approach is fine for many situations, especially for handling development cycles for a bunch of change requests or errors. However, they argue that the approach and especially the use of feature lists does have some disadvantages. The first flaw that they highlight is that the value of the product being developed is equal to the value of the users required features. This is seen as a flaw because users are often keen on features that make their life easier, make the product nicer, or just pops up when asked. The outcome is that the feature lists often contain a lot of ‘nice-to-have’ features such as 'the list must be divided into two horizontal panels', 'the colours must be customisable', etc. Secondly, in this approach there is nothing that ensures that the features listed are supporting the desired business benefits (Sharma *et al.*, 2012). To address these risks for failure they suggest that the desired business benefits should be outlined and the desired business process should be outlined. Finally, they recommend that the feature list must contain features ensuring the desired business benefits and these should be prioritised.

Simplicity

Karlesky & Voord (2008) suppose that unnecessary complexity results in unnecessary cost. Complexity causes the code to be error prone, obscures the testing process, slows down progress, and obscures code (consequently increasing maintenance and documentation effort). The simplest idea that will reach the goal at hand yields neat code, faster progress, and greater system efficiency than a more interesting, complicated solution.

Only implement needed software

Karlesky & Voord (2008) also hold the belief that conventional ideas of SD and PM constrain change. Developers and project managers who have adopted this viewpoint develop subsystems and software architecture to satisfy every expectable requirement of the layers of software to follow. Karlesky & Voord

(2008) confidently declare that to a large extent, the functionality delivered by such foundational layers of software will never be utilised by the software developed later in the project and that any functionality built early in a project that is not required later in the project is left-over. They further assert that the unnecessary effort taken to present unnecessary complication takes away the ability to make changes to the software. It also slows down progress toward the final goal of delivering valuable features. APM takes a different approach. Developers execute only the software required at the time they are producing it. Relying on groups of regression tests, developers can refactor existing code and add functionality when actually needed at later stages of the project.

Estimation

Karlesky & Voord (2008) state that the cause for most projects that are managed using traditional PM methods not meeting their timelines and thus either failing or being completed with serious budget overruns or with the lack of key features is the struggle in estimation of software. They further declare that APM focuses on the shortcomings of traditional software estimation by finding a fundamental limit in humans. They believe humans are not particularly capable of assessing how much time a given task or group of tasks will take to complete. This inability is exacerbated in large projects containing a substantial number of tasks. Nevertheless, humans are reasonably competent at estimating relative complexity (e.g. A is twice as complex as B). Based on these two notions instead of applying time-based estimation, APM uses complexity-based estimates. One of the drawbacks of this approach is that it can be misleading when used as a measure of productivity (Ceschi *et al.*, 2005). Different teams use this approach in different ways. It is a relative measure which means team A may assign a story a different complexity from team B (Ceschi *et al.*, 2005). Unless all teams are calibrated, confusion can result.

Documentation

Karlesky & Voord (2008) alluded to the fact that conventional ideas of SD and PM favour extensive documentation. Extensive planning and architectural documents are generated before the project commences. More documentation is contained within the source code while it is written. More documentation still is generated when the project is complete. Karlesky & Voord (2008)further draw attention to the main problem with software documentation which is its short "shelf life". Insignificant changes to the source code of a system can invalidate important sections of the documentation. In contrast APM favours reduced, adjustable, timely documentation. To the point that contractual obligations will allow, preliminary and final documentation must be simple, high-level summaries of significant features and subsystems that are not likely to change. Unit and system test collections represent executable documentation on the system's source code behaviour and architecture. When tests are reviewed, the existing documentation is reviewed as well. Development teams can make use of adjustable, combined documentation systems such as wikis to document and effortlessly modify important processes, setup instructions, and instruct boundaries. Such systems are useful in collaborating among the team itself and the larger organization. Keramati & Mirian-Hosseinabadi (2008) argue that the minimal documentation strategy means there is less information available to new starters in teams about the features and how they're expected to work. Keramati & Mirian-Hosseinabadi (2008) state that this documentation could cause potential misunderstandings if the teamwork and communication aren't at their best.

Risk Management & Scope Management

According to Karlesky & Voord (2008) Risk and scope management are easy to achieve in APM. The high risk areas of the project are selected to be done first. As each iteration is finalised, the Burndown chart is reviewed and selections are made on existing and new functionality. Velocity calculations give such insight that resource and feature planning can be dynamically and strategically changed to meet release schedule and budget constraints.

2.3.2 Challenges of making use of Agile Software Development Methodologies

From the sections above it is apparent that there are several strong promoters of AM and there appears to be a lot of benefits that come with taking on these methodologies. However, the sections above also highlighted that the agile approach to SD is not without its drawbacks. In fact, the features that are considered to be beneficial and to be good reasons for adopting SDM are often seen as limitations to the use of AM. Moczar (2013) claimed that in spite of the agile approach's wide acceptance and its potential to deliver solutions to perennial IT concerns, from his participation in agile projects as a lead architect ,team member and manager he believes that these methodologies have not only proven to be unsuccessful but have in fact exacerbated the situation in the SD field. He reinforces his account with what he refers to as "the most destructive agile principles". He states that these principles result in the converse of what is promised by Agile. The principles under scrutiny are outlined below:

Early and continuous delivery of valuable software

Moczar (2013) believes this is an imperative principle for ensuring customer collaboration. However the risk associated with it is the focus on continuous delivery and the resulting impact on generating an unsurmountable defect backlog. This leads to programmers who were initially enthusiastic about the idea of collaboration being psychologically worn down. This is due to the endless defect list resulting in a burn out which contradicts the agile promise that projects "should be able to maintain a constant pace indefinitely.

Responding to change over following a plan

According to Moczar (2013), in theory developers code at the same time as retaining a strong user engagement to enable the users to outline requirements and express any changes in requirements as the project advances. The methodology, however, does not account for the cost associated with each change hence there is a requirement to distinguish between major and minor changes. This leads to individuals raising significant changes in the advanced stages of the project exploiting the rationale that, being an agile project it should be able to manage this. There is only one way in which the project can manage this and that is by increasing the number of iterations. As that happens, defects that may have been simple to resolve at one point become more complex to resolve, as the code base keeps changing. Moczar (2013) concludes that this principle encourages inadequate and immature planning while obscuring its effects. He affirms that as the iterations progress and the defect list grows, it raises dissatisfactions from the customer's side due to the lack of quality and the expected deliverables not being delivered. He further highlights the disparity between this principle and the traditional practices, where the project is centred on distinct requirements and changes are controlled using a Change Management process. Although it can be complex, it does at least declare time and monetary costs more clearly. Yatzeck (2012) however argues that a lot of work has been done on agile to manage this challenge. She explains that in agile the planned work is laid out at a high level before any work is started. This is done to make sure the team is in general agreement about how much the team will get done, but once the general size of the effort is agreed, details are purposely left to be determined later. At the planning stage of the project, the team might specify for example, that a user interface is with about 10 fields is required. However they don't specify what the exact fields are, what the validations on the fields will be, nor where they will be located in the database, until it is time to do actual development on that story. This is done to avoid scope creep. Essentially, all that the team has invested in at this point are words on an index card and since the team hasn't done detailed analysis of the story, it is not a major issue to swap an index card. She adds that in agile stakeholders are not permitted to make any changes when development has already been completed. Secondly, the agile team allows product owners to change user stories without penalty if the team estimates that the new story is roughly the same amount of effort as the old one (Yatzeck, 2012). According to Yatzeck (2012) when agile teams put together a release plan, they should leave a 20% contingency completely open for newly discovered scope. If they discover as they go along that in order to get what is needed, they need some additional effort, this can still be done until the contingency has been used up.

Self-Managed teams

Firstly, Moczar (2013) acknowledges the value in empowering people and the fact that a significant number of traditionally managed projects do not effectively utilise people. However he believes that responsible and reliable PM cannot be swapped with self-organisation.

The last argument raised by Moczar (2013)on the insufficiency of self-organisation is important yet debatable. PM is undeniably an indispensable part of any SD venture; however the self-organized teams inspired by the AM are not aimed at replacing PM but it is a new responsibility that is being undertaken by agile project managers (Hoda *et al.*, 2008). The core principles of PM which are management of the teams, customer relationships, cost reduction, risk management, maintaining project time line and budget are still in existence in AM, the only thing that has changed is how they are implemented (Hoda *et al.*, 2008). Agile project managers are different from the conventional/traditional project managers in that they promote self-organized teams with 'light-touch' management while traditional project managers follow a control and command method to PM (Hoda *et al.*, 2008).

Gandomani & Nafchi (2016)'s observation on the limitations of the agile approach harmonize with Moczar (2013)'s views as they consider the agile approach to be heavily reliant on close business participation. If possible, the interaction is required on a daily basis. This might bring about a problem because it places business people under unnecessary pressure as they also have to manage their daily operational responsibilities. In response to this challenge a study was done by Bass (2015) on eight international companies based in London, Bangalore and Delhi. The aim of the study was to scale AM to large offshore enterprise development programmes with a focus on tailoring the role of a product owner. In this study Bass (2015)suggests that to combat the problem of product owners not being able to commit the time necessary to guide the teams in developing the correct product; there should dedicated teams of product owners. In this study Bass (2015) discovered that apart from the primary role of communicating customer needs to agile teams, there are other functions within the role of a product owner. Some of these functions are, gathering requirements from business clients, ensuring that requirements bring value to business, and manage and approve release plans. Having the product owner teams and dividing the functions amongst the various proxy product owner ensures that the required collaboration with business is attained without placing business under pressure.

Additionally Imran *et al.* (2016) add to the limitations of agile principles by highlighting that the flexibility to change the path of the project as required and to ensure that the correct product is delivered possesses two fundamental trade-offs. The first is the possibility of scope creep which is accompanied by the risk of a project that never reaches the end. This brings about a challenge in forecasting what the project will deliver both at the beginning and during the project. This creates more difficulties in defining a business case for the project, and makes it even harder to discuss fixed price projects. The lack of a definite vision and the absence of governance with regards to establishing timelines and interchanging scope the project is placed in a risky position(Imran *et al.*, 2016). Yatzeck (2012) believes that project changes cannot be defined as scope creep if they result in the delivered product being exactly what the project stakeholders need at release time and the code quality is such that the team can continue to keep the software matched to the business needs indefinitely. Essentially, she believes that the important thing is that the software meets current business needs and will continue to do so. The second drawback stems from the technique used to carry out requirements elicitation in the agile approach. This technique is said to reduce waste of effort on deliverables that do not last (i.e. that end up not being part of the finished product) and this saves time and thus money. However this results in the collection of inadequate requirements. Inadequate requirements may possibly result in a problem of new team members having insufficient information on the software features and how they are required to function(Imran *et al.*, 2016).Imran *et al.* (2016) also touch on the possibility of misinterpretations and misunderstandings surfacing if the teamwork and communication are not at the best standard. Finally, the fact that testing is incorporated with the other activities throughout the lifecycle increases the cost of resources on the project.

Furthermore, the principles of ASDM are said to encourage insufficient documentation, poor architecture and lack of risk management which makes it difficult to directly use them for medium and large development projects (Qureshi, 2012). This contention is recorded in the case study report by Dybå and Dingsøyr (2008). In the literature they referenced 33 projects and only 4 of those had 20 or more members only one project team had a size greater than 23, at 60 members. The study by Chow & Cao (2008) investigating critical success factors in 109 agile projects also serves as a substantiation to this claim. Out of the 109 projects, almost 80% of project teams had less than 20 members. To further validate this claim it is important to highlight a commonly known fact that the Scrum Methodology endorses project teams of no more than six members (Leffingwell & Smit, 2005).From the evidence offered and also recognised by Batra *et al.* (2010), agile approaches are more suitable for projects with high levels of uncertainty and unpredictable requirements and varying project objectives. In all other cases, the use of the AM for development can be challenging.

Qureshi (2012) appears to be in agreement with these statements as he emphasises that the principles of ASDM only offer support for the development of small software projects characterised by having small teams. Yet there is no guidance offered on how the agile approach can be adapted to suite medium and large software projects. He believes that some of the reasons why this approach must be amended to make it more suitable for medium and large projects are: inadequate architectural design and organization, over emphasis on early results, insufficient documentation and low levels of test coverage. As an example, the Extreme Programming (XP) methodology advocates for coding/development and operational software in the place of comprehensive documentation/architecture design. He also adds that there is a tendency to trade-off architectural design by concentrating on features that have a direct and instantly evident influence on the current increment. This practice offers the expected results for small projects but overlooks significant architectural aspects for medium and large projects where cost and effort of a change in the architectural design is substantial. The challenge of ASDM principles not being suitable is not only limited to medium and large projects, AM such as XP are not suitable for safety critical projects. This is because during the project lifecycle of safety critical software, all requirements need to be known up front and every portion of code must be mapped out to the requirements. This contradicts the agile philosophy where there is a less of emphasis on documentation, traceability and other formal techniques (Poppendieck, 2002).

Aside from the challenges presented for implementing ASDM for medium and large projects, Boehm & Turner (2005) identified three areas of difficulty that traditional developers and managers are confronted with. These are associated with scaling up and integrating AM to traditional, top-down systems development organizations. The first of these is development process conflicts. In this area managers are confronted by the predicament of integrating AM with traditional business processes. This needs to be done while avoiding destroying the agility or undermining the years spent defining and improving systems and software engineering process asserts. Boehm & Turner (2005) found four subcategories that result in this predicament and these are listed below:

Variability

In situations where agile and traditional teams are generating software for the same product, they can turn up with considerably different fragments that may pose challenges when making an effort to integrate. Without the required resources for coordination, an agile team's domain conventions, GUIs, or commercial off-the-shelf selections could vary significantly from other developers' corresponding conventions (Boehm & Turner, 2005).

Different life cycles

Making use of different life cycles also introduces problems. Agile processes stress instant delivery of functionality, while traditional methods emphasise promoting development over a prolonged period. The traditional lengthier life cycles need amendments to the agile processes (Boehm & Turner, 2005).

Legacy systems

Applying agile methods to legacy systems, whether within maintenance or as new development, presents a number of concerns. Legacy systems are often not easy to refactor or modify to accommodate agile changes that want to create expertise in increments. Legacy systems could also establish difficult and often inflexible business processes that are entrenched in the culture and are not easy to do away with (Boehm & Turner, 2005).

Requirements

Differences between how agile and traditional approaches execute the requirements process can also bring about challenges. Requirements in the agile methodology are largely functional and relatively informal. This could or could not work in a systems engineering verification and validation method.

The second area of difficulty that was identified was business process conflicts. There is a difference in the manner that daily business processes are done (Boehm & Turner, 2005). The following business processes instigate difficulties (Boehm & Turner, 2005):

Human resources

Companies need to learn to adjust to human-resource affairs such as timekeeping, role descriptions, team-oriented versus individual incentives, and essential skills. AD team members often go above and beyond the limits between standard development position descriptions and might need significantly more skills and information to function well. Ironically, HR departments and procedures often hinder efforts at empowering people to get involved in non-traditional approaches that need organizations to re-examine legally appraised and audited policies and procedures (Boehm & Turner, 2005).

Progress measurement

Standard contracts, objectives, and progress tracking techniques may fail in supporting the fast pace of agile processes. Contracts and payments are often focused on principles that are not valuable in an agile setting for example preliminary and critical design reviews (Boehm & Turner, 2005).

Process standard ratings

An area of conflict for reputable and traditional organizations is often on the influence that agile will bear on their rankings with regard to the Capability Maturity Model Integration (CMMI), International Organization for Standardization (ISO), or other process standards. Boehm & Turner (2005) believe that agile is aligned to the level 5 concept of striving for continuous improvement in performance. However they also express that a great number of AM lack the necessary documentation and infrastructure to support such lower-level certification which makes it less effective.

The last area of difficulty examined by Boehm & Turner (2005) is people conflicts and they believe it’s the most significant in bringing about an improvement in the management of engineering and development of personnel. People issues are also said to be the main focus of the agile movement and to a large extent the paradigm shift is directed at empowering people by promoting reasonable goals, shorter feedback cycles, ownership and flexibility. The difficulties are outlined below:

Management attitudes

The change from traditional to agile management attitudes can be challenging. Important management processes such as earned value and statistical process control were established from a manufacturing concept and tend to toss employees all over the place as disposable parts. Managers also have a habit of relating to employees with certain roles and that could instigate difficulties in the multitasking traits of agile team members. Project managers in most AM perform two main functions: protector and coach. They embody a barrier between the organization and the team to reduce unnecessary discomfort during a sprint or development lifecycle and provide adept technical assistance when the need arises. Many traditional managers also play these roles, but AM place more emphasis on them (Boehm & Turner, 2005).

Logistical issues

Particular logistical issues have a direct impact on people in agile settings. Agile teams need to be collocated virtually all the time. The typical agile workspace needs pair-programming stations, walls for status charts and assignments, a layout that enables easy conversations between team members to share information, and sufficient tools to sustain continuous integration and regression testing (Boehm & Turner, 2005).

Handling successful pilots

Boehm & Turner (2005) observed that the undesirable effects of how organizations deal with the success of pilot projects are often ignored when reporting results. They refer to a statement made by Alistair Cockburn during a USC workshop. The response he observed to a successful agile pilot is that they either dismiss or promote the manager or break up the team and they are of the opinion that that the outcomes of that are: damage to team relationships, both technical and personal, it dilutes the acquired knowledge and lessons learned and it sends out the message that trying new things might be risky to your career.

Change management

Resistance within the organization may surface as soon as there is an indication of change to the existing culture. This can be noted in a number of behaviours including the cultural victimisation of change agents or early adopters and the intentional disruption of projects through direct or indirect means. On top of this there might be employees who refuse to make use of new methods. This is detrimental in an agile environment because agile teams rely on trust and shared implicit knowledge to support pair programming and shared ownership. This also institutes controversy to any efforts to measure the results (Boehm & Turner, 2005).

The other management challenge related to adopting agile is the change in the nature of PM (Nerur *et al.*, 2005). As previously mentioned AM involve a change from command-and-control management to leadership and collaboration. The major challenge at this point is getting the project manager to give up the authority he/she previously possessed (Nerur *et al.*, 2005). AM encourage lean thinking and reducing operating cost, mainly documentation. A considerable amount of the knowledge in AD is unspoken and is inherent in the minds of the development team members. This forms a dependency on the development teams from the organization’s side and can possibly shift the balance of power from the management to the development teams (Nerur *et al.*, 2005). Unlike the TPM practices where decisions are made by the project manager, in an agile environment decisions are made by the development team that consists of software developers and the customer. Decision making in such cases is very difficult.

Thamhain (2014) weighs in on the challenges associated with implementing AM. Based on his study on the practices of APM he highlights that agile principles which prescribe that the management process must be iterative, incremental, rely on self-organised teams and evolve regarding its work structure and processes as needed during the project lifecycle are very difficult to fulfil in large projects. He believes that this is because larger projects need more execution formality and discipline to deal with the specific complexities, contractual requirements and project interfaces. The second challenge that he draws attention to is that not all project activities fit agile. Certain activities and their deliverables, for instance documentation or training are not part of agile and applying agile to these activities calls for key modifications. Lastly he points out that because agile is based on adaptable team organizations, it is often seen as being unsuitable for projects that need to be implemented to produce particular outcomes within the given time and resource constraints. These projects practise traditional standardized management processes. Project managers maintain that you cannot have it both ways, maintaining a flexible organization with evolving work processes and control over established project requirements and deliverables. This argument is especially powerful for large projects and special categories of projects, such as government contracts. For such projects the overall requirements and project scope must be established up-front and becomes the basis for performance measurements throughout the project lifecycle.

In response to the challenges presented by agile for large and complex projects. Thamhain (2014) believes that much of the criticism of agile is based on the assumption that it does not work in particular organisational cultures or work processes. He asserts that the main problem is that people believe that processes are fixed and are not under the control of the project manager. He dispels this belief by stating that asserting that agile is not a rigid template but rather a guideline that must be adapted to a specific project situation, especially if the situation is outside the framework for which agile was originally designed for. There are studies that have been done that attest to Thamhain (2014)'s view on the adaptability of agile to suite different kinds of projects. A good example of such a study is the study done by Nord & Tomayko (2006) based on a large program with the aim of assisting it to modernise its large transaction-based system that operates 24x7 while adopting ASDM. In this study they found that embracing the principles of ASDM and software architecture provides improved tactics for risk management. They also found that a systematic-architecture-centric review of project factors is crucial for understanding of risks and dealing with the challenges arising in large-scale SD. They propose the use of an incremental evaluation approach in ASDM, where small, short architecture evaluation sessions could be applied in agile sprints. This creates a strong architectural base for large and complex projects. Additionally, Leffingwell (2017) developed an ASD framework popularly known as the Scaled Agile Framework (SAFe). This framework was developed with the goal of enabling large enterprises to use agile with their large and complex programmes (Leffingwell, 2017). It supports both software and systems development from the small scale of under 100 practitioners to the largest software solutions and complex systems. Figure 8 below summarises this framework.

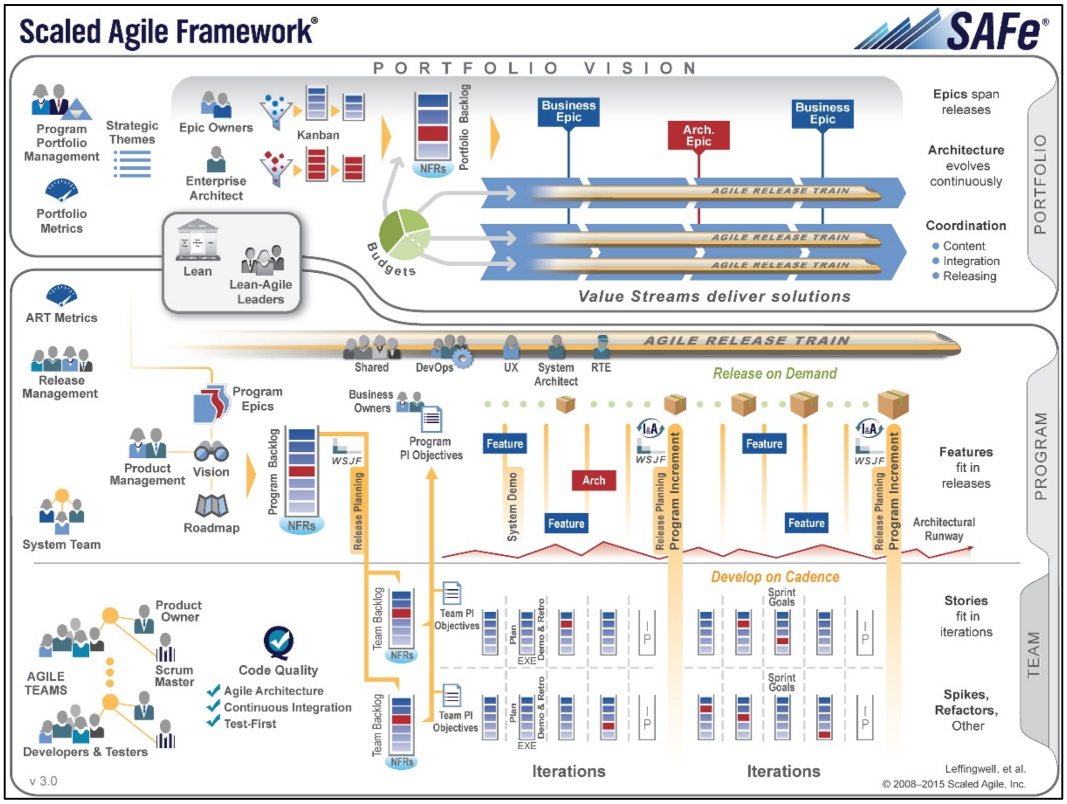


Figure 8: Scaled Agile Framework (SAFe) (Leffingwell, 2017)

Traditionally, scrum, extreme programming and other agile methods tend to concentrate, and end, at the team level. SAFe offers a distinct and integrated view of the work to executives, which enables them to drill down for details or drill up for trends and analysis.

Team

A team in SAFe may consist of 8 to 10 people, with everything they require in order to deliver software, end-to-end. These teams work in a two-week sprint using Extreme Programming (XP) methods. Unlike traditional development scrums the teams do not work autonomously. As an example, their team backlog comprises items drawn from the Program backlog, and the length of their sprints are coordinated with all the other teams on the same "Agile Release Train" (see the next section) (Leffingwell, 2017).

Program

Several SAFe teams create what is referred to as an agile release train. This is organized around a program. Agile release trains synchronize their iteration boundaries and deliver incorporated, working systems every two weeks (Leffingwell, 2017).

Portfolio

A portfolio is a collection of these programs, the total amount of budget within IT going into software development. SAFe calls this "Program Portfolio Management," and suggests that one office should have the responsibility for strategy and investment funding, program management and funding (Leffingwell, 2017).

The following subsection will be evaluating the PM technique applied in ASDM.

2.3.3 Project Management in Agile Software Development Methodologies

Agile Project Management (APM) is a paradigm shift from the traditional plan-then-execute-PM models which follow the principles of the normal four-stage (initiate, plan, execute, close-out) project life-cycle phases to a new five-phase (envision, speculate, explore, adapt, close) project life cycle, as explained by Highsmith (2009). TPM methodologies are intended to avoid change through comprehensive planning and documenting as much as possible before the system is developed. On the other hand APM accepts that change is unavoidable and that it is not to be avoided but managed (Karlesky & Voord, 2008). APM enables software project managers and employees to adapt to changing circumstances, instead of improving rigid formal controls as in traditional linear development methods (Augustine *et al.*, 2005).

In this section APM will primarily be defined with reference to Highsmith (2009). APM has four important points (Highsmith, 2009):

- Opportunities created by the agile revolution and its impact on product development;
- Values and principles that change APM;
- Specific practices that embody and amplify these principles; and
- Practices to help entire organisations, not just project teams, embrace agility.

SD teams are transforming where they have to in order to adjust from pre-emptive to adaptive styles of development (Highsmith, 2009). To build ground-breaking software products, APM has five business objectives (Highsmith, 2009):

- Constant innovation: deliver on current client and business expectations;
- Product flexibility: deliver on future client and business expectations;
- Better time to market: meet market opportunities and increase the return on investment;
- People and process flexibility: respond swiftly to required product and business changes; and
- Dependable results: support business development and effectiveness.

The three fundamental APM values (Highsmith, 2009) include the following:

1. Delivering value above meeting constraints: This value places emphasis for changing how performance is measured on projects (Highsmith, 2009). Traditional project managers emphasise delivering according to the time, cost and quality

requirement restrictions as outlined in the project scope. Agile project managers on the other hand focus on delivering value and continuously questioning whether different interpretations of scope are worth the value they deliver (Highsmith, 2009).

2. Leading the team above managing tasks: Agile leaders provide leadership to teams while non-agile ones focus on managing tasks (Highsmith, 2009). The major focus of APM is to form self-organising teams and to manage them with a "lead-by-serving mentality". There are four main subjects related to constructing teams:

- Constructing self-organising project teams;
- Leadership;
- Collaborative teamwork (including decision-making); and
- Client collaboration.

3. Adapting to change above conforming to plans: Traditional project managers see the plan as the objective and concentrate on following the plan with minimal changes, whereas agile project managers see client value as the goal and put all their effort on successfully adapting to unavoidable changes (Highsmith, 2009). The project plan turns into a way of achieving certain goals not the goal itself when quality and client value are the main objectives. Even though the restrictions described in project plans are very important, project plans are not consecrated; "they are meant to be flexible; they are meant to be guides" that do not confine the team (Highsmith, 2009).

APM principles resulting from the adaptive principle statements are summarised as (Highsmith, 2009):

- Accept change and react appropriately instead of following outdated plans.
- Adapt processes and practices as necessary.

The Agile Project Management and Delivery Framework

The APM delivery framework is established to support an organisation's business objectives; it "stresses implementation and it is descriptive rather than deterministic” (Highsmith, 2009). In order to accomplish business objectives, the framework must (Highsmith, 2009):

- Support a controlled and self-organising project team;
- Endorse consistency and reliability as far as possible, given the level of ambiguity that might exist in the project;
- Support an adapt, envision and explore culture;
- Include practices that provide support for each project phase;
- Include learning;
- Be flexible;
- Support a transparent view into the process; and
- Supply checkpoints for management for evaluation.

The APM delivery framework as presented in Figure 9 depicts the five phases of APM. The phases need to be viewed as one phase flowing into the next and not like encapsulated separated phases – “the APM terms were selected to imply iterative evolution” (Highsmith, 2009).

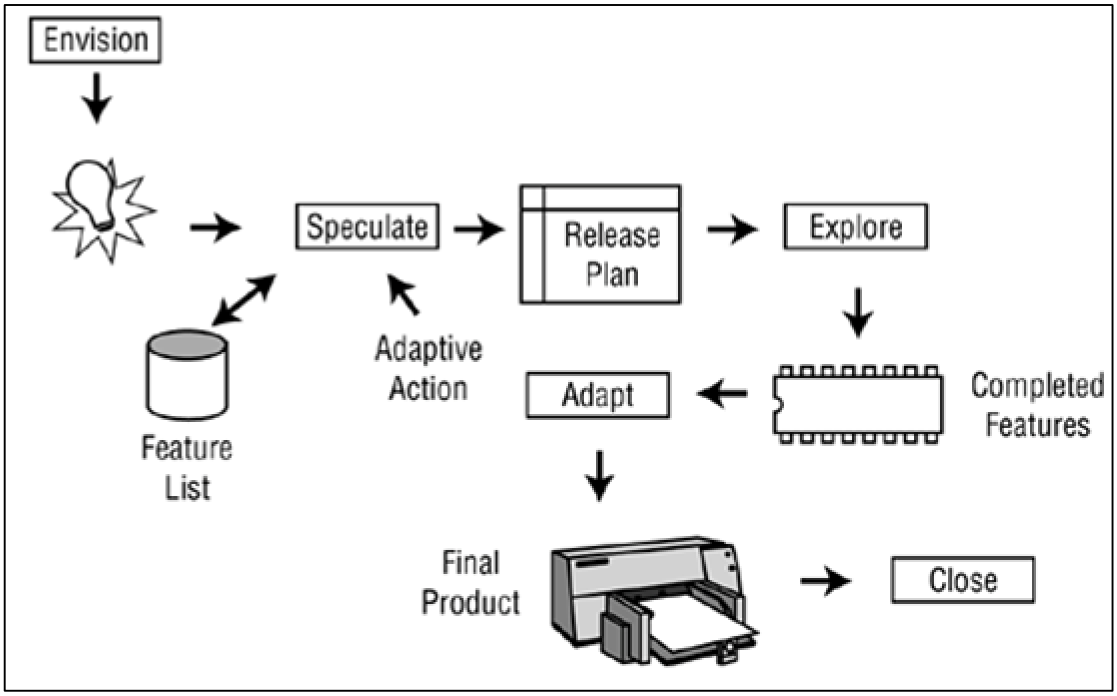


Figure 9: APM delivery framework (Highsmith, 2009)

APM is made up of five phases, namely: envision, speculate, explore, adapt and close. The APM delivery framework doesn't make use of the traditional phase names such as initiate, plan, execute, develop, test, etc. This has a great implication, as it shows that APM accepts change as frequently as required. The traditional initiation phase is substituted by the envision phase to indicate the importance of a vision.

The customary planning phase, which is normally connected with prediction and comparative certainty, is substituted by the speculate phase in order to emphasise the uncertainty of the future and the need for flexibility, as it is impossible to predict the end result (Highsmith, 2009) Many traditional project managers confronted by uncertainty attempt to plan the uncertainty away. However it is important to learn to take risks and adapt rather than plan and build (Highsmith, 2009).

The build, design and test phases are substituted by the explore phase, which is a non-linear and non-waterfall model that is centred on iterative development. In this phase, the uncertainties that were named in the speculate phase are examined, solved and explored.

The adapt phase makes sure that the team members stay focused on the vision and adjust to the current environment. During the last phase, close-out, the project is handed over and knowledge is transferred.

Envision

Any good leader emphasises the significance of a vision. For this reason, APM describes the project's vision in the first phase to make sure that everyone concerned comprehends what has to be achieved and by when in order to decide whether the project has been delivered successfully. The cycle of the envision phase as presented in Figure 10 operates closely with the cycle of the explore phase.

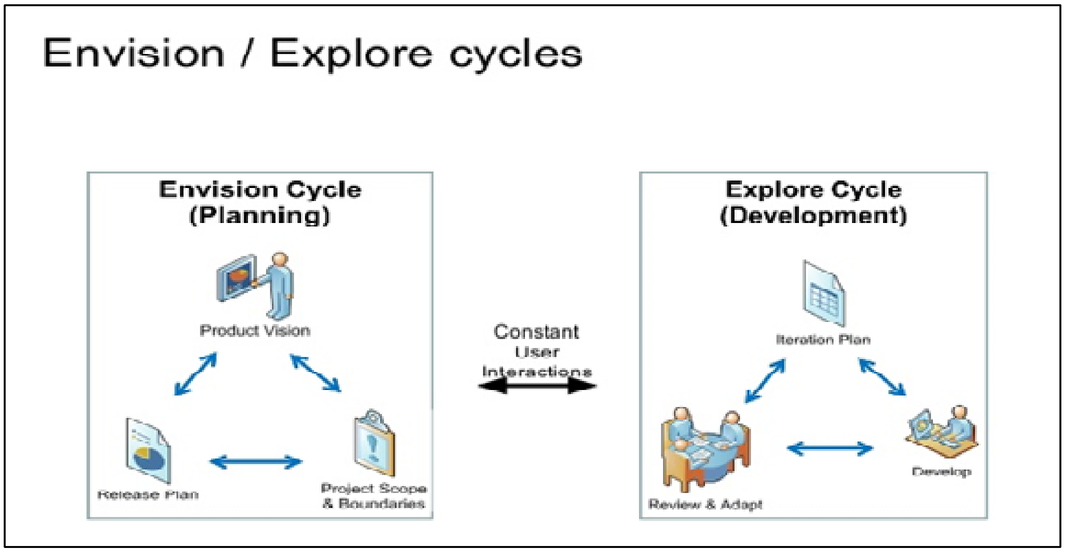

Envision / Explore cycles
Envision Cycle (Planning)
Product Vision
Release Plan
Project Scope & Boundaries
Constant User Interactions
Explore Cycle (Development)
Iteration Plan
Review & Adapt
Develop

Figure 10: Envision/ Explore cycles (Highsmith, 2009)

During the envision phase, the project scope, project community and stakeholders, product vision, and the way in which the team works together are established. Firstly, the team must envisage what must be delivered. In order to do this, the project objectives, constraints, boundaries and vision must be outlined and understood. Secondly, the team has to ascertain which stakeholders and members of the business community will be engaged. Lastly, the team must choose the method in which they will work together in order to deliver the anticipated and outlined vision of the project.

Release planning (as seen in Figures 9 and 10) is typically done throughout the speculate phase. The following questions are answered in this phase (Highsmith, 2009):

- What is the client's product vision?
- What are the key capabilities required of the product?
- What are the project's business objectives?
- What are the project's quality objectives?
- What are the project's constraints (scope, schedule and cost)?
- Who are the right participants to include in the project community?
- How will the team deliver the product (approach)?

In smaller IT projects, the envision and speculate phases can potentially be done during the project kick-off meeting when the whole project team is in attendance. In larger IT projects, some project activities, such as procurement and training, can be incorporated in the first iteration (also called iteration 0) of the speculate phase.

According to (Highsmith, 2009), it is imperative to remember that:

- The team members should continuously regulate the bare necessities of the process and documentation needed.
- All the practices associated with the way in which a team delivers are adapted to enhance performance as the project advances.

- The project community will advance its methods where everyone works together.

Crafting a vision ensures that everyone linked to the project works towards the same goal as a whole. It is also imperative that each team member has a clear understanding of his/her role in each of the outlined project goals, or else they will disregard their responsibilities and this may lead to the project failing.
The envision phase has four categories of practices:

1. “Product vision:
   - product vision box and elevator test statement;
   - product skeleton architecture and guiding principles;
2. Project objectives and constraints:
   - project data sheet;
3. Project community:
   - obtaining the right people;
   - participant identification;
   - client team–development team interface;
4. Approach:
   - process and practice tailoring.”

Speculate

Envisaging what will occur in the future in the course of IT projects may end up being a costly exercise because of the changes in the project environment. As soon as clients get a view of what can be accomplished, they always wish for more. For this reason, it is better to speculate than to plan because the future is unknown, keeping in mind that the speculation is based on what is already known. The term “speculation” may appear to indicate irresponsible risk taking, while it actually means estimating project expectations based on incomplete facts and information (Highsmith, 2009). APM is more than just planning and doing – it is about crafting a vision and exploring it because only some information is available and this must be scrutinised to determine the course of action in the next iteration (Highsmith, 2009).

During the speculate phase, planning is not overlooked, however speculating is a more suitable term given the existence of uncertainty (Highsmith, 2009). Planning in the business world is a worn-out word and suggests that planning can surmount uncertainty. Planning cannot surmount uncertainty however; rather, project managers and teams can speculate and adjust according to their project's conditions and expectations. The speculate phase involves (Highsmith, 2009)

- gathering the main client requirements for the product;
- outlining the volume of work as a backlog of product features;
- developing an iterative release plan that is feature based;
- incorporating risk mitigation strategies into the plan; and
- assessing the cost of the project and gathering other essential administrative information.

The speculate phase delivers a release plan, which is not centred on activities like traditional IT projects, but on stories, like XP. During this phase, the product organisation, backlog of stories and competences, and the release plan are well-defined, comprehended and produced (Highsmith, 2009).

Feature (time-boxed) planning and development are performed by involving the client through clarifying the entire process of releasing the product(s) and project. Change is also managed more easily when clients are involved because they appreciate the difficulties and limitations faced by the project team. Features, which are presented in the form of story cards, are used as a method of communication for collective understanding between the development team and the client. The features are then broken down for development and implementation.

During the envision phase, a product breakdown structure is produced by identifying and recording the features needed to release the product. During the speculate phase, the list of features is expanded by creating story card(s) for each feature, where the front of the story card contains requirements for planning purposes and the back contains technical information that assists team members in calculating the time and effort needed to deliver on the client's requirements. The story cards are then examined in greater detail to be developed and tested within a specific planned iteration during the explore phase.

When listing and expanding the features, it is imperative to consider the vision, aims and objectives of the project, which is why the release plan is so crucial. The release plan symbolises a roadmap of how the team aims at achieving the product vision within the project objectives and constraints (Highsmith, 2009). Agile project speculation helps the project team in (Highsmith, 2009):

- Defining the way in which the product and its features will develop in the current release;
- Balancing expectation with adjustment as the project develops;
- Concentrating on the highest-value features early in the project;
- Considering the business goals, project objectives and client expectations;
- Offering the necessary cost and schedule information to management;
- Establishing priorities for trade-off decisions;
- Organising interrelated activities and features across teams;
- Reflecting on options and adaptive actions; and
- Delivering a base-line for evaluating events that take place during the project.

Explore

This phase cannot be compared to an explorer discovering the unknown, because the vision of what the project team must achieve without reckless risks is outlined. During the explore phase, stories (features) are planned, developed, tested and delivered in small iterations while the objective is to continuously decrease the uncertainty and risk of the project. In Figure 10, the release plan implemented during the envision phase relates to the iteration plan of the explore cycle. The figure also depicts three main activities with sub-activities that operate in conjunction with the following activity areas in this phase (Highsmith, 2009):

- Planning and delivering stories by management of the workload and using suitable risk mitigation strategies and technical practices;
- Aiding and crafting a self-organised and collaborative project environment; and
- Managing communication and collaboration amongst stakeholders, clients and product management.

The first major activity is integration planning and monitoring, which involves iteration planning, workload management, and monitoring the iteration progress – which is managed by the iterations manager. Iteration planning is done by the team by making use of each story card in the release plan to plan for the first or next iteration(s). The tasks to complete the listed story cards for the iteration that must be executed are listed, after which the team re-estimates the work effort and reprioritises the story cards where necessary. The workload is administered by each team member, as he/she is responsible for delivering the stories when the iteration is complete. The progress of a specific iteration is supervised on a daily basis using the daily stand-up meetings, as is the case with Scrum.

The second major activity is technical practices, which entails simple design, continuous integration, laborious automated testing, and opportunistic refactoring of which most are specific to the product engineering domain – in order to keep costs of change low and to provide a product that is of high quality. The four technical practices work in concert with each other and are critical for the effective adaptability of a project and the project team. The technical division is brought onto a project when a project team is under pressure to complete stories (deliverables). This decreases the speed of development and it negatively influences the team's capability to deliver. The aim of simple design is to keep the team focused on what they know, instead of anticipating what they do not know. There are two essential approaches to managing change (Highsmith, 2009):

- Anticipation: Plan for the future and predict what types of changes are possible.
- Adaptation: Wait until requirements or development issues have been defined and build them into the product.

Continuous integration at the opening stages of the project and during development is intended at ensuring that the product features (stories) are unified as a whole to lessen the load of testing and the high cost of misalignment. The less frequent the iterations, the more prone the development effort will be to major setbacks late in the process and the more difficult, and expensive, it will be to find and fix them (Highsmith, 2009).The aim of laborious automated testing is to make sure that the quality of the product continuously remains high during the development process. A team will be more efficient if it has tested and running features for each iteration. During opportunistic refactoring, the project team concentrates on repetitive and continuous improvement, as the product is intended to make it more adaptable in order for the team to release features on a continual basis. The third major activity relates to the project community. It involves coaching and team development, hands-on decision-making, collaboration and coordination. During coaching and team-building, the team explores, experiments and learns from the mistakes they make. The agile project manager's aim is to construct a high performing and focused team by realising each team member's individual talents. The aim of hands-on decision-making is to get the team members and stakeholders involved and to offer project practices to contribute to decision-making as the project proceeds. In order ensure that collaboration and coordination between stakeholders takes place during a project, practices such as stand-up meetings (Scrum), stakeholder coordination and daily interaction with the project team are applied.

Adapt

Adapt or die is a familiar expression, which means that if something does not adapt to its environment it will not survive; this principle is relevant to IT projects. According to (Highsmith, 2009) adapt suggests adjustment or change rather than success or failure. In order to adapt, there must be an understanding of the risks, changing requirements, project processes and the market. Every agile project team must assess and adapt in the following areas (Highsmith, 2009):

- Product value;
- Product quality;
- Team performance; and
- Project status.

The term "corrective action" is frequently used between teams and project managers. This term suggests that the team has underperformed or made an error and that action must be taken in order to re-align the project with the project plan, which in turn indicates that the project plan is always correct. APM replaces the term with "adaptive action". This term denotes that the team responds to events or incidents (instead of trying to correct) because the project plan was initially developed based on assumptions and speculations. This follows the ASD Manifesto's value of responding to change rather than following a plan. The team has to answer four challenging questions, as it is more difficult to adapt to events than correct a project plan (Highsmith, 2009)

- Is value, in the form of a releasable product, being delivered?
- Is the quality goal of building a reliable, adaptable product being met?
- Is the project progressing satisfactorily within acceptable constraints?
- Is the team adapting effectively to changes imposed by management, clients or technology?

In the adapt phase, the accomplishments of an iteration and project are evaluated by the project team and stakeholders, such as the client. The feedback offered gives information on the actual versus the predicted time, cost and quality estimates, and up-to-date project progress and status information. The lessons learnt and output are utilised during the next iteration's re-planning effort. Performing review and adaptive action sessions at the end of an iteration is vital, as this allows the team to reflect, learn and adapt where necessary, and to create a change in pace and urgency in executing the project iterations. After creating a vision in the envision phase, the speculate, explore and adapt phases loop for each iteration to perfect the product in development. There may be occasions in which the vision of a project in the envision phase must be revised, as new and changing requirements or information is gathered that may influence the direction the project should take.

Close

Closing out a project is often perceived as a time-consuming phase with not much value to the client, even though it is in fact just as important as the other phases of the project. By putting together a closeout report, the project team summarises what has been carried out successfully and finally agrees that the expectations of the client have been met. Even if a project initiates another project, it is still imperative to obtain sign-off and close-out of what has been successfully completed to warrant that the client does not later complain that what has been done is actually not what they expected or that the project is not working because the previous project's expectations were not met. The main aim of this phase, and the small close-outs of each iteration, is the lessons learnt, which must be used for the planning and execution of the next iteration or project. When closing out a project, it is a good idea to have a celebration and short award ceremony.

Evaluation of Agile Project Management

Grey (2011) believes APM is still relatively new in relation to PRINCE2 and PMBOK, which have been established and enhanced over the last 20 years. As APM is so new, it is challenging to do research on where it has been used in practice effectively, what the advantages and disadvantage are, or whether it can be integrated with other methodologies (Grey, 2011). In addition, Highsmith (2009) admits that the APM is not a complete life cycle and still needs to go though necessary revisions for improvement.

Application Area

According to Puri (2009), there is a serious need for agile projects to be managed because there is a view that when one uses ASDMs, one does not require PM. In today's fast-paced and ever-changing business environment that necessitates a PM methodologies that can adjust, business and project managers have begun to apply APM for much more than just software development Moore (2010). Instead of using TPM Methodologies, new PM approaches should consist of the four major values of APM and ASDMs (Rico, 2010):

- Client collaboration;
- Iterative development;
- Self-organising teams; and
- Adaptability to change.

APM grants the opportunity to release diverse kinds and different sized projects across various industries because of its adaptive nature. It is acclaimed by many CEOs, directors and academics from corporations and institutions of higher education across the world (Highsmith, 2009), which proves that this PM Methodology has something of value to offer.

Advantages

The benefit of APM is that it can be used to manage dynamic projects with a high level of uncertainty, urgency and complexity more effectively (Hass, 2010). In these kinds of projects, there is a need to amalgamate the roles of a business analyst and a project manager into one role (Wysocki, 2011). APM presents the following advantages (Virine & Trumper, 2009):

- A creative environment is created that is free of frustrated developers;
- To seek and determine preventative and corrective actions for project risks; and
- To manage adaptively and to review every decision made.

In the midst of organisational turmoil, APM has the capability to survive by constantly adapting to the changes in the project environment. APM acknowledges that change is unavoidable and that change must not be avoided but managed and controlled, whereas traditional project management approaches have an objective of reducing change by comprehensively planning and documenting as much as possible before development begins.

APM is lean, flexible, easy to use, and individuals do not have to go through rigorous training to utilise it correctly. Furthermore, APM is unlike the other PM approaches, as it allocates iteration managers to support project managers by managing the internal team members and the individual iterations.

The three core values described by Highsmith (2009) revolutionise the approach towards project management, according to which the project manager has to adapt, lead the team and deliver value, rather than following a plan, managing tasks and meeting constraints, the aspects focused on by all TPM Methodologies. The aim of the simple design is to ensure that the team is focused on what they know, rather than anticipating what they do not know. The vision is clarified and improved on a regular basis as the project proceeds. APM is differentiated by speculation and iterative development – instead of planning. The project team speculates based on what they already know, but planning is not ignored. It just accepts that the future is unknown, while other PM methodologies seek to avoid uncertainty through planning. APM is the only PM approach studied that incorporates short daily meetings, like Scrum, in order to make sure that there is regular stakeholder and team member communication and problem resolution. Regular and continuous testing is embedded throughout the PM Methodology to ensure continuous, high-quality product delivery.

Disadvantages

Agile Project Management is formed on the same fundamentals as ASDMs, which subjects it to the same atmosphere of criticism and evaluation. Many of APM's components are similar to those of XP and Scrum, which makes it susceptible in the sense that if someone or an organisation is not in favour of XP or Scrum as ASDMs they may reject as a whole (Grey, 2011).

It still needs to go through comprehensive evaluation and an enhancement process in order for it to be established as a PM Methodology that will be accepted globally, like PMBOK and PRINCE2 (Grey, 2011).

It also lacks particular management areas that are vital for good project management, such as human resource management, issue management and procurement management. One focus of APM is to keep documentation to a minimum, which might cause problems, especially in a high-tech project environment in which documentation is viewed as the main reference source for any information or decisions to be made. However, if documentation is not viewed as important, this may cause confusion and misalignment owing to team members not being sufficiently guided by documentation. For this reason, there needs to be a balance between the importance of documentation and the volume of documentation. Documentation must still be viewed as very important because it is used as the benchmarking mechanism in many organisations to govern the execution of a project. The only documentation that must be avoided is documentation that will never be read (Grey, 2011).

APM is developed to be executed at a technical level, the project manager must thus be careful not to focus only on individual iterations and must bear the project's vision in mind continuously. It does not give much attention to setting up a business case, which makes evaluation at project close-out a problem. When the project is evaluated to determine whether it was successful, the business case should be used as a base-line to check whether the client's expectations have been satisfied. If the business case is included in the vision, then benefit realisation might be more accurate and effective (Grey, 2011).

2.3.4 Existing frameworks implementing project management principles in agile

Integration of PRINCE2 and Scrum framework

Tomanek *et al.* (2015) present a conceptual framework that integrates principles and processes from PRINCE2 and Scrum. The conceptual framework consists of the alignment of principles and processes. The integrated process model is depicted in Figure 11 below

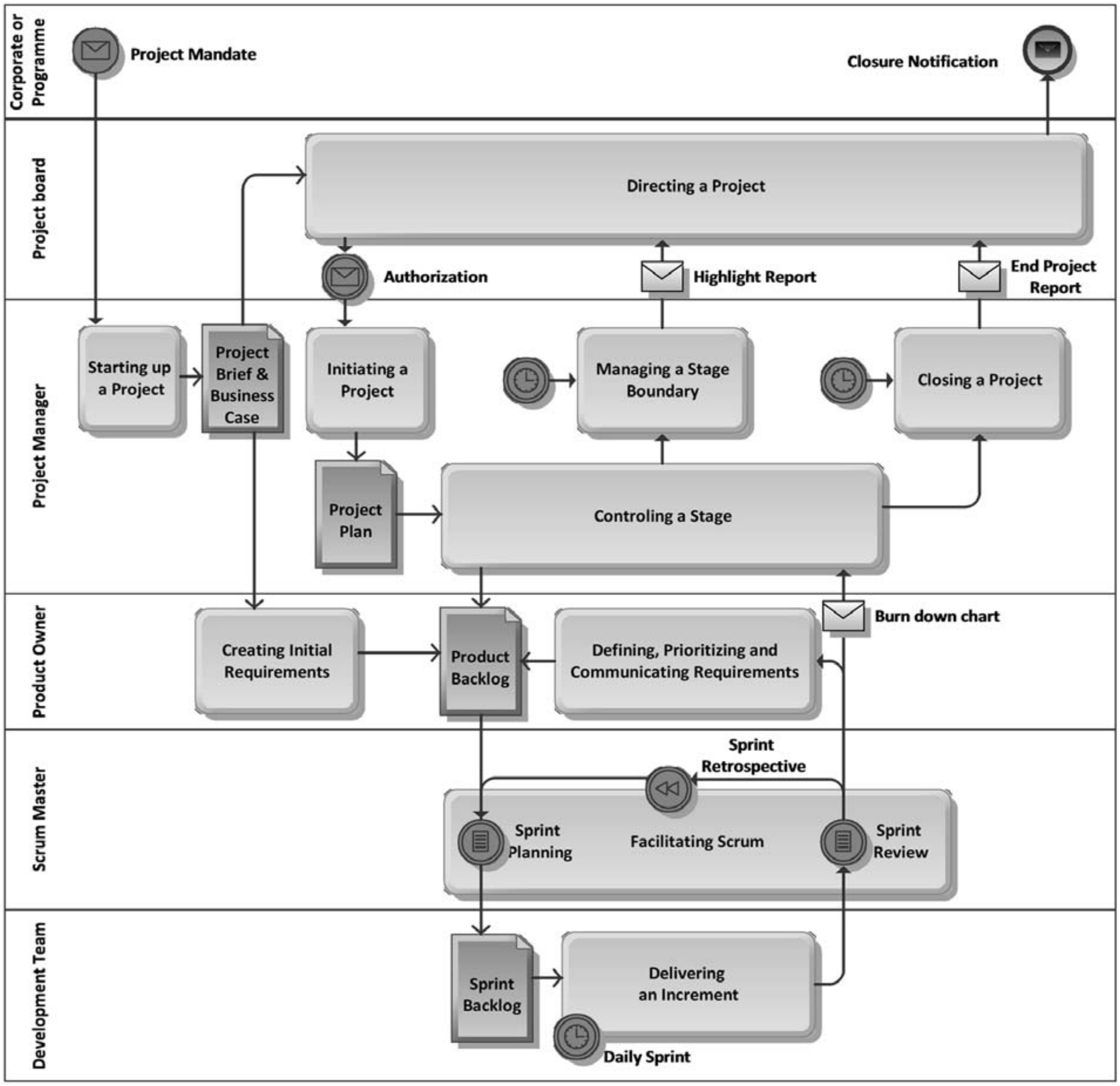


Figure 11: Integration of Scrum and PRINCE2 Conceptual framework (Tomanek *et al.*, 2015)

Tomanek *et al.* (2015) believe that the Scrum development process aligns into the PRINCE2 process framework. They propose that Scrum as the development framework can substitute the managing delivery process defined by PRINCE2. In this way PRINCE2 covers the delivery process of the project products. Scrum offers the benefit of providing guidance on how to develop the product in an effective manner and adjust to the changing environment. The project commences as any other project with the project manager compiling the requirements and expected costs. In case where the product owner exists in the current line organization then the project manager would collaborate with him or her to validate and clarify the requirements (Tomanek *et al.*, 2015). The project manager also designs and assigns the PM team. Expected project benefits together with estimated costs are clearly stated in the business case and presented to the project board. The project board can then decide if the project should be initiated and authorize project execution (Tomanek *et al.*, 2015).

When the project has been authorised to begin, the project manager plans the following steps in greater detail. The project manager plans the phases and associated sprints. In the interim the product owner should transfer the initial requirements from the business case into the product backlog .At this stage the scrum master can be selected and together with the project manager should facilitate the first sprint and to have the first sprint planning event (Tomanek *et al.*, 2015).

The product owner with the development team discusses the product backlog and agrees on the requirements that can be released in the next sprint. All selected requirements are moved to the sprint backlog and defined in further detail by the development team. When the sprint backlog is created, the team starts planning, developing and testing the product increment. They meet daily and during 15 minutes they discuss what has been done, what will be achieved quickly and whether they face any barriers. These daily stand up meetings are facilitated by the scrum master and if the product owner is available then the product owner should be present at these daily sprints as well (Tomanek *et al.*, 2015).

When the delivery portion of scrum is completed, the development team displays the product increment to the Product owner in the sprint review meeting. The product owner can decide if the product should be released to production or further developed. After the sprint review meeting, the product owner can explore the product increment, create new requirements, update the existing ones and prioritize all of them in the product backlog (Tomanek *et al.*, 2015).

Hybrid Agile Project Management Methodology (Grey, 2011)

Grey (2011) developed a hybrid PM methodology by combining the strengths, addressing the weaknesses and bridging the gaps of the ASDMs and TPM methodologies.

This hybrid framework which is presented in Figure 12 below is made up of seven phases: pre initiation phase, vision and definition phase, preparation phase, collaborative development phase, close out and handover phase, adapt, direct, monitor and control phase (Grey, 2011).

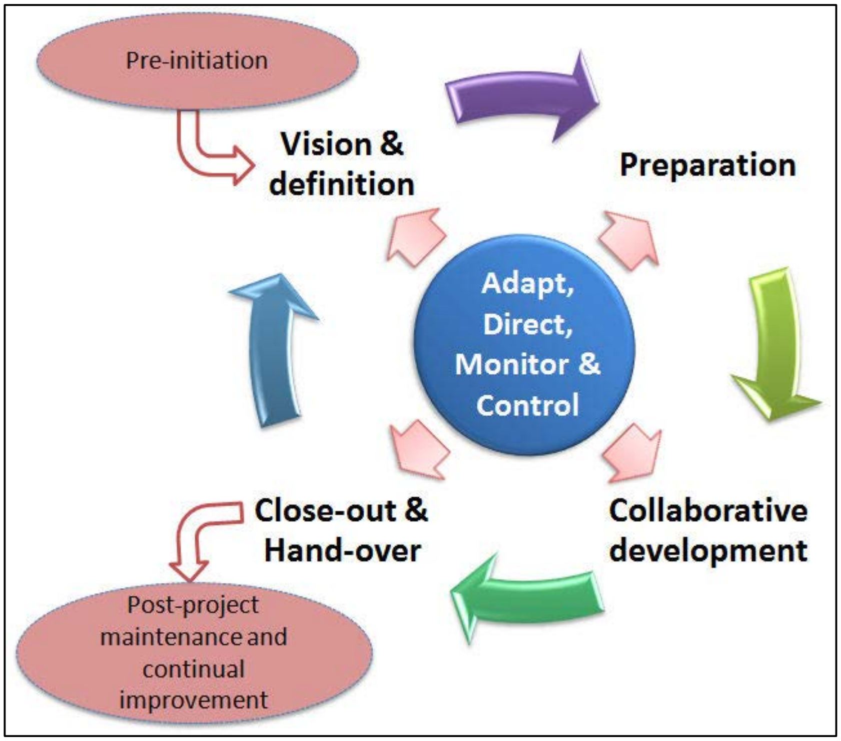

Pre-initiation
Vision & definition
Preparation
Adapt, Direct, Monitor & Control
Collaborative development
Close-out & Hand-over
Post-project maintenance and continual improvement

Figure 12: Hybrid Agile Project Management Methodology (Grey, 2011)

Pre-initiation phase

This phase makes sure that everything is ready for the project to be commenced effectively (Grey, 2011). The key objectives of this phase some of which were drawn from PRINCE2, include the following (Grey, 2011):

- The commencement of the project is defended in a business case that outlines a preliminary project scope, or a preliminary feasibility study is done to ensure that it would be practical to initiate the project.
- The required members of senior management and stakeholders are assigned to offer approval on collaborative increments completed.
- The various ways in which the project can be developed are assessed and the method to be used to develop the project collaboratively is identified.

- People are selected to be part of the project team or to be the project manager who will have the responsibility of executing the work required as agreed on in the project's vision and definition. When individuals are selected from different areas within the organisation, it is essential to make sure that team members are allocated appropriately and that the time and duration of their allocation to a project are communicated to rest of the organisation. This will make sure that resources are not over-utilised within an organisation. The project manager and team members selected must have applicable skills and knowledge to perform their tasks.
- The decision is taken concerning whether the project is to begin or not. Time is not spent on starting a project that is based on variable assumptions and expectations regarding scope, time and cost constraints, and acceptance criteria. The project must only start when senior management approve of the project as being important and they must give the reassurance and commitment to providing full support and guidance for the duration of the project.
- The tasks to be completed during the vision and definition phase are speculated on.

The project with the highest priority within an organisation should be the project that must be implemented next. This selection and prioritisation of projects is typically done in the portfolio management function of an organisation, where management confirms that the organisation's strategic objectives are met by the prioritised projects that must be implemented (Grey, 2011). This is done to make sure that the projects in implementation are aligned with the strategic direction of the organisation. The business case or feasibility study will then be used as a base-line to develop the project charter in the vision and definition phase (Grey, 2011).

<u>Vision and definition phase</u>

This phase is an integration of the initiation process of PMBOK, the IP process of PRINCE2, and the envision phase of APM (Grey, 2011). The vision makes certain that each person that is involved in the project knows the objectives and his/her responsibilities in order for everyone to work towards the same objective as a whole. During this phase, the project vision, preliminary scope, community and collaborative development approach are established. The key objectives of this phase include the following (Grey, 2011):

- Stating the project's vision
- Outlining the essential key competencies and objectives
- The benefits expected, related risks, issues and the reasons for executing the deliverables must be defined
- Speculating on the time-frame and cost of developing the goals of the project collaboratively
- Defining who takes key decisions and provides approval
- Defining the method in which quality, risks and progress will be measured, controlled and tracked
- Outlining an effective communication structure and levels of authority; adapting this model further to fit the project environment;
- Speculating on the activities to be done during the preparation phase;
- Inspiring the team to make certain that the objectives and capabilities required are implemented successfully;
- Make sure that the vision of the project is aligned with the vision of the organisation (where the project's vision is not aligned with the vision of the organisation, it must be reconsidered and adjusted to ensure that it speak to the expectations of the organisation as well.)

In order to observe the principle of documenting as little as possible and instead focusing on carrying out the work, it is crucial to continuously define the basic fundamentals of the process and documentation required. There must be an emphasis on maintaining simplicity and avoiding spending time on developing long preliminary project plans and schedules, which will probably change anyway (Grey, 2011).

The document that will be delivered by this phase is the project charter, which will make use of the business case, or the selected solution in the feasibility study, which was completed in the preinitiation phase, together with the aforementioned key objectives, as the base-line (Grey, 2011). In most cases, the same document used for the business case can just be updated after the vision and project's definition have been clarified. The document will justify the project, put major processes of understanding in place, and identify the main stakeholders for the project. Furthermore, it will define the project's focus, major objectives and deliverables and state by whom they must be delivered within the associated deliverable cost, time and quality constraints (Grey, 2011).

Preparation phase

The key objectives of this phase include (Grey, 2011):

- Gathering as many requirements as possible beforehand;
- Defining the workload by grouping and prioritising deliverables, deliverables sets and
- Incremental release plans;
- Developing an incremental release plan and project schedule guideline;
- Incorporating risk mitigation and issue resolution strategies into the plan; and
- Speculating on and estimating project costs, timelines, roles and execution responsibilities.

The project charter that covers the vision and definition of the project is used to collect as many user and business requirements as possible in advance. The reason that as many requirements as possible must be collected prior to collaborative development taking place is to reduce the chance of new and changing requirements. It also ensures that planning and speculation are done based on more known requirements and facts than unknown requirements, which will lead to a more precise project schedule guideline (Grey, 2011). The project schedule guideline is made up of assumptions and speculations that will result in the prioritised speculated release backlog, which will be the beginning of the collaborative development phase (Grey, 2011).

Requirements are collected through workshops, JAD sessions (as with ASD), questionnaires or story cards (as with XP). The requirements are then analysed and clarified as deliverables and activities to be done in order to fulfil the required client expectations. Deliverables and activities are categorised, grouped and prioritised to create the release plan (Grey, 2011). The defined deliverables and activities are assembled into logical deliverables sets after the deliverables have been prioritised based on their level of importance. The deliverables sets are then grouped into incremental release plans after they have been prioritised based on their level of importance. The incremental release plans are then also prioritised after which the one with the highest priority is selected to be executed as the next collaborative increment in the collaborative development phase. Each collaborative increment is time-boxed, as with ASDM (Grey, 2011).

In order to adapt to the different levels of complexity in projects, one deliverable can be completed as part of a deliverables set, and one deliverables set can even be completed as an incremental release plan. This means that one deliverable can be viewed as an incremental release plan in small projects. After grouping and prioritising activities, deliverables, deliverables sets and incremental release plans, the release plan is used to create the project schedule guideline (Grey, 2011). The project schedule guideline will contain the typical Work Breakdown Structure of who does what, how much each deliverable will cost and by when a certain deliverable must be completed. The scheduled guideline will be used to guide the project as a whole by keeping the vision of the project in mind. Some deliverable dates, deliverables as a whole, and persons responsible for completing deliverables will change as the project progresses. For this reason, it is not called a scheduled plan but a scheduled guideline. The scheduled guideline will then be used as the speculated release backlog in the collaborative development phase (Grey, 2011).

Customers are involved during every step of the preparation phase to ensure that the team understand the requirements and expectations, and to ensure the client understand the threats and constraints the project team might be facing.The collaborative phase is directly dependant on the preparation phase (Grey, 2011).

Collaborative development phase

According to Grey (2011) this phase was named collaborative development because the word “collaborative” suggests that teamwork, communication and continuous stakeholder involvement are essential to ensure a successful development and delivery of each collaborative increment and the project as a whole. Clients are engaged and consulted at every step, and preparation is integrated throughout the collaborative development phase.

The collaborative development phase begins with the speculated release backlog, which contains prioritised incremental release plans. When an incremental release plan with the highest priority is selected, it becomes a collaborative increment, which will be executed and signed off to release a new functionality of value that satisfies client objectives and expectations (Grey, 2011).

Every collaborative increment is initiated by a kick-off meeting. Everyone participating in the execution and approval of the deliverables set(s) and collaborative increment is requested to attend this meeting so that it can be ensured that effective incremental development preparation is done by describing the method that will be used to execute the deliverables, who will be responsible and by when the execution needs to be done. An environmental assessment is also done to make sure that the project is up to date on the latest technologies and trends. This will ensure that the organisation continues to be competitive and benefits from using the latest technologies (Grey, 2011).

When the incremental development preparation has been completed, each deliverables set is executed using the iterative refactoring cycle, whereby the design is developed until it is of good quality and value. The iterative refactoring cycle can be done repeatedly for a specific deliverables set, or for every deliverable in the deliverables set. More than one deliverable can thus run concurrently through the cycle. The cycle begins with a speculate and analyse step to ensure that if any changes occurred they are taken into consideration while the project has progressed. The necessary analyses are also done prior to the deliverable(s) being physically created (or designed) (Grey, 2011).

The deliverable(s) is evaluated and tested after the create step has been completed. During this step the project manager and project team members establish whether the agreements that were reached have been fulfilled and whether the users' expectations have been met. If the deliverable(s) does not meet the acceptance and quality criteria, the deliverable(s) is improved during the improve and adapt step and tested again during the test and evaluate step. Even if the deliverable passed the test and evaluate phase the first time, it is crucial to verify that there is no room for improvement within the time and cost constraints so as to deliver an even better deliverable(s) that would ensure absolute client satisfaction (Grey, 2011). During these steps, the distinctive features (defined in the pre-initiation phase) of the project must also be reviewed to ensure that the nature of the project has not altered. In circumstances where an extreme change has to be made to the deliverable(s) because of the project's nature that changed, or due to new or changing requirements, an adaptive action must be executed to rectify the problem by including the new or changed requirement(s) in the create step, which must be executed again together with the remainder of the step in the iterative refactoring cycle (Grey, 2011). The improve and adapt phase also ensures that functionalities are included by the project team in a deliverable(s) that clients have not identified and that may result in a failed deliverable(s) (Grey, 2011).

After the completion of the improve and adapt step, the deliverable(s) are reviewed in the benefit realisation step to determine whether the value expected in accordance with the project schedule guideline and the client expectations has been realised. In many projects, deliverables are completed successfully but they do not bring value to the organisation or to the project as a whole (Grey, 2011). The benefit realisation step in the iterative refactoring cycle makes sure that functionalities of good quality and value are delivered during the release new functionality step. This will ensure that the project team does not only releases products quickly, but also releases products of quality as quickly as possible. The functionalities can be delivered in the form of prototypes (Grey, 2011).

Learning is augmented in the iterative refactoring cycle because as deliverables advance through the different cyclical steps many lessons are learnt that must be recorded on a regular basis. In the learn step, the project team can rely on prior experiences and solutions to problems to make sure that the same mistakes are not repeated during the execution of the next deliverables set (Grey, 2011) .

.

During the iterative refactoring cycle, short daily stand-up meetings are held. The reason for holding stand-up meetings is to avoid people discussing matters that are not urgent or relevant to the project when they sit down. During this meeting, problems and project changes are discussed, quick solutions are found, and support is provided by the project team if necessary. Furthermore, it guarantees that the team members will stay focused on the vision and the task at hand. Issues and risks are quickly resolved to ensure that any negative impact on the project is minimised (Grey, 2011) .

After the completion of the deliverables sets, a post-incremental meeting (derived from Scrum's post-sprint meeting) is conducted, in which the project team, clients and other stakeholders go through a final benefit realisation exercise after all the deliverables sets have been completed within a specific collaborative increment (Grey, 2011).

After the successful completion and sign-off of each collaborative increment, a full (system) implementation step can be executed (only if necessary) to integrate the different collaborative increments in order to release them as a whole into the organisation (Grey, 2011).

The system and processes are then handed over to the production environment in the close-out and hand-over phase, after which the system must be maintained and continually improved in the post-project maintenance and continual improvement phase(Grey, 2011) .

Close-out and handover phase

The close-out and hand-over phase is as crucial as the other phases of the project. It is not a remote phase, as it can be used in any phase of this model (Grey, 2011). It can also be applied when closing a project down ahead of time for whatever reason. In this phase, the project is officially closed by delivering to the client all the signed-off deliverables, deliverables sets, collaborative increments and final implementation sign-off documentation (if relevant). According to Grey (2011) the reason for attaining formal closure is to ensure that the client or stakeholders do not later say that certain expectations or functionalities were not provided. Additionally, the final results of the project must be considered against the vision, value and benefits that were committed to in the project schedule guideline and project charter (Grey, 2011).

Adapt, direct, monitor and control phase

This phase assists in integration, which happens throughout the various stages of the project (Grey, 2011). The stakeholders, management, sponsor or project board (if appointed) administers by placing particular project controls in place in order to make cognisant decisions grounded on facts and not assumptions. They have the ability to offer ad hoc guidance as the project advances. A culture of "respect for all" must be nurtured together with the project manager, who must constantly monitor, control and address any political or cultural issues that might arise as the project progresses. Human behaviour could also be considered, if there is enough time, to support the management of political issues and cultural behaviours (Grey, 2011).

Like in PRINCE2, the stakeholders, management, sponsor or project board (if appointed) can assign project or organisational assurance to execute some of the evaluating and reviewing functionality. They lead the project by considering exclusions as highlighted by the project manager, who plays the role of a communication channel by communicating all relevant project and progress information to programme or organisational management (Grey, 2011).

In order to successfully control and monitor a project, it is essential to account on the progress of a project in order to identify possible problems and issues that arose, and to identify adaptive actions before they become a threat to the successful delivery of a project. The escalation process of informing the stakeholders, management and sponsor or project board of identified risks, issues and changes to the project must also be formalised during this process (Grey, 2011).

Post-project maintenance and continual improvement phase

There is always room for improvement to a solution, which is the reason why it is vital to monitor continually whether enhancement or an upgrade to the current implemented solution is necessary(Grey, 2011) . As soon as users get a view what is possible, they always desire more. For this reason, there are new releases of software and hardware products in the market to offer quicker and better solutions for clients. The proposed hybrid APMM (ver. 0) life cycle is then re-initiated by the vision and definition phase if an improvement or upgrade is required. If pre-initiation planning is required, it can be done before the vision and definition phase is executed (Grey, 2011) .

2.4 Summary of the Chapter

A view on the historical progression of PM shows that development projects have been part of society even before the notion of PM was formalized. It also shows that as time has progressed PM was born out of the research that was continuously taking place. Down through the years the PM tools and methods were introduced and standardized. Today projects make up the strategic management of businesses in building and maintaining competitive advantages. The project portfolio of organizations consists of initiatives associated with the pursuit of innovation and efficiency, which includes various levels of complexity and uncertainty. Given the recognized poor performance of projects, generally intensified by the growing number of complex and uncertain projects, new approaches to PM emerged in contrast to the conventional approach. The major reason behind the development of new approaches to PM has been the criticism directed towards the traditional approach to PM. One of the criticisms of the traditional approach to PM is that this method is not able to manage modern day projects that exist in competitive, fast moving global markets. A number of constraints were introduced earlier. One of these is the scope constraint that is severely affecting the practice of PM. The demeanour of ASDM to allow changes in scope is proving to be quite a challenge for traditional PM. This study is an attempt to explore possible ways in which the benefits of PM can be maintained whilst still working within the agile framework of SD. In Chapter 3, the theoretical frameworks underpinning the current study will be discussed.

3.0 THEORETICAL FRAMEWORK

3.1 Introduction

The theoretical framework of a research project conveys the theoretical foundation on which the research is based, and establishes a link between the theoretical components and practical aspects of the research undertaken (Abraham, 2008). The theoretical framework, therefore holds repercussions for all the decisions taken during the research process (Mertens, 2014). According to Crotty (1998) the initial phase when developing a research proposal is to identify the methods and methodologies that will be employed in the research project and to justify that choice. The methodologies are concerned with the approach, plan of action, procedure or research design that enables the attainment of the anticipated research outcomes (Crotty, 1998). Therefore, it is imperative that the method selected is compatible with the objectives of the research project. From a broader perspective, the rationale behind the choice of methodologies and methods goes further than responding to the research questions. This rationalisation relates to identifying the fundamental suppositions about reality and considerations of human knowledge that the researcher takes into the research and the theoretical perceptions which lie beneath the chosen methodology. The theoretical framework, therefore reveals the methods, methodology, theoretical perspective and the nature and limits of human knowledge underpinning the research (Crotty, 1998).

3.2 A viable theoretical framework

The researcher identified the need for a theoretical framework to underpin the different aspects of the study. A close examination of the research objectives reveals a dualistic nature inherent in the study. The first aspect of the study entails an elicitation of practitioners' perspectives on the use of agile methodologies (AM) and project management (PM) at Bank A. In order to achieve this objective, a theoretical framework is required to guide the accurate gathering of this knowledge. The second aspect of the study consists of the development of a framework that optimally integrates AM with PM (subsequently referred to as the "agile-PM" framework for the purpose of the current study). The afore-mentioned component of the study seeks to establish practitioner acceptance of the proposed agile-PM framework in order to refine the framework into a viable model for implementation at Bank A. It is envisaged that the proposed agile-PM framework will serve as a complementary resource to project managers and software developers and possibly mitigate the challenges that may be experienced when using AM and PM strategies for software development (SD).

So ideally, on the one hand, the theoretical framework should inform and validate the activity of establishing the perceptions of IT staff at Bank A on the use of AM and PM for SD. In this regard, the researcher has opted to use phenomenography as a guiding framework. On the other hand, the researcher would like to obtain an academically defendable indicator of the acceptance of the proposed agile-PM framework. In this regard an acceptance-oriented framework would be a viable option.

In the subsequent sections, a critical analysis of the choice of academic framework is done with the objective of providing an academic defence for the selected academic framework to underpin the current study.

3.3 Phenomenography

Phenomenography is a framework that has been intended to discover individuals' qualitative experience of phenomena (Khan, 2014). It often depicts the manner in which people understand, differentiate, identify, envisage, perceive or experience various aspects of the world around them (Carbone *et al.*, 2007). Therefore, it deals with people's perceived experience of a certain phenomenon.

The primary objective of phenomenography is to determine the qualitatively different means of subject experience. It is also aimed at conceptualising, deducing or comprehending a range of phenomena and characteristics of the world. Researchers in this paradigm make the assumption that people experience certain phenomena in a restricted number of qualitatively different ways (Bowden, 2000). Therefore phenomenography seeks to discover the qualitatively different, but logically connected notions and interpretations that a group of people hold for a particular context (Bowden, 2000).

Phenomenography was developed by a researcher in the educational field (Marton, 1981).The first phenomenographical study that was associated with Information Systems (IS) research was from the same field: it is the study that was done by Booth (1992) on learning programming. Although phenomenography was established over twenty years ago, it is seldom used by researchers in the IS discipline. Kaapu *et al.* (2014)used it in three studies in the IS context, furthermore, it has been used in two doctoral dissertations: Isomäki (2002) studied the system designers' views of human beings and Vartiainen (2005) studied views of morality in IS education.

The diverse terrain of IS development requires and necessitates an appreciation of multiple perspectives on the development of these systems (Davidson *et al.*, 2001). In the context of the study it is envisaged that these multiple perspectives will be used to provide an informed response to the question: What are the main challenges of implementing ASDM and what are the practitioners' experiences of using AM and PM? Phenomenography is an experiential and qualitative methodology that enables an evaluation and comprehension of an individual's experience of some phenomenon. According to Yates *et al.* (2012) phenomenography differs largely from most theoretical and methodological approaches used in IS research. This is due to the fact that this method recognises that knowledge is both qualitatively different and differentially disseminated. This means that a change in connotations can be addressed at the point of an individual's subjective awareness. Kaapu *et al.* (2014) studied consumer's views on information privacy and in this study phenomenography assisted in understanding the variability of conceptions regarding privacy. When contrasted with a similar study carried out by Cheung & Lee (2006) on consumers' trust in Internet shopping by theoretically grounded integrative model, it was found that Cheung and Lee's findings did not offer empirical support for the outcome of perceived privacy control of Internet mechanism on consumer trust. This comparison is used just a case in point to emphasise the effectiveness of phenomenology as a research technique.

According to Bowden (2005) experience is not an isolated entity; rather it is relational. Phenomenography does not think of the research subject (the individual who has experienced the phenomenon) and the aspect of the world (phenomenon) as separate entities; rather a relation is formed between the two. Therefore, experience establishes a relation between a person and a given phenomenon and is referred to as a 'relational approach' (Limberg, 2000). So a relational approach is merely the close relationship between subjects and aspects of the domain through which a researcher can gain an understanding of the subject's experience. This relation is illustrated in Figure 13 below.

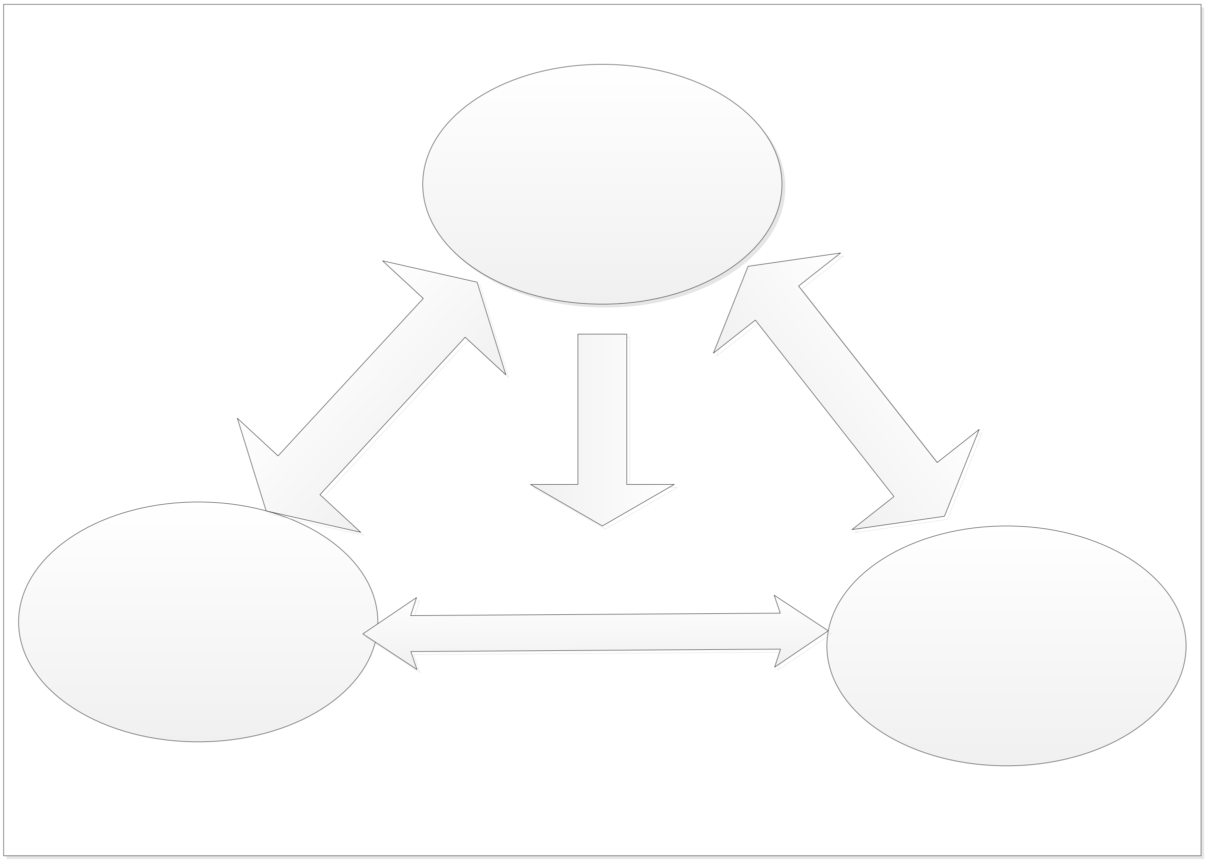


Figure 13: Relationship between objects with subjects and researcher (Bowden, 2005).

Figure 13 above shows that phenomenography embodies an intention to investigate the relationship between the subjects and aspect of the world (objects) in a given situation by the researcher (phenomenographer). Hence the subject and aspect of the world in a study are not independent but are connected with each other. In order to gain an understanding of people’s experience, González (2010) refers to referential and structural parts of the experience . The referential aspect of the experience is stating or emphasising the direct phenomenon or a specific meaning of the phenomenon. It is defined as a particular phenomenon which we are undergoing (experiencing) as the way it is. The structural aspect is defined as how people acted towards something (an action), how they go about in carrying out an action, how something is acted upon or carried out (González, 2011). The structural aspect of an experience consists of an outer structure and an internal structure of a phenomenon (González, 2011). The external structure of the way of experiencing a specific phenomenon concerned is to discern it from the outer context. This is referred to as an external horizon. In contrast, the internal structure in the way of experiencing a certain phenomenon is to determine the fragments of that phenomenon and how they are interconnected as a whole object, which is referred to as the internal horizon (González, 2011). Therefore external and internal horizons, together shape the structural aspects of people's experience of phenomenon. Figure 14 below provides an illustration of the linkage between the internal and external horizons.

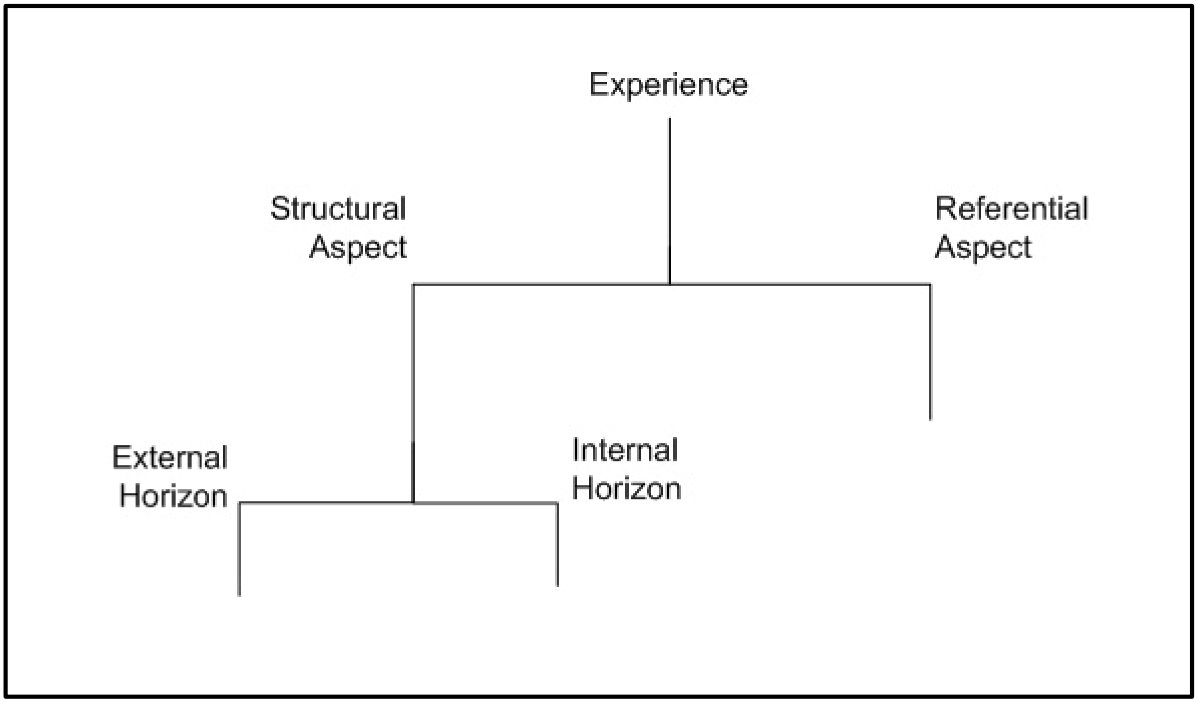


Figure 14: Component of experience (González, 2011)

Although the structural and referential aspects are different they are not separate, rather they are dependent and interconnected (Marton & Pong, 2005). In phenomenography the referential aspect is referred to as the 'what' aspect of an experience whereas the structural aspect is called the 'how' aspect of an experience. This observation has a direct bearing on the requirements of the current study where there is strong linkage to a subset of the research questions listed below:

- What are the PM related challenges of implementing AM at Bank A?
- How can the PM related challenges of implementing AM at Bank A, be mitigated?

Based on the evidence provided, phenomenography has been identified as a suitable methodology for the qualitative portion of the current study because of its ability to provide a platform to present and contrast different views of reality.

3.4 Review of Technology Acceptance Theories

The Technology Acceptance Model (TAM), proposed by Venkatesh & Davis (2000) is one of the most widespread theories that is used to explain information systems usage (Ibrahim *et al.*, 2011). There has been many adaptations of the original TAM due to the intense focus placed on it. The main advantage of the original TAM is its simplicity (Chau, 1996). However, as is typically the case with academic discourse that embodies a social element, simplicity is not always appropriate and does not factor-in the complexities inherent in human and organisational dynamics (Ibrahim *et al.*, 2011). According to Park *et al.* (2012) in order to address these shortcomings, many studies have been carried out which have resulted in changes to the originally proposed model. Park *et al.* (2012) also state that whilst there have been many adaptations of TAM, the most prominent acceptance based models used in IS research is the original TAM

(Davis, 1989), TAM2 (Venkatesh & Davis, 2000) and Unified Theory of Acceptance and Use of Technology (UTAUT) (Venkatesh *et al.*, 2003). In order to contextualise the theoretical framework that will be used to underpin the current study, the afore-mentioned acceptance models will be elaborated upon.

3.4.1 Technology Acceptance Model (TAM)

According to Riemenschneider & Hardgrave (2001) the best way to explain the acceptance of a Software Development Methodology (SDM) is by theoretical behavioural models that are supported by the Theory of Planned Behaviour (TPB) and the Theory of Reasoned Action (TRA), TAM and the Diffusion of Innovation's theory (DOI). Ibrahim *et al.* (2011) and Sánchez & Hueros (2010) assert that TAM is the most commonly used and referenced Technology Acceptance Model. Furthermore Chan & Thong (2009) state that the concepts of TAM, which are Perceived Ease of Use (PEOU) and Perceived Usefulness (PU) are adequately generic to be utilised in the investigation of how acceptable a SDM is. Figure 15 below illustrates this theory.

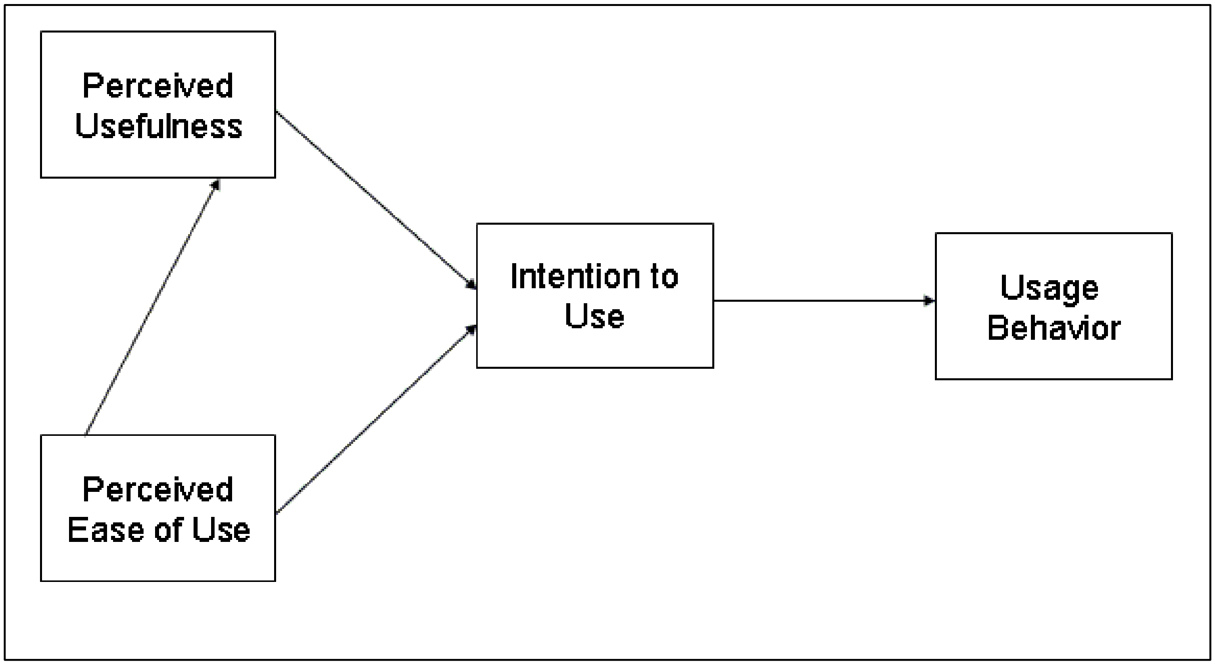


Figure 15: Technology Acceptance Model (Venkatesh & Davis, 2000)

TAM classifies the components that influence changes in the individual's behavioural attitude when using new technologies (Venkatesh *et al.*, 2003). In TAM, the key aspects concerning a person's use/intended use of an innovation are Perceived Usefulness (PU) and Perceived Ease of Use (PEOU). PU is the user's interpretation on exactly how the innovation enables them to effectively perform their work and bring about an improvement in results. In contrast PEOU takes into consideration the effort that is exerted by the user in using the system (Dillon & Morris, 1998).

Technology acceptance was described by Dillon & Morris (1998) as "the evident inclination within a user group to utilise an information technology (IT) for the tasks it was intended for". The main subjects in research focus primarily on instrumental influences, which explore acceptance decisions concerning views as to how utilising a technology will result in objective improvements in performance (Thompson *et al.*, 2006). An argument put forward by Thompson *et al.* (2006) is that this method might have had a restrictive outcome on technology research. They expanded their research to incorporate ideas concerned with non-instrumental influences on technology acceptance. TAM proposes that PU and PEOU influence a person's behavioural intention to use (BU) a technology or innovation piece. As said by Hu *et al.* (1999) there are various influences that contribute to the initial acceptance of technology, but fundamental determinants (e.g. PEOU and PU) play an inordinate role in continued acceptance.

TAM has been criticised for failing to include social influence as an external variable in the process of users' acceptance of a technology (Malhotra & D.Galletta, 1999). It is only regarded at the individual level and lacks application in various personal settings. Overlooking the impact of social ties on the users' adoption of technology has reduced the explanatory power of TAM. This has led to TAM being continuously extended to other technology acceptance models, such as UTAUT and the TAM 2 Model (Venkatesh & Davis, 2000).

Legris *et al.* (2003) conducted an extensive analysis of the literature on TAM and its applications and discovered a number of concerns. The first of these concerns is that the majority of the studies that validate TAM involved students and there was a lack of an application in business environments in most of these studies. The second concern was that most the studies involved an introduction of office software and there was a lack of business applications. The other concern discovered by Legris *et al.* (2003) is that the factors considered in the adaptation of IT are also influenced by organisation dynamics not included in TAM. Studies show that TAM explains only about 40% of IT usage (Hu *et al.*, 1999) and although TAM is a useful model it needs to be expanded to include social and human factors (Agarwal & Prasad, 1997). For these reasons TAM was not selected as the Technology Acceptance theoretical framework for the current study.

The following section looks at the TAM2 model.

3.4.2 TAM2

Due to the limitations of TAM, TAM2 was developed. Two practices, social influence (SI) practices (subjective norm, voluntariness and image) and cognitive instrumental practices (job relevance, output quality, result demonstrability and perceived ease of use), were included into TAM. This was done in order to rationalize the influences of the different factors on PU and behavioural intention (Liu, 2013). Subjective norm and image are the two factors of PU that correspond to the SI practices. TAM2 conceives that there are three SI instruments namely compliance, internalization and identification that play a role in interpreting the SI practices. Compliance stands for a situation whereby an individual behaves in a particular manner in order to attain certain rewards or escape penalty (Juinn & Tan, 2013). Identification on the other hand refers to an individual's belief that behaving in a certain way will boost their social status within a particular group, because within that particular group it is believed that conduct is important (Juinn & Tan, 2013). Internalization is defined as the combination of a referent's belief into one's own belief structure (Liu, 2013). TAM2 submits that subjective norm and image will have a positive influence on PU through processes of internalization and identification, respectively.

Four constructs namely job relevance, output quality, result demonstrability and PEOU describe the impact of cognitive instrumental practices on PU. A comprehensive discussion was offered by Venkatesh & Davis (2000) on how and why people shape perceptions of usefulness based on cognitive instrumental processes. This was based on three theoretical paradigms- work motivation theory, action identification theory and behavioural decision theory. They indicated that people develop PU judgment to a certain extent by cognitively associating the capability of a system with what they need to do in their job. TAM2 conceives that the foundation for establishing perceptions concerning the usefulness of the system is individuals' conceptual valuation of the match between significant work goals and the outcomes of carrying out job responsibilities using a system. It also suggests that PEOU and result demonstrability will have a positive direct impact on PU. Job relevance and output quality will have a moderating impact on PU, as a result - a higher output quality means the stronger the effect job relevance will have on PU. This model is illustrated in Figure 16 below.

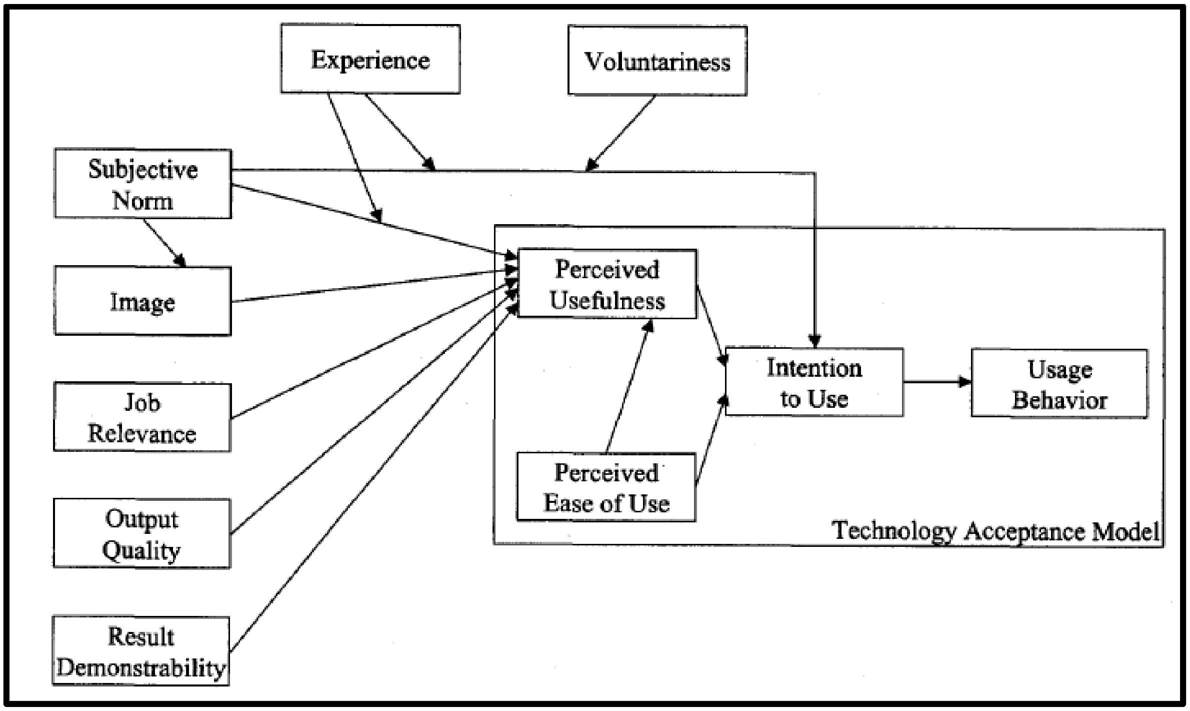


Figure 16: TAM2 model (Venkatesh & Davis, 2000)

3.4.3 Diffusion of Innovation Theory

The Diffusion of Innovation Theory is an Acceptance Model developed by Rogers (1995) with the primary intention of providing an explanation of the way in which any technological innovation transitions from the stage of invention to extensive use (or not). Although it is not solely related to IT, diffusion theory presents a conceptual framework aimed at examining acceptance at a universal level (Dillon & Morris, 1996). Diffusion theory hypothesises five attributes of innovations that influence their diffusion: relative advantage (the degree to which a technology presents improvements over the existing tools), compatibility (consistency with societal procedures and norms amongst its users), complexity (the ease of use or learning), trialability (the opportunity to test an innovation before committing to use it), and observability (how clear it is to see the technology's outputs and its gains) (Rogers, 1995). These attributes are illustrated in Figure 17 below.

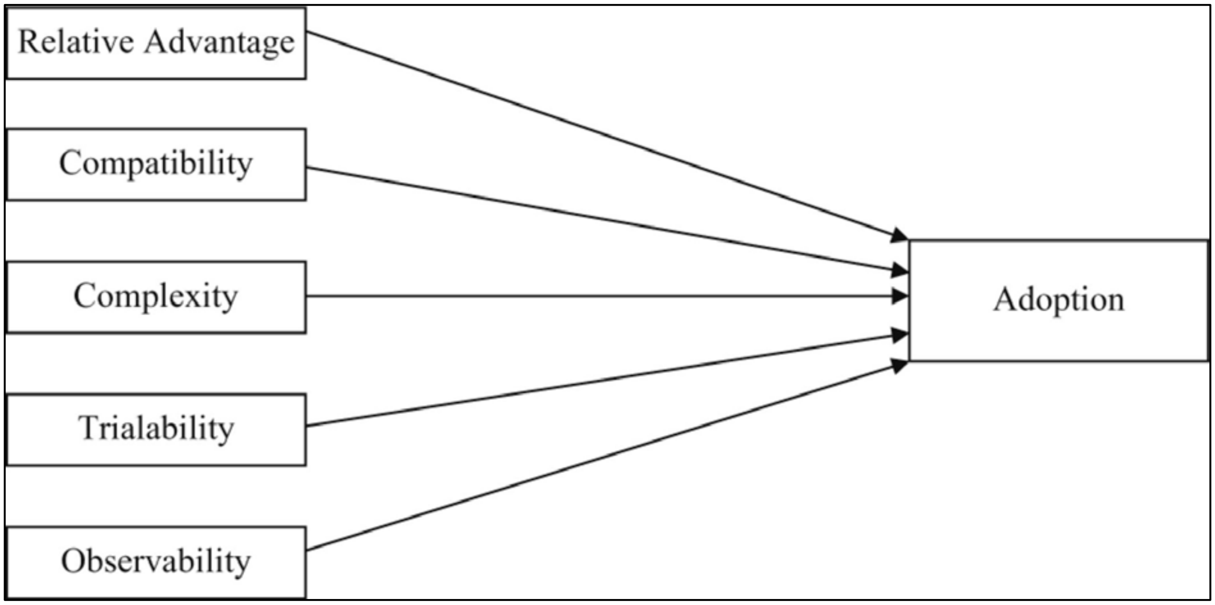
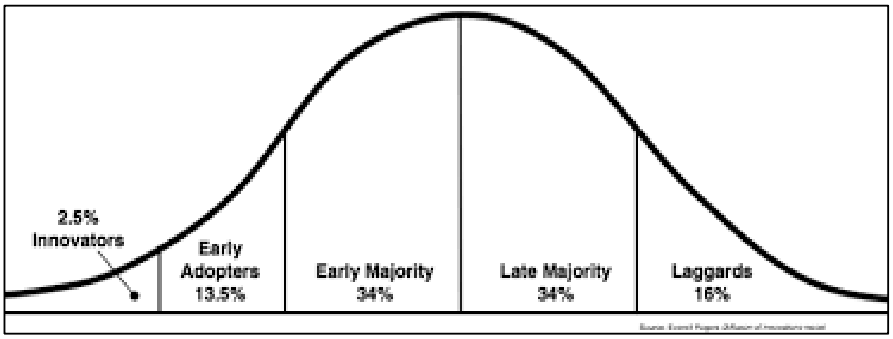


Figure 17: Variables determining the rate of adoption of innovations (Rogers, 1995)

According to Rogers (1995) each of these attributes cannot predict either the extent or the rate of diffusion individually. However diffusion studies have proved that innovations which provide advantages, compatibility with current procedures and principles, simplicity, possible trialability, and observability, are most likely extensively and rapidly diffused than an innovation with the collection of contrasting characteristics. An initial meta-analysis of the innovation diffusion literature discovered that three of these characteristics had the highest impact on adoption: compatibility and relative advantage were positively associated with innovation adoption (Dillon & Morris, 1996). Complexity was negatively associated with adoption at marginally significant levels (Tornatzky & Klein, 1982). However, the authors critiqued the initial conceptualizations of these concepts. Relative advantage, specifically, was quoted as mainly ambiguous. This was due to the fact that the principles used in judging what is "advantageous" are often not defined, as an example, an innovation could be considered as advantageous because it costs less or is less complex.

Innovation diffusion theory proposes that aspects at the individual user level are also important. Innovation is split into five groupings by Rogers (1995). The five groupings are based on how quickly individuals adopt or accept an innovation. These five groupings are: innovators, early adopters, early majority, late majority, and laggards. These classifications are mapped out over a normal distribution where each category (innovators and early adopters are combined into one for this purpose) signifies a standard deviation of diffusion. A depiction of this can be seen on Figure 18 below.



Source: Everett Rogers Diffusion of innovations model

Figure 18: Relationship between the types of adopters (Tiwari, 2015)

Consequently, the partition between early and late majority is the mean, with laggards and late adopters forming 50% of the population. Based on this, Rogers (1995) estimates that early adopters and innovators together constitute only 16% of the entire population. Early adopters have an inconsistent impact on the acceptance of any technology. Profiling studies of these categories have discovered a number of personalities such as risk-taking, adventure seeking and socioeconomic factors such as wealth, education that allegedly differentiate their affiliates.

According to Tiwari (2015) the Diffusion of Innovation theory is at its best a descriptive tool. There is uncertainty about the degree to which it can develop a readily refutable hypotheses. Many of its components may be specific to the culture in which it was derived for example North America in the 1950s and 1960s and therefore making it less relevant in, for example, East Asian and African countries, and as time goes on. Nevertheless, it offers one important foundation on which research and practice can be placed. According to Legris *et al.* (2003) the Diffusion of Innovation theory does not take into account an individual's resources or social support to adopt new behaviours (or innovation). Bagozzi (2007) supports this view by pointing out that in the adopter's categories of this theory, the category of a set of adopters is omitted. He elaborates on this by stating that some adopters may have the features of innovators/early adopters but may not quickly adopt an innovation. Due to these limitations this theory was not selected as the framework that would be used in the study to measure acceptance. The selected framework needs to take into account majority of the factors that influence adoption so as to provide the researcher with a clear view on whether there is a problem with the proposed framework or the issue is with the individual.

3.4.4 Unified Theory of Acceptance and Use of Technology (UTAUT)

The UTAUT model is an acceptance and adoption model developed by Venkatesh *et al.* (2003). The UTAUT model merges eight different models, namely TRA, TAM, the motivational model, TPB, a model combining TAM and TPB, the model of PC utilization, DOI, and the social cognitive theory (Venkatesh *et al.*, 2003). Venkatesh *et al.* (2003) developed UTAUT based on their recognition of specific factors they regarded as significant in stimulating an individual's decision on whether or not to adopt a new technology (Liu, 2012). Figure 19 below illustrates the different components of this model and their relationships.

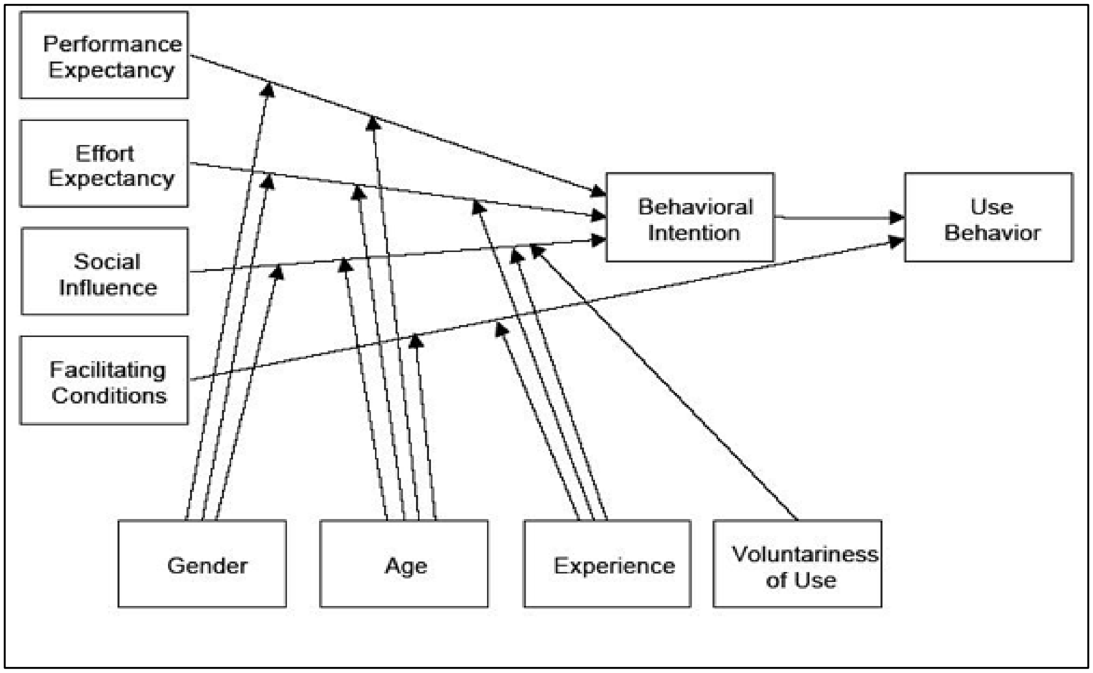


Figure 19: Unified Theory of Acceptance and Use of Technology (UTAUT) (Venkatesh *et al.*, 2003)

Performance expectancy (PE) speaks of the user's evaluation of the possible job benefit that the use of the technology may carry (Venkatesh *et al.*, 2003). This estimation or evaluation is made up of PU of the technology, extrinsic motivation to use the technology, usefulness of the technology to job-fit, relative advantages of the technology over other technologies, and outcome expectancy. To be more specific about these five constructs contained within PE: the PU assesses the degree to which an individual contemplating using a particular technology will enhance his or her job performance; extrinsic motivation considers the outer drive, such as improved job performance, pay or promotions, to use a particular technology; job-fit focuses on the functional side of technology in improving an individual's job performance; relative advantage looks at the benefit that the new technology might bring compared with what has already been accomplished by previous systems; outcome expectancy concentrates on the significance of behaviour which can be broken down into job-related performance expectations and personal expectations that concern individual goals (Venkatesh *et al.*, 2003).

Effort expectancy (EE) is similar to the idea of PU of technology defined in the TAM model. It comprises of three concepts: PEOU, complexity, and ease of use, which derive from previous studies (Juinn & Tan, 2013). PEOU is the degree to which a user considers it extra effort to use a particular technology; complexity describes a situation whereby people consider the new system as a comparably more difficult tool to understand and use; ease of use is the extent to which using an innovation is perceived as being difficult (Juinn & Tan, 2013).

The SI construct is the extent to which an individual perceives that significant people believe he or she should make use of the new system (Juinn & Tan, 2013). It consists of the subjective norm construct, the social factor construct, and the image construct. Subjective norm is when a person's decision about whether to adopt an innovation hinges on other people whose opinion is deemed to be important to him or her. The social factor construct is when an individual decides to adopt a technology under the influence of the whole social situation. The image construct is when the use of an innovation is perceived to improve an individual's image or status in their social system (Venkatesh *et al.*, 2003).

Facilitating conditions (FC) focus on the role that organizational and technical infrastructures play in the innovation adoption decision of an individual. It is a compound of three different constructs: perceived behavioural control, facilitating conditions, and compatibility. Perceived behavioural control is an individual's self-efficacy, resource facilitating conditions, and technology facilitating conditions. Facilitating conditions describe the surrounding environment, including both technical aspects and rule aspects, which have the potential to enhance or hinder innovation adoption for individuals. The compatibility construct primarily describes the compatibility of the innovation with already existing values, needs, and experiences of potential adopters (Venkatesh *et al.*, 2003).

Besides the four main constructs, there are other four moderators: gender, age, experience, and voluntariness of experience. Even though they are not determinant factors they can have an impact on using behaviour by impacting those four determinant constructs (Liu, 2013).

Gender can moderate PE, EE, and SI. Research shows that men are inclined to having higher PE than women because they tend to be task-oriented, and task achievement is important to them (Liu, 2013). Additionally, studies have indicated that EE is more important to women than to men (Liu, 2013). Women tend to be more perceptive to other people's opinions than men do and as a result SI is more prominent in adopting technology to women than it is in men (Venkatesh *et al.*, 2003). Age, is also an important mediator factor which can impact all the main constructs. For PE, younger people are inclined to being appealed to by extrinsic rewards than older people. EE is a more noticeable factor in adopting an innovation among older people than younger people (Venkatesh & Davis, 2000). Also, older people are said to be more likely to place increased salience on SI, with the influence deteriorating with experience (Venkatesh & Davis, 2000). In addition, with regards to FC, older people are more biased to environmental setup due to the fact that their way of learning is more passive and founded on experience. Experience can have an impact on an adopter's EE, SI, and FC. It discusses the amount of operation adeptness of a technology a user gains over a period of time. For individuals with little experience with a new system, EE is a significant factor in forecasting behavioural intention (BI). Conversely, if the experience is at a later stage, EE will not have much of an impact on BI. Similarly, SI plays an important role in improving BI in the early stages of experience, while its influence will diminish as people's experience concerning the new technology progresses into a later stage (Venkatesh *et al.*, 2003). The FC becomes a more significant factor compared to BI as experience with the new systems increases, so do the obstructions concerning to the viable usage (Venkatesh *et al.*, 2003). Voluntariness of use can only impact the SI's effect on BI. SI can exert its influence to fullness under a mandatory context because it has a direct impact on intention, while more effort is spent to impact BI under voluntary context (Venkatesh *et al.*, 2003).

This study will exploit the adaptability of these concepts to put together a construct that will direct the study in exploring the perception of the practitioners on the usefulness of the agile-PM framework. The theory of technology acceptance that will be used for the current study is the UTAUT. UTAUT was identified as the most suitable theory for this study because it has a number of advantages over the two previously discussed theories. Firstly it has been developed from the "experience" obtained from previous technology acceptance theories, which makes it a comparably complete model. Secondly, its descriptive power in technology using behaviour is up to 70 percent, which is a significantly higher rate than other technology acceptance theories (Wu *et al.*, 2008). With such precision and comprehensive application in rationalising technology adoption behaviour, the UTAUT model outdid other theories and turned out to be a better alternative for researchers in the area of technology use behaviour (Wu *et al.*, 2008). Thirdly the consideration of multiple personal circumstances explored in the constructs adds value to the study as it is focused on multiple Information Systems professionals of different backgrounds, length of experience and with different expectations and viewpoints. This further expands the explanation power of this model. Lastly, other studies have made use of the UTAUT model to explain the adoption and acceptance of technology in a banking environment. These include the study by Martins *et al.* (2013) on understanding the Internet Banking Adoption and it made use of UTAUT to explain behaviour intention and usage behaviour of Internet banking. A similar study was carried out by Yu (2012) and it was mainly focused on the factors affecting individuals to adopt mobile banking and it employed the UTAUT model. Similarly Saibaba & Murthy (2013) used the UTAUT model to evaluate the factors that influence the behavioural intention to adopt Internet Banking.

Most importantly the UTAUT model has been used in similar contexts, to evaluate the adoption/ acceptance of a software development methodology. Lambert (2012) carried out a study based on the UTAUT model focusing on 'behavioural intent' to adopt agile software development methodologies (ASDM). This study investigated the relationship between adoption and the impact on the project performance attributes. Independent variables included PE, EE, SI and FC with the dependant variable being BI to adopt ASDM. The research found positive correlations between PE, EE, SI, FC, and BI to adopt ASDM. Algharibi & Arvanitis (2011) adapted the UTAUT model and utilised it to validate captured user needs and requirements of particular interactive software technologies, within the framework of Clinical Trial Management Systems (CTMS). This is very similar to how the current study aims to utilise the UTAUT model. In this study Algharibi & Arvanitis (2011) made two main additions to UTAUT. Firstly they condensed age, gender, experience level, speciality, and voluntariness to use the system into one dimension called Individual Factors. The second change was introducing Technology Anxiety and Adaption Timeline as new dimensions.

The current study will make use of the PE, EE, FC and SI to measure BI. Actual use of the agile-PM framework cannot be evaluated at this stage as this is a proposed framework/model that is not currently in use at this organisation. The moderating factors have also been excluded from the model. As mentioned above moderators are variables that influence the strength or weakness of the correlations between independent and dependent constructs in the model (Serenko *et al.*, 2006). The moderators in this study were excluded because of the statistical difficulties of detecting these moderator effects. Further no relationships were anticipated to entirely be invalidated due to moderation and it was projected that any strong correlation between the underlying variables would be evident, even when moderated. As Chin *et al.* (1996) point out, analysis of moderators with co-variance based techniques such as SEM is “tedious and technically demanding”. In reality, it is difficult to find the moderator effects even when advanced approaches are utilised (Jaccard *et al.*, 1990; McClelland & Judd, 1993) and when they are found interpretation is challenging as even the sign of the regression coefficient of the moderator may not point to anything (Mossholder *et al.*, 1990). Due to time and scope constraints, the statistical challenges could not be mitigated.

Figure 20 below is a depiction of the adapted UTAUT model that was used to evaluate practitioners’ acceptance of the agile-PM framework.

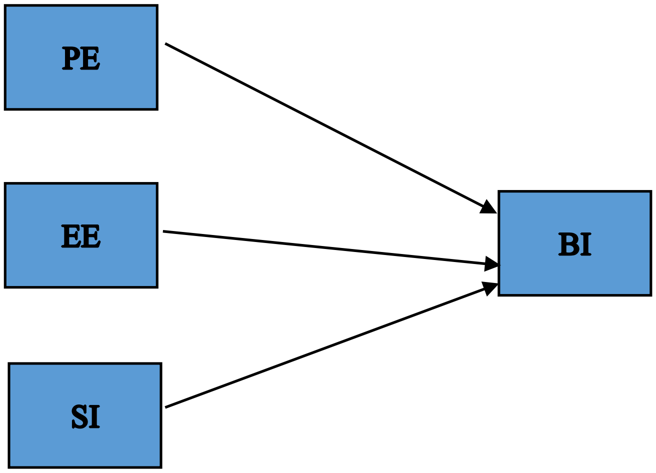


Figure 20: Adapted UTAUT model

Use of the UTAUT model for understanding software practitioners' (SP) perceptions of the proposed framework and potential future use is based on the following assumptions:

- PE will positively impact SP's intention to use the framework
- EE will positively impact SP's intention to use the framework
- SI will positively impact SP's intention to use the framework

3.5 Summary of the Chapter

This chapter examined the literature on phenomenography and on the most important models of technology acceptance such as, TAM, TAM2, DOI theory and UTAUT. Based on the limitations and applicability of each of the models, this research utilized phenomenography for the qualitative portion and UTAUT for the quantitative portion of the study. The next chapter will present the research methodology including the research methods, selection and justification of the proposed methods and discussion on the research model.

4.0 RESEARCH METHODOLOGY

4.1 Introduction

Research is cited by Welman *et al.* (2006) as a method that involves collecting scientific data by utilising various objective methods and procedures. The use of the word ‘objective’ suggests that these methods and procedures do not rely on opinions or personal feelings and that at each phase of the research process specific methods are employed. These methods make up ways for obtaining samples, evaluating variables, gathering information and analysing that information (Welman *et al.*, 2006). The purpose of a research project is said to determine the method that is selected amongst the various methods (Welman *et al.*, 2006). The research methodology considers and describes the rationality behind research methods and techniques.

The main aim of this chapter is to give details about the research design and methodology employed in addressing the research objectives indicated in Chapter 1. It also explains the research instruments used (i.e., questionnaire and interview), the research population, and the techniques used to establish validity and reliability of the data.

4.2 The Research Design

Brink & Wood (1998) state that the motivation behind a research design is to establish an approach for responding to research questions and “is a blueprint for action”. It is the general plan that gives the strategies that will be used by the researcher to obtain objective, accurate and interpretative information. According to Welman *et al.* (2006) research design can take one of two forms – qualitative or quantitative- based on the type of data to be collected, it is also partially affected by the data collection technique, presentation and analysis. Quantitative research uses surveys, questionnaire and experiments for data collection. The data is evaluated and presented in numbers, allowing the data to be defined using statistical analysis (Hittleman & Simon, 1997). Quantitative researchers use a sample of subjects to measure variables and describe relationships between variables. They do this by making use of statistics such as relative frequencies, correlations, or differences between means; their emphasis is on the testing of theory. In contrast qualitative research uses a number of different knowledge claims, investigation approaches, and data collection methods (Creswell, 2003). Qualitative data sources include questionnaires and interviews, observation and participant observation, manuscript and official documentation, and the researcher's thoughts and responses (Myers, 2009). Interviews, direct observation of behaviours, public documents and written opinions are some of the data collection methods used in qualitative methods (Sprinthall *et al.*, 1991). Other sources of data are recorded accounts about people, opinions and events, or a combination of these.

Labaree (2013) defines twelve kinds of research design and these are philosophical design, sequential design, longitudinal design, observational design, exploratory design, historical design, descriptive design, experimental design, cross-sectional design, causal, design cohort design, case study design and action research design. Sekaran & Bougie (2009) believe the selection of a research design is guided by the complexity of knowledge of the research problem, the research questions and the feasibility of the research. A case study approach was selected for the current study. The motivation for this choice is of research approach is primarily the context that was presented for the study at Bank A as well as the researcher’s access to resources and opportunities that could facilitate the study at Bank A.

4.2.1 Case Study Research

The case study research design is intended for the collection of data about particular proceedings of a single organisation (Hair *et al.*, 2007). This type of research design also does particularly well at establishing an understanding of complex issues for researchers and can reinforce what was previously understood through previous research (Soy, 1997). This research design type has also been commonly used to provide the basis for the use of ideas and expansion of methods in reality (Soy, 1997) . Yin (1984) defines the case study research method as a practical investigation that explores a contemporary development in its realistic setting; when the boundaries between the trend and context are not clearly marked; and in which different sources of verification are utilised. The data is usually acquired from focus groups, personal interviews and company history.

Cornford & Smithson (1996) critiqued the case study based on reasons of non-representativeness and insufficient statistical generalizability. What they found to be a reason for concern was the wealth and intricacy of information that is gathered which results in numerous interpretations and perhaps 'researcher biases'. Miles & Huberman (1994) also pointed out the fact that there is no recommended method for analysing case study data. Yin (2003) counters these reviews by drawing attention to the fact that case studies are suitable for generating and refining generalizable theories. Furthermore the use of multiple case studies can result in generalizations in terms of propositions. In addition he states that case studies are utilized for investigative generalizations in the case where the researcher aims to generalize a particular set of results to a broader theoretical proposition. Moreover Yin (2003) considers this approach to be vastly useful in supporting the researcher to acquire 'thick descriptions' of the concept that is under investigation. This gives access to some of the restraints of differing and diverse interpretations and this would not be possible with experimental or quantitative research.

The research problem and research questions are making an effort at exploring new ground and discovering an innovative means of achieving a particular objective. Therefore, due to the type of questions that are presented, a research method that is primarily interpretative and qualitative is required and from the account given above, the case study research is considered to be the most suitable method. A brief look at the research questions and objectives of this study shows that the researcher is expected to have no influence on what is being studied and should not influence the elements being studied in any way. This will enable the researcher to get an accurate view of the opinions and attitudes of the practitioners on agile and PM. Henceforth the researcher will be able recommend a suitable framework that amalgamates these two thus the case study approach is suitable because it draws focus to the real-life context.

The aspect of practicality of the research when selecting a research design which was put forward by Sekaran & Bougie (2009) was taken into consideration when deciding on this research design. Case studies are carried out within a certain time frame and interviews are done at a suitable time that can be agreed upon by the respondents and the researcher or if questionnaires are being used they could be mailed to the respondent (Yin, 2003). This makes it particularly suitable for this study because it provides the respondents with the flexibility to propose a suitable time to participate in the research so as to eliminate disruption of company operations.

Lastly, the case study is historically a commonly recognized research technique in the field of Information Systems. This was confirmed in the study done by Scott & Ives (1992) which indicated that this research technique was the most frequently used strategy between 1970 and 1979. This was evident in 532 journal articles/papers. Additionally the case study research method has been used in performing research of this nature. Batra *et al.* (2010) did a study to examine the possibility of combining AM with the traditional waterfall software development approach. This study was based on a case study carried out in the cruise line industry. Another study which was designed to explore how agile practices can be altered to accommodate large and distributed projects was conducted by Lan *et al.* (2004). They made use of a case study research methodology and they reported on the preliminary insights acquired from the case study. This validates the fact that the case study research design is extensively used for studies of this nature hence it is suitable for this study.

4.2.2 The Research work plan

While the research design alludes to the logical aspect of the study, the logistical aspects of the study will be detailed as part of the research work plan. The first step in the research was an attempt at answering the following research questions:

- What are the Project Management (PM) related challenges of implementing Agile at Bank A?
- How can the PM related challenges of implementing AM at Bank A, be mitigated?

In order to obtain a more complete picture of the challenges of implementing AM from a PM perspective and devising a solution on how the challenge of integrating PM principles into AM can be mitigated interviews were conducted with key stakeholders, mainly PMs. The interview component of the study heralds an incursion into the qualitative research paradigm. However, there is no intention to classify this study strictly along the dimensions of a qualitative study. This part of the study was underpinned by phenomenography. The outcome of this phase of the research was a framework that integrates AM and PM which was developed using the responses from the interviews with stakeholders. A questionnaire which was aimed at establishing the respondents' acceptance of the agile-PM framework was distributed to the stakeholders. Acceptance of the agile-PM framework was underpinned by the UTAUT model so that the researcher could establish the respondents' intention to use the model. The research question that this step was aimed at answering was:

- What is the acceptance by software practitioners (SP) at Bank A, of a framework that guides the integration of PM principles into AM?

Figure 21 below illustrates the process described above.

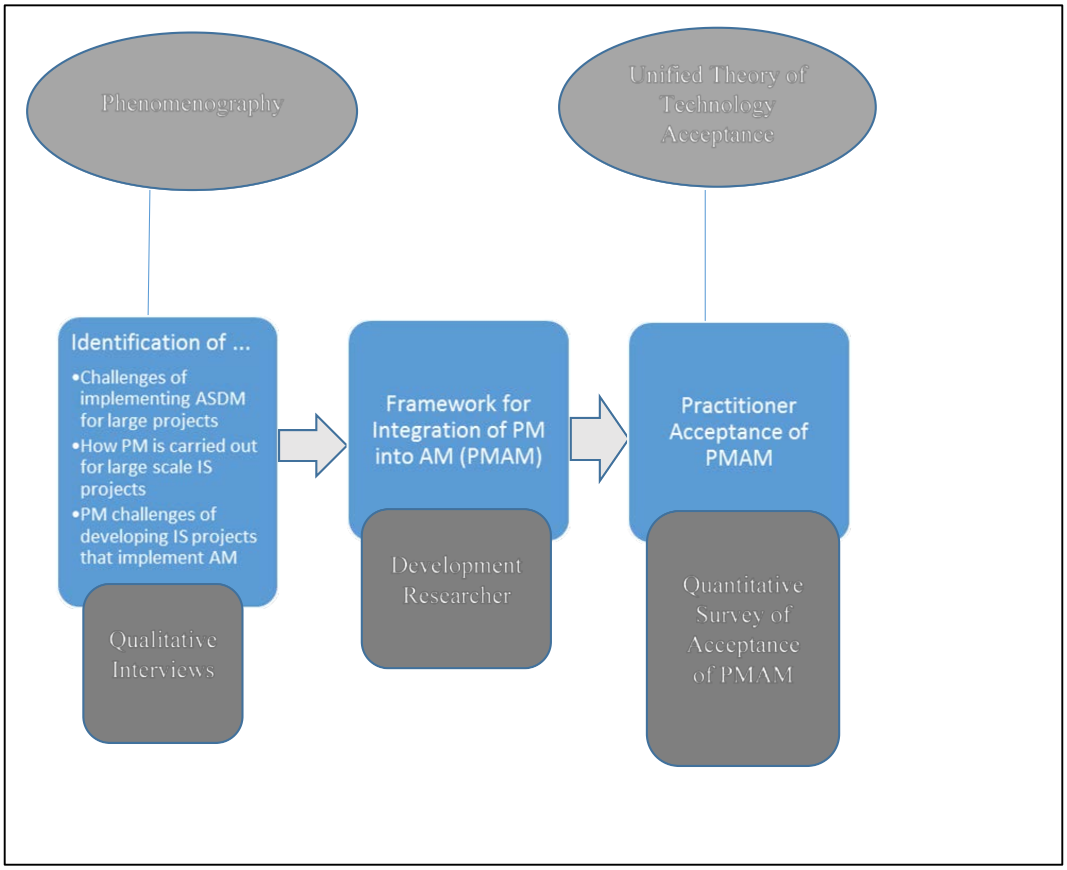

Phenomenography
Unified Theory of Technology Acceptance
Identification of ...
•Challenges of implementing ASDM for large projects
•How PM is carried out for large scale IS projects
•PM challenges of developing IS projects that implement AM
Qualitative Interviews
Framework for Integration of PM into AM (PMAM)
Development Researcher
Practitioner Acceptance of PMAM
Quantitative Survey of Acceptance of PMAM

Figure 21: The Research work plan

4.3 Sampling Technique

Here emphasis is given to the target population and the sampling technique that should be used for this proposed study by first defining the population that the sample was drawn from (the target population) and there after describing the technique that was used to draw out the sample.

4.3.1 Target Population

The Target population is the entire set of components (people or objects) from which the research data is to be acquired to satisfy the purpose of the study (Hair *et al.*, 2007). As previously stated the current study was carried out at a Banking institution in the Johannesburg area. This organization was chosen on the basis of its adherence to Traditional Project Management (TPM) techniques. Also, the company has recently embraced the principles of Agile Software Development Methodology (ASDM) and is making a transition from a waterfall-like approach to an agile approach to software development (SD). The SD staff members (developers, testers, analysts and project managers) who are currently in agile feature teams in this company were identified as the population. The size of this population is 108 employees.

4.3.2 Sampling Strategy

Sampling is related to the method of selecting a portion of the complete population. That portion needs to be representative of the entire population to enable the researcher to obtain data about the phenomenon that is of interest in the study. A sample is a fragment of the population that has been chosen to take part in a study (Weiss & Weiss, 2012). The method of sampling is commonly categorized into two; one technique is known as probability sampling and gives rise to samples where the probability of selection of each component is guaranteed. In this type of sampling technique each component of the population has an equal chance of being selected and included in the sample (Weiss & Weiss, 2012). The other technique is referred to as non-probability sampling and with this technique the probability of being selected is unknown. The components of the population do not have an equal chance of being selected (Polit & Hungler, 1995). Cases of probability sampling techniques include simple random sampling, stratified random sampling, systematic sampling and cluster sampling. Non-probability sampling examples include convenience sampling, purposive sampling, judgmental sampling, snowball sampling and quota sampling (Welman *et al.*, 2006).

According to Sekaran & Bougie (2009) factors that should be given attention when picking a sample technique to be used in the study are objectives of the research study, the desired precision level, the risk tolerance in estimating the precision level, variability in the population, cost and time limitations and population size.

The sampling technique used for this study was a purposive, non-probability sampling technique. A purposive sampling technique is one that uses facts that the researcher has concerning the population and the purpose of the study to select the sample. The members to be included in the sample are chosen on the basis of some known characteristic (Sekaran & Bougie, 2009). So a purposive sample of staff members that have had experience in SD project(s) making use of TPM techniques. This technique was used to ensure that the objectives of the study were achieved. As an example the population of SD staff included members that had been recently employed from University and had not been exposed to any projects within the organization. The need for individuals who have a thorough and practical understanding of the two notions (Agile and TPM) is required since precision and getting a realistic view of the case is important in this study.

Also the purposive sample consisted of team members from each phase of the project lifecycle to ensure that the data collected on PM and AM can be analyzed from the perspectives of the different phases of the SDLC. Selection of the suitable respondents also assisted in retaining the focus of the study. Purposive sampling is believed to be a good fit for the goals of qualitative research particularly case studies. This is because case studies are characterized by a lack of concern for the fundamental predispositions of the larger group but are more interested in a specific group or a specific concept (Given, 2008).

The purposive sampling technique was used in both the quantitative and qualitative portions of this study. The sampling frame was obtained by firstly consulting all the IT heads within the various business units at Bank A. The IT heads identified all the feature teams within their structure and shared details of the various portfolio managers with the researcher. The researcher approached these portfolio managers for a list of staff members within their teams that had experience in TPM principles. Once the researcher met with the selected participants it was verified that they indeed have been exposed to TPM principles from a project perspective.

For the qualitative portion of the study 20 staff members were targeted while 84 staff members were targeted for the quantitative portion at the 95% level of confidence. The samples consisted of project managers, portfolio managers, business analysts, software developers, testers, production support managers and project administrators. According to Glaser & Strauss (1967) qualitative sample sizes must be large enough to gain views for the majority of insights. Gaining most or all of the insights will result in the realisation of saturation. Saturation comes about when adding more respondents to the study does not result in other views or additional data. They recommend the theory of saturation for reaching a suitable sample size in qualitative studies. For phenomenological studies Creswell (1998) recommends 5 to 25 interviews and Morse (1994) recommends at least 6. This proves that the selected qualitative sample size was suitable.

4.4 Research Instruments

Research instruments refer to the methods of gathering and assessing evidence on the variables of interest, by making use of a recognized methodical process that enables the researcher to formulate answers to listed research questions, test hypotheses, and evaluates results. Research instruments are commonplace to all fields of study including humanities, physical and social sciences, business, etc. (Welman *et al.*, 2006).

Semi-structured interviews were conducted for the qualitative component of this study and for the quantitative component questionnaires were employed.

4.4.1 Face-to-face semi-structured interviews (Qualitative)

Semi-structured interviews contain a number of important questions that are intended to identify the subjects that are of concern and will be investigated. It is acceptable for the interviewer or respondent to digress in order to explore an idea in greater detail. The flexibility provided by this approach as opposed to the rigidity of structured interviews enables respondents to expand on information they believe is important(Gill *et al.*, 2008).

The decision to make use of face-to-face semi structured interviews was persuaded by the nature of the study and the need to acquire comprehensive data that can be used in qualitative analysis and accordingly prompt responses to the research questions. The research additionally looks to acquire an in-depth understanding of what is going on instead of the recurrence of event of things that are already understood.

Making use of a face-to-face interview firstly gave the certainty of the person who gave the responses as opposed to maybe using a questionnaire which will be completed in the absence of the researcher and the respondent may not be the one who was targeted. This would cause a problem in the accuracy of the data collected as it is necessary that the data comes from people who were direct participants in the projects. The importance of obtaining the data from staff members that were directly involved in the project is also illustrated in the sampling method that is utilised which is purposive sampling. Secondly, the responses were recorded with a recording device which will produced an accurate report than one that could have been formulated by writing out the responses. That accuracy is important in this study as in any other study in order to ensure the credibility and validity of the research. The third advantage is the probing of respondents for more detailed information.

The semi-structured nature of the interview enabled the gathering of detailed information and allowed for the researcher to get the respondent to elaborate on issues that were seen as being important. This is particularly important in this study because there was only one chance to interview the respondents because the researcher wanted to minimise the interruptions that this may cause in the organisation with the time spent on these interviews. This nature of an interview also allowed the respondents the freedom to express their views.

According to Sekaran & Bougie (2009) the benefit of face-to-face interviews is that the researcher can clarify uncertainties and to ensure that the questions are properly understood by reiterating or rephrasing the questions when required to do so. Furthermore, a 100 percent response rate is ensured (Sekaran & Bougie, 2009). In contrast, face-to-face interviews are subject to geographical limitations, a limitation that was managed by including staff members at the Johannesburg campus of the bank only. Additionally, respondents may have felt uncomfortable about anonymity of their responses when they interacted directly with the interviewee (Sekaran & Bougie, 2009). Nevertheless, this possible fear was dealt with by informing the respondents prior to the interview that no reference would be made to specific individuals in the study.

Pretesting of interviews

The interview questions were pre-tested to ensure that each question was clear and unambiguous. The pre-testing of interview questions was conducted with project managers and business analysts at the bank in order to test appropriateness, comprehension and identify any inadequacies before the actual interviews took place. These candidates were selected for the pre-testing of the interview questions because they had shown an interest in the study. They were also knowledgeable in this field of study. A few issues were highlighted by the pre-testing candidates and the changes were made accordingly.

This step was carried out in order to ensure validity of the research study. Validity is defined as the level to which a research study measures what it was set out to measure (Sekaran & Bougie, 2009). Welman *et al.* (2006) further state that for validity to be achieved it should be ensured that the selected design is able to give answers to the research questions and thus serve the purpose for which the research was established for to begin with. The research plan must eventually provide feedback to the research hypothesis that was expressed at the beginning of the study (Welman *et al.*, 2006).

4.4.2 Questionnaires (Quantitative)

The primary aim of the questionnaire was to attain a quantifiable measure of end users' acceptance of the agile-PM framework. In the current study, end user acceptance has been operationalized via the theoretical framework for the quantitative part of the study to the constructs of Performance Expectancy (PE), Effort Expectancy (EE), Social Influence (SI), Facilitating Conditions (FC) and Behavioural Intention (BI). According to Sekaran & Bougie (2010) the following step is to find a representative group of items that are an adequate measure of each of the constructs. This is described as verifying the internal validity of the questions in the questionnaire. One possible way of accomplishing internal validity is by utilising measures that have already been developed, validated and reputed to be "good" instead of developing a unique measure that still needs to be validated

(Sekaran & Bougie, 2010). Questions 1 to 15 of the questionnaire are aligned to the UTAUT model which is one of the theoretical frameworks of the current study and assist in achieving a measure of quantification of user acceptance of the proposed model/framework. The items used to measure PE, EE, SI, FC and BI stem from validated instruments used by Venkatesh *et al.* (2003) . Question 16 of the questionnaire is particularly aimed at determining what adjustments the respondents feel should be made to the proposed framework to make it more usable in their environment.

Questionnaire Design for Performance Expectancy

The current study adopted the items for the measurement of PE from the instrument developed by Venkatesh *et al.* (2003). Wording modifications were made to the user acceptance scale to fit the software development model that is under study. All items were measured on a 5 point Likert scale. This instrument was validated by Venkatesh *et al.* (2012), this was done by testing for content validity, reliability and construct validity in a study involving 4, 127 users making use of mobile internet technology.

According to Juinn & Tan (2013) PE is focused on the following constructs and theories:

- Perceived Usefulness
- Extrinsic Motivation
- Job-fit
- Relative advantage
- Outcome Expectations

Questions 1 to 4 of the questionnaire (shown in Table 11 below) used in the current study (Appendix B), are aligned to the constructs used by Venkatesh *et al.* (2003) to measure PE.

Table 11: Questionnaire items to measure PE

The proposed model would be useful for me to use in software development projects
Using the proposed model would enable me to ensure that those aspects of software development that are important to me are upheld
Using the proposed model would help me to do my software development related job activities more quickly
Using the proposed model would increase my productivity in terms of software development related activities

Questionnaire Design for Effort Expectancy

The items to measure EE were also adapted from the instrument used by Venkatesh *et al.* (2003). According to Juinn & Tan (2013), EE is focused on the following constructs and theories:

- Perceived ease of use
- Complexity
- Ease of use

Questions 5 & 6 of the questionnaire used in the current study (refer to Appendix B for the questions) are aligned to the constructs used by Venkatesh *et al.* (2003). These questions are displayed in Table 12 below.

Table 12: Questionnaire items to measure EE

I predict that learning how to apply this model to my software projects would be easy to do
I predict it will be easy for me to become skilful at using the proposed model

Questionnaire Design for Social Influence

According to Juinn & Tan (2013), SI is based on the following constructs and theories:

- Subjective norm
- Social factors
- Image

Questions 7, 8 & 9 of the questionnaire used in the current study (refer to Appendix B for the questions) are aligned to the constructs used by Venkatesh *et al.* (2003). These questions are displayed in Table 13 below.

Table 13: Questionnaire items to measure SI

People who are important to me in my job domain will endorse my preference to make use of the proposed model
People who have an influence in my behaviour will endorse my preference to make use of the proposed model
People whose opinions I value will endorse my preference to make use of the proposed model

Questionnaire design for facilitating conditions

According to Juinn & Tan (2013), FC is centred on the following constructs and theories:

- Perceived behavioral control
- Facilitating conditions
- Compatibility

Questions 10, 11, 12 & 13 of the questionnaire used in the current study (refer to Appendix B for the questions) are aligned to the constructs used by **Venkatesh *et al.* (2003)**. These questions are displayed in Table 14 below.

Table 14: Questionnaire items to measure FC

I have at my disposal, the resources necessary to make us of the proposed model
I have the required knowledge to make use of the proposed model
The proposed model is compatible with other processes that I use for work related activities
I will be able to obtain help from others when I have difficulties in using the proposed model

<u>Questionnaire design for Behavioural intention</u>

Question 14 of the questionnaire used in the current study (refer to Appendix B for the questions) are aligned to the constructs used by Venkatesh *et al.* (2003). These questions are displayed in Table 15 below.

Table 15: Questionnaire items to measure BI

I intend to start making use of the proposed model for upcoming software development projects
Please select the type of software development projects that you think will be most appropriate for use of the proposed model

Pre-testing of the questionnaire

A pilot study was performed by handing out the questionnaires to approximately 5 people. The pilot study was designed to make sure that all questions were unambiguously understood by the respondents prior to conducting the final survey. The questionnaire consisted mainly of closed ended questions and one open ended question. The open ended question purely served an information purpose with the aim of offering respondents a platform to openly articulate their views on how the proposed model could be improved. Although this information was not part of the statistical analysis for the current study, it was insightful for the researcher to acquire this information in order to expand her understanding of the type of model/framework that could be most suitable for this kind of environment. The use of closed ended questions is promoted by Sekaran & Bougie (2010) as these type of questions assist respondents to make the correct selection by going over the options given. Minor suggestions from the pilot study were integrated into the questionnaire to help the respondents to better understand the questions. The pilot study also helped in determining the time taken to complete the questionnaire.

Distribution of the questionnaire

The questionnaires were personally administered to the selected participants by circulating hard copies of the questionnaires. The advantage of personally administering questionnaires is that the researcher can collect completed responses in a shorter time frame (Sekaran & Bougie, 2010). For the participants that were inaccessible, the researcher found it necessary to email them in order achieve a better response rate.

4.5 Ethical Clearance

Ethical behaviour is of great importance when conducting research. Ethical considerations consist of issues such as honest reporting of results and plagiarism however other matters surface when the research involves human subjects. The theories of ethical behaviour when conducting research are shared globally and they are concerned with honesty and respect for individual rights (Welman *et al.*, 2006). The university has an ethical committee that needs to grant approval on all research projects that concern humans and animals. In fact, a number of organisations have such a committee in place.

To warrant that the study follows these ethical requirements a few steps were taken. The first of these steps was to get ethical clearance from the UKZN's ethical committee which granted permission to conduct the study. Secondly, authorisation to conduct the research at Bank A was obtained with the gate keeper's permission signed by the CIO and the Head of Human Resources.

The interview questions (Appendix A) and user questionnaire (Appendix B), were issued to the Ethical Committee at UKZN for review and full ethical clearance was granted (refer to Appendix C). Approval was obtained from the Office of the CIO and HR (refer to Appendix D). Letters of informed consent on the questionnaire and interviews provided a background to the study, assurance on the anonymity of the results and the researchers contact details, if required. The information collected was also exported to a compact disc for storage.

4.6 Summary of the chapter

This chapter outlined the research design adopted for this study and the data collection techniques used to extract the relevant information in order to meet the research objectives. Qualitative data was collected in the first phase of data collection and quantitative data was obtained for the second phase. Interviews were used as the qualitative data collection instrument, followed by questionnaires which was the quantitative data collection instrument. A pre-test of the questionnaire and interview questions was conducted to ensure reliability and validity of the study. SPSS was used to capture the quantitative data and NVivo was used to capture the qualitative data. The next chapter deals with the findings and analysis of the study.

5.0 FINDINGS AND ANALYSIS

5.1 Introduction

This chapter contains the findings and analysis of this study. The results of the interviews will be presented first, followed by the results of the survey, as the findings from the survey build on the findings from the interviews.

5.2 Response Rate

The response rate will be presented according to each phase of the study. The first phase of the study which may be deemed to be an exploratory phase entailed interviews with significant role players who may be regarded as experienced in terms of their involvement with project management (PM) and agile software development (ASDM). The interview component of this study was aimed at answering the following research questions:

- What are the PM related challenges of implementing Agile at Bank A?
- How can the PM related challenges of implementing AM at Bank A be mitigated?

Twenty individuals were targeted for the face- to-face interviews and only ten agreed to be interviewed representing a 50% response rate which is said to be acceptable in a qualitative study that entails interviews (Kevin, 1999; Owen & Jones, 1994; Saunders *et al.*, 2000).

The second phase of the study entailed the presentation of a framework/model that proposes the integration of PM principles into ASDM. A total of 84 questionnaires were administered and 61 were returned representing a response rate of 73% and according to (Kevin, 1999; Owen & Jones, 1994; Saunders *et al.*, 2000) this is also acceptable.

5.3 Analysis of the Qualitative Data

Qualitative data analysis is intended at assisting the researcher to derive substantial conclusions from the sizeable amount of data that has been gathered (Sekaran & Bougie, 2009). The analysis of qualitative data is conducted in three steps which are data reduction, data display, and the drawing of conclusions. The methods that are used for analysing qualitative data are content analysis, conceptual analysis, relational analysis and narrative analysis. This study will make use of content analysis. Content analysis is referred to by Sekaran & Bougie (2009, p. 82) as "*an observational research method that is employed in the systematic evaluation of representational content of every form of recorded communications*". Content analysis is done on text through coding it into classifications and analysing through conceptual analysis which identifies the occurrence of and the rate of occurrence of notions such as words, themes, or characters or else the analysis is performed through relational analysis. Relational analysis originates from conceptual analysis by noting the connections between concepts in text. According to Sekaran & Bougie (2009) content analysis is suitable for analysing data collected from document analysis, interview, focus groups, and observation. Data reduction, data display and the drawing of conclusions was performed during data analysis. These three stages of analysis were done on the data that was collected through interviews using content analysis as already mentioned. There are various forms of content analysis and for this current study thematic analysis was used. During thematic analysis, the data is examined in its entirety to find the common concepts that occur and to collapse these instances into main themes that summarize all the views that have been collected (Brikci & Green, 2007).

5.3.1 Interview Protocol

There were three parts to the interview process. There was an introductory part, a main part which consisted of the questions and answers and the conclusion of the interview. The introductory part of the interview consisted of greeting the respondents and verbally obtaining informed consent from the respondents. The purpose of the study was also explained to the participants, their willingness to participate was verbally obtained, permission to record the session was verbally obtained and they were informed of their right to withdraw at any time of the study and confidentiality of their personal data.

The main part of the interview consisted of the questions and answers. In addition to pretesting the interview questions as discussed in the previous chapter; to further ensure validity the questions were phrased in a way that respondents understood what was being asked for. At every stage of the data collection process a check was done of how relevant the information is and how well it contributes to answering the research questions, this was possible in this portion of the study due to the qualitative nature of the data. Also by explaining to the participants what the study entails and what it is all about prevented them from attempting to outperform all other participants due to misinterpreting the purpose of the evaluation. The environment was made conducive for the participants to be able to think and recall past events well. This was done through the quiet and private setting of the interview rooms at the Bank A offices. During the question ad answers section the researcher was recording the responses with the permission of the participant using her mobile phone. The interview questions can be found on Appendix A.

At the end of the interview the participants were thanked for their time and asked if they had anything that they would like to add which gave them an opportunity to address matters that they thought about but were not addressed during the interview.

The data collected is being analysed and reported with honesty and the personal data of participants will be kept confidential.

5.3.2 Thematic Analysis

For the thematic analysis the researcher made use of the model/guidelines prescribed by Braun & Clarke (2006) for performing thematic analysis. The six phase model is summarised in Table 16 below.

Table 16: Guidelines for thematic analysis

Phase	**Description**
Phase 1: Familiarising yourself with your data	This phase involves repeated reading of the data, and reading the data in an active way – searching for meanings and patterns. If the data is verbal it will need to be transcribed into written form
Phase 2: Generating initial codes	This phase entails the development of initial codes from the data. The codes identify a feature of the data that appears interesting to the researcher. Coding can be done manually or using a software program
Phase 3: Searching for themes	This phase is concerned with arranging the different codes into possible themes and organising all

	the significant coded data extracts within the identified themes.
Phase 4: Reviewing themes	This phase involves the refinement of the themes. During this phase, the researcher may discover that some proposed themes are not really themes, this could be the case if there isn’t enough data to support them. Other themes may even collapse into each other and some may need to be broken down into separate themes.
Phase 5: Defining and naming themes	When the researcher has successfully refined and is satisfied with his/her thematic map, they define what the essence of each theme is about. The researcher also needs to establish what characteristic of the data each theme captures. Lastly the researcher needs to start thinking of the names to give to each theme during the final analysis.
Phase 6: Producing the report	At this stage the researcher must have a set of fully worked-out themes and can now undertake final analysis and write-up of the report.

Following the guidelines described on Table 16 the first step taken by the researcher was transcribing the data that was recorded from all the interviews.

During this process the initial views were recorded as this is regarded as an important step in analysis (Riessman, 1993). The transcribed data was then read and re-read several times and, furthermore, the recordings were listened to several times to confirm the accuracy of the transcription. This process of “repetitive reading” and using the recordings to listen to the data, gives rise to data immersion and speaks to the researcher's closeness with the data (Braun & Clarke, 2006). Following on from this initial stage and building on the transcripts and concepts produced through transcription and data immersion is the coding phase. These codes categorised aspects of the data that the researcher considered to be relevant to the research questions. In addition, as is essential to the method, the whole data set was given equal attention so that full consideration could be given to repeated patterns within the data. The third stage consisted of examining the data for themes; these explained larger sections of the data by linking different codes that may have been very similar or may have been considered the same aspect within the data. All initial codes pertinent to the research questions were merged into a theme. Braun & Clarke (2006) also propose the development of thematic maps to support the creation of themes. These helped the researcher to envision and study the links and relationships between themes. Figure 22 provides a portion of the thematic map that was drawn by the researcher.

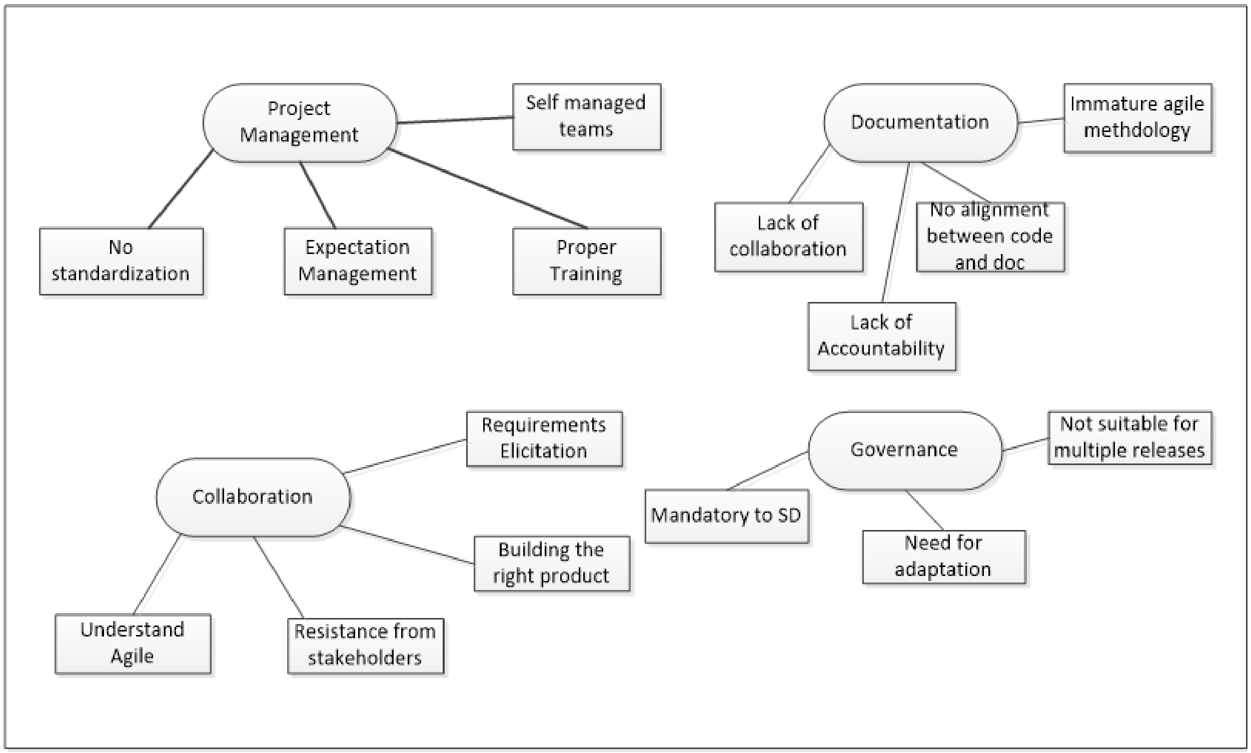


Figure 22: Thematic Map

At this point any themes that had insufficient data to support them or were too diverse were removed. This fine-tuning of the themes was done on two levels, mainly with the coded data making sure they made up a comprehensible pattern, secondly once a comprehensible pattern was established the themes were studied in relation to the data set as a whole. This ensured the themes correctly replicated what was apparent in the data set as a whole (Braun & Clarke, 2006). Further coding was done at this stage to ensure no codes had been overlooked in the earlier stages. Once an evident idea of the various themes and how they fitted together was established, analysis progressed to phase five. This includes defining and naming the themes, each theme needs to be clearly defined and supplemented by an in depth analysis. Considerations were made not only of the story told within individual themes but how these related to the overall story that was evident within the data. Furthermore, it was very important to come up with short but punchy names that communicated a direct indication of the fundamental nature of the theme (Braun & Clarke, 2006). The final stage or the report production involved choosing examples of transcript to illustrate elements of the themes. These quotations clearly identified subjects within the theme and presented a simple example of the point being made.

5.3.3 Presentation of Qualitative Data

The thematic analysis process that was applied to the transcripts elicited key concepts that have become the main themes for the data analysis. It is envisaged that these themes represent a holistic conflation or an aggregation of the views expressed by the interviewees at a more granular level. As suggested in Shroff (2008), the identification of themes has been conducted in conjunction with the objective of ensuring that the themes have a relevance to the main research questions. These themes have been labelled as: "Project Management Approach," "Need for Collaboration," "User Requirements," "Governance process," and "Documentation". It should be noted that the evidence used to support the identification of a specific theme is not mutually exclusive and may overlap with other themes. Also, the presentation of evidence in support of each theme will be done by using a strategy of indirect reporting and the use of verbatim transcripts taken from the interview sessions. Whilst these verbatim transcripts may be perceived as being somewhat monotonous, the researcher holds the opinion that the impact that the verbatim transcripts make will be diluted by indirect reporting of the concept under scrutiny.

Theme 1: Project Management Approach

This theme is defined by the views expressed by the various software professionals on the PM process within the organisation in reference to the way that software was previously developed and the context for PM in view of the implementation of agile methodology for software development (SD).

Firstly, the respondents highlighted the importance of ensuring that the project manager has a good understanding of the PM principles in order for them to ensure that they are properly implemented in the project regardless of what methodology is being followed. There were specific references made to the concerns that ASDM are inadvertently regarded as a replacement of traditional project management (TPM) principles. The comment, documented in the transcript below, is representative of this general view held by many of the project managers.

> *“Agile by itself does not ensure that the project management practice is implemented so it rests on the project manager to ensure that the principles are properly implemented”*
>
> **– Project Manager**

However despite the project managers displaying an understanding of what their role is in ensuring that PM principles are upheld, there is evidence of confusion on what is expected of project managers in agile projects. The preceding assertion is corroborated by the verbatim transcript from an interview with one of the project managers.

> “Our *project management style in the new agile way of working is still immature and there is no structure, we just do whatever work is given at whatever time and do as we are told by program managers because we are not really sure how project management should be done…I’m not really sure if the role of a project manager is still relevant”*
>
> – **Project Manager**

This may result in the lack of a standardised PM approach within the organisation due to every team using the approach that they believe is best. Respondents have indicated that this causes confusion and presents a challenge when it comes to synchronising changes and deployments due to the different timelines and processes followed. This also makes it difficult for project team members to gauge on how well they know the methodology because each project is doing it differently. A major source of concern in this regard is the confusion that seems to emanate from the integration of TPM principles and an agile-like mentality. This tension between the 2 different paradigms of development is reflected in the verbatim transcripts below.

"*Currently at the bank there is a split between traditional project management principles and agile. There are certain project managers that are still making use of the traditional way of managing a project while others have fully adopted Agile. This can be confusing when you are required to move from one project to the other*"

– **Business Analyst**

"*In large programs where there are many project streams and half of those are making use of traditional ways of managing projects e.g. waterfall methodology and the other half is using agile it becomes extremely difficult to deliver the project due to misalignment*"

– **Project Manager**

The PM theme is used to encapsulate the discourse on PM in the context of agile software development (ASD). This reporting around this theme focuses on the identification of areas of concern that project managers have with regards to the use of agile methodology (AM) and its compromising influence of TPM principles. Holistically, it may be inferred from the responses that the current approach to PM breeds frustration amongst the project team because of the perceived incompatibility between PM principles which enforce accountability and a high level of structure over the perceived lack of structure to AM. This incompatibility manifests in an incongruous view of the trajectory of a software project, between the project manager(s) and the rest of the team. Project team members see the role of a project manager in a completely different way from how they view their role. There is also a difference in expectations on what PM should look like in the new (agile) ways of working. Project managers have not yet received proper training on AM and resulting in a lack of knowledge of how their role is expected to evolve in an agile context. A possible reason for this untenable situation is that the advent of AM in the bank is not being driven from a strategic and core level resulting in fragmented support for the methodology. This fragmented approach to the implementation of AM has been identified as a major impediment to the successful implementation of the methodology. The preceding assertion has a pivotal role in the context of the current study and has been identified as a major theme that has emanated from the interview sessions. The need for a collaborative effort in ensuring that the agile intervention is successful is discussed as part of the next theme.

Theme 2: Need for Collaboration

Collaboration between business stakeholders and project teams enables business stakeholders to understand what agile entails and what is required of them. This makes it easy for business stakeholders to provide the support needed by agile teams, including project managers, to effectively transition into new ways of working. The preceding statements are an embodiment of the viewpoints expressed in the interviews. According to many of the interviewees, collaboration between the different sectors in the bank were pivotal to ensure that the correct software system is developed. The collaboration will be instrumental in simplifying the requirements elicitation process thereby paving the way for the development of a successful system. This sentiment is highlighted in the following verbatim transcripts. For example some of the respondents believe that the requirements elicitation process could be simplified if there was more collaboration

> *"Requirements Elicitation would be simpler if there was more collaboration between IT and Business"*
>
> **– Business Analyst**

The following verbatim transcript demonstrates the beliefs that the majority of the respondents had on the importance of collaboration in ensuring that they deliver quality products that meet user requirements.

> *"Collaboration is extremely important in these projects especially between business stakeholders, Business Analysts and Designers to ensure that what eventually gets developed is what needs to be developed to avoid reworks"*
>
> **– Developer**

A major cause of concern in this regard is the negative impact that the lack of collaboration seems to have on agile adoption at the organisation. This is demonstrated in the 2 transcripts below:

> *"The disconnection between IT and business is hindering our progress, we don't see each other as partners and as one team. When something goes wrong there's no accountability, there's always pointing of fingers and blame shifting. We are just not jellying as well as we should be"*
>
> **- Project Manager**
>
> *"Business stakeholders need to adopt the agile mind-set and be more collaborative because in agile you need to constantly question the requirements by adapting and confirming with the user. People refuse to collaborate, they want to conform to their old roles and this is a major challenge"*
>
> **– Program Manager**

To substantiate the preceding assertion, 80% of the respondents thought that the resistance to collaborate is caused by business stakeholders' lack of understanding of what it entails to adopt an agile strategy. This fact is reflected in the following verbatim transcript which serves as an example of the views of respondents:

“There is no understanding of what it means to “go agile” from the business side hence the resistance when it comes to collaborating” **– Project Manager**

The reporting on this theme focuses on the evaluation of how a lack of collaboration impacts on agile adoption at this organisation. Taking all the views of the respondents into account, it can be concluded that the lack of collaboration between the business and IT project teams is an impediment for this organisation effectively adopting ASDM. The respondents believe that this impediment is a result of the lack of understanding by business stakeholders of what it means to be agile as a bank. 90% of the respondents expressed the importance of collaboration in ensuring that there is shared accountability for every decision taken in the projects. They also believe that collaboration is vital in ensuring that the correct software systems are delivered through proper requirements elicitation. Further, the respondents believe it is not possible to develop software systems that assist in solving business problems because of the disconnection between the two areas and lack of understanding of business processes which is further exacerbated by the lack of collaboration. The need for a collaborative effort in ensuring that the agile intervention is successful has been identified as one the major themes that emerged from the interviews. The next theme examines how a structured approach to user requirements elicitation and in managing those requirements can aid in the successful adoption of agile.

Theme 3: User Requirements

There was a united consensus that user requirements played an important role in developing the correct product that meets the needs of users. Elements of this theme have been demonstrated throughout the discussion of the previous theme, this demonstrates how they view the current process of requirements elicitation and how it can be improved. Firstly, there were references made to the concern that the requirements elicitation process is highly dependent on the individual performing it, the process is not repeatable which makes it difficult to measure continuous improvement. The comment in the transcript documented below represents, the collective view of most of the business analysts that were interviewed:

> *"There is no standard way of carrying out requirements elicitation across the bank, the way this is done varies greatly from Business Analyst to Business Analyst and from Area to Area. How well it's done is always dependent on the strength of the individual Business Analyst. So there is no repeatable process and quality is not guaranteed"*
> **– Business Analyst**

There is evidence that displays that this is not the only challenge being experienced when it comes to the requirements elicitation process. Respondents expressed concern regarding the fact that not all requirements are addressed particularly non-functional requirements are not covered during this process. This general concern amongst the respondents is captured in the verbatim transcript below:

> *"The agile methodologies that are currently being used in the bank do not address non-functional requirements, we often have to go back to the waterfall methodology to accommodate this. I would love for agile methodologies to address non-functional requirements for example IT Security as this is important due to regulatory requirements imposed on us as a bank"*
> **– Project Manager**

However despite the requirements elicitation process addressing non-functional requirements to meet security, regulatory and compliance obligations, there is evidence of dissatisfaction with how requirements change management is handled. This was believed to be particularly important due to the fact that there are finite resources allocated to each project. Project managers expressed that budget and time are fixed for each project and they are required to manage the scope hence proper change management is mandatory to ensure that the important functionality is deployed first. This is also to ensure customer satisfaction. These sentiments are reflected in the example transcript below:

> *“We are currently failing to prioritise the important functionality and understanding what is most important to customers so that the important functionality is released first”*
>
> **– Program Manager**

This theme focused on attempting to establish whether there are any aspects of the requirements elicitation process that are hindering agile from being successfully adopted at this organisation. The responses revealed that the project stakeholders are not satisfied with how requirements are gathered, they expressed concern regarding the lack of standardisation and repeatability of this process. Respondents highlighted the difficulty in measuring how well it’s being carried out and continuous improvement. In addition to the lack of consistency in how these requirements are gathered, a bigger concern was that non-functional requirements, IT security and compliance requirements as an example are not addressed. Non-functional requirements such as the aforementioned examples cannot be overlooked by Bank A as it is a financial institution. Despite meeting regulatory and compliance requirements through addressing non-functional requirements, there is still a greater need to provide customer satisfaction which is often accomplished by ensuring that the functionality that is important and adds value to them is delivered first and the respondents believe that the lack of a structured approach to requirements change management prevents this from happening. The next theme will be focused on how the governance process is preventing a successful implementation of ASD in this organisation.

Theme 4: Governance process

This theme is defined by the ideas expressed by the respondents on the governance process within the organisation in reference to the way that software was previously developed and the context for governance in view of the implementation of AM for SD.

Firstly, there was a unanimous understanding amongst the program managers that governance is mandatory and it cannot be circumvented regardless of the software development methodology that is being applied. Program managers believe that governance is important because they need to ensure that proper integration with existing infrastructure is accomplished each time something new is deployed. Additionally they expressed the importance of ensuring that new deployments do not cause outages on the bank's existing systems or inappropriate changes to security patterns as this would have a very negative impact on the bank's customers. The following verbatim transcript expresses this common viewpoint:

> *"The larger the project the more governance that needs to be enforced …We simply cannot get away with not enforcing governance especially being a financial institution and being heavily regulated. We need to ensure that integration is done properly and no unforeseen damages are caused on existing systems due to changes being deployed carelessly especially from an IT security perspective"*

– **Program Manager**

The other respondents seemed to agree with the idea of governance however they felt that the governance process should be adapted to be more suitable for AM. The following transcript captures the essence of this assertion:

> *"The Governance overhead makes it extremely challenging to effectively implement agile for example we have ESCAB requests before any change to production systems can be done, Impact Assessments, IT Security sign-offs, Release Board Approvals and all these gates need to be passed through before a release. This type of Governance worked for Waterfall because there was one release so this would typically be done once for each project however in Agile we have about 12 releases for each project which means this has to be done 12 times for one project"*
>
> **– Project Manager**

This theme reveals the importance of governance and further provides a clear understanding on why the governance portion cannot be overlooked. Program managers expressed this importance in terms of maintaining the existing systems in a secure state and ensuring that these systems are always available for customers to use. However, despite the criticality of having a strict governance process the other respondents were in consensus that there is a need for an adaptation to the governance process. This adaptation is seen as necessary due to the fact that the current governance process was made to work well for the waterfall methodology and its few releases. AM have more frequent releases and there are currently too many approvals that have to be obtained from various governance boards and this impacts on project timelines. Finally, the role played by the current documentation strategy in thwarting the progress towards a successful agile implementation will be discussed. This is the final theme that emanated from the interview responses.

Theme 5: Documentation

This theme is aimed at looking into the documentation strategy that is currently being followed for SD projects.

Firstly project managers express their frustrations with the minimal documentation strategy advocated for by AM. They believe that due to the immaturity of the agile teams and lack of collaboration between IT and business stakeholders, it is presenting more challenges than benefits. The challenge that was highlighted was the issue of accountability. As an example business stakeholders will make modifications to their requirements and this is not recorded anywhere that they requested this change and they sometimes cannot account for these changes when the software has been deployed and they are not happy with what has been deployed. The transcript below expresses this viewpoint:

> *“In Agile we are told to document as minimal as possible however I believe this causes a problem for the project teams. Business makes so many changes to their requirements and proving that what you delivered is what they requested is often problematic because it’s not documented anywhere so I feel that documentation should be comprehensive. I also believe that this is related to maturity because business should be seen as being part of the team instead of being an external party however right now they’re still external to the project team and they refuse to take accountability for anything”*
>
> **– Project Manager**

Respondents also felt that Quality Assurance (QA) sessions need to be considered for documentation to ensure that the correct information has been recorded and it has been done at an acceptable standard that the developers can use. The transcript below gives captures this:

> *"I believe it would be very helpful if we were to have QA sessions for the documentation to ensure that the documentation is at the right standard to be passed on to developers"*
> **– Program Manager**

This theme revealed that the minimal documentation strategy enforced by AM is seen as a challenge given the immaturity and lack of collaboration between the teams. This means that everything has to be documented to ensure that there is accountability for every decision and every implementation made during the project. It was also noted that QA is needed to ensure that correct information has been captured.

The results detailed above highlight some important findings as to what PM related challenges of implementing Agile at this Bank are encountered by software development practitioners. These findings were used in formulating a framework that would attempt to address the highlighted challenges. The formulation of this framework is discussed in the following subsection.

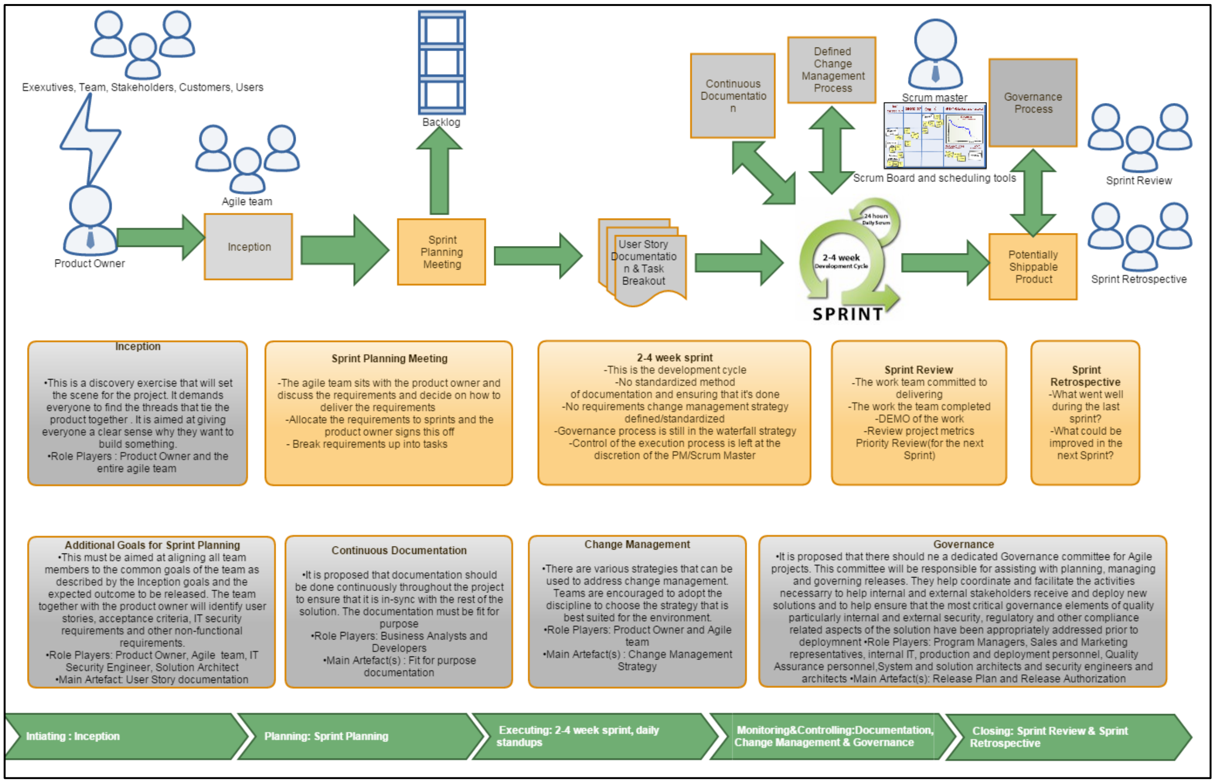
Agile-PM Framework

To formulate the framework the researcher used the Agile Manifesto (Appendix E) and the PMBOK process groups (PMPG) (Appendix G).

The Agile Manifesto focuses on the individuals and their interactions over processes and tools, working software over comprehensive documentation, customer collaboration over contract negotiation and responding to changes over following a plan. In the proposed framework the researcher used the values of the agile manifesto as the baseline for development, emphasising more of the working software, documenting only appropriate needs, with collaboration and interaction between the team and the business stakeholders/customer.

PMBOK defines five PM processes which form a series of actions directed towards a particular result. These are initiation, planning, executing, monitoring and controlling & closing. The planning process includes devising and maintaining a workable plan to further track the progress of the project and the stakeholders' needs. Any change in the project is well evaluated and managed, and can be adjusted in the plan accordingly. Executing is the process whereby the actual implementation of the plan is performed, resulting in the produced products, services or results. Monitoring and controlling is performed on a regular basis to ensure that the development is progressing in the right direction to achieve the project goal. The closing process is an orderly end of the project or any phase of it. In the proposed framework, PMPG suggests the flow of the project. It also shows the interdependencies of the phases involved in the project and the way the TPM can be incorporated into the agile process. Table 26 on Appendix G demonstrates how the phases and processes in the proposed framework map onto the PMBOK PM processes.

Figure 23 below presents a summary of the framework and the full framework can be found on Appendix G.


Exevutives, Team, Stakeholders, Customers, Users
Product Owner
Agile team
Inception
Backlog
Sprint Planning Meeting
User Story Documentation & Task Breakout
Continuous Documentation
Defined Change Management Process
Scrum master
Scrum Board and scheduling tools
Governance Process
24 hours Daily Scrum
2-4 week Development Cycle
SPRINT
Potentially Shippable Product
Sprint Review
Sprint Retrospective
Intiating : Inception
Planning: Sprint Planning
Executing: 2-4 week sprint, daily standups
Monitoring&Controlling:Documentation, Change Management & Governance
Closing: Sprint Review & Sprint Retrospective

Inception

•This is a discovery exercise that will set the scene for the project. It demands everyone to find the threads that tie the product together . It is aimed at giving everyone a clear sense why they want to build something.
•Role Players : Product Owner and the entire agile team

Sprint Planning Meeting

-The agile team sits with the product owner and discuss the requirements and decide on how to deliver the requirements
-Allocate the requirements to sprints and the product owner signs this off
- Break requirements up into tasks

2-4 week sprint

-This is the development cycle
-No standardized method of documentation and ensuring that it's done
-No requirements change management strategy defined/standardized
-Governance process is still in the waterfall strategy
-Control of the execution process is left at the discretion of the PM/Scrum Master

Sprint Review

-The work team committed to delivering
-The work the team completed
-DEMO of the work
-Review project metrics Priority Review(for the next Sprint)

Sprint Retrospective

-What went well during the last sprint?
-What could be improved in the next Sprint?

Additional Goals for Sprint Planning

•This must be aimed at aligning all team members to the common goals of the team as described by the Inception goals and the expected outcome to be released. The team together with the product owner will identify user stories, acceptance criteria, IT security requirements and other non-functional requirements.
•Role Players: Product Owner, Agile team, IT Security Engineer, Solution Architect
•Main Artefact: User Story documentation

Continuous Documentation

•It is proposed that documentation should be done continuously throughout the project to ensure that it is in-sync with the rest of the solution. The documentation must be fit for purpose
•Role Players: Business Analysts and Developers
•Main Artefact(s) : Fit for purpose documentation

Change Management

•There are various strategies that can be used to address change management. Teams are encouraged to adopt the discipline to choose the strategy that is best suited for the environment.
•Role Players: Product Owner and Agile team
•Main Artefact(s) : Change Management Strategy

Governance

•It is proposed that there should ne a dedicated Governance committee for Agile projects. This committee will be responsible for assisting with planning, managing and governing releases. They help coordinate and facilitate the activities necessarry to help internal and external stakeholders receive and deploy new solutions and to help ensure that the most critical governance elements of quality particularly internal and external security, regulatory and other compliance related aspects of the solution have been appropriately addressed prior to deploymnent •Role Players: Program Managers, Sales and Marketing representatives, internal IT, production and deployment personnel, Quality Assurance personnel,System and solution architects and security engineers and architects •Main Artefact(s): Release Plan and Release Authorization

Figure 23: Proposed Agile-PM framework

The framework was aimed at answering the second research question

- How can the PM related challenges of implementing AM at Bank A be mitigated?

A discussion on how this framework provides an answer to this research question will be provided in Chapter 6.

Table 17 below compares the proposed Agile-PM framework with the two frameworks that were discussed in Chapter 2.

Table 17: Comparison of Agile-PM framework with similar frameworks

	Scrum & PRINCE2	Hybrid APMM	Agile/PM Framework
Management Style	Leadership, Collaboration & Control	Leadership, Collaboration & Control	Leadership, Collaboration & Control
Change Management	Not addressed	Explicit – iterative refactoring cycle	Explicit – Dedicated Change Management strategy options
Risk Management	Not addressed	Explicit – Covered in the prep phase, a risk mitigation and issue resolution plan is delivered	Explicit – Covered in the governance strategy
Documentation	Follows the agile minimum documentation strategy. Documentation produced includes: business case, product backlog, highlight report and burn-down chart	Comprehensive and continuous documentation strategy. Documentation produced includes : business case, project charter, project schedule guideline and product backlog	Useful and fit for purpose. Continuous documentation strategy and documentation forms part of the acceptance criteria. Documentation produced includes : user manuals, training materials, operation manuals, product backlog and system overviews
User requirements	Interactive input (high-level requirements delivered upfront); non-functional	Interactive input. Collected through workshops, JAD sessions, questionnaires or story cards; no	Collaborative effort and alignment of user requirements with architectural vision of organisation given

	requirements not addressed	specific reference made to non-functional requirements	particular attention. Critical non-functional requirements e.g. IT Security requirements are also addressed
Primary Objectives	Quick value and high safety of the project as a whole	Deliver IT projects in constantly changing project and business environments	Make ASDM suitable for projects with high PM demands
Development model	Evolutionary delivery model	Lifecycle model	Lifecycle model
Size	All project sizes	All project sizes	All project sizes

The following portion of the study was aimed at evaluating user acceptance towards the proposed framework and the following section presents the results for this evaluation.

5.4 Quantitative Data Analysis

This section presents the analysis and findings of the quantitative data collected from the survey questionnaires. The data collected for the current study was exported into MS Excel from the questionnaire. The quantitative data was inspected for abnormalities, cleaned and imported into SPSS. The questionnaire consisted mostly of Likert Scale type of questions that ranged from "Strongly Agree" to "Strongly Disagree". With regards to the data preparation, the data was captured in Microsoft Excel format and then subsequently imported into SPSS for further analysis. Coding of the data involved assigning a value of 5 to "strongly agree" and a value of 1 was assigned to "strongly disagree.

In the following sections the demographic statistics are reviewed, and secondly the descriptive statistics presented, which include reliability, frequency and mean analysis. Afterwards the correlations between construct items is analysed via Pearson's correlation and Regression Analysis.

5.4.1 Demographic Statistics

Survey sample characteristics are illustrated in Table 18, comprising of 61 responses. These included only agile practitioners with one to five years of experience in using Agile (96.72%) and 5 to 10 years of experience in agile (3.28 %). The demographic profiles of respondents show that male is the slightly more dominant gender group with 50.8% of respondents, while females constituted 49.2%. Regarding age composition, it is clear that respondents are predominantly young people, with 46.2% of the respondents between 26 and 30 years of age; also 27.9% between ages of 31 and 35.

Table 18: Profile of respondents

	Frequency	***Percentage***
Gender		
Female	30	49.2 %
Male	31	50.8 %
Total	61	100 %
Age		
21-25	9	14.8 %
26-30	26	46.2 %
31-35	17	27.9 %
36-40	6	9.8 %
41-45	1	1.6 %
46-50	2	3.3 %
Total	61	100 %
Years of Experience in Agile		
Under 1	0	0.0 %
1 -5	59	96.72 %

5- 10	2	3.28 %
Over 5	0	0.0 %
Total	61	100 %

In terms of job title/position, significant ratios of respondents were from the Business Analysis (24%) or Project Management (21%) disciplines, as it is presented in the graph on Figure 24.

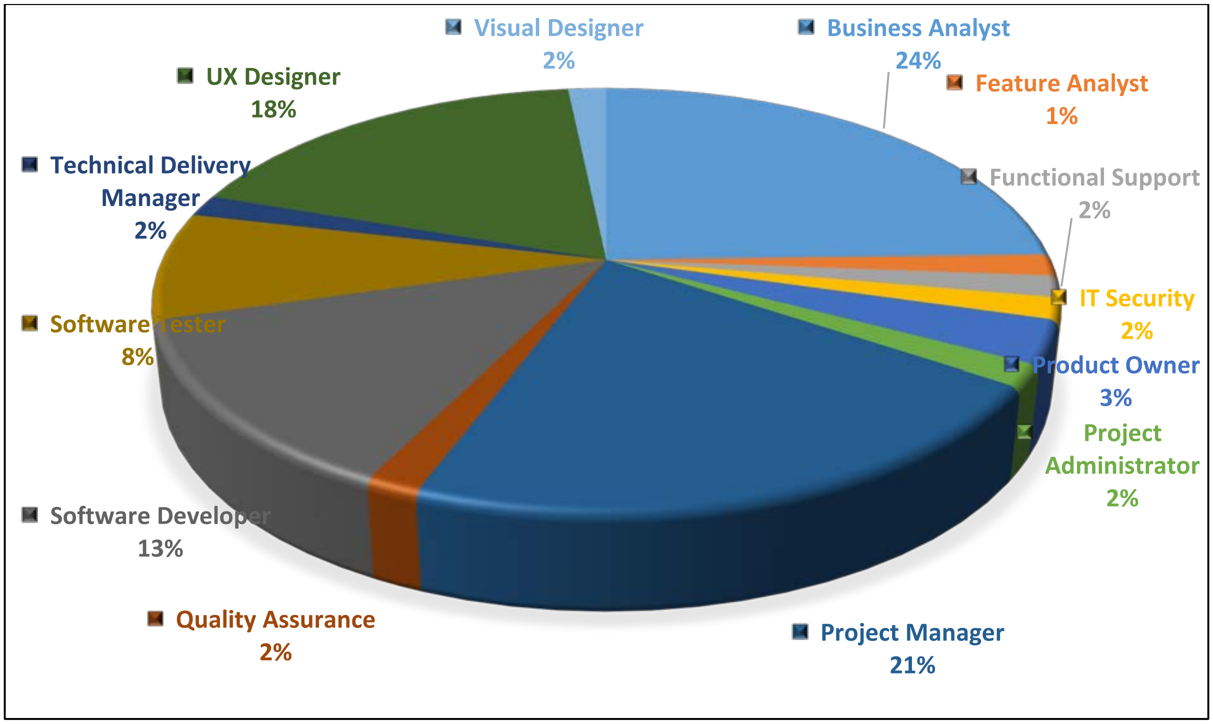


Figure 24 : Respondents occupation statistics

5.4.2 Construct Validity and Reliability

According to Sekaran & Bougie (2010) when many items are used to measure a single construct, the factor analysis technique should be used to confirm that the data obtained is a valid measure for that construct. Factor analysis, also used by Davis (1989) in his study, is *"…a multivariate technique that confirms the* *dimensions of the concept that have been operationally defined"* (Sekaran & Bougie, 2010). In the case of the current study, the four dimensions/factors identified in accordance with the UTAUT theoretical framework were Performance Expectancy (PE), Social Influence (SI), Effort Expectancy (EE) and Facilitating Conditions (FC). SPSS was used to conduct a principal component analysis using varimax rotation, and specifying a four factor solution (to match each of the dimensions of the study as mentioned above). Table 19 shows the results of the factor analysis. It can be seen in Table 19 that all items load highly on their predicted factors

Table 19: Four dimensional factor analysis

	Rotated Component Matrix[a]				
		Component			
		1	2	3	4
Performance Expectancy	The proposed model would be useful for me to use in software development projects	0,886	0,220	0,149	0,247
	Using the proposed model would enable me to ensure that those aspects of software development that are important to me are upheld	0,885	0,187		0,130
	Using the proposed model would help me to do my software development related activities more quickly	0,832	0,138		
	Using the proposed model would increase my productivity with regards to software development activities	0,808	0,244	0,159	0,105

Social Influence	People who iinfluence my behaviour will endorse my preference to make use of the proposed mo2el	0,189	0,895		0,104
	People whose opinions that I value will endorse my preference to make use of the proposed model	0,221	0,876	-0,102	0,129
	People who are important to me in my job domain will endorse my preference to make use of the proposed model	0,261	0,845	0,243	
Effort Expectancy	I predict that learning how to apply this model to my software projects would be easy to do		0,142	0,867	0,152
	I predict it will be easy for me to become skillful at using the proposed model	0,283	-0,142	0,815	
Facilitating Conditions	I have at my disposal, the resources necessary to make use of the proposed model		0,284	0,391	0,725
	I will be able to obtain help from others when I have difficulties in using the proposed model	0,253	0,140	-0,179	0,722
	I have the required knowledge to make use of the proposed model	0,221		0,220	0,702

	Extraction Method: Principal Component Analysis. Rotation Method: Varimax with Kaiser Normalization.
	a. Rotation converged in 6 iterations.

Having confirmed the construct validity of the measurement instrument used in the study, the next step was to perform a Cronbach’s Alpha computation to ensure “interterm consistency” reliability is achieved, i.e. testing the consistency of respondents’ answers to all questions in specific groups or subsections (Sekaran & Bougie, 2010). Cronbach’s Alpha is commonly used in research to measure the internal validity or the “…cohesiveness of the individual question” (Alhujran, 2009). In this study Cronbach’s coefficient alpha value was assessed to examine the internal research consistency of measuring (Field, 2005). Hinton *et al.* (2004) propose four degrees of reliability scale: excellent (0.90 and above); high (0.70 to 0.90); high moderate (0.50 to 0.70) and low (0.50 and below). Pallant (2005) states that Cronbach’s coefficient alphas of 0.70 and above are deemed acceptable. Moreover Hair *et al.* (2007) mentioned that construct reliability should be 0.7 or higher to indicate adequate convergence or internal consistence (Hair *et al.*, 2007). According to the current realest model as well as Venkatesh *et al.* (2003), the construct constituting UTAUT should have a good internal consistency with a reported Cronbach’s alpha value greater than 0.70. To prove that the scales used in the survey questionnaire satisfied the model constructs consistently and accurately, a reliability coefficient was run on SPSS for each set of constructs and results are presented in Table 20 which shows the Cronbach’s alpha value for each variable. The results of the analysis show that all of the constructs got a high reliability of more than 0.7 Cronbach’s value result varied between 0.748 for EE and 0.950 for FC. Overall, the result show that all alpha values of the study instrument are reliable and exhibit appropriate construct reliability.

Table 20: Cronbach's alpha reliability results

Constructs	No. of items	Cronbach's Alpha	Comments
Performance Expectancy (PE)	4	0.920	Excellent Reliability
Social Influence (SI)	3	0.895	High Reliability
Effort Expectancy (EE)	2	0.748	High Reliability
Facilitating Conditions (FC)	4	0.950	Excellent Reliability

5.4.3 Descriptive Statistics

It is reported in Sekaran & Bougie (2010) that several questions may be used to measure a single concept. In order to obtain a measure of quantification, "… scores on the original question have to be combined into a single score" (Sekaran & Bougie, 2010). In accordance with this suggestion, the analysis of the responses was conducted by collapsing the individual measures of the perception variables into 4 single independent variables that represented the mean of the individual responses. The independent variables represented Performance Expectancy (PE), Effort Expectancy (EE), Social Influence (SI) and Facilitating Conditions (FC).

The mean value derived for PE for the framework is 3.90 (ranked on a scale from 1 to 5 where 1 indicates strong disagreement that the framework is perceived as possibly bringing job benefit and 5 indicates strong agreement that the framework will bring about job benefit). This value is generally endorsed as acceptable (*M*=3.90), as reflected in Figure 25. This was achieved for an N value of 61 respondents. The results are significant at the 95% confidence level (the standard error is 0.028), reflecting that the mean value computed for PE is representative of the population of agile practitioners at Bank A. The result is indicative of a reliably strong consensus that the agile-PM framework proposed in this study will have a positive influence on the software practitioners' job performance when projects involving agile methodology are undertaken.

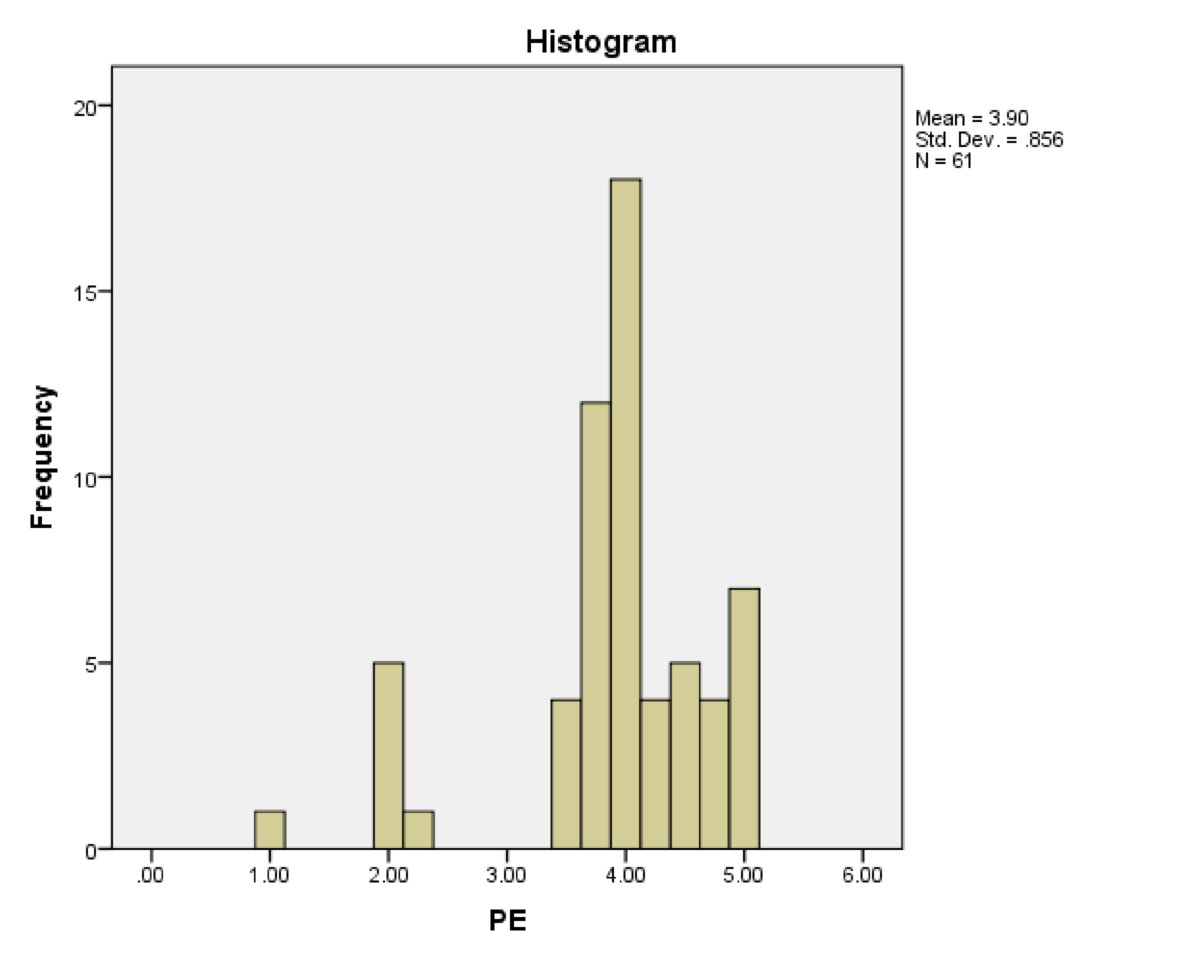


Figure 25: Mean values for PE

The mean value obtained for EE of the framework is 3.84 (also ranked from a scale from 1 to 5, where 1 indicates strong disagreement that the use of the framework will be easy and effortless and 5 indicates strong agreement that the use of the framework will be easy and effortless)as is reflected in Figure 26. The outcome suggests a consistently strong agreement that the agile-PM framework proposed in this study will be easy to use when projects involving agile methodology are undertaken.

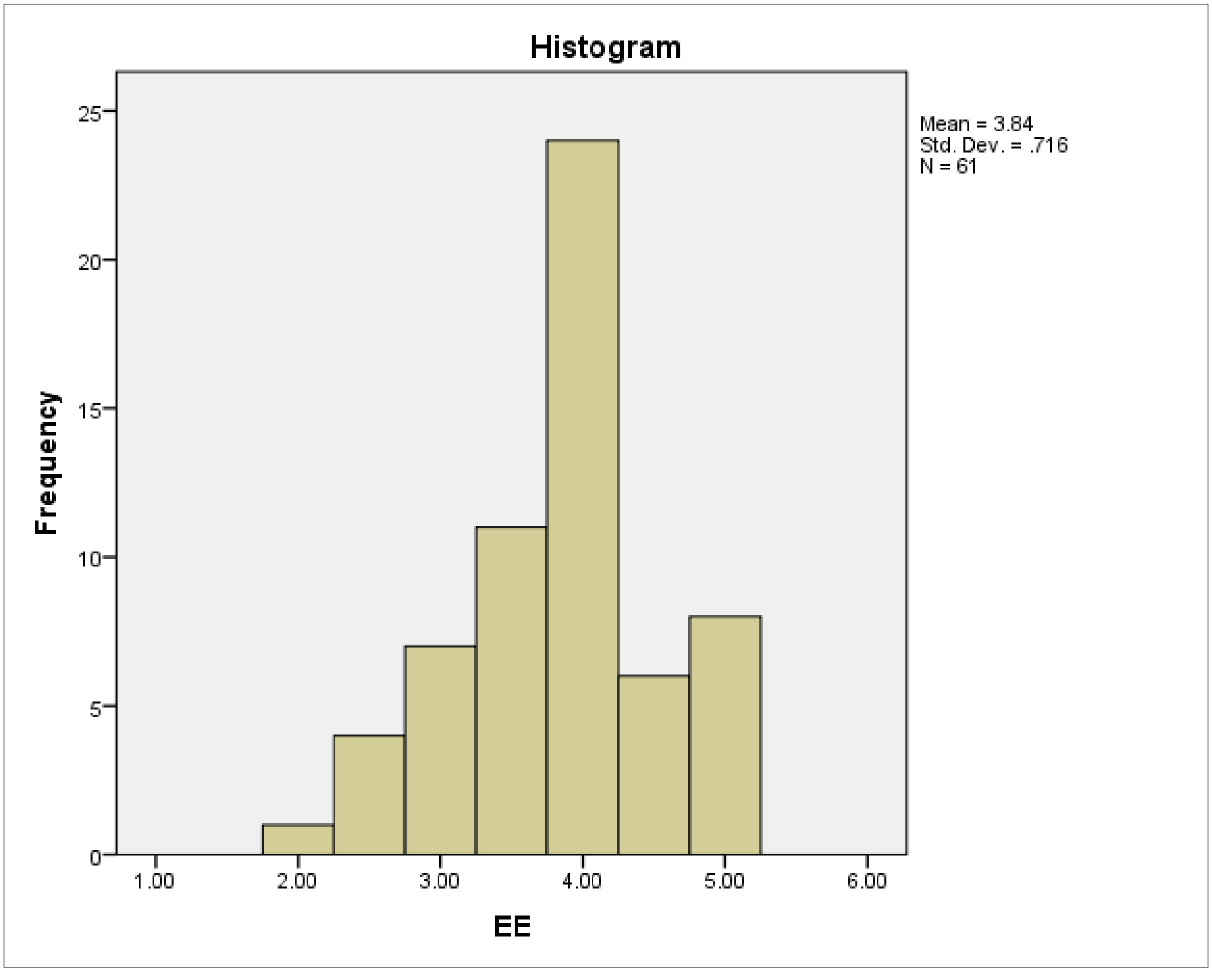


Figure 26: Mean values for EE

SI is generally endorsed as very good (M=3.65; significant at the 95% confidence level)), as reflected in Figure 27. SI was also ranked from a scale from 1 to 5, with 1 indicating a strong disagreement that significant people believe they should make use of the proposed framework and 5 indicates strong agreement that significant people believe they should make use of the proposed framework. The resulting mean value indicates a strong consensus amongst the respondents that significant people believe that they should make use of the proposed framework.

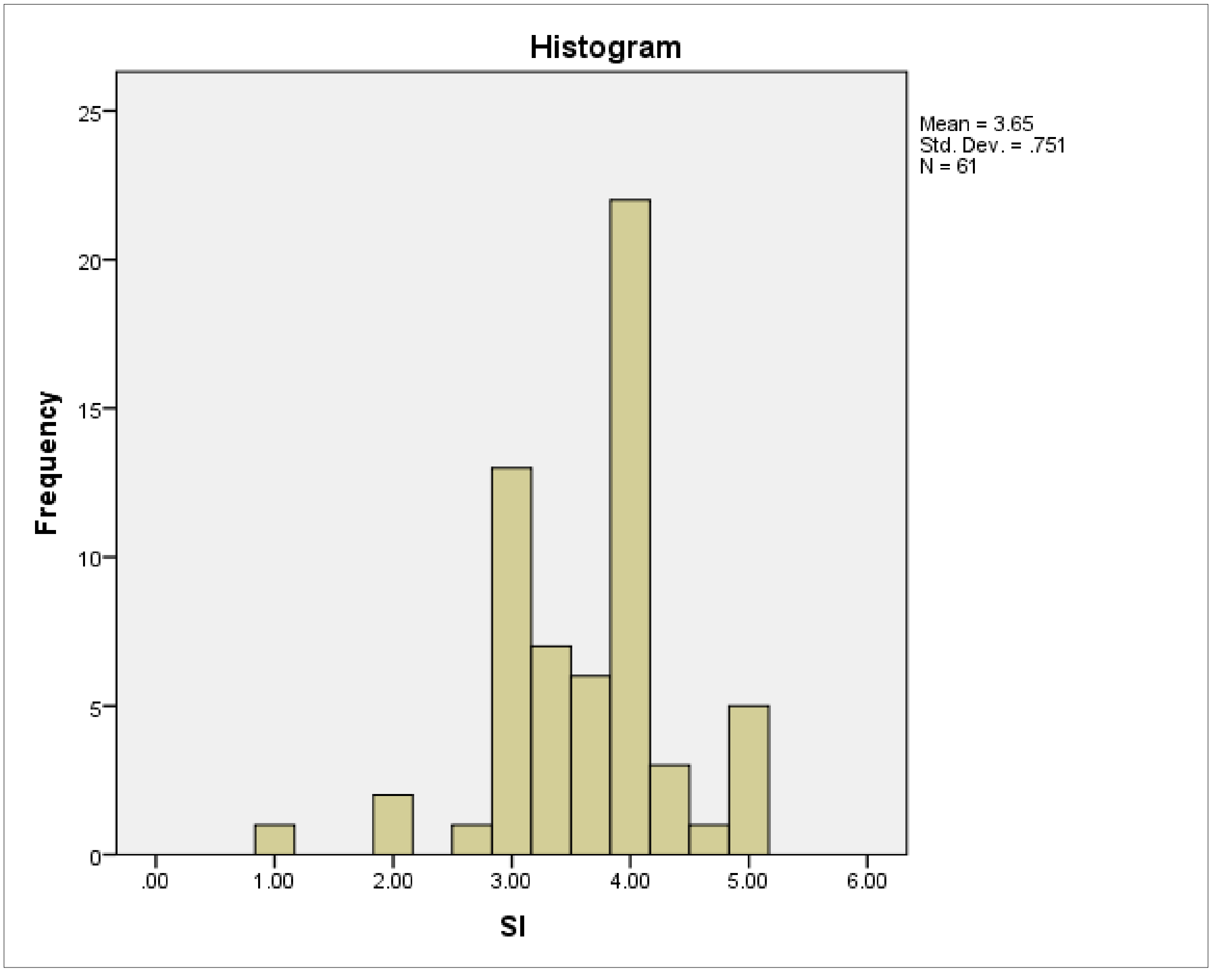


Figure 27: Mean values for SI

FC are generally endorsed as very good (*M*=3.82; significant at the 95% confidence level)), as reflected in Figure 28. This was also measured on a scale from 1 to 5, with 1 indicating a strong disagreement that organisational and technical infrastructures will support the use of the proposed framework and 5 indicates strong agreement that organisational and technical infrastructures will support the use of the proposed framework. M = 3.82 is endorsed as very good because it indicates a strong agreement amongst the respondents that the organisational and technical infrastructures will support the use of the proposed framework.

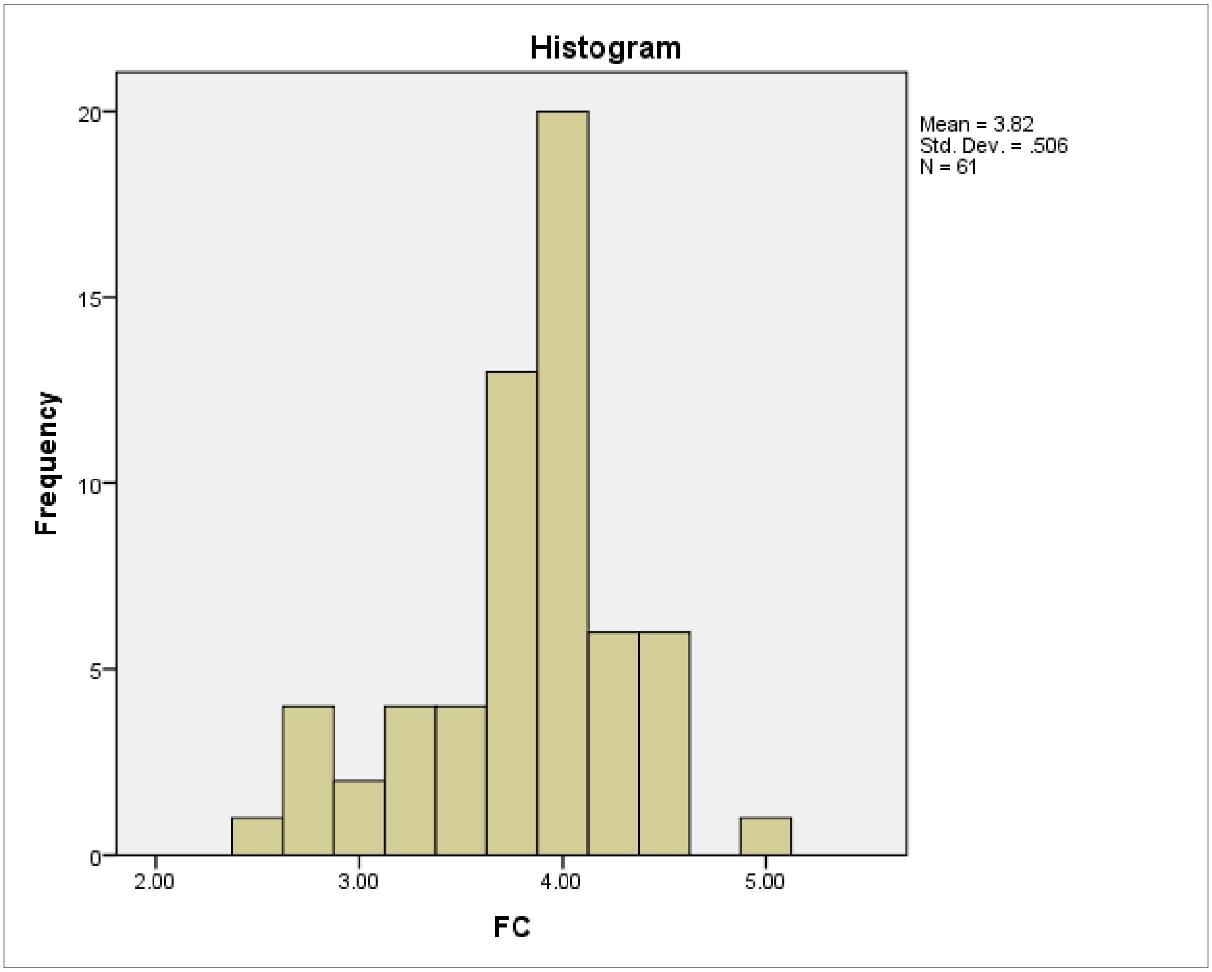


Figure 28: Mean values for FC

5.4.4 Pearson’s Correlation

At first, the relationship between PE and BI for the framework was studied, as the UTAUT model theorises that there is a positive correlation between these two variables of the model. A Pearson correlation analysis was conducted in the context of the data for the current study in order to confirm such a relationship thereby presenting a case for consistent acceptance of the proposed agile-PM framework. From Table 21 it is clear, it can be seen that a correlation co-efficient of .671 has been recorded, thereby indicating a relatively strong positive correlation between PE and BI.

Table 21: Pearson's correlation between PE and BI

		BI	PE
BI	Pearson Correlation	1	.671**
	Sig. (2-tailed)		,000
	N	61	61
PE	Pearson Correlation	.671**	1
	Sig. (2-tailed)	,000	
	N	61	61

**. Correlation is significant at the 0.01 level (2-tailed).

Since the EE questions were all positively worded, a high value for EE would indicate that the effort required to make use of the proposed framework is relatively low, indicative of a positive outcome. This should then correlate quite well with BI. As can be seen in Table 22, the correlation co-efficient of 0.274 is positive, thereby suggesting a positive correlation, the strength of the correlation is quite weak, however the results show significant correlation, therefore the relationship between EE and BI for the framework is not being eliminated at this point.

Table 22: Pearson's correlation between EE and BI

		BI	EE
BI	Pearson Correlation	1	.274*
	Sig. (2-tailed)		,033
	N	61	61
EE	Pearson Correlation	.274*	1
	Sig. (2-tailed)	,033	
	N	61	61

*. Correlation is significant at the 0.05 level (2-tailed).

Thirdly, the relationship between SI and BI for the framework is described. The SI questions were all positively worded hence a high value for SI would indicate that significant people believe that the respondents should make use of the proposed framework, indicative of a positive outcome. The results of the Pearson's correlation indicate a strong, positive correlation between SI and BI (r = 0.443; N= 61; p < .001).

Table 23: Pearson's correlation between SI and BI

		BI	SI
BI	Pearson Correlation	1	.443**
	Sig. (2-tailed)		,000
	N	61	61
SI	Pearson Correlation	.443**	1
	Sig. (2-tailed)	,000	
	N	61	61

5.4.5 Regression Analysis

A multiple linear regression was conducted to confirm Pearson's analysis. This statistical tool is used for the investigation of relationships between variables, when it is important to ascertain the causal effect of one variable upon another variable. To explore this relationship regression is employed to estimate the quantitative effect of the causal variables upon the variables that they influence(Freedman, 2005).

Table 24: Model Summary

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate
1	.691[a]	.477	.450	.6725

a. Predictors: (Constant), SI, EE, PE

Table 24 above provides the R and R square values. The R-value represents the simple correlation which is 0.691 and indicates a high degree of correlation. The R square value indicates how much of the total variation in the dependent variable can be explained by the independent variables. In this case 47.7% of the total variation in BI can be explained by EE, PE and SI.

Table 25: ANOVA

Model		Sum of Squares	df	Mean Square	F	Sig.
1	Regression	23.532	3	7.844	17.343	.000[b]
	Residual	25.780	57	.452		
	Total	49.311	60			

a. Dependent Variable: BI

b. Predictors: (Constant), SI, EE, PE

Table 25 reports how well the the regression equation fits the data (i.e. predicts the dependent variable). This table indicates that the regression model predicts the dependent variable significantly well. As seen on the “Sig.” column. Here, $p < 0.0005$ which is less than 0.05 and indicates that overall, the regression model statistically significantly predicts the outcome variable.

Table 26: Coefficients

Model		Unstandardized Coefficients		Standardized Coefficients	t	Sig.	95.0% Confidence Interval for B	
		B	Std. Error	Beta			Lower Bound	Upper Bound
1	(Constant)	.300	.610		.491	.625	-.922	1.521
	PE	.602	.120	.568	5.027	.000	.362	.842
	EE	.090	.128	.071	.703	.485	-.166	.346
	SI	.208	.130	.173	1.601	.115	-.052	.469

a. Dependent Variable: BI

Table 26 contains the standardized beta coefficient value which indicates that one standard deviation increase in PE score brings about 0.568 standard deviation increase in behavioural intention (BI) towards acceptance of the framework. Additionally EE with Beta = 0.071 also has a positive impact on BI towards the acceptance of the framework. Lastly SI also has a positive impact on BI towards the acceptance of the framework, with Beta= 0.173.

In line with Venkatesh *et al.* (2003) suggestion, the results from the quantitative portion of the study confirm that PE has a significant positive effect on behavioural intention (BI) to use the proposed agile-PM framework. Besides that this study found that EE positively affects behavioural intention (BI) to use the framework as well. Lastly, SI was also found to have a positive influence on behaviour intention (BI) to use the proposed framework. Therefore it can be concluded that PE, EE and SI are significant factors to determine the user's acceptance. The convergence of this analysis is discussed in more detail in **Chapter 6**

5.5 Summary of the chapter

This chapter revealed the findings and analysis of the study, which aimed to determine the PM related challenges of implementing Agile at Bank A. It also aimed at determining how these PM related challenges of implementing AM at Bank A can be mitigated through formulation of a framework that integrates PM principles into AM and evaluate acceptance of the proposed framework by SD practitioners. The results of this study have addressed all three research questions as outlined in Chapter 1. The results from the interviews gave a detailed outline of the challenges encountered by SD practitioners and the questionnaire provided an evaluation of the acceptance towards the proposed framework. The qualitative data served the primary purpose of facilitating the formulation of the framework based on the concerns expressed by the SD practitioners at Bank A. Thematic analysis was used for analysing the qualitative data.

The quantitative portion was aimed at determining the level of acceptance of the proposed framework by the SP. SPSS was used for analysing the quantitative data. Factor analysis and Cronbach Alpha was used to establish construct validity and reliability of the data so that the discussion of the results is based on valid and reliable data. Survey data was analysed using descriptive statistics, Pearson's correlation and regression analysis. A discussion of the results in conjunction with the research questions asked in the study is now presented in Chapter 6.

6.0 CONCLUSION AND RECOMMENDATIONS

6.1 Introduction

The agenda for the current research effort has been prompted by the lack of empirical evidence to justify the integration of traditional project management (TPM) principles with agile methodologies (AM). The current study is an attempt to explore the idea of formally integrating project management (PM) with AM. Although Scrum has elements of PM embedded within it, with a Sprint review meeting, a product backlog meeting and daily sprint meetings where PM has the potential to be exercised. There has been no formally endorsed integration of PM into the Scrum methodology. This integration has been achieved in an ad hoc manner leaving too much for interpretation. It is within this context that the current study delves into the domain of PM and AM. The distinguishing feature of the current study is that it leverages off experiential data obtained from project managers (PMs) and software practitioners (SPs) who have encountered challenges in the development of software systems because of the inadequacy of good PM control or the lack of a software development methodology that was dynamic enough to handle changing customer requirements.

This study employed a hybrid methodology combining both quantitative and qualitative research methodologies in achieving its objectives. The main objective of the qualitative component of the study mainly was to provide an insight into the PM challenges facing software practitioners (SP) in successfully adopting AM and to facilitate the formulation of the framework which integrates PM to AM. The quantitative aspect of the study was an effort made to obtain a statistically representative view of the acceptance by SP at Bank A towards the proposed framework.

The remainder of the current chapter is divided into sections addressing the main research problem and associated sub problems.

6.2 Discussion of Research Questions

6.2.1 Research Question 1: What are the PM related challenges of implementing AM at Bank A?

The second research question was aimed at determining what the PM specific challenges are for implementing AM at Bank A. This sub problem was addressed by the interview data and the analysis conducted, particularly Questions 1 to 9. On the basis of the interviews, it was established that the following are the most crucial PM challenges when it comes to the implementation of agile at Bank A:

Project Management Approach

The current approach to PM breeds frustration amongst the project team because of the perceived incompatibility between PM principles which enforce accountability and a high level of structure over the perceived lack of structure to AM. This incompatibility manifests in an incongruous view of the trajectory of a software project, between the project manager(s) and the rest of the team. Project team members see the role of a project manager in a completely different way from how they view their role. There is also a difference in expectations on what PM should look like in the new (agile) ways of working. Project managers have not yet received proper training on AM and resulting in a lack of knowledge of how their role is expected to evolve in an agile context. A possible reason for this untenable situation is that the advent of AM in the bank is not being driven from a strategic and core level resulting in fragmented support for the methodology.

Lack of collaboration between project teams and business stakeholders

The lack of collaboration between the business and IT project teams was identified as an impediment for this organisation effectively adopting ASDM. The respondents believe that this impediment is a result of the lack of understanding by business stakeholders of what it means to be agile as a bank. Collaboration is thought to be essential in ensuring that there is shared accountability for every decision taken in the projects. Respondents also expressed that collaboration is vital in ensuring that the correct software systems are delivered through proper requirements elicitation. Further, the respondents believe it is not possible to develop software systems that assist in solving business problems because of the disconnection between the two areas and lack of understanding of business processes which is further exacerbated by the lack of collaboration.

User Requirements

The responses revealed that the project stakeholders are not satisfied with how requirements are gathered, they expressed concern regarding the lack of standardisation and repeatability of this process. Respondents highlighted the difficulty in measuring how well it's being carried out and continuous improvement. In addition to the lack of consistency in how these requirements are gathered, a bigger concern was that non-functional requirements, IT security and compliance requirements as an example are not addressed. Non-functional requirements such as the aforementioned examples cannot be overlooked by Bank A as it is a financial institution.

There was evidence of dissatisfaction with how requirements change management is handled. This was believed to be particularly important due to the fact that there are finite resources allocated to each project. Project managers expressed that budget and time are fixed for each project and they are required to manage the scope hence proper change management is mandatory to ensure that the important functionality is deployed first.

Governance process

Respondents expressed the need for an adaptation to the governance process. This adaptation is seen as necessary due to the fact that the current governance process was made to work well for the waterfall methodology and its few releases. AM have more frequent releases and there are currently too many approvals that have to be obtained from various governance boards and this impacts on project timelines.

Documentation

The minimal documentation strategy enforced by AM is seen as a challenge given the immaturity and lack of collaboration between the teams. This means that everything has to be documented to ensure that there is accountability for every decision and every implementation made during the project. It was also noted that QA is needed to ensure that correct information has been captured.

6.2.2 Research Question 2: How can the PM related challenges of implementing AM at Bank A be mitigated?

Sub problem 1 highlighted PM challenges that are faced by SP in the new agile way of working which makes it particularly difficult for them to effectively adopt the AM. In response to these challenges the researcher came up with a framework (Appendix G) that could be incorporated into the agile ceremonies that are already being used in the organization. It is important to note that this is not a methodology but a framework, a set of good practices, a guideline for incorporating PM principles into AM. A more detailed breakdown of the framework which can be found on Appendix G presents critical aspects contained in the framework that need to be incorporated into the Scrum process that is currently being used by Bank A as their software development methodology (SDM). These critical aspects are aimed at assisting with curbing the challenges presented above.

Below is a discussion of how the proposed critical aspects of the agile/PM framework will alleviate the challenges highlighted in this study.

Inception

The inception meeting will assist in achieving team alignment. Team members will gain an understanding on why the project is important to the business stakeholders, how it fits into the bigger picture of the organisation, what the highest priority items are and what trade-off the project sponsors are willing to make. It will also help the business stakeholders to be aligned with the team. Where it is not possible to interact extensively with business stakeholders on a daily basis due to geography, schedules and stakeholder availability, this single day spent on the inception will serve as an opportunity for that interaction to take place. The collective thinking and making of decisions encouraged in this phase will potentially alleviate challenges related to the lack of collaboration currently being experienced.

Sprint planning

The sprint planning meeting will address the following challenges that were highlighted by SP

- Establishing a standardized way of eliciting requirements
- Ensuring that the documentation produced has been quality assured and signed off by the relevant stakeholders and this

will ensure that accountability is ensured on all the requirements

- Having the system architect participating in this phase ensures that that governance is introduced during the early stages of the sprint and problems identified are remediated early in the process.
- IT security participation ensures that the governance is addressed and also ensures that the crucial non-functional requirements are included and addressed.

Documentation

The proposed documentation strategy will assist in ensuring that documentation that is fit for purpose is produced. The feedback cycle between doing the work and documenting what you've done is short and this ensures that there is alignment between what is documented and what has been developed.

Governance

Ensuring that there is a dedicated governance team with the correct representation will assist in that they can be able to come up with a governance strategy. This strategy will guide the governance process, ensure that all governance checks are done prior to the releases. This should be done while ensuring that this strategy accommodates agile projects with their multiple releases.

Change Management

This artefact will assist in standardising the change management process in agile projects and in ensuring that this is implemented in a way that is suitable for the projects.

6.2.3 Research Question 3: What is the acceptance by stakeholders of a framework/model that guides the integration of PM into AM?

The last sub problem aimed to determine the level of user acceptance with the proposed framework. This was addressed by Questions 1 to 15 of the questionnaire where information was gathered on the user satisfaction and acceptance of the proposed model. The academic framework was pivotal in ensuring that the appropriate data on the Performance Expectancy (PE), Effort Expectancy (EE), Social Influence (SI), Facilitating Conditions (FC), and Behavioural Intention (BI) was obtained. These questions were formulated based on the variables and constructs in the unified theory of acceptance and use of technology (UTAUT) – constructs used were PE, EE, SI, FC and BI.

The objective of this question is to establish whether software development practitioners perceive the proposed framework as a model that would add value in their day to day operations, that is easy to learn how to use, whether their decision to make use of it would be supported by people whose opinion matters in their lives and whether it would be supported by the current environment. This information will be reliable predictors of whether the staff members have an intention to use the framework.

A Pearson correlation analysis was conducted in the context of the data for the current study in order to confirm the relationships between EE and BI; PE and BI; SI and BI which are theorised by the UTAUT model. This analysis indicated a strong correlation between PE and BI with a correlation coefficient of 0.671, the correlation between EE and BI is 0.274, this correlation is small however it is not zero which makes the association significant, the correlation between SI and BI is 0.443 which is also significant.

A multiple linear regression was also conducted to confirm Pearson's analysis and to ascertain the causal effect of the independent variables (EE, PE, and SI) on the dependent variable (BI). The R-value represents the simple correlation which is 0.691 and indicates a high degree of correlation. The R square value indicates how much of the total variation in the dependent variable can be explained by the independent variables. In this case 47.7% of the total variation in BI can be explained by EE, PE and SI.

The Behavioural Intention (BI) to use as gleaned from the mean values established for PE (3.80) which is at the 78 percentile, EE (3.84) this is at the 77 percentile, and SI (3.65) which is at the 73 percentile. The results indicate a high acceptance of the proposed framework.

On the basis of these results, it can be concluded that there is a high level acceptance for the proposed framework.

6.3 Conclusion of the study

Both traditional and agile project management approaches have similar principles and values for software development. Both are focused on accomplishing the goal, directing the team, and attaining customer satisfaction. However, it was established by the study that the methodology, collaboration, governance, requirements elicitation and change management, and documentation play an important role in PM. The methodology being applied currently is an adaptation of Scrum however it does not address the aforementioned PM needs. To overcome these problems, teams are suggested to conduct an inception meeting prior to kicking off any projects to align the team goals and ensure that there is initial collaboration between business stakeholders and project teams. Requirements elicitation will initially be conducted in the sprint planning and all the relevant stakeholders are given an opportunity to sign-off on the documented requirements ensuring accuracy and accountability.

Documentation should be done during the iteration towards the end when the majority of the development work has been done. This helps the design architects and other stakeholders involved in the documentation process to avoid having to rework the documentation. A dedicated governance team should be established and will be responsible for assisting the project team in planning, managing and governing releases and will have the authority to guide the teams toward the business goals.

Therefore an agile-PM framework has been proposed with reference to the standards enshrined in PMBOK. This aspect of the proposed framework is elaborated in Appendix G (Table 26). It is envisaged that the proposed framework will assist project managers in understanding their roles as well as their obligation to the teams and to business stakeholders. The business stakeholders will understand the significance of collaboration. Project teams will understand why maintaining appropriate documentation is necessary for future reference and why governance is vital when deploying software. The framework was evaluated on user acceptance and the results indicate that there is a high level of acceptance by software development practitioners.

The study has shown that agile and TPM principles can complement each other and are not necessarily bipolar choices as suggested in Vidgen and Wang (2009). Agile without structure can cause chaos, particularly in large, complex projects where planning, control, and coordination are critical.

The main research problem: “How can principles of PM be integrated in ASDM at Bank A?” was addressed successfully by the data collected and through the responses provided by the individual sub research questions.

6.4 Limitations and Recommendation for future study

Even though the relevant information could be gathered and a successful attempt was made at compiling the literature chapter, there were limited literature resources available, that examined these two topics concurrently. In view of the diverse nature of AM and PM techniques, finding software practitioners who have had equal exposure to both these aspects of SD proved to be a challenge.

In much of this study, there has been reference to Agile Methodology (AM) which is a broad reference to a set of agile methods that are unique in the sense that each agile method has a specific methodology attached to it. In the current study, only Scrum, which represent just one of the agile methods, is used as the basis for this study on the integration of PM into AM. This strategy is based on the assumption that all agile methods are conceptually and philosophically similar, hence the generic reference to agile methodology is used. The restriction to just a single agile method is time and resource constrained. However, a study that examines the integration of PM principles with a more diverse set of agile methods would be a good complement to the current study.

From a research design perspective, it should be noted that the purpose of case study research is to obtain deeper insight into the phenomenon of inquiry. As such the generalisability of the study may be open to criticism. A viable strategy to mitigate this criticism may be to conduct a multi-case study so that different organisational and contextual perspectives are obtained.

In terms of the research design, the development of the framework should ideally have been subjected to a Delphi-like inquisition. However, due to time and scope constraints, such an exercise is suggested for further inquiry on the validity of the proposed framework.

6.5 Summary of the Chapter

Whilst this paragraph represents a culmination of the current chapter of the study, the researcher is very much aware that it does not represent a culmination of the discourse on project management and its relevance to software development methodology. The advent of agile methodology, very much perceived to be a "silver bullet" that will contribute towards the development of successful software systems, has been accompanied by the added dilemma of ensuring that the software development is accountable for the resources consumed. Any strategy that purports to integrate 2 diametrically opposing concepts will always be open to criticism and suspicion. Whilst criticism is always a welcome aspect in any discourse, the challenge is to provide a forum so that the criticism is structured and can be viewed constructively. The current study has provided such a forum for the discourse on the integration of PM and AM. In the final chapter of the current study, the main aspects of this discourse have been documented. These are:

- A discussion of ***Research Question One (RQ1)*** that alludes to the challenges of integrating Project Management into Agile Methodology at Bank A. This discussion makes reference to the phenomenological approach used to answer this question and to provide an indication of the challenges that were identified by virtue of the thematic analysis that was conducted on the data.
- A discussion of ***Research Question Two*** that alludes to the mitigation of the challenges of integrating PM into AM. The data obtained to answer RQ1 is used in a strategic manner so that the experiential knowledge of project managers and software practitioners are encapsulated into a framework that leverages off the benefits of Scrum methodology as well as the expert knowledge of project managers to develop a framework that embodies an integration of PM into AM. The development of this framework is also influenced by the researcher's knowledge of the PMBOK standard to ensure that the framework has a universal domain of applicability. The framework proposed as part of the current study is used as an "instrument" to mitigate the challenges associated with the integration of PM into AM.

- A discussion of ***Research Question Three*** that alludes the acceptance by project managers and software practitioners of the framework that is proposed in the current study. This discussion has a dual purpose. It makes reference to the use of the UTAUT theoretical model that has received seminal status in terms of its ability to measure user acceptance and use of a technology. In the context of the current study, the integrative PM-Agile framework is subjected to user acceptance testing informed by the UTAUT model. As has been documented in the actual discussion, the proposed model shows a high rate of user acceptance which, according to the UTAUT correlational sub-system, is a reliable predictor of users' intentions to want to make use of the PM-Agile framework to alleviate the shortcomings of the agile approach if it used in an isolated manner.

REFERENCES

Abdelaziz, A. A., El-Tahir, Y., & Osman, R. (2015). *Adaptive Software Development for developing safety critical software.* Paper presented at the Computing, Control, Networking, Electronics and Embedded Systems Engineering (ICCNEEE), 2015 International Conference on.

Abraham, M. R. (2008). *Importance of a theoretical framework for research.* Paper presented at the ACS Symposium series.

Agarwal, R., & Prasad, J. (1997). The role of innovation characteristics and perceived voluntariness in the acceptance of information technologies. *Decision Sciences, 28*(3), 557-582.

Algharibi, A. J., & Arvanitis, T. N. (2011). Adapting the unified theory of acceptance and use of technology (UTAUT) as a tool for validating user needs on the implementation of e-trial software systems. *Proceedings of the 25th BCS Conference on Human-Computer Interaction*, 526-530.

Alhujran, O. (2009). *Determinants of E-Government Services adoption in developing countries: a field study and a case study.* (PhD), University of Wollongong.

Ambler, S. W., & Lines, M. (2012). *Disciplined Agile Delivery: A Practitioner's Guide to Agile Software Delivery in the Enterprise*: IBM Press.

Augustine, S., Payne, B., Sencindiver, F., & Woodcock, S. (2005). Agile project management: steering from the edges. *Communications of the ACM, 48*(12), 85-89.

Bagozzi, R. P. (2007). The legacy of the technology Acceptance model and a proposal for a paradigm shift. *Journal of the Association for Information Systems, 5*(4), 244-254.

Balada, J. (2013). *Scrum Adoption for Information System Development within Complex Environments.* Paper presented at the 39th International Conference on Current Trends in Theory and Practice of Computer Science, Spindleruv Mlyn.

Bass, J. M. (2013). *Agile method tailoring in distributed enterprises: Product owner teams* Paper presented at the IEEE 8th International Conference on Global Software Engineering.

Bass, J. M. (2015). How product owner teams scale agile methods to large distributed enterprises. *Empirical Software Engineering, 20*(6), 1525-1557.

Batra, D., Xia, W., VanderMeer, D., & Dutta, K. (2010). Balancing agile and structured development approaches to successfully manage large distributed software projects: A case study from the cruise line industry. *Communications of the Association for Information Systems, 27*(1), 21.

Beck, K., M. Beedle, Bennekum, A. V., A. Cockburn, W.Cunningham, M. Fowler, . . . Jeffries, R. (2001). Agile manifesto. Retrieved 3 August 2013, from http://agilemanifesto.org/

Blanchard, B. S., Fabrycky, W. J., & Fabrycky, W. J. (1990). *Systems engineering and analysis* (Vol. 4): Prentice Hall Englewood Cliffs, NJ.

Boehm, B. (2002). Get ready for Agile Methods with Care. *IEEE*.

Boehm, B., & Turner, R. (2005). Management challenges to implementing agile processes in traditional development organizations. *Software, IEEE, 22*(5), 30-39. doi: 10.1109/MS.2005.129

Booth, S. A. (1992). *Learning to program: A phenomenographic perspective*. Universitatis Gothoburgensis

Bowden, J. (2000). *The nature of phenomenographic research in J. Bowden & E. Walsh*. RMIT University Press. Melbourne.

Bowden, J. (2005). *Reflections on the Phenomenographic Team Research Process*. RMIT University Press. Melbourne.

Braun, V., & Clarke, V. (2006). Using thematic analysis in psychology. *Qualitative Research in Psychology, 3*(2), 77-101.

Bredillet, C., Tywoniak, S., & Dwivedula, R. (2015). What is a good project manager? An Aristotelian perspective. *International Journal of Project Management, 33*(2), 254-266.

Brikci, N., & Green, J. (2007). A guide to using qualitative research methodology. *MEDICINS SANS FRONTIERS*.

Brink, P. J., & Wood, M. J. (1998). *Advanced Design in Nursing Research*: SAGE Publications.

Burke, R. (2013). Project Management:planning and control techniques. *International Project Management Journal*, *5*(1).

BusinessTech. (2015). Standard Bank unveils new innovation hub. Retrieved 10 July 2015, 2015, from http://businesstech.co.za/news/banking/78467/standard-bank-unveils-new-innovation-hub/

Cadle, J., & Yeates, D. (2004). *Project management for information systems*: Pearson education.

Carbone, A., Mannila, L., & Fitzgerald, S. (2007). Computer science and IT teachers' conceptions of succssful and unsuccessful teaching: A phenomenographic study. *Computer Science Education*, *17*(4), 275-299.

Ceschi, M., Sillitti, A., Succi, G., & De Panfilis, S. (2005). Project management in plan-based and agile companies. *IEEE Software*, *22*(3), 21-27.

Chan, F. K. Y., & Thong, J. Y. L. (2009). Acceptance of agile methodologies: A critical review and conceptual framework. *Decision Support Systems, 46*(4), 803-814. doi: http://dx.doi.org/10.1016/j.dss.2008.11.009

Charvat, J. (2003). *Project Management Methodologies: Selecting, Implementing, and Supporting Methodologies and processes for Projects*: John Wiley & Sons Inc.

Chau, P. Y. (1996). An empirical assessment of a modified technology acceptance model. *Journal of Management Information Systems, 13*(2), 185-204.

Chertok, B. Y. (2010). *Rockets and People Volume III: Hot Days of the Cold War*. In A. A. Siddiqi (Series Ed.) History Series, Vol. 3.

Cheung, C. M. K., & Lee, M. K. O. (2006). Understanding Consumer Trust in Internet Shopping: A Multidisciplinary Approach. *Journal of the American Society for Information Science and Technology, 57*(4), 479-492.

Chin, W. W., Marcolin, B. L., & Newsted, P. R. (1996). *A partial least squares latent variable modelling approach for measuring interaction effects: Results from a Monte Carlo simulation study and voice mail emotion/adoption study*. Paper presented at the 17th International Conference on Information Systems.

Chow, T., & Cao, D.-B. (2008). A survey study of critical success factors in agile software projects. *Journal of Systems and Software, 81*(6), 961-971. doi: http://dx.doi.org/10.1016/j.jss.2007.08.020

Cicmil, S., Williams, T., Thomas, J., & Hodgson, D. (2006). Rethinking project management: researching the actuality of projects. *International Journal of Project Management, 24*(8), 675-686.

CIO. (2014). Project Level Definitions.   Retrieved 26 June 2015, 2015

Clark, W. (1923). *The Gantt chart: a working tool of management*

Collyer, S., Warren, C., Hemsley, B., & Stevens, C. (2010). Aim, fire, aim—Project planning styles in dynamic environments. *Project Management Journal, 41*(4), 108-121.

Conforto, E. C., & Amaral, D. C. (2016). Agile project management and stage-gate model—A hybrid framework for technology-based companies. *Journal of Engineering and Technology Management, 40*, 1-14.

Cooper, R. G., & Sommer, A. F. (2016). The Agile–Stage-Gate Hybrid Model: A Promising New Approach and a New Research Opportunity. *Journal of Product Innovation Management*.

Coram, M., & Bohner, S. (2005). *The impact of agile methods on sofware project mnagement*. Paper presented at the 12th IEEE International Conferenceand Workshops.

Cornford, T., & Smithson, S. (1996). *Project Research in Information Systems: A Student's Guide*: Palgrave Macmillan Limited.

Creswell, J. W. (1998). Qualitative inquiry and research design: Choosing among five designs: Thousand Oaks, CA: Sage.

Creswell, J. W. (2003). *Qualitative, quantitative, and mixed methods approaches* (2nd ed.). Thousands Oaks, CA.: Sage.

Crotty, M. J. (1998). *The foundations of Social Research: Meaning and Perspective in the Research Process*: Sage Publications.

Cui, Y., & Olsson, N. O. (2009). Project flexibility in practice: An empirical study of reduction lists in large governmental projects. *International Journal of Project Management, 27*(5), 447-455.

Davidson, A. L., Schofield, J., & Stocks, J. (2001). Professional cultures and collaborative efforts: A case study of technologists and educators working change. *The Information Society, 17*(1), 21-32.

Davis, F. D. (1989). Perceived usefulness, perceived ease of use and user acceptance of

information technology. *MIS Quarterly, 13*, 319-339.

Dikert, K., Paasivaara, M., & Lassenius, C. (2016). Challenges and Success Factors for Large-Scale Agile Transformations: A Systematic Literature Review. *Journal of Systems and Software, 119*, 87-108.

Dillon, A., & Morris, M. (1996). User acceptance of new information technology: theories and models *Annual Review of Information Science and Technology, 31*, 3-32.

Dillon, A., & Morris, M. (1998). From "can they" to "will they?": Extending usability evaluation to

address acceptance *Proceedings Association for Information*

Systems Conference. Baltimore, MD.

Dybå, T., & Dingsøyr, T. (2008). Empirical studies of agile software development: A systematic review. *Information and Software Technology, 50*(9–10), 833-859. doi: http://dx.doi.org/10.1016/j.infsof.2008.01.006

Dybå, T., Dingsøyr, T., & Moe, N. B. (2014). Agile Project Management *Software Project Management in a Changing World* (pp. 277-300): Springer.

Ebbesen, J. B., & Hope, A. (2013). Re-imagining the iron triangle: embedding sustainability into project constraints. *PM World Journal, 2*(III).

Elbanna, A. (2014). *Identifying the Risks associated with Agile Software Development: an Empirical Investigation.* Paper presented at the MCIS.

Ferreira, C., & Cohen, J. (2008). *Agile systems development and stakeholder satisfaction: a South African empirical study*. Paper presented at the Proceedings of the 2008 annual research conference of the South African Institute of Computer Scientists and Information Technologists on IT research in developing countries: riding the wave of technology, Wilderness, South Africa.

Field, A. (2005). *Discovering statistics using SPSS* (2nd ed.). London: Sage Publications.

Fitsilis, P. (2008). Comparing PMBOK and Agile Project Management software development processes *Advances in Computer and Information Sciences and Engineering* (pp. 378-383): Springer.

Fitzgerald, B., Hartnett, G., & Conboy, K. (2006). Customising agile methods to software practices at Intel Shannon. *European Journal of Information Systems*, *15*, 200-213.

Flora, H. K., & Chande, S. V. (2014). A Systematic Study on Agile Software Development Methodologies and Practices. *International Journal of Computer Science and Information Technologies*, *5*(3), 3626-3637.

Freedman, D. A. (2005). *Statistical Models: Theory and Practice*: Cambridge University Press.

Gandomani, T. J., & Nafchi, M. Z. (2016). Agile transition and adoption human-related challenges and issues. *Computers in Human Behavior, 62*(C), 257-266.

Gandomani, T. J., Zulzalil, H., Ghani, A. A. A., Sultan, A. B. M., & Nafchi, M. Z. (2013). Obstacles in moving to agile software development methods; at a glance. *Journal of Computer Science, 9*(5), 620.

GatherSpaceTeam. (2015, August 18). Agile software development.

Gill, P., Stewart, K., Treasure, E., & Chadwick, B. (2008). Methods of data collection in qualitative research: interviews and focus groups. *Br Dent J, 204*(6), 291-295.

Given, L. M. (2008). *The Sage Encyclopedia of Qualitative Research Methods* (Vol. 2). Thousand Oaks,CA: Thousand Oaks.

Glaser, B. G., & Strauss, A. L. (1967). The discovery of grounded research: strategies for qualitative research. *New Brunswick (USA) and London (UK): Aldine Transaction*.

González, C. (2010). What do university teachers think eLearning is good for in their teaching? *Studies in Higher Education*, *35*(1), 61-78.

González, C. (2011). Extending research on 'conceptions of teaching': comonalities and differences in recent investigations. *Teaching in Higher Education, 16*(1), 65-80.

Grey, J. (2011). *The development of a hybrid agile project management methodology*. North-West University.

Gupta, S., Dumas, M., McGuffin, M. J., & Kapler, T. (2016). MovementSlicer: Better Gantt charts for visualizing bahaviours and meetings in movement data. Retrieved 14 February 2017, from IEEE

Hair, J. F., Money, A. H., Samouel, P., & Page, M. (2007). *Research methods for business*. The Atrium: John Wiley and sons Ltd.

Hass, K. (2010). Managing Complex Projects that are Too Large, Too Long and Too Costly. Retrieved 17 October 2013, 2013, from http://www.projecttimes.com/articles/managing-complex-projects-that-are-too-large-too-long-and-too-costly.html

Hass, K. B. (2007). The blending of traditional and agile project management. *PM World Today*, *9*(5), 1-8.

Highsmith, J. (2009). *Agile project management: creating innovative products*: Pearson Education.

Highsmith, J. (2013). *Adaptive software development: a collaborative approach to managing complex systems*: Addison-Wesley.

Hinton, P. R., Brownlow, C., McMurray, I., & Cozens, B. (2004). *SPSS explained*. East Sussex, England: Routledge, Inc.

Hittleman, D. R., & Simon, A. J. (1997). *Interpreting Educational Research. An introduction for consumers of research* (2nd ed.). New Jersey: Prentice-Hall,Inc.

Hoda, R., Marshall, S., & Noble, J. (2008). *Agile Project Management.* Paper presented at the New Zealand Computer Science Research Student Conference, Christchurch, New Zealand.

Hu, P. J., Chau, P. Y. K., Sheng, O. R. L., & Tam, K. Y. (1999). Examining the technology acceptance model using physician acceptance of telemedicine technology. *Journal of Management Information Systems, 16*(2), 91-112.

Iamandi, O., Popescu, S., Dragomir, M., & Morariu, C. (2015). A Critical Analysis of Project Management Models and its Potential Risks in Software Development. *Calitatea, 16*(149), 55.

Ibrahim, R., Khalil, K., & Jaafar, A. (2011). Towards Educational Games Acceptance Model (EGAM): A Revised Unified Theory of Acceptance and Use of Technology (UTAUT). *International Journal of Research and Reviews in*

Computer Science, 2(3), 839.

Imran, G., Dayang Norhayati Abang, J., Siva, D., & Ahmed, S. (Eds.). (2016). *Emerging Innovations in Agile Software Development*. Hershey, PA, USA: IGI Global.

Isomäki, H. (2002). *The prevailing conceptions of the human being in information systems development: Systems designers' reflections*: Tampereen yliopisto.

Jaccard, J., C.Wan, & R.Turrisi. (1990). The detection and interpretation of interaction effects between continuous variables Multiple Regression, Multivariate Behavioral Research. *25*(4), 467-478.

Joslin, R., & Müller, R. (2015). Relationships between a project management methodology and project success in different project governance contexts. *International Journal of Project Management, 33*(6), 1377-1392.

Juinn, P., & Tan, B. (2013). Applying the UTAUT to Understand Factors Affecting the Use of English E-Learning Websites in Taiwan *Sage Publications*.

Kaapu, T., Saarenpaa, T., Tiainen, T., & Paakki, M.-K. (2014). The Truth is Out There - Phenomenography in Information Systems Research.

Karlesky, M., & Voord, M. (2008). Agile Project Management (or, Burning Your Gantt Charts). Boston, Massachusetts: Atomic Object.

Karlstro¨m, D., & Runeson, P. (2006). <integrating agile into stage gate.pdf>. *Empirical Software Engineering, 11*, 203-225. doi: 10.1007/s10664-006-6402-8

Keil, M., Arun, R., Cheney Mann, J. E., & Zhang, G. P. (2003). Why software projects escalate: the importance of project management constructs. *Engineering Management, IEEE Transactions on, 50*(3), 251-261. doi: 10.1109/TEM.2003.817312

Kennaley, M. (2010). *SDLC 3.0: Beyond a Tacit Understanding of Agile*: Fourth Medium Press.

Keramati, H., & Mirian-Hosseinabadi, S.-H. (2008). *Integrating software development security activities with agile methodologies.* Paper presented at the Computer Systems and Applications, 2008. AICCSA 2008. IEEE/ACS International Conference on.

Kerzner, H. (2013). *Project management: a systems approach to planning, scheduling, and controlling*: John Wiley & Sons.

Kerzner, H. R. (2013). *Project Management: A systems Approach to Planning, Scheduling, and Controlling* (Eleventh ed.). New York: Wiley.

Kevin, J. B. (1999). *Methods of Business Research* (2nd ed.). Reading, MA: Addison-Wesley.

Khalid, A., Zahra, S., & Khan, M. F. (2014). Suitability and Contribution of Agile Methods in Mobile Software Development. *I.J. Modern Education and Computer Science, 2*, 56-62.

Khan, S. H. (2014). Phenomenograpgy: A Qualitative Research Methodology in Bangladesh. *International Journal on New Trends In Education and Their Implications, 5*(2).

Klastorin, T. (2003). *Project Management: Tools and Trade-Offs* (pp. 242).

Kruchten, P. (2013). Contextualising agile software development. *Journal of Software: Evolution and Process, 25*(4), 351-361.

Kwak, Y., & Anbari, F. (2008). *Impact on Project Management of Allied Disciplines: Trends and Future of Project Management Practices and Research* (pp. 170).

Labaree, R. V. (2013, 29 October 2013). Organizing Your Social Sciences Research Paper Retrieved 30 October 2013, 2013, from http://libguides.usc.edu/content.php?pid=83009&sid=615849

Lambert, T. (2012). *Cross sectional study of agile software development methods and project performance*: ProQuest / UMI.

Lan, C., Mohan, K., Peng, X., & Ramesh, B. (2004, 5-8 Jan. 2004). *How extreme does extreme programming have to be? Adapting XP practices to large-scale projects.* Paper presented at the System Sciences, 2004. Proceedings of the 37th Annual Hawaii International Conference on.

Landry, J. P., & McDaniel, R. (2015). *Agile Preparation within a traditional project management course*. Paper presented at the Electronic Journal of Information Systems, Wimington, North Carolina USA.

Larman, C., & Vodde, B. (2008). *Scaling lean and agile development: Thinking and organizational tools for large-scale scrum: Successful large, multisite and offshore products with large-scale scrum*. Upper Saddle River, NJ, USA: Addison Wesley.

Larson, E. W., & Gray, C. F. (2011). Project management: The managerial process.

Lee, S., & Yong, H.-S. (2013). Agile software development framework in a small project environment. *Journal of Information Processing Systems, 9*(1), 69-88.

Leffingwell, D. (2007). *Scaling software agility: Best practices for large enterprises*. Boston, MA, USA: Addison Wesley.

Leffingwell, D. (2015, 01 January 2015). SAFe 4.0 for Lean Software and Systems Engineering. Retrieved 30 July, 2016, from http://www.scaledagileframework.com/

Leffingwell, D. (2017). SAFe reference guide.

Leffingwell, D., & Smit, H. (2005). A CIO's Playbook for Adopting the Scrum Method of Achieving Software Agility: Rally Software Development Corp.

Legris, P., Ingham, J., & Collerette, P. (2003). Why do people use information technology? A critical review of the technology acceptance model. *Information & Management, 40*(3), 191.

Limberg, L. (2000). Phenomenography: a rational approach to research on information needs, seeking and use. *The New Review of Information Bahaviour Research, 2000*(1), 51-67.

Liu, J. (2012). Unified theory of acceptance and use of technology *Breaking the mold: An educational perspective n diffusion of innovation*.

Liu, J. (2013). *E-Learning in English Classroom: Investigating factors impacting on ESL (English as Second Language) colege students' acceptance and use of the Modular Object-Oriented Dynamic Learning Environment*. Iowa State University. Digital Repository @ Iowa State University.

Malhotra, Y., & D.Galletta. (1999). *Extending the technology acceptance model to account for social influence: Theoretical bases and empirical validation.* Paper presented at the Hawaii international conference on system sciences.

Manifesto, C. (2012). The Standish Group.

Marchewka, J. T. (2014). *Information Technology Project Management* (5th ed.): Wiley.

Marques, L. (2012). Complexity and Adaptive Project Management. Retrieved from Revista BSP website:

Martins, C., Tiago, O., & Ale, P. (2013). Understanding the Internet Banking Adoptio: A Unified theory acceptance and use of technology and perceived Risk application. *International Journal of Information Management*.

Marton, F. (1981). Phenomenography—describing conceptions of the world around us. *Instructional science, 10*(2), 177-200.

Marton, F., & Pong, W. Y. (2005). On the unit of description in phenomenography. *Higher Education Research and Development, 24*(4), 335-348.

Maylor, H. (2001). Beyond the Gantt chart:: Project management moving on. *European Management Journal, 19*(1), 92-100. doi: http://dx.doi.org/10.1016/S0263-2373(00)00074-8

McClelland, G. H., & Judd, C. M. (1993). Statistical difficulties of detecting interactions and moderator effects. *Psychological Bulletin, 114*(2), 376-390.

Mertens, D. M. (2014). *Research and Evaluation in Education and Psychology: Integrating Diversity with Quantitative, Qualitative and Mixed Methods* (Fourth ed.): Sage Publications.

Miles, M. B., & Huberman, A. M. (1994). *Qualitative Data Analysis: An Expanded Sourcebook*: SAGE Publications.

Moczar, L. (2013). Why Agile Isn't Working Bringing Common Sense to Agile Principles. Retrieved 1 Septemebr 2013, from http://www.cio.com/article/734338/Why_Agile_Isn_t_Working_Bringing_Common_Sense_to_Agile_Principles

Mohammed, N., Munassar, A., & A.Govardhan. (2010). A comparison between five models of software engineering. *International Journal of Computer Science Issues, 7*(5), 94-101.

Moore, S. (2010). Strategic Project Management. Enabling a Productive Organization: Wiley & Sons, New Jersey.

Morris, P. (1994). *The Management of projects*

Morse, J. M. (1994). Designing funded qualitative research.

Mossholder, K. W., Kemery, E. R., & Bedeian, A. G. (1990). On using regression coeffecients to interpret moderator effects. *Educational and Psychological Masurement, 50*(2), 255-263.

Munassar, N. M. A., & Govardhan, A. (2010). A comparison between five models of software engineering. *IJCSI, 5*, 95-101.

Murphy, B., Bird, C., Zimmermann, T., Williams, L., Nagappan, N., & Begel, A. (2013, 10-11 Oct. 2013). *Have Agile Techniques been the Silver Bullet for Software Development at Microsoft?* Paper presented at the 2013 ACM / IEEE International Symposium on Empirical Software Engineering and Measurement.

Myers, M. D. (2009). *Qualitative Research in Business & Management* University of Auckland, New Zealand SAGE Publications Ltd

Nerur, S., Mahapatra, R., & Mangalaraj, G. (2005). Challenges of migrating to agile methodologies. *Commun. ACM, 48*(5), 72-78. doi: 10.1145/1060710.1060712

Nord, R. L., & Tomayko, J. E. (2006). Software architecture-centric methods and agile development. *IEEE Software, 23*(2), 47-53.

Ntuthuko. (2013). Top five reasons for rating a bank as innovative. Retrieved 26 June 2015, 2015, from http://www.innovationagency.com/top-five-reasons-for-rating-a-bank-as-innovative/

O'Sheedy, D. G. (2012) <A study of agile project management methods used for IT implement.pdf>. Southern Cross University, Lismore, NSW: Southern Cross University, Lismore, NSW.

Oberlender, G. D. (1993). *Project management for engineering and construction* (Vol. 2): McGraw-Hill New York.

Owen, F., & Jones, R. (1994). *Statistics* (4th Ed.). London: Pitman.

Pallant, J. (2005). *SPSS survival manual: A step by step guide to data analysis using SPSS* (2nd ed.). Crows Nest, NSW: Allen & Unwin.

Papatheocharous, E., & Andreou, A. S. (2013). Evidence of Agile Adoption in Software Organizations: An Empirical Survey. In F. McCaffery, R. V. O'Connor & R. Messnarz (Eds.), *Systems, Software and Services Process Improvement: 20th European Conference, EuroSPI 2013, Dundalk, Ireland, June 25-27, 2013. Proceedings* (pp. 237-246). Berlin, Heidelberg: Springer Berlin Heidelberg.

Papke-Shields, K. E., Beise, C., & Quan, J. (2010). Do project managers practice what they preach, and does it matter to project success? *International Journal of Project Management, 28*(7), 650-662.

Park, S. Y., Nam, M. W., & Cha, S. B. (2012). University students' behavioral intention to use mobile learning: Evaluating the technology acceptance model. *British Journal of Educational Technology, 43*(4), 592-605.

PENNSTATE. (2008). Supplemental Topic A: Project Management.

Phillips, J. (2013). *PMP, Project Management Professional (Certification Study Guides)*: McGraw-Hill Osborne Media.

PMI. (2000). *A guide to the project management body of knowledge* (5th ed.). Newtown Square, Pennsylvania: Project Management Institute.

PMI. (2004). *A Guide to the Project Management Body of Knowledge:PMBOK® Guide*

PMI. (2008). *A guide to the project management body of knowledge (PMBOK® guide)* (pp. 459).

PMI. (2015). Capturing the value of Project Management.

Polit, D. F., & Hungler, B. P. (1995). *Nursing Research: Principles and Methods* (5th ed.): Lippincott Williams & Wilkins.

Poppendieck, M. (2002). LeanEssays: XP in a Safety-Critical Environment. Retrieved 2 September 2013, from http://www.leanessays.com/2002/08/xp-in-safety-critical-environment.html

Puri, C. (2009). *Agile management: feature driven development*: Global India Publications.

Qureshi, M. R. J. (2012). Agile software development methodology for medium and large projects. *IET Software, 6*(4), 358-363. http://digital-library.theiet.org/content/journals/10.1049/iet-sen.2011.0110

Raithatha, D. (2007). *Making the whole product agile - A product owners perspective.* Paper presented at the 8th International Conferene on Agile Processes in Software Engineering and Extreme Programming, Como.

Rasmusson, J. (2010). *The Agile Samurai: How Agile Masters Deliver Great Software*: Pragmatic Bookshelf.

Rico, D. F. (2010). Lean and agile project management: for large programs and projects *Lean Enterprise Software and Systems* (pp. 37-43): Springer.

Riemenschneider, C. K., & Hardgrave, B. C. (2001). EXPLAINING SOFTWARE DEVELOPMENT TOOL USE WITH THE TECHNOLOGY ACCEPTANCE MODEL. *Journal of Computer Information Systems, 41*(4), 1.

Riessman, C. K. (1993). *Narrative Analysis*. London: Sage.

Rogers, E. M. (1995). *Diffusion of innovations* (Vol. 4th Edition). New York: The Free Press.

Rose, K. H. (2013). A Guide to the Project Management Body of Knowledge (PMBOK® Guide)—Fifth Edition. *Project Management Journal, 44*(3), e1-e1.

Rosser, L., Marbach, P., Lempia, D., & Osvalds, G. (2013). Systems Engineering for Software Intensive Projects Using Agile Methods. *co-authors: Marbach, Phyllis*.

Rubin, K. S. (2012). *Essential Scrum: a practical guide to the most popular agile process*: Addison-Wesley.

Saibaba, S., & Murthy, T. N. (2013). Factors Influencing the Behavioural Intention to Adopt Internet Banking. *Journal of Arts, Science & Commerce, 4*(4), 77-91.

Saladis, F. P., & Kerzner, H. (2011). *Bringing the PMBOK guide to life: a companion for the practicing project manager*: John Wiley & Sons.

Salo, O., & Abrahamsson, P. (2008). Agile methods in European embedded software development organisations : A survey on the actual use and usefulness of Extreme Programming and Scrum. *Software*, 58 - 64.

Sánchez, R. A., & Hueros, A. D. (2010). Motivational factors that influence the acceptance of Moodle using TAM. *Computers in Human Behavior, 26*(6), 1632-1640. doi: http://dx.doi.org/10.1016/j.chb.2010.06.011

Saunders, M., Lewis, P., & Thornhill, A. (2000). *Research methods for Business Students* (2nd ed.). Harlow, Essex: Pearson Education.

Schwalbe, K. (2015). *Information technology project management*: Cengage Learning.

Scott, H., & Ives, B. (1992). *MIS Research Strategies* (R. D. Galliers Ed.). London, England: Balckwell Scintific Publications.

Sekaran, U., & Bougie, R. (2009). *Research methods for business: a skill building approach* (5th ed.). Padstow, Cornwall: TJ international Ltd.

Sekaran, U., & Bougie, R. (2010). *Research Methods for Business : A skill building approach*: John Wiley&Sons.

Serenko, A., Turel, O., & Yol, S. (2006). Moderating roles of user demographics in the American customer satisfaction model within the context of mobile services. *Journal of Information Technology Management, 17*(4), 20-32.

Serrador, P., & Pinto, J. K. (2015). Does Agile Work? - A quantitative analysis of agile project success. *International Journal of Project Management, 33*(5), 1040-1051.

Seymour, T., & Hussein, S. (2014). The history of project management. *International Journal Management & Information Systems, 18*(4), 233 -250.

Sharma, S., Sarkar, D., & Gupta, D. (2012). Agile processes and methodologies: A conceptual study. *International journal on computer science and Engineering, 4*(5), 892.

Shenhar, A., & Dvir, D. (2007). Reinventing Project Management: The Diamond Approach to Successful Growth and Innovation, by Aaron Shenhar and Dov Dvir. *Inf. Knowl. Syst. Manag., 6*(4), 363-363.

Shroff. (2008). Introduction to Interview Analysis and Presentation *Politis*.

Soy, S. K. (1997, 02/12/2006). The case study as a research method. Retrieved 30 October 2013, from https://www.ischool.utexas.edu/~ssoy/usesusers/l391d1b.htm

Sprinthall, R., Schmutte, G., & Sirois, L. (1991). *Understanding Educational Research*. New Jersey: Prentice Hall.

Stretton, A. (2007). A Short History of Modern Project Management. *PM World Today, IX*(X), 1-18.

Svejvig, P., & Andersen, P. (2015). Rethinking project management: A structured literature review with a critical look at the brave new world. *International Journal of Project Management, 33*(2), 278-290.

Sylvester, T. (2013, June 11 2013). Waterfall, Agile & the “Triple Constraint”. Retrieved from http://tom-sylvester.com/lean-agile/waterfall-agile-the-triple-constraint/

Thamhain, H. (2013). Managing risks in complex projects. *Project Management Journal, 44*, 20 -35. doi: 10.1002/pmj.21325

Thamhain, H. J. (2014). Can We Manage Agile in Traditional Project Environments. *Software, IEEE*, 2497-2504.

Thompson, R., Compeau, D., & Higgins, C. (2006). Intentions to use information technologies: An integrative model. *Journal of Organizational and End User Computing, 18*(3), 25-43.

Thomsett, R. (2002). *Radical project management*: Prentice Hall Professional.

Tiwari, P. (2015). Diffusion of Innovations Theory: Characteristics, Types and Barriers to Diffusion. Retrieved 21 December, 2015

Tomanek, M., Cermak, R., & Smutny, Z. (2015). A Conceptual Framework for Web Development Projects Based on Project Management and Agile Development Principles. *arXiv preprint arXiv:1502.04297*.

Tornatzky, L. G., & Klein, K. J. (1982). Innovation Characteristics and Innovation Adoption Implementation: A Meta-Analysis of Findings. *IEEE Transactions on Engineering Management*, *29*(1).

Turk, D., France, R., & Rumpe, B. (2014). Limitations of agile software processes. *arXiv preprint arXiv:1409.6600*.

Vartiainen, T. (2005). *Moral conflicts in a project course in information systems education*: University of Jyväskylä.

Venkatesh, V., & Davis, F. D. (2000). A Theoretical Extension of the Technology Acceptance Model: Four Longitudinal Field Studies. *Management Science, 46*(2), 186-204. doi: doi:10.1287/mnsc.46.2.186.11926

Venkatesh, V., Morris, M. G., Gordon, B. D., & Davis, F. D. (2003). User Acceptance of Information Technology: Toward a Unified View. *MIS Quarterly, 27*(3), 425-478. doi: 10.2307/30036540

Venkatesh, V., Thong, J. Y. L., & Xu, X. (2012). Consumer Acceptance and Use of Information Technology: Extending the Unified Theory of Acceptance and Use of Technology. *MIS Quarterly, 36*(1), 157-178.

Vijayasarathy, L., & Turk, D. (2008). Agile software development: A survey of early adopters. *Journal of Information Technology Management, 2*(19).

Virine, L., & Trumper, M. (2009). *Project decisions: The art and science*: Management Concepts Inc.

Waardenburg, G. v., & Vliet, H. v. (2013). When agile meets the enterprise. *Information and Software Technology, 55*(12), 2154 - 2171.

Weiss, N. A., & Weiss, C. A. (2012). *Introductory statistics*: Pearson Education London.

Welman, C., Kruger, F., & Mitchell, B. (2006). *Research Methodology*: OUP Southern Africa.

Williams, L. (2010). Agile software development methodologies and practices. *Advances in Computers, 80,* 1-44.

Wu, Y. L., Tao, Y. H., & Yang, P. C. (2008). The use of theory of acceptance and use of technology to confer the behavioural model of 3G mobile telecommunication users. *Journal of Statistics & Management Systems, 11*(5), 919-949.

Wysocki, R. K. (2011). *Effective project management: traditional, agile, extreme*: John Wiley & Sons.

Yates, C., Partridge, H., & Bruce, C. (2012). Exploring information experiences through phenomenography. *Library and Information Research, 36*(112), 96-119.

Yatzeck, E. (2012). How to control scope creep in Agile.

Yin, R. K. (1984). *Case study research: Design and methods*. Newbury Park, CA: Sage.

Yin, R. K. (2003). *Case Study Research: Design and Methods*: SAGE Publications.

Yu, C.-S. (2012). Factors Affecting Individuals to Adopt Mobile Banking: Empirical Evidence from the UTAUT Model. *Journal of Electronic Commerce Research, 13*(2), 104-121.

APPENDIX A: INTERVIEW QUESTIONS

UKZN HUMANITIES AND SOCIAL SCIENCES RESEARCH ETHICS COMMITTEE (HSSREC)

APPLICATION FOR ETHICS APPROVAL
For research with human participants

Information Sheet and Consent to Participate in Research

Date:

Greetings,

My name is Yonga Mapongwana (Student No. 209501068) and I am currently studying for a Master of Commerce (MCom) degree at the University of KwaZulu-Natal (UKZN), in the School of Management, Information Technology and Governance. The discipline of my study is in Information Technology (IT). The contact details for myself as well as my supervisor and the academic department at UKZN are listed below:

Researcher Name: Yonga Mapongwana; e-mail: Yonga.Mapongwana@Standardbank.co.za ; Office Contact Number: +27 11 721 5017
Mobile Contact Number: +27 82 500 5715
Supervisor Name: Mr S Ranjeeth; e-mail: ranjeeths@ukzn.ac.za ;
Office contact Number: +27 33 260 5641
Department of Information Systems & Technology: +27 33 260 5704; + 27 31 260 7051

You are being invited to consider participating in a study that involves research on the integration of traditional software development and project management techniques. The title of my study is:

An Integration of Traditional Project Management Principles into Agile Software Development Methodology: A Case study of Standard Bank South Africa

The aim and purpose of this study is to determine user acceptance of a software development model that integrates Agile Software Development Methodology with Project Management strategy. The first part of the study is directed at obtaining an insight into the software development and project management strategy used for the development of software systems at Standard Bank. This aspect of the study will entail the conducting of interviews with key stakeholders who are in a position to provide information regarding software development and management practice at Standard Bank. The interviews will be used as a platform to obtain an understanding of the challenges that may compromise the development of quality software systems. This study has the main objective of proposing a model of software development that will serve the purpose of mitigating the challenges of software development and management at Standard Bank. The duration of your participation if you choose to participate and remain in the study is expected to be approximately 40 minutes.

We hope that the study will be beneficial to Standard Bank by virtue of the envisaged contribution it will make to the process of software development and project management. It is also envisaged that the outcome of the study will make an academic and practitioner-based contribution to the general discourse on Agile Software Development Methodology.

This study has been ethically reviewed and approved by the UKZN Humanities and Social Sciences Research Ethics Committee (approval number_____).

In the event of any problems or concerns/questions you may contact the researcher by making use of any of the contact details provided above, or by contacting the UKZN Humanities & Social Sciences Research Ethics Committee. The contact details are as follows:

HUMANITIES & SOCIAL SCIENCES RESEARCH ETHICS ADMINISTRATION
Research Office, Westville Campus
Govan Mbeki Building
Private Bag X 54001
Durban 4000 KwaZulu-Natal, SOUTH AFRICA
Tel: 27 31 2604557- Fax: 27 31 2604609
Email: HSSREC@ukzn.ac.za

Your participation in the study is voluntary and by participating, you are granting the researcher permission to use your responses. You may refuse to participate or withdraw from the study at any time with no negative consequence. There will be no monetary gain from participating in the study. Your anonymity will be maintained by the researcher and the School of Management, I.T. & Governance and your responses will not be used for any purposes outside of this study.

All data, both electronic and hard copy, will be securely stored during the study and archived for 5 years. After this time, all data will be destroyed.

If you have any questions or concerns about participating in the study, please contact me or my research supervisor at the numbers listed above.

Sincerely

.Y. Mapon

....

Yonga Mapongwana

CONSENT TO PARTICIPATE

I ………………………………………………………………………………………. (Name) have been informed about the study entitled *An Integration of Traditional Project Management Principles into Agile Software Development Methodology: A Case study of Standard Bank South Africa* by *Yonga Mapongwana*.

I understand the purpose and procedures of the study.

I have been given an opportunity to ask questions about the study and have had answers to my satisfaction.

I declare that my participation in this study is entirely voluntary and that I may withdraw at any time without affecting any of the benefits that I usually am entitled to.

I have been informed about any available compensation or medical treatment if injury occurs to me as a result of study-related procedures.

If I have any further questions/concerns or queries related to the study I understand that I may contact the researcher at the details provided in **Page 1** of this document.

If I have any questions or concerns about my rights as a study participant, or if I am concerned about an aspect of the study or the researchers then I may contact:

HUMANITIES & SOCIAL SCIENCES RESEARCH ETHICS ADMINISTRATION
Research Office, Westville Campus
Govan Mbeki Building
Private Bag X 54001
Durban
4000
KwaZulu-Natal, SOUTH AFRICA
Tel: 27 31 2604557 - Fax: 27 31 2604609
Email: HSSREC@ukzn.ac.za

I hereby provide consent to:

Audio-record my interview / focus group discussion	YES / NO
Video-record my interview / focus group discussion	YES / NO
Use of my photographs for research purposes	YES / NO

______________________________ ______________________________

Signature of Participant **Date**

______________________________ ______________________________

Signature of Witness **Date**
(Where applicable)

______________________________ ______________________________

Signature of Translator **Date**
(Where applicable)

General Instructions for the Interview

During the interview, you are at liberty to request clarification or repetition of the question. There is no time limit set for answering a particular question or for the duration of the interview session. It is advisable to complete the interview in a single sitting.

Section A – Demographics & Background Information

- Please, state your name, title, age and position within the company.
- How many years have you been in this role for in the company?
- Please provide some details regarding your experience in the domain of software development and/or project management.
- Please provide some details regarding your experience with Agile Software Development Methodology (ASDM).
- Please provide some details regarding some of the software development projects that you have been involved with

Section B – Current Practice

1. What type of projects does the bank apply ASDM for?
2. In your opinion, how would you classify a project as a large software development project and a small software development project?
3. How is project management practice implemented for small and large projects?
4. If projects implement ASDM, how is project management practice enforced in such cases?
5. Do you think that ASDM is nicely aligned towards proper project management practice?
6. Do you think that the current software development practice at Standard Bank enhances the prospect of developing software that is:
    a. Functional/ high utility value

b. Usable/ abides by usability principles that will ensure end user satisfaction in the use of the systems

c. Reliable – minimal reports of failure

d. Performs efficiently with regards to resource consumption and response times

e. Easy to maintain and compatible with the various platforms on which it will be used.

7. How would you rate the software systems developed at Standard Bank in terms of:

a. The costs incurred to develop the system

b. The costs incurred to maintain the system

c. The time taken to develop the systems

d. End user acceptance of the systems

e. The business value generated by the systems

8. From a project management perspective, do you feel that ASDM will alleviate challenges associated with:

a. Documentation

b. Managing requirements changes

c. Managing a prescribed budget

d. Managing a prescribed schedule

e. Managing the cost of maintenance/ rework of software systems

f. Staff turnover

g. Fast pace of technology

9. Generally, what are the current challenges associated with software development at Standard Bank, from the perspective of:

a. Software development methodology

b. Project Management

Section C – Future Practice

This section seeks to establish whether the respondents were in favour of considering an adjustment/adaptation of currently implemented software development and project management practice at Standard Bank. In order to add a bit of structure to the discussion, the various phases of the Systems Development Life Cycle (SDLC) are used as points of reference.

1. What changes (if any) would you suggest to the ***software development practice*** at Standard Bank from the following perspectives?
   a. Requirements elicitation
   b. Analysis
   c. Design
   d. Implementation
   e. Maintenance
   f. Documentation
2. What changes (if any) would you suggest to the ***project management practice*** at Standard Bank from the following perspectives:
   a. Budget
   b. Schedule
   c. Control
   d. Documentation

3. With regards to development of a model for software development that embodies an integration of ASDM and Project Management practice, what is your opinion on the following:
   a. The general idea of attempting such an integration to produce an integrated model of software development that incorporates proper software development methodology as well as project management principles.
   b. Who are the main stakeholders that should be approached in order to ensure that such a venture is successful?

c. What are some suggestions that you feel will be pivotal in ensuring the success of such a model from a:
    i. Technical perspective
    ii. Social/organisational perspective
d. Do you think that implementation of such a model will be sustainable at Standard Bank? What can be done to ensure the sustainability of such a model?

Thank You for Your Participation

APPENDIX B: QUESTIONNAIRE

INFORMATION SHEET AND CONSENT TO PARTICIPATE IN RESEARCH

Date:

Greetings,

My name is Yonga Mapongwana (Student No. 209501068) and I am currently studying for a Master of Commerce (MCom) degree at the University of KwaZulu-Natal (UKZN), in the School of Management, Information Technology and Governance. The discipline of my study is in Information Technology (IT). The contact details for myself as well as my supervisor and the academic department at UKZN are listed below:

Researcher Name: Yonga Mapongwana; e-mail: Yonga.Mapongwana@Standardbank.co.za ; Office Contact Number: +27 11 721 5017
Mobile Contact Number: +27 82 500 5715
Supervisor Name: Mr S Ranjeeth; e-mail: ranjeeths@ukzn.ac.za ;
Office contact Number: +27 33 260 5641
Department of Information Systems & Technology: +27 33 260 5704; + 27 31 260 7051

You are being invited to consider participating in a study that involves research on the integration of traditional software development and project management techniques. The title of my study is:

An Integration of Traditional Project Management Principles into Agile Software Development Methodology: A Case study of Standard Bank South Africa

The aim and purpose of this study is to determine user acceptance of a framework that integrates Agile Software Development Methodology with Project Management strategy. The study is expected to include approximately 50 Standard Bank employees who are actively involved in software development and project management activities at Standard Bank, South Africa. The study will require participants to provide survey-based responses to questions regarding the viability of using the proposed integrated software development/project management framework to underpin future software development activities at Standard bank. The duration of your participation if you choose to participate and remain in the study is expected to be approximately 40 minutes.

The study will require your exclusive attention to the details of the proposed model so that you will be able to provide an informed response to the survey based questions. We hope that the study will be beneficial to the employees at Standard Bank by virtue of the envisaged contribution it will make to the process of software development and project management at Standard Bank. It is also envisaged that the outcome of the study will make an academic and practitioner-based contribution to the general discourse on Agile Software Development Methodology.

This study has been ethically reviewed and approved by the UKZN Humanities and Social Sciences Research Ethics Committee (approval number_____).

In the event of any problems or concerns/questions you may contact the researcher by making use of any of the contact details provided above, or by contacting the UKZN Humanities & Social Sciences Research Ethics Committee. The contact details are as follows:

HUMANITIES & SOCIAL SCIENCES RESEARCH ETHICS ADMINISTRATION
Research Office, Westville Campus
Govan Mbeki Building
Private Bag X 54001
Durban 4000 KwaZulu-Natal, SOUTH AFRICA
Tel: 27 31 2604557- Fax: 27 31 2604609
Email: HSSREC@ukzn.ac.za

Your participation in the study is voluntary and by participating, you are granting the researcher permission to use your responses. You may refuse to participate or withdraw from the study at any time with no negative consequence. There will be no monetary gain from participating in the study. Your anonymity will be maintained by the researcher and the School of Management, I.T. & Governance and your responses will not be used for any purposes outside of this study.

All data, both electronic and hard copy, will be securely stored during the study and archived for 5 years. After this time, all data will be destroyed.

If you have any questions or concerns about participating in the study, please contact me or my research supervisor at the numbers listed above.

Sincerely

Y. Mapon
....

Yonga Mapongwana

CONSENT TO PARTICIPATE

I …………………………………………………………………………………………. (Name) have been informed about the study entitled

An Integration of Traditional Project Management Principles into Agile Software Development Methodologies by *Yonga Mapongwana*.

I understand the purpose and procedures of the study.

I have been given an opportunity to ask questions about the study and have had answers to my satisfaction.

I declare that my participation in this study is entirely voluntary and that I may withdraw at any time without affecting any of the benefits that I usually am entitled to.

I have been informed about any available compensation or medical treatment if injury occurs to me as a result of study-related procedures.

If I have any further questions/concerns or queries related to the study I understand that I may contact the researcher at the details provided in **Page 1** of this document.

If I have any questions or concerns about my rights as a study participant, or if I am concerned about an aspect of the study or the researchers then I may contact:

HUMANITIES & SOCIAL SCIENCES RESEARCH ETHICS ADMINISTRATION

HUMANITIES & SOCIAL SCIENCES RESEARCH ETHICS ADMINISTRATION
Research Office, Westville Campus
Govan Mbeki Building
Private Bag X 54001
Durban 4000 KwaZulu-Natal, SOUTH AFRICA
Tel: 27 31 2604557- Fax: 27 31 2604609
Email: HSSREC@ukzn.ac.za

_________________________ _________________________
Signature of Participant **Date**

_________________________ _________________________
Signature of Witness **Date**
(Where applicable)

_________________________ _________________________
Signature of Translator **Date**
(Where applicable)

General Instructions

You are expected to study the proposed model that illustrates the integration of Project Management principles into Agile Software Development Methodology. The questions below have been designed to firstly establish a context with regards to your capacity as an employee at Standard Bank and to ascertain your acceptance of the proposed software development/project management model.

Please read and complete the following questionnaire. In those sections where options are provided, please indicate your response by making a cross (X) in the boxes provided.

PART 1: Demographic & Background Information

Job Title/Position		
Department		
Gender	*MALE*	*FEMALE*
Age		
Years of Experience as a Software Developer		
Years of experience in using Agile Software Development Methodology		
Years of Experience in Software Project Management		

PART 2 (Performance Expectancy):

In this section, please provide your response with respect to the following statements concerning the ***possible job benefits that you may experience if you had to make use of the proposed model.***

1. **The proposed model would be useful for me to use in software development projects.**

Strongly agree	Agree	Neutral	Disagree	Strongly disagree

2. **Using the proposed model would enable me to ensure that those aspects of software development that are important to me are upheld.**

Strongly agree	Agree	Neutral	Disagree	Strongly disagree

3. Using the proposed model would help me to do my software development related job activities more quickly.

Strongly agree	Agree	Neutral	Disagree	Strongly disagree

4. Using the proposed model would increase my productivity with regards to software development activities.

Strongly agree	Agree	Neutral	Disagree	Strongly disagree

PART 3 (Effort Expectancy):

In this section, please provide your response with respect to the following statements concerning ***the effort that it will take to implement the proposed model*** *in future software development projects.*

5. I predict that learning how to apply this model to my software projects would be easy to do.

Strongly agree	Agree	Neutral	Disagree	Strongly disagree

6. I predict it will be easy for me to become skillful at using the proposed model.

Strongly agree	Agree	Neutral	Disagree	Strongly disagree

PART 4 (Social Influence):

In this section, please provide your response with respect to the following statements concerning the extent to which you perceive that ***significant people believe that you should make use of the proposed model*** *for job related activities.*

7. **People who are important to me in my job domain will endorse my preference to make use of the proposed model.**

Strongly agree	Agree	Neutral	Disagree	Strongly disagree

8. **People who have an influence my behaviour will endorse my preference to make use of the proposed model.**

Strongly agree	Agree	Neutral	Disagree	Strongly disagree

9. People whose opinions that I value will endorse my preference to make use of the proposed model.

Strongly agree	**Agree**	**Neutral**	**Disagree**	**Strongly disagree**

PART 5 (Facilitating Conditions):

In this section, please provide your response with respect to the following statements concerning the ***role that the organization and technical infrastructures*** *may play in your adoption decision regarding the proposed model.*

10. I have at my disposal, the resources necessary to make use of the proposed model.

Strongly agree	**Agree**	**Neutral**	**Disagree**	**Strongly disagree**

11. I have the required knowledge to make use of the proposed model.

Strongly agree	**Agree**	**Neutral**	**Disagree**	**Strongly disagree**

12. The proposed model is compatible with other processes that I use for work related activities.

Strongly agree	**Agree**	**Neutral**	**Disagree**	**Strongly disagree**

13. I will be able to obtain help from others when I have difficulties in using the proposed model.

Strongly agree	**Agree**	**Neutral**	**Disagree**	**Strongly disagree**

PART 4 (Behavioural Intention):

14. I intend to start making use of the proposed model for upcoming software development projects.

Strongly agree	**Agree**	**Neutral**	**Disagree**	**Strongly disagree**

PART 5 (Type of Usage):

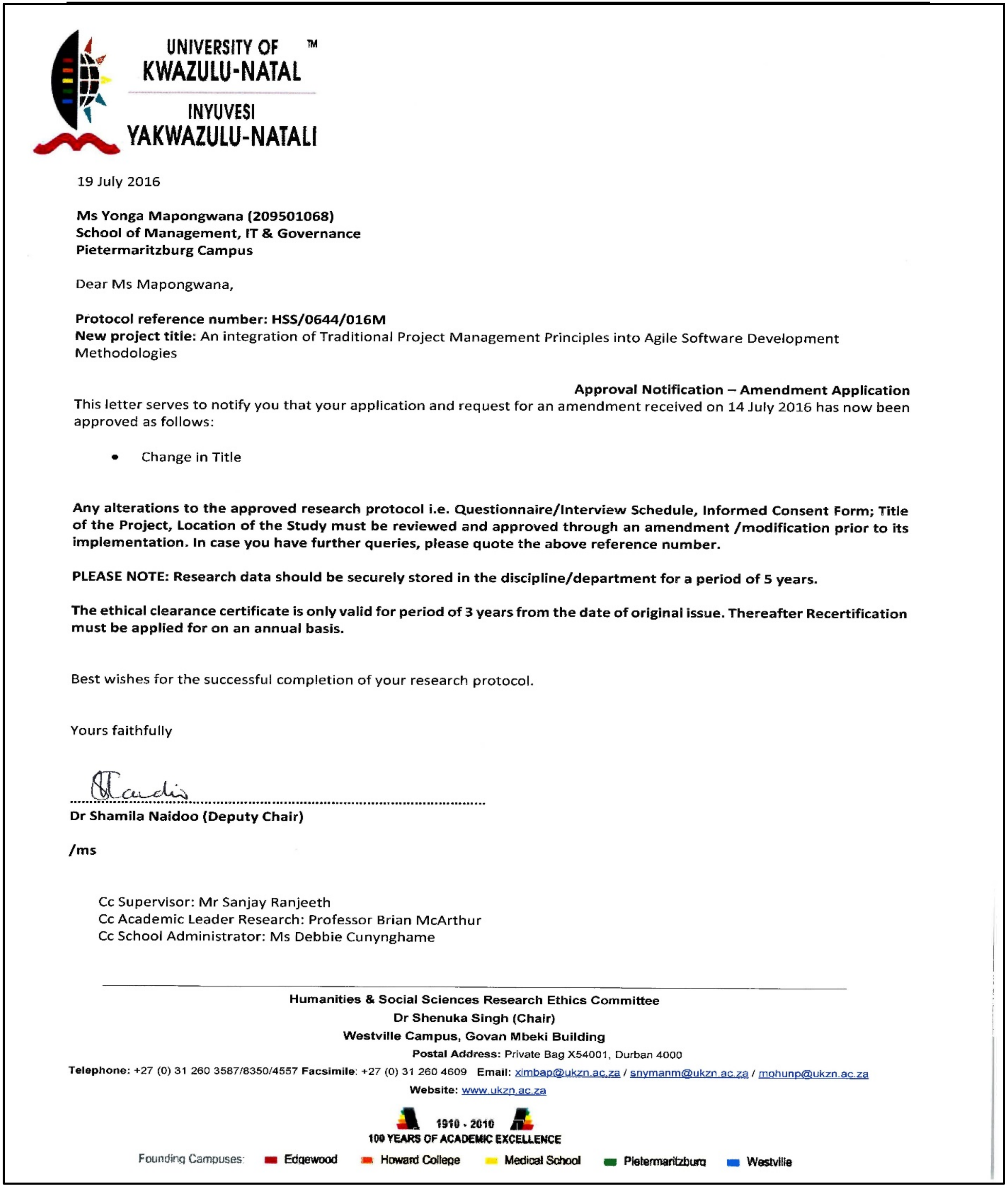
15. Please select the type of software development projects that you think will be most appropriate for use of the proposed model.

All Projects	**Small Projects**	**Large/Complex Projects**	**Medium scale projects**	**No projects**

PART 6 (Suggested Enhancement):

Please make suggestions with regards to how you think that the proposed model may be improved.

APPENDIX C: ETHICAL CLEARANCE



UNIVERSITY OF ™
KWAZULU-NATAL
INYUVESI
YAKWAZULU-NATALI

19 July 2016

Ms Yonga Mapongwana (209501068)
School of Management, IT & Governance
Pietermaritzburg Campus

Dear Ms Mapongwana,

Protocol reference number: HSS/0644/016M
New project title: An integration of Traditional Project Management Principles into Agile Software Development Methodologies

Approval Notification – Amendment Application

This letter serves to notify you that your application and request for an amendment received on 14 July 2016 has now been approved as follows:

- Change in Title

Any alterations to the approved research protocol i.e. Questionnaire/Interview Schedule, Informed Consent Form; Title of the Project, Location of the Study must be reviewed and approved through an amendment /modification prior to its implementation. In case you have further queries, please quote the above reference number.

PLEASE NOTE: Research data should be securely stored in the discipline/department for a period of 5 years.

The ethical clearance certificate is only valid for period of 3 years from the date of original issue. Thereafter Recertification must be applied for on an annual basis.

Best wishes for the successful completion of your research protocol.

Yours faithfully

....

Dr Shamila Naidoo (Deputy Chair)

/ms

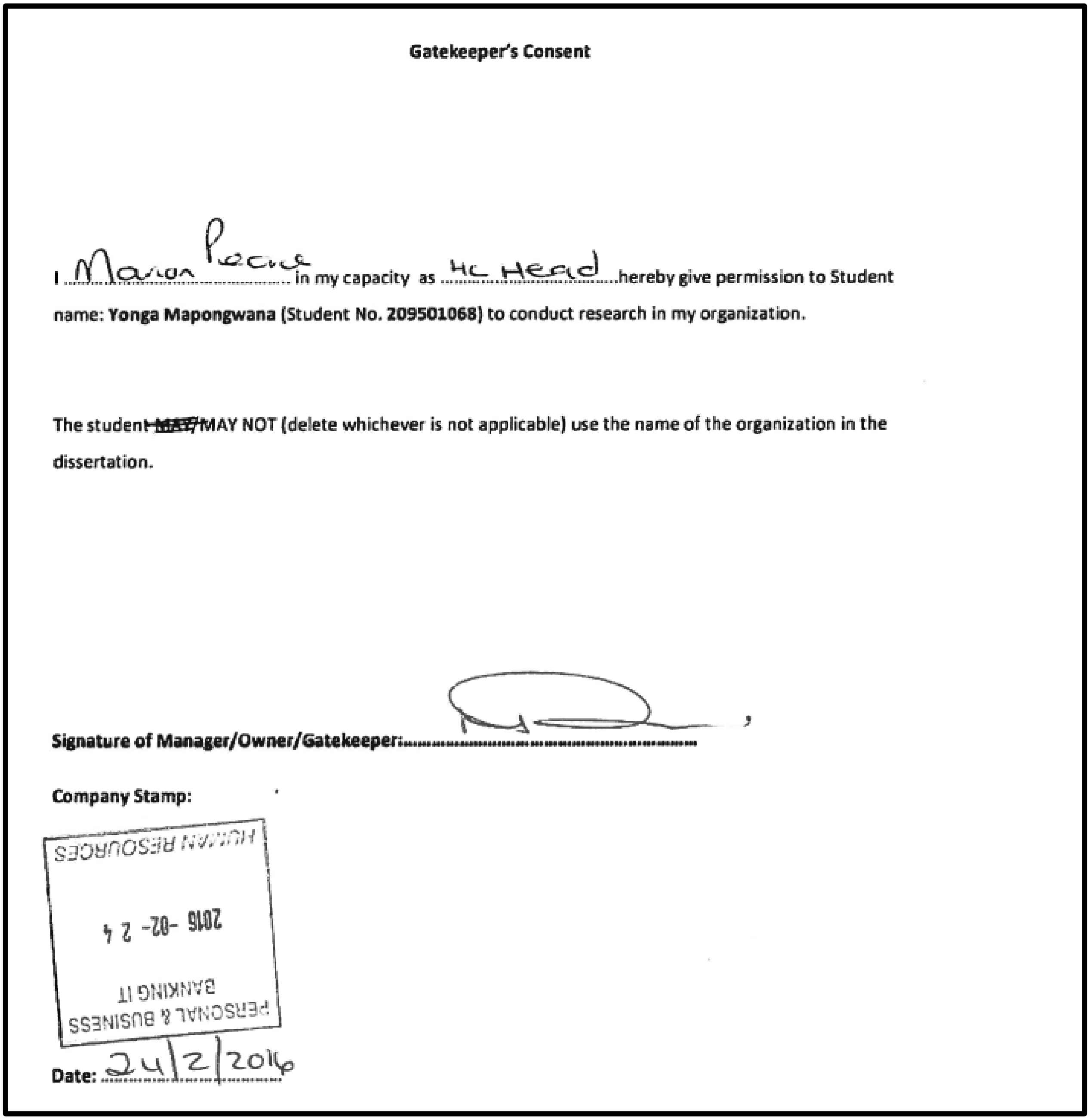
Cc Supervisor: Mr Sanjay Ranjeeth
Cc Academic Leader Research: Professor Brian McArthur
Cc School Administrator: Ms Debbie Cunynghame

Humanities & Social Sciences Research Ethics Committee
Dr Shenuka Singh (Chair)
Westville Campus, Govan Mbeki Building
Postal Address: Private Bag X54001, Durban 4000
Telephone: +27 (0) 31 260 3587/8350/4557 **Facsimile:** +27 (0) 31 260 4609 **Email:** ximbap@ukzn.ac.za / snymanm@ukzn.ac.za / mohunp@ukzn.ac.za
Website: www.ukzn.ac.za

1910 - 2010
100 YEARS OF ACADEMIC EXCELLENCE

Founding Campuses: 🟥 Edgewood 🟧 Howard College 🟨 Medical School 🟩 Pietermaritzburg 🟦 Westville

APPENDIX D: GATEKEEPER'S CONSENT



Gatekeeper's Consent

I Marion Pearce in my capacity as HC Head hereby give permission to Student name: **Yonga Mapongwana** (Student No. **209501068**) to conduct research in my organization.

The student ~~MAY~~/MAY NOT (delete whichever is not applicable) use the name of the organization in the dissertation.

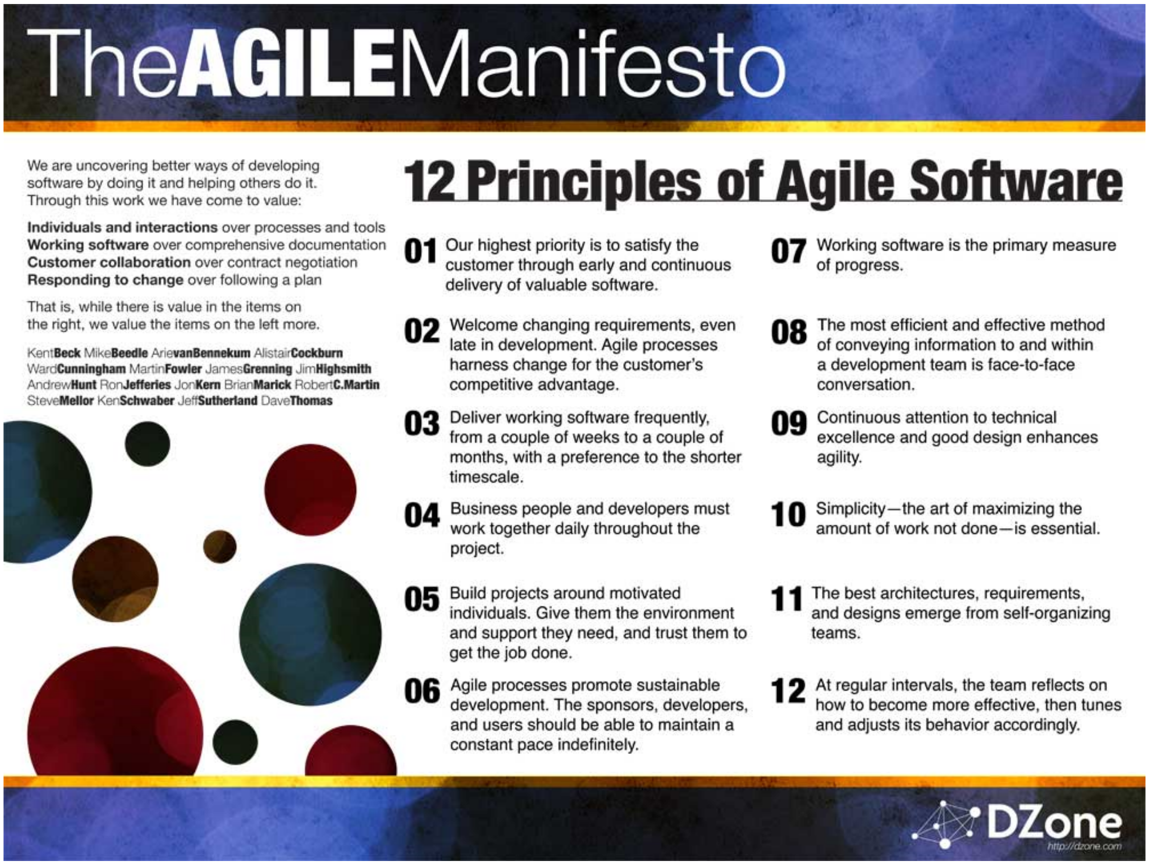
Signature of Manager/Owner/Gatekeeper:

Company Stamp:

PERSONAL & BUSINESS
BANKING IT
2016 -02- 2 4
HUMAN RESOURCES

Date: 24/2/2016

APPENDIX E: AGILE MANIFESTO



The**AGILE**Manifesto

We are uncovering better ways of developing software by doing it and helping others do it. Through this work we have come to value:

Individuals and interactions over processes and tools
Working software over comprehensive documentation
Customer collaboration over contract negotiation
Responding to change over following a plan

That is, while there is value in the items on the right, we value the items on the left more.

Kent**Beck** Mike**Beedle** Arie**vanBennekum** Alistair**Cockburn** Ward**Cunningham** Martin**Fowler** James**Grenning** Jim**Highsmith** Andrew**Hunt** Ron**Jefferies** Jon**Kern** Brian**Marick** Robert**C.Martin** Steve**Mellor** Ken**Schwaber** Jeff**Sutherland** Dave**Thomas**

12 Principles of Agile Software

01 Our highest priority is to satisfy the customer through early and continuous delivery of valuable software.

02 Welcome changing requirements, even late in development. Agile processes harness change for the customer's competitive advantage.

03 Deliver working software frequently, from a couple of weeks to a couple of months, with a preference to the shorter timescale.

04 Business people and developers must work together daily throughout the project.

05 Build projects around motivated individuals. Give them the environment and support they need, and trust them to get the job done.

06 Agile processes promote sustainable development. The sponsors, developers, and users should be able to maintain a constant pace indefinitely.

07 Working software is the primary measure of progress.

08 The most efficient and effective method of conveying information to and within a development team is face-to-face conversation.

09 Continuous attention to technical excellence and good design enhances agility.

10 Simplicity—the art of maximizing the amount of work not done—is essential.

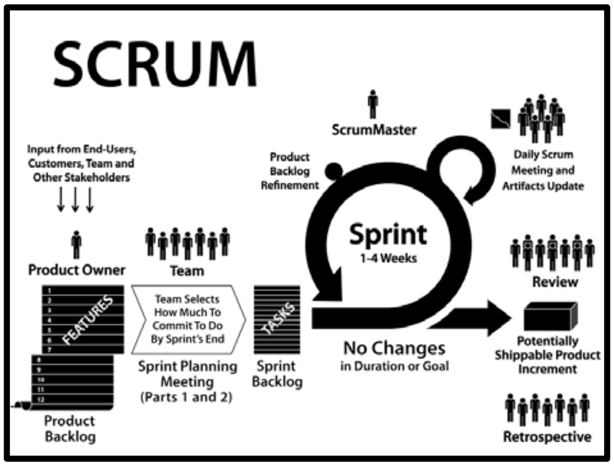
11 The best architectures, requirements, and designs emerge from self-organizing teams.

12 At regular intervals, the team reflects on how to become more effective, then tunes and adjusts its behavior accordingly.

DZone
http://dzone.com

APPENDIX F: SCRUM FRAMEWORK

Scrum is an iterative, incremental framework for projects and product or application development. It structures development in cycles of work called **Sprints**. These iterations are no more than one month each, and take place one after the other without pause. The Sprints are *timeboxed* – they end on a specific date whether the work has been completed or not, and are *never extended*. At the beginning of each Sprint, a cross-functional team selects **items** (customer requirements) from a prioritized list. The team commits to complete the items by the end of the Sprint. During the Sprint, the chosen items do not change. Every day the team gathers briefly to inspect its progress, and adjust the next steps needed to complete the work remaining. At the end of the Sprint, the team reviews the Sprint with stakeholders, and demonstrates what it has built. People obtain feedback that can be incorporated in the next Sprint. Scrum emphasizes working product at the end of the Sprint that is really "done"; in the case of software, this means code that is integrated, fully tested and potentially shippable. Key roles, artifacts, and events are summarized in Figure 1. A major theme in Scrum is "inspect and adapt." Since development inevitably involves learning, innovation, and surprises, Scrum emphasizes taking a short step of development, inspecting both the resulting product and the efficacy of current practices, and then adapting the product goals and process practices. Repeat forever.


SCRUM
Input from End-Users, Customers, Team and Other Stakeholders
Product Owner
Team
ScrumMaster
Product Backlog Refinement
Daily Scrum Meeting and Artifacts Update
Sprint
1-4 Weeks
Review
FEATURES
1
2
3
4
5
6
7
8
9
10
11
12
Team Selects How Much To Commit To Do By Sprint's End
TASKS
No Changes
in Duration or Goal
Potentially Shippable Product Increment
Sprint Planning Meeting
(Parts 1 and 2)
Sprint Backlog
Product Backlog
Retrospective

Scrum Roles

In Scrum, there are three roles: The Product Owner, The Team, and The ScrumMaster. Together these are known as The Scrum Team. The **Product Owner** is responsible for maximizing return on investment (ROI) by identifying product features, translating these into a prioritized list, deciding which should be at the top of the list for the next Sprint, and continually re-prioritizing and refining the list. The Product Owner has profit and loss responsibility for the product, assuming it is a commercial product. In the case of an internal application, the Product Owner is not responsible for ROI in the sense of a commercial product (that will generate revenue), but they are still responsible for maximizing ROI in the sense of choosing – each Sprint – the highest-business-value lowest-cost items. In practice, ‘value’ is a fuzzy term and prioritization may be influenced by the desire to satisfy key customers, alignment with strategic objectives, attacking risks, improving, and other factors. In some cases, the Product Owner and the customer are the same person; this is common for internal applications. In others, the customer might be millions of people with a variety of needs, in which case the Product Owner role is similar to the Product Manager or Product Marketing Manager Position in many product organizations. However, the Product Owner is somewhat different than a traditional Product Manager because they actively and frequently interact with the Team, personally offering the priorities and reviewing the results each two- or four-week iteration, rather than delegating development decisions to a project manager. It is important to note that in Scrum there is one and only one person who serves as – and has the final authority of – Product Owner, and he or she is responsible for the value of the work.

The Team builds the product that the Product Owner indicates: the application or website, for example. The Team in Scrum is “cross-functional” – it includes all the expertise necessary to deliver the potentially shippable product each Sprint – and it is “self-organizing” (self-managing), with a very high degree of autonomy and accountability. The Team decides what to commit to, and how best to accomplish that commitment; in Scrum lore, the Team is known as “Pigs” and everyone else in the organization are “Chickens” (which comes from a joke about a pig and a chicken deciding to open a restaurant called “Ham and Eggs,” and the pig having second thoughts because “he would be truly committed, but the chicken would only be involved”). The Team in Scrum is seven plus or minus two people, and for a software product the Team might include people with skills in analysis, development, testing, interface design, database design, architecture, documentation, and so on. The Team develops the product and provides ideas to the Product Owner about how to make the product great. In Scrum the Teams are most productive and effective if all members are 100 percent dedicated to the work for one product during the Sprint; avoid multitasking across multiple products or projects. Stable teams are associated with higher productivity, so avoid changing Team members. Application groups with many people are organized into multiple Scrum Teams, each focused on different features for the product, with close coordination of their efforts. Since one team often does all the work (planning, analysis, programming, and testing) for a complete customer-centric feature, Teams are also known as *feature teams*.

The **ScrumMaster** helps the product group learn and apply Scrum to achieve business value. The ScrumMaster does whatever is in their power to help the Team and Product Owner be successful.

The ScrumMaster is *not* the manager of the Team or a project manager; instead, the ScrumMaster serves the Team, protects them from outside interference, and educates and guides the Product Owner and the Team in the skillful use of Scrum. The ScrumMaster makes sure everyone (including the Product Owner, and those in management) understands and follows the practices of Scrum, and they help lead the organization through the often difficult change required to achieve success with agile development. Since Scrum makes visible many impediments and threats to the Team’s and Product Owner’s effectiveness, it is important to have an engaged ScrumMaster working energetically to help resolve those issues, or the Team or Product Owner will find it difficult to succeed. There should be a dedicated full-time ScrumMaster, although a smaller Team might have a team member play this role (carrying a lighter load of regular work when they do so). Great ScrumMasters can come from any background or discipline: Engineering, Design, Testing, Product Management, Project Management, or Quality Management.

The ScrumMaster and the Product Owner cannot be the same individual; at times, the ScrumMaster may be called upon to push back on the Product Owner (for example, if they try to introduce new deliverables in the middle of a Sprint). And unlike a project manager, the ScrumMaster does not tell people what to do or assign tasks – they facilitate the process, supporting the Team as it organizes and manages itself. If the ScrumMaster was previously in a position managing the Team, they will need to significantly change their mindset and style of interaction for the Team to be successful with Scrum. Note there is no role of project manager in Scrum. This is because none is needed; the traditional responsibilities of a project manager have been divided up and reassigned among the three Scrum roles. Sometimes an (ex-) project manager can step into the role of ScrumMaster, but this has a mixed record of success – there is a fundamental difference between the two roles, both in day-to-day responsibilities and in the mindset required to be successful. A good way to understand thoroughly the role of the ScrumMaster, and start to develop the core skills needed for success, is the Scrum Alliance’s Certified ScrumMaster training.

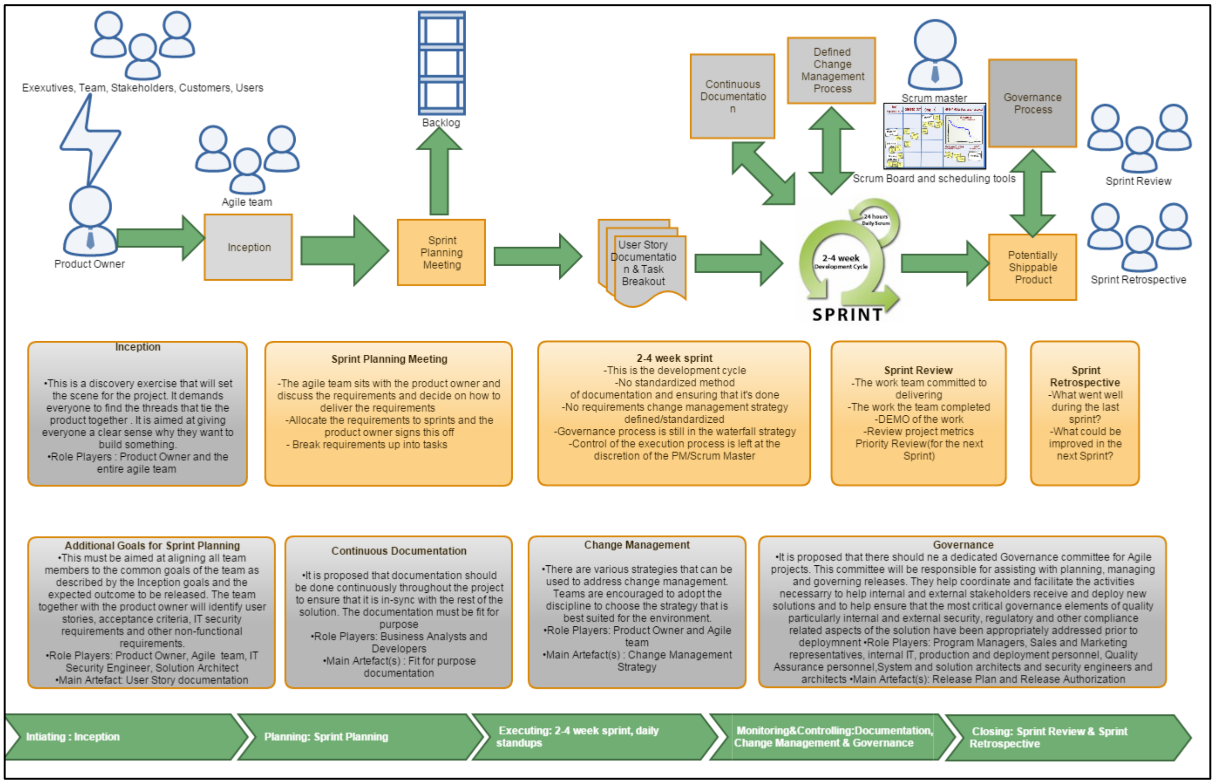
In addition to these three roles, there are other contributors to the success of the product, including functional managers (for example, an engineering manager). While their role changes in Scrum, they remain valuable. For example:

- They support the Team by respecting the rules and spirit of Scrum

- They help remove impediments that the Team and Product Owner identify
- They make their expertise and experience available

In Scrum, these individuals replace the time they previously spent playing the role of "nanny" (assigning tasks, getting status reports, and other forms of micromanagement) with time as "guru" and "servant" of the Team (mentoring, coaching, helping remove obstacles, helping problem-solve, providing creative input, and guiding the skills development of Team members). In this shift, managers may need to change their management style; for example, using Socratic questioning to help the Team discover the solution to a problem, rather than simply deciding a solution and assigning it to the Team.

Appendix G: AGILE/PM FRAMEWORK


Exexutives, Team, Stakeholders, Customers, Users
Backlog
Agile team
Product Owner
Inception
Sprint Planning Meeting
User Story Documentation & Task Breakout
Continuous Documentation
Defined Change Management Process
Scrum master
Scrum Board and scheduling tools
Governance Process
Sprint Review
24 hours Daily Scrum
2-4 week Development Cycle
SPRINT
Potentially Shippable Product
Sprint Retrospective

Inception

•This is a discovery exercise that will set the scene for the project. It demands everyone to find the threads that tie the product together . It is aimed at giving everyone a clear sense why they want to build something.

•Role Players : Product Owner and the entire agile team

Sprint Planning Meeting

-The agile team sits with the product owner and discuss the requirements and decide on how to deliver the requirements

-Allocate the requirements to sprints and the product owner signs this off

- Break requirements up into tasks

2-4 week sprint

-This is the development cycle

-No standardized method of documentation and ensuring that it's done

-No requirements change management strategy defined/standardized

-Governance process is still in the waterfall strategy

-Control of the execution process is left at the discretion of the PM/Scrum Master

Sprint Review

-The work team committed to delivering

-The work the team completed

-DEMO of the work

-Review project metrics Priority Review(for the next Sprint)

Sprint Retrospective

-What went well during the last sprint?

-What could be improved in the next Sprint?

Additional Goals for Sprint Planning

•This must be aimed at aligning all team members to the common goals of the team as described by the Inception goals and the expected outcome to be released. The team together with the product owner will identify user stories, acceptance criteria, IT security requirements and other non-functional requirements.

•Role Players: Product Owner, Agile team, IT Security Engineer, Solution Architect

•Main Artefact: User Story documentation

Continuous Documentation

•It is proposed that documentation should be done continuously throughout the project to ensure that it is in-sync with the rest of the solution. The documentation must be fit for purpose

•Role Players: Business Analysts and Developers

•Main Artefact(s) : Fit for purpose documentation

Change Management

•There are various strategies that can be used to address change management. Teams are encouraged to adopt the discipline to choose the strategy that is best suited for the environment.

•Role Players: Product Owner and Agile team

•Main Artefact(s) : Change Management Strategy

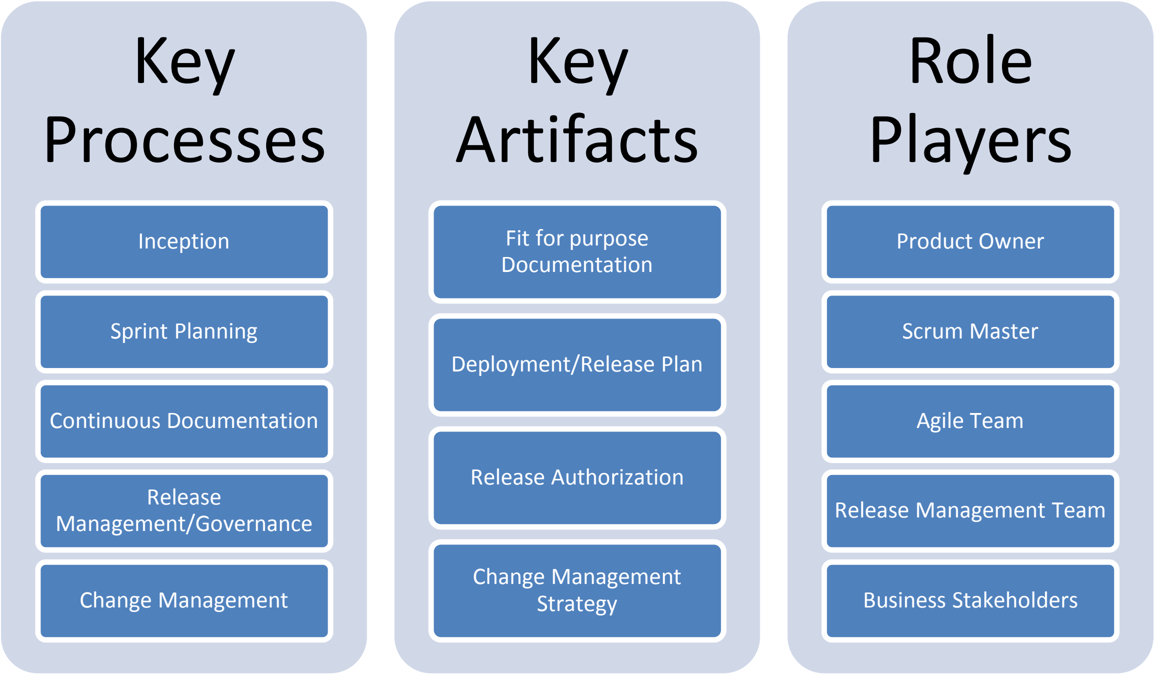
Governance

•It is proposed that there should ne a dedicated Governance committee for Agile projects. This committee will be responsible for assisting with planning, managing and governing releases. They help coordinate and facilitate the activities necessarry to help internal and external stakeholders receive and deploy new solutions and to help ensure that the most critical governance elements of quality particularly internal and external security, regulatory and other compliance related aspects of the solution have been appropriately addressed prior to deploymnent •Role Players: Program Managers, Sales and Marketing representatives, internal IT, production and deployment personnel, Quality Assurance personnel,System and solution architects and security engineers and architects •Main Artefact(s): Release Plan and Release Authorization

Intiating : Inception
Planning: Sprint Planning
Executing: 2-4 week sprint, daily standups
Monitoring&Controlling:Documentation, Change Management & Governance
Closing: Sprint Review & Sprint Retrospective

Agile/PM Framework

Below is an illustration detailing the critical aspects that have been incorporated into the Scrum process that is currently being used by Bank A in the Agile/PM framework.



Key Processes
- Inception
- Sprint Planning
- Continuous Documentation
- Release Management/Governance
- Change Management

Key Artifacts
- Fit for purpose Documentation
- Deployment/Release Plan
- Release Authorization
- Change Management Strategy

Role Players
- Product Owner
- Scrum Master
- Agile Team
- Release Management Team
- Business Stakeholders

Key Components to embed to ensure PM in Agile Strategy

- Inception: Adapted from Rasmusson (2010) the inception happens prior to starting any work on the project. It is aimed at achieving two things:

o Ensuring that the entire agile team is on the same page before the project has even started.

o Ensuring that all the tough questions are addressed up front.

The inception must cover the following key areas which Rasmusson (2010) refers to as the inception deck:

o Commander’s Intent: This is a mission statement. It is basically owned by the Product Owner, however the entire team needs to collaborate to formulate it. It is a one sentence statement that regulates every decision that will be made during the course of the project. It is what the team will refer to when they need something shipped and are evaluating as to whether a feature should or shouldn’t be released into production. Whilst the Product Owner is still in control of his /her product backlog items, especially when it comes to prioritising and grooming the product backlog, the Commander's Intent encourages everyone to think collectively as a team and make decisions as a team. During the course of the project the team constantly questions whether something meets the Commander's Intent? If yes, then it is in scope. If not, it is out of scope

- Elevator Pitch: This is a two sentence statement that ensures that everyone in the team has the same understanding about what is being done for who and why; what sets you apart from competitors. The objective of an Elevator Pitch, is that it challenges the team together with the business stakeholders to assess whether they should be proceeding with the project or not.
- The product box: If the software product were to be bought off the shelf, what would it look like? Would you buy it? This is aimed at translating what could be product features into benefits. Although no one knows what those features are yet, this enables the team to envision at a high-level what the product might look like and what is captivating about it.
- Pre-mortem: This is similar to a post-mortem but done before the project begins. It is triggered by the question- "What keeps us up at night regarding the project?"

The pre-mortem helps in encouraging the team members to be transparent and express what their fears are and what things they simply don't want to see happen. These need to be expressed and find ways to ensure that they do not happen.

- What is going to give? : In situations where the team is faced with too much to do and not enough time, is it better to :
- Cut scope?
- Add more people to the project?
- Push out the release date? Sacrifice quality?

  - The Definition of Done (DoD) : What will it take to release/deploy at the end of each sprint

  The role players in the inception meeting are the business stakeholders including the product owner and the agile team.

- Sprint Planning: In this session the Product owner is tasked with defining the user stories, the system architect is required to be present to ensure that the user stories are aligned to the architectural vision of the organisation. The IT Security Engineer will define IT security requirements aligned to each user story. The Product owner together with the team will then elaborate on the user story detail and define the acceptance criteria for each. The team may choose to further divide the stories into tasks and into hours to better refine their understanding of the work ahead. The team can now commit to a set of goals for the iteration. The Business Analyst will be tasked with documenting the user stories, with IT security requirements, and their acceptance criteria and all the stakeholders must sign off this document. Formally documenting the user stories is done to ensure that the user stories are quality assured and that every stakeholder signs off on what was discussed and what has been agreed on for accountability purposes. This will also make it easier to on-board a new member into the team and give other people not in the project team a point of reference. The following is what should be included in the document:

  - Narrative of the user stories
  - Scope (Impacted systems, Impacted Areas) – this could include a context diagram
  - Scenarios
  - Acceptance Criteria
  - IT Security requirements and other non-functional requirements

- Documentation: It is proposed that documentation should be done during the iteration towards the end when the majority of the development work has been done. This helps the design architects and other stakeholders involved in the documentation process to avoid having to rework the documentation. The documentation should be useful and fit for purpose – this includes user manuals, training materials, operation manuals and system overviews. Documenting continuously throughout the project ensures that the documentation is in sync with the rest of the solution, the team will know that they have sufficient documentation to support what they've built to date. The implication is that the solution will in fact be potentially shippable at the end of each iteration. Continuous documentation also ensures that the critical details of what needs to be captured are remembered because the feedback cycle between doing the work and documenting what you've done is short. Evolving requirements could mean that the business analysts will need to update the deliverable documentation to reflect these changes. To ensure that team members responsible for the documentation i.e. Business Analysts do not put off the documentation process it must be made an acceptance criteria.

- Governance: Adapted from the Scaled Agile Framework (SAFe) by Leffingwell (2015)the Governance team is a dedicated team which is responsible for assisting the project team in planning, managing and governing releases and has the authority to guide the teams toward the business goals. This team will also be responsible for coordinating the implementation of all the capabilities and features over the multiple iterations within a release especially as new issues, roadblocks, dependencies, overlaps, over-scopes and gaps in vision and backlogs are uncovered. They help coordinate and facilitate the activities necessary to help internal and external stakeholders receive and deploy the new solution and they help ensure that the most critical governance elements of quality – particularly internal and external security, regulatory and other compliance-related aspects of the solution – have been appropriately addressed prior to deployment. Primary responsibilities of this committee include:

- Ensuring that the organization’s release governance is understood
- Communicating release status to external stakeholders
- Ensuring that an appropriate deployment/distribution plan is in place
- Coordinating with marketing and with Product and Solution Management on internal and external communications
- Validating that the solution meets relevant quality and governance criteria
- Providing final authorization for the release

The governance committee comprises of individuals from the following areas:

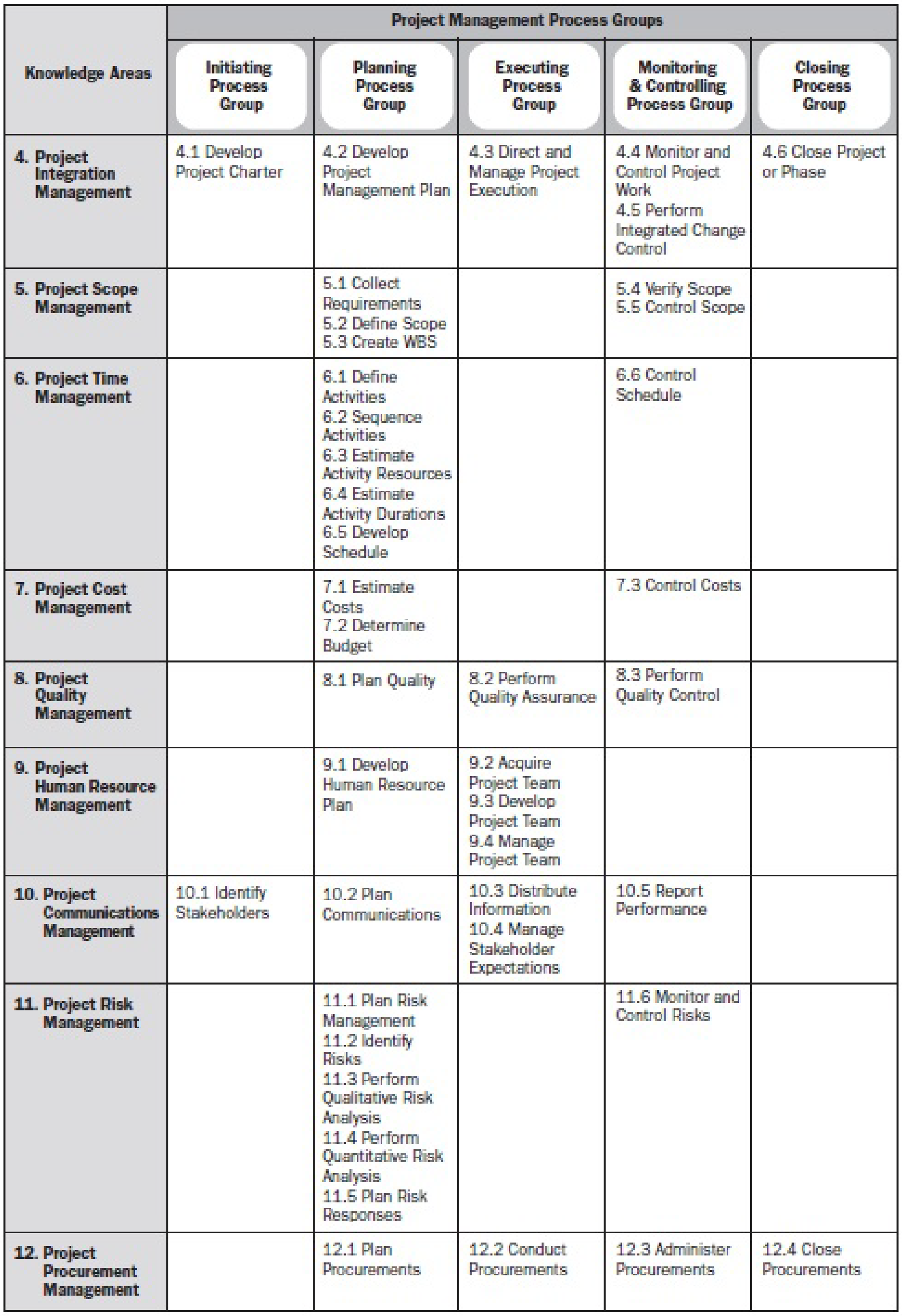
- Program Managers
- Senior representatives from sales and marketing
- Internal IT, production and deployment personnel
- Senior and solution level Quality Assurance personnel who are responsible for the final assessment of solution level quality, performance and suitability for use
- System and Solution Architects
- Security Engineers and Architects

- *Change Management*

  Requirements change all the time and stakeholders change their minds for a number of reasons. It could be that they’ve missed a requirement and while working on an existing they realise that it’s missing a feature; it could also be that the market place has changed, a competitor releases a new product which contains functionality that your product doesn’t have and lastly legislation changes could require new features or changes to existing features of the software. Regardless of what the reason for the change is, it is important to realise that “freezing” requirements early on in the project lifecycle guarantees that you’re not developing what people need but what they initially thought they wanted. As much as it is important to embrace changes in requirements, it is equally important that the change is managed correctly and done responsibly during the project lifecycle. There are different strategies that can be used to address change management. The Disciplined Agile Delivery (DAD) process framework (Ambler & Lines, 2012) describes a number of common strategies that may be adopted for managing change. These options include formal change management, Scrum’s product backlog strategy, work item stacks, work item pools, and no strategy at all. Please refer to Appendix H for a full description of the strategies with their advantages and disadvantages. The team will have to adopt the discipline to choose the strategy that is best suited for the environment. Ambler & Lines (2012)provide contextual advice to assist teams navigate these important process decisions but are ultimately decisions that the team needs to make.

Mapping to Project Management Body of Knowledge

Table 3-1 in the PMBOK® Guide (PMI, 2000) shows the Knowledge Areas down the side, the Process Groups along the top and then maps the difference processes in the relevant boxes where those two axes cross. This table is displayed below.



Knowledge Areas	Project Management Process Groups				
	Initiating Process Group	Planning Process Group	Executing Process Group	Monitoring & Controlling Process Group	Closing Process Group
4. Project Integration Management	4.1 Develop Project Charter	4.2 Develop Project Management Plan	4.3 Direct and Manage Project Execution	4.4 Monitor and Control Project Work 4.5 Perform Integrated Change Control	4.6 Close Project or Phase
5. Project Scope Management		5.1 Collect Requirements 5.2 Define Scope 5.3 Create WBS		5.4 Verify Scope 5.5 Control Scope	
6. Project Time Management		6.1 Define Activities 6.2 Sequence Activities 6.3 Estimate Activity Resources 6.4 Estimate Activity Durations 6.5 Develop Schedule		6.6 Control Schedule	
7. Project Cost Management		7.1 Estimate Costs 7.2 Determine Budget		7.3 Control Costs	
8. Project Quality Management		8.1 Plan Quality	8.2 Perform Quality Assurance	8.3 Perform Quality Control	
9. Project Human Resource Management		9.1 Develop Human Resource Plan	9.2 Acquire Project Team 9.3 Develop Project Team 9.4 Manage Project Team		
10. Project Communications Management	10.1 Identify Stakeholders	10.2 Plan Communications	10.3 Distribute Information 10.4 Manage Stakeholder Expectations	10.5 Report Performance	
11. Project Risk Management		11.1 Plan Risk Management 11.2 Identify Risks 11.3 Perform Qualitative Risk Analysis 11.4 Perform Quantitative Risk Analysis 11.5 Plan Risk Responses		11.6 Monitor and Control Risks	
12. Project Procurement Management		12.1 Plan Procurements	12.2 Conduct Procurements	12.3 Administer Procurements	12.4 Close Procurements

The table below is an adaptation of Table 3-1 in the PMBOK® Guide. The researcher used the mapping contained on the table to enforce and visualise the flow and integration of each process area. Doing so displays how the proposed Guideline combined with the Scrum methodology currently being used at Bank A map into the Project Management Body of Knowledge (PMBOK).

Project Management Process Group and Knowledge Area Mapping

Knowledge Areas	Project Management Process Groups				
	Initiating	Planning	Executing	Monitoring &Controlling	Closing
1. Integration	1.1 Inception	1.2 Sprint Planning	1.3 Daily Stand-up		1.4 Sprint Retrospectives
2. Scope	2.1 Release Plan 2.2 Vision & Roadmap	2.3 User Story Documentation 2.4 Product backlog 2.5 Sprint backlog			2.6 Sprint Review 2.7 Demo
3. Time		3.1 User story points			

		3.2 Task out stories 3.3 Sprint plan			
4. Cost					
5. Quality	5.1 Definition of Done	5.2 Acceptance Criteria	5.3 Continuous Documentation		
6. HR	6.1 Product Owner Agile team	6.2 Cross functional team	6.3 Self-organised team	6.4 Scrum Master	
7. Communication			7.1 Direct business participation 7.2 Co-located teams 7.3 Face to face	7.4 Sprint burndowns 7.5 Release Charts	
8. Risk		8.1 IT Security Requirements		8.3 Change Management	

		8.2 System Architects		8.4 Governance	
9. Stakeholder			9.1 Product Owner		9.2 Authorization of Release

APPENDIX H: CHANGE MANAGEMENT STRATEGIES

Formal Change Management

With a formal approach to change management, the work to be performed is typically defined in detail and agreed to early in the project; and any changes to that planned work are then managed throughout the lifecycle. In simple situations, the product owner will be responsible for considering and acting on change requests for requirements or defect reports; although at scale, a change control board (CCB) may exist and meet regularly (ideally, at least once an iteration) to manage any change requests. The team lead is typically responsible for making decisions pertaining to requests from other teams or personal requests. Part of deciding whether to accept a change may include analysis to determine the impact/cost of the change versus the priority and business value for the customer. The potential advantages of a formal approach to change management on an agile project include its applicability to regulatory situations where formal change management is mandated. Past surveys have found that roughly one-third of agile teams work in environments where one or more regulations apply. This approach also works well for environments where the requirements seldom change, although this proves to be very rare in practice given the hyper-competitiveness of today's marketplace.

There are many disadvantages to formal change management. First, it motivates stakeholders to accept a big requirements up front (BRUF) — an approach where detailed specifications are created and agreed to early in the project, thereby taking on all the disadvantages of BRUF (and there are many). Second, it can motivate onerous requirements traceability efforts to aid impact-analysis efforts for the CCB. Granted, it's possible to largely automate traceability if you adopt an acceptance test-driven development (ATDD) approach and a tool that supports it. Third, formal change management can add significant overhead to the effort, particularly when requirements change often, increasing project cost and extending the delivery schedule. Fourth, formal change management often evolves into a change prevention strategy on the part of IT staff, something that is ethically questionable at best. Fifth, for small changes, the overhead of considering the change may be greater than the cost of actually implementing it in practice. This indicates that you need to consider adopting a change triage strategy as well, further complicating your process.

Scrum Product Backlog

A common agile approach to change management is Scrum's product backlog strategy. A foundational concept in Scrum is that requirements, and optionally defect reports, should be managed as an ordered queue called a "product backlog." The contents of the product backlog will vary to reflect evolving requirements, with the product owner responsible for prioritizing work on the backlog based on the business value of the work item. Just enough work to fit into the current iteration is taken off the top of the stack by the team at the start of each iteration as part of the iteration planning activity.

This approach has several potential advantages. First, it is simple to understand and implement. Second, because the team is working in priority order, it is always focusing on the highest business value at the time, thereby maximizing potential return on investment (ROI). Third, it is very easy for stakeholders to define new requirements and refocus existing ones.

There are also potential disadvantages. You will need an additional strategy to manage other work item types, work such as people assisting other teams or taking time to attend training classes. The product backlog must be groomed throughout the project lifecycle to maintain priority order, and that effort can become a significant overhead if the requirements change rapidly. It also requires a supporting strategy to address non-functional requirements (NFRs), with a product backlog strategy, practitioners new to agile will often adopt an overly simplistic approach that focuses only on managing functional requirements. Finally, this approach requires a product owner who is capable of managing the backlog in a timely and efficient manner — something that organizations new to agile often struggle with.

Work Item Stacks

Development does not occur in a vacuum. Potential defects and enhancement requests are often reported from operations and support teams working with existing versions of a solution or from independent test teams working in parallel. Because your team is likely one of many within an organizational ecosystem, you may receive requests to review the work of other teams, to collaborate with them to ensure that your solution works well with what they're producing, and other similar requests. Individual team members will have personal requests to attend training classes, take vacations, and attend conferences, and so on. All these work items should be managed and acted upon by your team accordingly via your change management strategy.

A work item stack, sometimes called a "work item queue," is an extension to Scrum's product backlog that includes all types of work items (requirements, defects, team collaboration requests, and personal requests). Work items are prioritized based on a variety of considerations, including both stakeholder value (an extension of business value to address all stakeholder concerns, not just business ones) and team health considerations. Mature teams will take risk into consideration when prioritizing work. For example, to reduce technical risk, a DAD team will prove that their architectural strategy works by creating a working, end-to-end skeleton that implements several high-risk requirements.

There are several potential advantages to this approach. First, it includes the benefits of the Scrum product backlog described earlier. Second, it explicitly manages all work item types in a single, consistent manner and thereby simplifies the overall change management process. And, it provides an explicit strategy for addressing high-risk work early in a project thereby increasing the chance of project success

There are also some potential disadvantages with work item stacks. The most serious is that it increases the responsibilities of the product owner — already a tough role — to address team health and project risk considerations. Second, like product backlogs, the work item stack needs to be groomed throughout the project lifecycle, thus adding overhead. Third, it still requires a strategy to address non-functional requirements. Finally, although human considerations such as training and vacation should be addressed, they are often deprioritized by the product owner in favour of function-oriented work items such as new requirements and defects.

Work Item Pools

One lean approach to work item management is to treat work items as if they're in an options pool, not an ordered stack. This strategy explicitly recognizes that there are different ways to prioritize work items — the "standard way" based on stakeholder value, time-dependent strategies with defined delivery dates, the occasional emergency/high-priority work item that must be expedited, and the intangible team health work items captured in personal requests. Anyone can identify work items and place them in the pool, although the product owner is most likely to focus on doing so. The entire team, including the product owner, is responsible for pulling work from the work item pool appropriately. A particular work item is determined whenever the team has the bandwidth to pull new work into their process.

The primary advantages of this approach are that it is flexible enough to address several prioritization schemes and that it doesn't require any investment in backlog/stack grooming.

The disadvantages are that it requires teams to be responsible and disciplined enough to pull work out the pool in a fair and appropriate manner. This discipline requires teams to consider a variety of issues, including stakeholder value, risk, team health, and enterprise issues. This method also can be very threatening to traditional organizations accustomed to telling teams what the priorities are. It requires strict control over the number of work items to be expedited (everything can't be of utmost priority, otherwise the other categories are never addressed). Another common problem is that teams new to agile may prioritize team health considerations or infrastructure work over delivering stakeholder value. This strategy really does require significant discipline to pull off.

No Strategy at All

Finally, there is always the option of having no change strategy at all. This non-existent strategy is valid for very small or simple projects. The primary advantage is that there is no overhead to this approach. The primary disadvantage is that there is significant potential for low value work to be implemented because nobody is prioritizing things. My experience is that the no-strategy approach is a sign of an ad-hoc project team posing as an agile team.