



**The Integration of Agile Methodology in the
Development of Software Project**

دمج منهجية الأجايل في تطوير مشاريع البرمجيات

by

MOHAMMED ALMHEIRI

**Dissertation submitted in partial fulfilment
of the requirements for the degree of
MSc PROJECT MANAGEMENT**

at

The British University in Dubai

August 2024

Declaration

I warrant that the content of this research is the direct result of my own work and that any use made in it of published or unpublished copyright material falls within the limits permitted by international copyright conventions.

I understand that a copy of my research will be deposited in the University Library for permanent retention.

I hereby agree that the contents of this dissertation for which I am author and copyright holder may be copied and distributed by The British University in Dubai for the purposes of research, private study or education and that The British University in Dubai may recover from purchasers the costs incurred in such copying and distribution, where appropriate.

I understand that The British University in Dubai may make a digital copy available in the institutional repository.

I understand that I may apply to the University to retain the right to withhold or to restrict access to my thesis for a period which shall not normally exceed four calendar years from the congregation at which the degree is conferred, the length of the period to be specified in the application, together with the precise reasons for making that application.



Signature of the student

Copyright and Information to Users

The author whose copyright is declared on the title page of the work has granted to the British University in Dubai the right to lend his/her research work to users of its library and as an open source and to make partial or single copies for educational and research use.

The author has also granted permission to the University to keep or make a digital copy for similar use and for the purpose of preservation of the work digitally.

Multiple copying of this work for scholarly purposes may be granted by either the author, the Registrar or the Dean only.

Copying for financial gain shall only be allowed with the author's express permission.

Any use of this work in whole or in part shall respect the moral rights of the author to be acknowledged and to reflect in good faith and without detriment the meaning of the content, and the original authorship.

Abstract

The adoption of agile methodologies in project management and software development has revolutionized traditional approaches by promoting flexibility, adaptability, and continuous improvement. This study explores the role of agile project management in addressing the complexities and challenges associated with conventional software development methodologies, such as the waterfall model. Agile methodologies, including Scrum, Lean Software Development, and Kanban, emphasize iterative progress, customer collaboration, and responsiveness to change, enabling teams to deliver high-quality software products within constrained timelines and budgets. This research aims to develop a framework for the successful integration of agile methodologies in software development, focusing on their impact on project management practices, performance indicators, and the evolution of project management standards like PMBOK. The findings highlight the importance of aligning agile practices with organizational goals to enhance communication, teamwork, and overall project success. Furthermore, the study underscores the need for a systematic approach to adopting agile methodologies, particularly in contexts where traditional methodologies have been prevalent. By examining the intersection of agile practices and traditional project management, this research provides valuable insights into the benefits and challenges of agile adoption, offering practical recommendations for organizations seeking to optimize their software development processes through agile project management.

Keywords: agile methodology, project management, software development

خلاصة

لقد أحدث تبني المنهجيات السريعة في إدارة المشاريع وتطوير البرمجيات ثورة في النهج التقليدية من خلال تعزيز المرونة، القدرة على التكيف، والتحسين المستمر. تستكشف هذه الدراسة دور إدارة المشاريع الخفيفة في معالجة التعقيدات والتحديات المرتبطة بالمنهجيات التقليدية لتطوير البرمجيات مثل نموذج الشلال. تركز المنهجيات السريعة، بما في ذلك Scrum و Lean Software Development و Kanban، على التقدم التكراري، والتعاون مع العملاء، والاستجابة للتغيير، مما يمكن الفرق من تقديم منتجات برمجية عالية الجودة ضمن الجداول الزمنية والميزانيات المحددة. تهدف هذه الدراسة إلى تطوير إطار عمل للتكامل الناجح للمنهجيات الخفيفة في تطوير البرمجيات، مع التركيز على تأثيرها على ممارسات إدارة المشاريع، مؤشرات الأداء، وتطور معايير إدارة المشاريع مثل PMBOK. تسلط النتائج الضوء على أهمية مواءمة الممارسات السريعة مع أهداف المنظمة لتعزيز التواصل، العمل الجماعي، ونجاح المشروع بشكل عام. بالإضافة إلى ذلك، تؤكد الدراسة على الحاجة إلى نهج منهجي لاعتماد المنهجيات السريعة، خصوصاً في السياقات التي كانت تسود فيها المنهجيات التقليدية. من خلال دراسة التداخل بين الممارسات السريعة وإدارة المشاريع التقليدية، تقدم هذه الدراسة رؤى قيمة حول فوائد وتحديات تبني المنهجيات السريعة، وتقدم توصيات عملية للمنظمات التي تسعى إلى تحسين عمليات تطوير البرمجيات لديها من خلال إدارة المشاريع السريعة.

الكلمات الرئيسية: المنهجية السريعة، إدارة المشاريع، تطوير البرمجيات

Dedication

I want to dedicate my work to my family, who have always encouraged and supported me along the way. Their tolerance, comprehension, and faith in my objectives have been a never-ending source of motivation. This research has been made possible by the invaluable advice and insights of my mentors and colleagues, to whom I also dedicate this work. Their commitment to quality and enthusiasm for the subject have greatly influenced this work. This dedication serves as a little mark of my appreciation for everything that they have contributed.

Acknowledgements

I want to dedicate my work to my family, who have always encouraged and supported me along the way. Their tolerance, comprehension, and faith in my objectives have been a never-ending source of motivation. This research has been made possible by the invaluable advice and insights of my mentors and colleagues, to whom I also dedicate this work. Their commitment to quality and enthusiasm for the subject have greatly influenced this work. This dedication serves as a little mark of my appreciation for everything that they have contributed.

Taking this opportunity, I extend my heartfelt thanks to everyone who contributed to this research. First and foremost, I am deeply grateful to my supervisor, Dr. Maria Papadaki, for her invaluable guidance, expertise, and constant support throughout this journey. Her constructive feedback and encouragement were crucial to the success of this study. I also want to extend my thanks to my colleagues and peers for their stimulating discussions and collaborative efforts, which significantly enhanced the quality of this work. Special appreciation goes to the research organizations that provided the essential data and insights. I am also deeply appreciative of the support provided by the library at The British University in Dubai, whose resources and services were instrumental in conducting this research. Finally, I wish to thank my family and friends for their unwavering support and encouragement, which provided me with the strength and motivation needed to complete this research.

Table of Contents

Chapter 1: Introduction.....	1
1.1 Background of the Study.....	2
1.2 Problem Statement	5
1.3 Aim and Objectives	7
1.3.1 Aim.....	7
1.3.2 General Objectives	7
1.3.3 Specific Objectives.....	7
1.4 Research Questions	8
1.5 Scope of the Study.....	8
1.6 Significance of the Study	9
Chapter 2: Literature Review	10
2.1 Introduction	10
2.2 Traditional Software Development	10
2.3 The Traditional Project Manager	11
2.4 Agile Software Development	15
2.5 The Project Manager in Agile Software Development	17
2.6 Relationship of Scrumban and Agile Methodology	19
2.7 Pros and Cons of Agile Methodology in Project Management and Software Development.....	21
2.8 Implementation of Agile Methodology in Software Development and Project Management	22
2.8.1 Existing Models and Tools.....	23

2.9	Related Theories.....	24
2.9.1	The Explicit Theory of Project Management (TETPM)	24
2.9.2	The Progress Monitoring Theory (PMT)	24
2.9.3	The Program Theory (TPT).....	25
2.10	Research Gaps	25
Chapter 3: Research Methodology		27
3.1	Introduction to Research Methodology	27
3.2	Research Design	28
3.2.1	Target Population	30
3.2.2	Research Philosophy	30
3.3	Empirical Model.....	32
3.3.1	Organization and Measurement of Variables	33
3.4	Data Collection Instruments	35
3.4.1	Online Survey Sample	36
3.5	Ethical Consideration	38
Chapter 4: Discussion, Findings, and Results.....		40
4.1	Introduction	40
4.1.1	Response Rate	41
4.1.2	Age of the Respondents	41
4.1.3	Agile Project Management Industry	42
4.2	Analysis Approach	43
4.3	Results	44
4.4	Methods	45
4.5	Findings	47

4.6 Discussion.....	48
4.7 Conclusions	49
4.8 Recommendations	51
Chapter 5: Research Implications, Recommendations, Limitations, And Lessons	
Learned	53
5.1 Introduction	53
5.2 Effectiveness of Agile Methodologies in Software Projects	55
5.3 Research Implications.....	58
5.4 Limitations of this Study	62
5.5 Conclusion	63
References	64

List of Tables

Table 1: Pros and Cons of Agile Methodology in Project Management and Software Development	21
Table 2: Response Rate of the Survey Directed to Different Respondents from Different Organizations	41
Table 3: Ages of the Respondents Who Were Interviewed Using Online Survey Tool	41
Table 4: Agile Project Management Industry	42

List of Abbreviations

SDLC - Software Development Life Cycle

RUP - Rational Unified Process

SLR - Systematic Literature Review

PMBOK - Project Management Body of Knowledge

XP - Extreme Programming

PM - Project Management

WBS - Work Breakdown Structure

ARN - Agile Research Network

TETPM - The Explicit Theory of Project Management

PMT - The Progress Monitoring Theory

TPT - The Program Theory

SLR - Structured Literature Review

ANN - Artificial Neural Network

Chapter 1: Introduction

Project methodology combines logically related practices, methods, and processes to determine the effective plan, control, and delivery of the project throughout the process implementation. It is considered as a method that allows controlling of the entire process management through efficient decision-making and providing the solutions to the problem to ensure successful approaches, techniques, technologies, and performance. The notion in this study entails a collection of procedures and organizing the necessary activities to finish the project. Gathering, analysis, design, coding, system testing, and delivering partially implemented software are all steps of the agile project management lifecycle that must be performed to produce the required software products. The adoption of agile project management approach is based on delivering maximum benefit in contradiction of business priorities in terms of timeline and project cost (Sharma et al, 2012). The concept of project management contains operations such as the steps of defining the requirements determination of work level, allocation of the required resources, executing the project activities, monitoring the progress, and regulating the project plans. Agile methodology promotes evolution of the projects and adaptive planning. Additionally, agile approach is well known for recurrent software development, promotes flexibility, and responsiveness to changes.

The adherence to the agile methodology that is linked with values defined in it may include project activities in software development that are either not executed or have explicit definitions. Agile approach as an option methodology for, classical project management applicable in software development assists teams in responding to impulsive iterative work inflections. Notably, project management activities involved in these methodologies are distinct from disciplined software development methodologies such as waterfall-based methodology, spiral methodology, and Rational Unified Process (RUP.) The rationale behind

the application of Agile methods is to eliminate any existing complexity and resolving the challenges relating to clarity of the project requirement and risks within the project. This paper aims to discuss the role played by agile methodology in project management and software development.

1.1 Background of the Study

In the year 1990s, software developers faced a crisis as a result of developing new software development methods. For a successful project completion, the developers considered creating an Agile manifesto and a statement of principles that concentrated on recurring threads in the various techniques (Beck et al., 2001). The agile manifesto and principles are a set of guidelines for understanding the issues that software engineers encounter. Historically, program planning has been considered a multi-layered procedure that can effectively execute a venture in various elements, such as budgeting, project pricing and cost containment, and performance assessment throughout program execution. In the actual environment, the agile manifesto has built a strategy for the rapid application development approach. The majority of the issues revolve around project planning and cost estimation. Over time, the development process in competitive businesses has seen more dynamic projects. Client transformation, scheduling, budgetary limits, communication concerns, and client participation are just a few of the difficulties these modifications present for the design of the procedure and software.

The waterfall technique can use traditional SDLC methods. Finding needs and agreeing on solutions throughout the project team is an important part of agile software development. Flexible architecture, evolving app development, early technology deployment, and ongoing advancement are all encouraged. It has also established several techniques to provide exceptional service to clients. To minimize collaboration, the main features of these

methods are centered on self-organization for group members who are assigned to their jobs. Furthermore, in this case, the core procedures may accept any modification in the specifications at any time during the program development lifecycle. It is worth noting that with agile project approaches, it becomes easier to resolve issues that organizations come across relating to delivering the product based on the needed timeline, budget, and quality of the project. Additionally, based on Agile, customers are involved through the development process to facilitate feedback and reflect on developing teams in the project.

Although software development techniques are required for an adaptive lifetime, conventional program management outlines several strategies and well-organized processes not used in software design. This happens whenever software development companies, especially in their initiatives, use Agile methodology. Agile project management might be the greatest option in this situation. With Agile project management, the method is geared towards defining the key elements applicable in traditional project management. Additionally, a traditional project provides a little scope for making changes to the products where the activities are completed in the project. PMBOK is a technique to achieve project success that may not be worth enough to control the process and the entire project progress can be predetermined. Classical project management is applicable in such types of projects. In such projects, a succession of actions such as initiation, planning, execution, monitoring and controlling, and closing might be defined (Wysocki, 2016). The usage of appropriate software development technique is critical in giving solutions depending on the problem domain. The goal of a Systematic Literature Review (SLR) is to examine all of the work that has been done on the study topic. As a result, doing SLR will be critical when accurately evaluating the results of past investigations. To identify the linked research, precisely the utmost essential methodology and frequent references will be considered in this research. However, this could

mean that research that have published in other periodicals on an irregular level are not taken seriously. Conventional definitions of program management include strategy, budgeting, and controlling costs, as well as quality standards throughout program execution (Gandomani et al., 2020). The foundation of OMI is built as a reflection of real-world program management. Notwithstanding this, there have been a variety of concerns identified and documented in this area. Design phase, cost analysis, or other PMBOK-related challenges are the most typical complaints. Moreover, it suggests that the PMBOK techniques are inadequate for project management and profitability.

Individual-based values have been adequately addressed under the new project management and directing criteria. As a result, project managers are advised to utilize the project team in the development of such initiatives. Traditional project management refers to many strategies and disciplined procedures that are not used in software development when iterative lifecycle approaches are required. This is most common when a software development team uses Agile methodologies, especially in their projects. Agile project management would be the best option in this situation. The goal of Agile project management is to define the fundamental characteristics that apply to traditional project management. With high-performance teams, such factors are visually controlled. Test-driven development, formation of a collaborative team, leadership, and collaboration. It is worth noting that with agile project approaches, it becomes easier to resolve issues that organizations come across relating to delivering the product based on the needed timeline, budget, and quality of the project. In this scenario, the concern is whether agile program management concepts can be adopted or integrated into PBMOK.

This inquiry had a largely favorable answer. In reality, project management standard procedures, particularly those linked to PMBOK principles, may be transferred to numerous

Agile-related initiatives. Based on the division of Agile manager's agile resource managers and project managers, the duo works together to ensure successful completion of their projects at hand. This study introduces Agile approach where there are some roles that can handle project management roles. With the deep focusing on rational factors has been evidently addressed as the main reason for the failure of the most projects (Gandomani et al., 2020). Agile project methodology seems to be the best choice for most of the companies; it is not only applied by software companies but also adapted by those companies that are non-software. Nevertheless, there have been little effort spent on how to apply Agile project management, especially in software projects and its specific roles and responsibilities.

1.2 Problem Statement

Traditional software development methodologies have been coupled with several problems. Ideally, most of the organizations put focus on delivering quality and providing customer satisfaction with the aim of implementing their projects. The key problem with the traditional project management approaches is associated with the complexity of implementation process. The problem lies on selecting between the traditional development methodologies and agile development methodologies. Based on both methodologies have positive and negative sides (Wysocki, 2011). Ideally, making the right decision play a significant role while beginning a fresh project (Alqudah & Razali, 2018). The key things to put into consideration while selecting the methodology has been a major challenge while choosing the software development methodology. Most of the organizations have been found themselves in challenges especially mapping their business needs-which include mapping on the impacts of the implementation of the defined requirements on customer business. Customer's perception and perspective of business impact the implementation of project and this delays the delivery of such projects.

Application of agile methodology require the problem to be well identified in advance. The solution to be offered by agile also needs to be well determined in advance. Though with agile methodology there is flexibility, the problem lies on the integration of agile methodology in the development of software project. Agile projects come with several challenges that are distinct from those experienced by projects following traditional methodology. Based on the research done, there are various issues affecting the implementation of agile projects in the software development process. Such problems include issues with communicating, management of day today operational problems, customers and team members, changing culture and mindset, among other challenges. Notably, selecting the right methodology and matching the methodology and project needs has been considered to be one of the key challenges as far as project management is concerned (Moniruzzaman & Hossain, 2013). Agile methodology is considered ideal when the product details cannot be defined or agreed in advance with any degree of accuracy. This issue requires a collaborative environment existing between the user or clients and the developer. One of the main challenges associated with agile methodology is based on the availability and popularity of the methodology is critical, customer availability, and organizational culture. In line with Agile manifesto affirms that project managers must deliver working software frequently.

In consideration of the aforementioned issues with software design, the recommended approach is centered on incorporating agile methodologies into the software advancement process. With Agile projects, the idea is to implement software projects incrementally and through iterative approach to development. Integration of agile in the software projects comprises of various stages which include project inception, planning, requirements refinement, product backlog, testing, and transition. The solution realized in the implementation process of agile approach focus on time, cost, quality, and features, which are

variable in nature unlike in the traditional methodology that are used in software development (Papadopoulos, 2015). Lastly, the solution derived from the integration of agile methodology in the software development projects is based on identifying factors which makes the transition from traditional to agile methodology necessary. Additionally, this solution can be realized whether the software project is small or big. Therefore, with the integration of agile methodology in the software development project; this is ideal in meeting the rapidly changing business needs of an organization.

1.3 Aim and Objectives

1.3.1 *Aim*

The goal of this investigation is to find out more about to how agile methodology is implemented in software development and project management context. The outcome of this investigation will be used to develop a framework, guidance and recommendations for improving the software projects development through the use of agile project management methodology.

1.3.2 *General Objectives*

- To undertake comprehensive research on the role played by agile methodology in project management.

1.3.3 *Specific Objectives*

- To understand how agile methodology is implemented in program management and software engineering to investigate the use of Agile methods in project management and software development
- To understand how agile methodology relates to Scrum, Lean Software Development, and Kanban.

- To analyze the performance indicators of the information systems using Agile project management methodology.
- To develop a framework for the successful integration of agile methodology in software development projects

1.4 Research Questions

In terms of project management, what function does conventional software development play?

What is the role of the Agile approach in software development and project management?

How does Agile methodology relate to Scrum, Lean Software Development, and Kanban?

What are the performance indicators of the information systems using Agile methods?

What is the framework that can improve the integration of Agile methodology into software Development?

1.5 Scope of the Study

The project scope aims to plan and provide documents that are required to complete the project goals and expected deliverables. It aids in defining a project's limits, a team's tasks and responsibilities, and the procedure through which the project will be verified. Agile software development covers technological parameters and workflow of long-term products growth strategy for the client. The involvement of the customer in the development process assists the project team to produce quality products. The output of agile software development is based on small incremental that changes the requirements of the project system. To figure out the mechanics of delivery, Agile put emphasis on collaboration between the client and project management team rather than negotiations between the two. Collaborating with consumers involves putting them on board throughout the development process, not just at the inception and completion of the project, making it much easier for teams to meet their

customer's needs. In Agile software development, the customer may be invited to be involved in the product demos at various intervals. Additionally, the Agile cycle's short iterations make it easier to make the necessary changes, facilitating the team to tailor the process to their needs rather than the other way around. Overall, Agile software development considers change to be a constant opportunity to improve the project and generate value.

1.6 Significance of the Study

It will be critical to offer a full knowledge of the implementation of agile project technique in program management and software engineering depending on the findings of this research. This study will also provide hands-on experience in using Agile methodology to manage medium-sized and large projects in organizations, as well as improving communication and teamwork throughout project execution. The Agile method will also assist teams in more effectively and efficiently managing work while producing a high-quality product within the constraint's constraint of the budget. Time and cost are the most factors that are determined in the project. Agile project management solutions such like "SCRUM and XP", amongst many others, adhere to Agile rules centered on constant progress, review board, and the supply of high-quality latest results. Testing is integrated into the Agile process across the lifecycle with standard tests to determine the solution that is functioning all through progress for constant merging. Clients can also be engaged all through the program using Agile development, from selecting features via iterative assessment and evaluation meetings to recurring program builds incorporating fractionally modified. Clients, on the other side, should accept that they are witnessing a project in development in return for the advantage of increased transparency. As a result, the goal of this study is to define the key elements of agile methodology deployment and usage, as well as to investigate the reasoning and timeframes for adopting agile techniques to ensure project accomplishment.

Chapter 2: Literature Review

2.1 Introduction

The primary objective of this category is to introduce a literature review of various components that help in providing a summary of the evolution of traditional software development strategies, the position of the project supervisor, the initiation of agile methodologies in the project managerial and applications advancement frame of reference, and the emergence of Agile strategies in the software industry. This review is geared towards assessing the literature revolving around Agile in project management and other approaches connected to it.

2.2 Traditional Software Development

The traditional methodology under review in this case is Waterfall, which is specification-driven approach that is pigeonholed by extensive planning, detailed requirements, upfront requirement gathering, among others. Ideally, Waterfall methodology integrates different aspects of traditional project management approach, which include linear arrangements of different software different steps which put emphasis on far-reaching phase-based approach. The main criticism which has been identified with this methodology is associated with poor flexibility to changes within the project environment (Warburton & Cioffi, 2014). The fact that linear Waterfall methodology does not fit well into vacillating software development setting, this amount to spiral alternate model as proposed by (Shastri, 2020). Ideally, this model focused on solving the issues relating to risks involving the customer within the development process. In software development operations, the application of waterfall development remains to be popular method of working. The model has been closely associated with a few issues. Some of the generally recognized are such as inclusion of

the inability of dealing with the with change and the fact that faults are commonly experienced too late in the project implementation process.

Despite the flaws highlighted, the paradigm is still widely used in the software industry, and some researchers believe it will continue to exist for a long time. According to studies, the following patterns have emerged. First, the model appears to be of little interest to scholars because it appears to be outdated. Recent research has placed a greater emphasis on agile and incremental development. Second, there isn't much actual evidence to back up what we think we know about the waterfall paradigm. Furthermore, there is a difficulty with the entire system's integration and testing. According to a review of 400 waterfall projects, the software being produced is either not deployed or is not used. Changes in needs and a lack of opportunities to clarify misunderstandings are the reasons behind this.

The absence of opportunities for customers to provide input on the system is the source of this. The waterfall paradigm, in particular, fails in large-scale, complicated projects and experimental initiatives. Waterfall development, on the other hand, has its own set of benefits. The waterfall approach is predictable, and it emphasizes meticulous preparation of the software system's design and structure, which is especially crucial when dealing with huge systems (Vasylykov. 2019) There is a risk that design decisions will be dependent on inferred knowledge rather than being formally recorded and assessed if architecture planning is not prioritized. As a result, there's a good chance you'll miss architectural flaws.

2.3 The Traditional Project Manager

Gaddis created the term "project management" in the 1950s as a method of managing experts. However, in the previous history, there has occurred a breakthrough with the development of PMBOK, that defines program managing ideas. According to the PMBOK, a program supervisor is a professional who is appointed by the performance company to oversee

the team accountable for achieving the project's goals. A program supervisor, unsurprisingly, is viewed as a connection amongst clients and the task organization. There's been a surge in the requirement for program managers from various industries in recent years. This is evident in the growing participation of professional associations such as PMI, as well as an increase in project management credential holders.

Notably, classical project management methodologies are successful in various sectors, with the exemption of new software product development. The whole project life cycle is divided into a number of phases in traditional project management, and the job is completed phase by phase. Classical project management techniques and tools like WBS and Gantt Charts are broadly used. Most goods, such as traditional infrastructure construction, allow the team to specify and establish all of the end product's requirements upfront. Due to this, the entire project can be planned. They do not anticipate several modifications in such projects. The entire project's progress may be predicted. Furthermore, a set of actions, such as commencement, planning, execution, monitoring and controlling, and closing, can be defined for such projects. All of the activities occur in a particular order. Traditional project management employs a classical development method, where one phase must be finished before the next can start.

According to the traditional method, project budget and schedule do is not fixed and may change from time to time, while the requirements are fixed based on this approach. With this approach, the inputs, processes, and outputs are interdependent; for instance, the output of one stage is used as an input of the other stage. In this methodology, the concepts are based on predictable tools and the predictable environment. For the case of Agile Project Management, it is the general mechanism applied for software development; it vividly depends on the teamwork synergy and the flexibility to adapt to any new requirements. With

Agile methodology, projects are broken down into small portions, which are finished in work-related phases that operate from the design phase, testing to quality-assurance phase (Špundak 2014). The agile development methodology is also referred to as Scrum. Application of Agile method is mostly where rapid feedback is required. The agile method mostly involves four phases, which include Scanning the emerging trends as well as the risks. Analyzing the information led decisions and solutions. The third is responding to the opportunities and risks. The fourth phase is change whereby it involves shaping the future environments.

Traditional vs Agile-Project-Management

Undoubtedly, there is a critical distinction that exist between the two methods. Such vital differences can be viewed in terms of their characteristics, such as: in terms of the organization structure, traditional methodology follows and put emphasis on the linear processes while for the case of the agile methodology, the focus is on iterative processes (Wysocki, 2011). In this case, the traditional approach is a sequential process whereby one process to be executed; its subsequent process must have completed executing. Additionally, the approach makes use of a top-down approach. For agile it does not follow the linear processes but rather ensure the flexibilities in terms of the processes.

The other key difference can be viewed in terms of project complexity; the traditional method applies to small projects with fewer requirements. With this approach, any introduction of the requirement makes it complicated for the project; hence such additional may force the project team to start the whole project from scratch. On the other hand, the agile method is the recommended method to follow in case of the complex projects since they can respond to such structures in the best way possible.

In terms of the adaptability, the traditional method operates in the sense that, once a phase is done, it cannot be reviewed in the future. In a situation of any change, the traditional

methodology does not adapt to such changes, the only option available is to start the whole process from the beginning. For the agile approach, the adaptability factor is high since it does not follow the linear mechanism. The projects that are complex comprises of various interrelated levels where a change in one level can cause an on the other level. In such situations, the project leaders can take calculated risks as there is a high probability of adaptability.

Based on the scope for feedback and change, traditional method processes are clearly defined at the begin of the project. Additionally, the delivery time for the projects using this methodology is fixed, and changes are allowed very rarely (Špundak 2014). Concerning the agile method, there is high acceptance based on this methodology. The process act in a flexible manner to allow regular feedback that helps provide better results within the specified project delivery time.

Ownership and transparency: the traditional method projects are under the ownership of the project managers; this is because they must plan and document all the processes for the product. On the other hand, the ownership is to the team leader who can share it. In most cases, all the team members put their ideas together for the success of the project. Considering the problem-solving aspect, in case of any problem during the development, the individuals are supposed to escalate the issues to the project managers as the approach make use of top-down approach (Wysocki, 2011). However, approaching the manager every single moment is not a viable option.

Additionally, there may be delays in case the issues escalated to the project managers are many. This may come with the additional cost. Agile methods can take decisions on their own. The team tries to solve the issues at hand to avoid any delay of the project delivery, because of this issue, team leaders rarely address their concerns to their managers.

Conclusively, when it comes to checking points and monitoring the progress, traditional approach, solely put the focus on the streaming of the processes, rather than the product itself. The traditional method emphasized the planning at the analysis and design stage of the product. Once the process is wrapped up, the team is supposed to follow stage by stage, with little guidance. In this case, the progress of the project is ascertained after the completion of the project. Moreover, there is no requirement for frequent checks, unless the manager receives any escalation. In agile methodology, the team is expected to have the checkpoints after the stipulated period. Based on this requirement, it is not hard to progress of the project as well as it is paramount in helping to maintain the accountability in their job operations. The agile approach operates within a dynamic environment; in such an environment, there are very few circumstances where there is no occurrence of the change.

2.4 Agile Software Development

The application of agile methodology has significantly been adopted in software development as well as in project management. Agile software methodology focuses on the development, where the demands and solutions evolve based on the collaborative effort of the project management team and its customers. The method is considered to be one of the modern approaches applicable in software development, which advocates on adaptive planning, Teamwork, flexible scheduling, scalability, early release, ongoing refinement, frequent client input, and periodic rethinking all contribute to the creation of technology additions that are executed in phases to answer to ever-changing client needs. Scrum, radical computing, evolving nature, and future-driven creation are examples of progressive and continuous software development models covered under agile software advancement method. Agile technology and program methodology should, in theory, have a transparent reaction to individuals and a quick adaptability to change. Inside the software business internationally, the

adoption of agile methodologies in program management has been particularly rapid in recent generations. Scrum was selected as one of the widely utilized approaches depending on the repeating frequency of agile research. In one of the most current rounds of the study, 58 percent of respondents said their projects used Scrum framework, that they viewed as a stand-alone approach. Furthermore, according to this poll, 7% of participants mentioned adopting Kanban and XP. 2016 (Siddique & Hussein). While Extreme programming emphasizes on programming approaches such as test-driven production, whereas Scrum concentrates on project management components of agile such as project evaluation and scheduling evolutionary development, continued development as well as encouraging flexibility to change.

In the recent past, there has been blossoming research associated with large-scale agile implementation. This is considered as part of reflection of the rising need in the software industries on the clear guidelines relating to implementation of agile methodology especially in large projects with multiple teams. Loiro et al. (2019) in their systematic review of literature have stated that large scale agile comprises of software development firms with fifty or more individuals or at least six teams. A review of agile development's history published in 2004 revealed some of its antecedents in other disciplines. It explored the relationship between agile development and the Capability Maturity Model in detail. Agile approaches, like object-oriented methods, are expected to solidify over time, according to the authors. They also expected that agile and traditional approaches will have a symbiotic connection, with criteria such as the number of people working on a project and the application domain, as well as criticality and innovativeness, determining which process to choose.

Additionally, in line with ever-changing software complexity and dynamic customer requirements, the software industry has switched from classical software development

paradigms to agile-based development. Agile approaches, in contrast to traditional models, are depicted by shorter development cycles, increased customer involvement, incremental delivery, and frequency redesigning with adjustments necessary by dynamic user requirements. Although different software development approaches follow similar set of Agile principles outlined in the Agile Manifesto (Becky et al., 2001), they differ in several ways. Begel and Nagappan (2007) conducted an empirical study at Microsoft that found that roughly 33% of respondents agreed to use Agile software development approaches, with Scrum being the most popular agile methodology at Microsoft. Heavyweight techniques, often known as conventional software development methods, are defined by thorough planning, process orientation, a predictive approach, and extensive documentation. Lightweight approaches, in contrast to standard software techniques, guarantees frequency in the delivery of software increments in small region cycles, as well as a team-oriented and adaptive approach. In recent years, lightweight techniques, often known as agile methodologies, have made significant inroads into the software industry.

2.5 The Project Manager in Agile Software Development

In Agile Software Development approaches such as Scrum, Kanban, XP, among others the role of project manager in the project management is crucial. For the case of Scrum, it has introduced two additional roles which are product owner and Scrum master (Papadopoulos, 2015). Notably, product owner is the client representative whereas the Scrum master is predominantly the internal facilitator. eXtreme programming has introduced new roles which include tester, consultant, tracker etc. Industry surveys and recent research affirm on the continued application and role of the project management in agile projects. Additionally, putting recognition on the co-existence of the project managers with the organized agile project teams. As a project manager, the key idea is on the collaboration, stakeholder

administration, and coaching of project teams. Considerable research undertaken has been geared towards checking diverse areas of agile teams. Siddique and Hussein (2016) in their recent research have addressed the issue of conflict in the project teams from project management perspective ascribing it to lack of enough experience and customer involvement within the project teams. Ideally, the consequences of such conflicts can lead to dropping in in the productivity as well as motivation.

Agile teams aren't meant to have project managers on them. Scrum and XP, on the other hand, establish positions such as product owner, scrum master, and coach. Research has revealed the presence of a project manager in agile projects, demonstrating a disconnect between theory and practice. The project manager is crucial in conventional software development projects that need multiple aspects of team management such as leadership, team building, motivation, communication, influencing, decision making, planning, and coaching. The late 1990s saw the introduction of Agile Software Development (ASD), which brought self-organized teams to software engineering. Self-organizing teams are described as having a high degree of autonomy in making choices, managing workloads, and allocating work amongst themselves. However, evidence is accumulating that the project manager job title remains in ASD businesses, demonstrating a gap between agile philosophy and reality. Furthermore, research on self-organized teams has revealed that new roles such as scrum master and product owner take on some of the previous management responsibilities. There hasn't been comprehensive research that examines the project manager's function in agile projects.

Furthermore, the findings of the pre-interview survey imply that the project manager's presence on an agile project may influence the team's frequency of implementing agile methodologies. When a project manager was present, the agile activities of a scrum of scrums

meetings and project velocity measurement increased the most frequently. In agile projects, the project manager's responsibilities include everyday activities such as facilitation and mentorship, hard, moderate, and soft management styles, and certain classic project management responsibilities such as tracking project progress and budgeting (Dingsyr et al., 2012).

Understanding the project manager's role in agile projects can assist practitioners better handle this role's expectations and make their agile transitions easier. Notably, there are various agile approaches and styles to consider. Scrum, extreme programming, Kanban, lean software development, feature driven development, agile unified process, dynamic systems development method (DSDM), and others are prominent and commonly studied in literature. Project management, project life cycle, team management, engineering, and delivery are some of the disciplines that agile approaches attempt to define. Not all methods are appropriate for all subjects. DSDM, for example, covers all disciplines, whereas Scrum just covers team management and the project life cycle. The importance of team management is emphasized in all agile techniques.

2.6 Relationship of Scrumban and Agile Methodology

Agile software and project development is abroad approach, that consists of various mechanisms where the requirements keep on changing. The evolution of agile approach is based on a collaborative approach of cross-functional project team members and end users. The methodology promotes evolutionary and adaptive planning. Moreover, agile software and project development methodology is beneficial for its unremitting software development and promotes ability to make changes and responsiveness to changes. Since agile project and software development methodology is broad in nature, this paper primarily deliberates on scrumban methodology, which is a critical approach, in helping to meet the needs of the

stakeholders that aims at minimizing the batching of work (Alqudah & Razali, 2018). As far as Agile project approach is concerned, scrumban approach has been identified to have various advantages, whereas other researchers have criticized this approach. This research looks at both the pros and cons of scrumban, as accessed from various journals. Various studies have been conducted on multiple aspects of agile project methodology and are now part of the current literature. According to Oivo and Liukkunen (2016), scrumban is a SCRUM upgrade whose main goal is to decrease time waste while also providing value to the product, primarily designed for clients. According to Elvadas (2015), the project management team searches for methods to improve the company's procedures, notably in software development activities. As a result, project management and software development have evolved over the previous few decades, shifting from traditional methodologies to agile software development.

The adoption of Scrumban has been acknowledged as a strategy that can acquire numerous advantages with the progressive refining of agile approaches to produce outstanding solutions. Scrumban, according to further research, is a hybrid of Scrum and Kanban that can provide a flexible and adaptable strategy. According to Oivo and Liukkunen (2016), the system improves frequent feedback throughout every iteration process. The Scrumban approach's appeal in today's world is due to its capacity to enhance effectiveness, particularly in project management. The rationale for the increase in efficiency is that the strategy is designed to decrease resource waste. According to Elvadas (2015), the benefits of this system come from the combination of Scrum and Kanban. Scrumban also reduces overhead difficulties for project management and the entire software development team, resulting in increased effectiveness and essential client satisfaction.

2.7 Pros and Cons of Agile Methodology in Project Management and Software

Development

Table 1

Pros and Cons of Agile Methodology in Project Management and Software Development

Pros of Agile Methodology	Cons of Agile Methodology
Agile methodology offers flexibility and adaptability. Automation of business processes may require changes to be made; hence this method is ideal for such as automation.	Integration with project management- The methodology may not be completely appropriate for the projects requiring a more plan-driven approach in achieving a specific level of predictability (Cobb, 2019).
The methodology facilitates creativity and innovation. In automating business processes, it is worth noting that in the competitive world that we are living in today, a high level of excellence is needed to develop the right products.	Organization transformation- Agile methodology require specific level of organizational transformation for the whole approach to become successful.
Improved quality is another merit. In agile methodology, quality is an integral part of the software development process, as compared to a linearal process.	
The method also offers a higher level of customer satisfaction and solutions due to their involvement and providing feedback in the development process.	

2.8 Implementation of Agile Methodology in Software Development and Project

Management

Implementation of agile methodology in software development is based on delivering the requirements in iterative and incremental manner throughout the lifecycle. Ideally, iterative approaches are mainly used in software development projects with the aim of promoting velocity and adaptability because the benefit of iteration is based on the premise that it is easier to make adjustments as the project is carried along. Implementation of Agile depends on the specific agile approach in place. For example, with Scrum, the focus is based on delivering the business value in the shortest time possible. Additionally, the goal of Scrum is to develop and test software applications rapidly (Permana, 2015). Implementation of Agile development methods in software projects have its roots in design and development.

Most of the projects fail to succeed because of not having project maturity, depicting the requirements for a well-defined and the required Project Management methodology for the organization. One of the key patterns in the streamlined software development process is the application of agile project management (PM) approaches. Since the publication of an agile manifesto, various applications of agile Project Management Methodologies have been tried, and various research have been undertaken on successful or failure implementation.

Individuals' factors, trainings, clients, team, organizations, culture, planning, and scheduling have all been linked to the effectiveness of the agile Project Management technique adoption. Identification of relevant methodology, identification of enterprise specific requirements, modification, and implementation of methodology are all elements in the methodology implementation process (Rasnacis & Berzisa, 2017). The application of the agile Project Management technique has been extensively studied from several perspectives. Identifying roles, behaviors, artifacts, and processes that must be suitable for the current circumstance has

been done as part of the adaptation process. Different aspects connected to the team, the intrinsic and extrinsic environment, the objectives, maturity levels, and past knowledges are used to identify the situation. The agile Project Methodology (Vaishnavi et al., 2004) has been suggested to be prepared to modify the procedures to each project team since aspects of the project team such as internal connections and rationale have not been evaluated in existing adaptation methodologies.

2.8.1 *Existing Models and Tools*

Software project management involves, knowledge, tools, and techniques which are critical for managing software development projects. Notably, there are several theories and models that provides the foundation of the application of agile methodology. Alaidaros et al. (2018) present an overview of existing models and tools applicable for managing and developing software projects. Recent studies on the same have revealed that there are several approaches established in managing and monitoring software development projects (SDP). Some of the existing tools that various research has focused on include for managing the software development projects include but not limited to Primavera, MS Project, Gantt Chart, Jira, among others. Further studies on existing tools have revealed advantages and disadvantages associated with some of these tools. Initially, agile approaches were established as a response to the observed problems of classical project management for software projects, such as a lack of contact with customers or the inability to account for adjustments to a project during its life cycle. Some scholars take a different approach to explaining agile approaches than others, instead of using existing theoretical or conceptual lenses. Adolph, Kruchten, and Hall (2012) created a grounded theory on how the application of agile software development approach. While the authors' grounded theory is insightful, they do not connect or tie it to other theories, leaving their findings unconnected to the existing

bodies of knowledge. Additionally, because agile project management is a subset of overall project management, it's vital to identify generic elements like top management buy-in (2008) and arising general project management theories from factors, concepts, and theories specific to agile project management.

2.9 Related Theories

Theories are critical in shaping a discipline and guiding the researchers on investigating the phenomenon. Ideally, theory and practice are mostly developed concurrently. Various theories relating to project management and agile methodology have provided different perspectives especially on the implementation of the software projects. Such theories have been discussed as follows:

2.9.1 *The Explicit Theory of Project Management (TETPM)*

This is the first theory of project management which comprises of various goals. TETPM is critical in providing prediction for the future behavior. Moreover, tools for performing analysis, designing, and monitoring can be established based on TETPM. Warburton and Cioffi (2014) have proposed an approach that can be applied in future developments.

2.9.2 *The Progress Monitoring Theory (PMT)*

The simultaneous progress monitoring is paramount in the development of software projects. Accordingly, MacGregor et al. (2001) assert that this theory has features relating to maximization heuristic and progress monitoring, where the first feature affirms that each decision provides attempts to maintain as much headways as possible towards achievement of the project goals that have been set. The second feature persistently do assessment on the level of progress of the project and whether it should be delayed or not.

2.9.3 *The Program Theory (TPT)*

This theory provides a systematic approach for collecting, analyzing, and using provided information to provide answers to the questions on the projects, policies, and programs, especially on their effectiveness and efficiency in the projects. The main focus of TPT is based on developing a logical model which is a crucial way for visualizing important aspects of monitoring and evaluation. This is critical in articulating the problem, resources, and process applicable in managing the software projects (Alaidaros et al., 2018). It is worth noting that this theory put more focus on the policies and programs in ensuring that the software projects have been implemented in a more efficient manner with optimal utilization of the project resources.

2.10 Research Gaps

The Agile Research Network was formed as a collaboration between The Open University (OU) and the University of Central Lancashire (UCLan) in the United Kingdom in response to the concerns mentioned above. The DSDM Consortium funds ARN, which is currently supported by the institutions. ARN aims to collaborate closely with agile practitioners in the workplace to investigate difficulties identified by them, such as determining the extent and nature of agile techniques' efficacy or understanding the implications and affects agile methods have on organizations and individuals (Gregory et al., 2014). ARN researchers differ from consultants in that they have time of investigating, observing, and comprehending the problem situation; they collaborate with collaborators but retain full control over any changes they wish to make; they structure their intents using a model and documenting their process throughout; and they apply a model in structuring their intervention and documenting their processes throughout.

They are not accountable to management, unlike a consultant. The ARN has created and

implemented a method to solve each of the above-mentioned challenges. Collaboration kick-off, investigation, implementation, and evaluation are the four main phases of the strategy. It is timed, and ARN researchers collaborate closely with the organization during each step.

Notably, collaborators are found through a call for challenges sent to the DSDM mailing list, and the challenges are debated and evaluated by the ARN team before the kick-off meeting. This means that rather than the ARN team approaching organizations, organizations approach the ARN team. Researchers spend time scoping and outlining the focus region in depth at the organization throughout investigation. To explore the problem and determine how improvements could be evaluated at the assessment stage, various research methodologies could be used (Sharp et al., 2014). Then, in the problem area, relevant existing research and literature is surveyed and summarized. Stakeholders are supplied with a customized literature evaluation and explicit recommendations. The organization determines whether or not to implement any of the proposals offered during the Implementation stage. If they do, the research group assists in the adoption of new working methods. Alternatively, if existing research is unable to make solutions or the proposals are inappropriate for the situation, a research agenda to further examine the challenge area might be formed. The research team assesses the improvements made at the organization during the Evaluation phase. Quantitative and qualitative measurements might be used in the evaluation.

Chapter 3: Research Methodology

3.1 Introduction to Research Methodology

The science of research methodology is how research is conducted and the development of a systematic approach to solving research challenges. Its goal is to provide a study strategy for characterizing, analyzing, and forecasting phenomena. The nature of the research questions and subject being researched, according to Denzin & Lincoln (2005), determines the research technique. It gives information based on the research approach employed to complete the project. The researcher in this study outlines the project's research plans, research methodologies, research paradigm, data collecting methods, sample selection, research procedure, kind of data analysis, ethical issues, and research limits (Fitsilis, 2008). The research methodology, in particular, describes the several research procedures which must be incorporated in the project research, such as team member selection, data gathering, and data processing. Both qualitative and quantitative methods can be used to analyze data. In this scenario, the qualitative technique is more flexible, relying on the researcher's decision based on open-ended interviews inquiries, literature review, and research papers. Quantitative analysis, on the other hand, uses numbers and statistics to comprehend frequencies, averages, and correlations because data is collected and processed in a statistically sound manner that allows the results to be easily standardized and disseminated across academics.

In Agile methodology, the researcher illustrates the research design that was selected for this research and the rationale for the selection. The instrument that was applied in collecting data has also been illustrated and the processes that were followed in carrying out this study have also been included (Hu et al., 2009). The methods used in examining the data have also discussed by the researcher. It comprises information relating to the respondents,

such as the study's eligibility requirements, who the respondents were, and how they were sampled.

The methodology that was used in doing the research was Structured Literature Review (SLR). SLR involves ways of synthesizing scientific evidence to answer a particular research question in a way that it can be transparent and reproducible. It has become a major methodology in development of software projects to advocate the design research supposed to adopt the method. The rationale for SLR in design research is explored to analyze the current practice in the project. The challenges in taking forward SLR method in design research are highlighted and directed to developing of software's as proposed. It is important to note that, this method holds potential for design research that helps in addressing some important issues, defining review methods, and adapting guidance to the needs of the customers and specificities. The structured literature review follows defined protocol where the criteria has been clearly stipulated before reviewing has been undertaken. It is a comprehensive, transparent search conducted over multiple databases and grey literature that can be replicated and reproduced by other researchers. It involves planning a well thought out search strategy which has a specific focus and answers that define question. Additionally, Structured Literature Review (SLR) can be considered as a means by which any central literature might be considered when scoping out a study.

3.2 Research Design

Kratochwill (2015), argues that a research design is the meticulous arrangement and analysis of data in order to match the study object. The study will embrace a cross-sectional descriptive design to realize the objectives. This research will use a descriptive approach. A descriptive design, according to Colorafi & Evans (2016), is focused with determining what,

when, where, and how of a phenomenon. This design focuses on gathering data that may be used to analyze occurrences and provide meaningful findings and suggestions. Research design is defined as the general approach that a developer selects to integrate the various elements of the study coherently and logically in ensuring that the researcher can successfully address the research questions, it can constitute a blueprint for the collection, measurement, and analysis of data. The design allows the researcher to dwell on research methods that are suitable for the subject matter and to set up the study for the success. Research design on this case is based on data collection, measurement, and analysis. When research is impactful it creates minimum bias on data as it increases accuracy in data collection. The researcher has to clearly understand various types of research design to select to implement on the study. In qualitative research, researcher determines the relationship between the data collected and observations based on mathematical calculations the research on this design rely on theory exist along with what the respondent have to say about it. This study the researcher will use experimental research design to establish the relationship among the cause and effect of the situation. It is important to note that, experimental research design observes the impact caused by the independent variable on the dependent variable.it is highly practical research method that contributes to solving problem at hand. Researcher can have respondents change their actions and study on how people around them react to gain a better understanding of social psychology. Agile as a design framework that is used for software development has now being used in other areas where there is need to complete large tasks such as project management tasks. It is an alternative to typical project management framework that contains linear process in a project. For the last decade, agile methodology has become so popular for delivering positive results and rapid changes in software development. The effects of agile software development according to Beux et al. (2018) have immediate feedback through the client

engagement, adoption, and effective changes. Notably, the success of software development involves all developers from the early stage of the design and implementation that plays a critical role in this process. In this research, the use of agile methodology is addressed to support the design and implementation of more effective software development of the projects. Additionally, agile development has been proved to be adequate to dominate the present situation to capitalize the project changes. On other hand, agile approach is presented as an asset when the project goal is clear, but the solution is unclear. Notably, the design and implementation of agile methodology. The data gathered from the experiment is quantitative and is collected using the Google Chromes developers' tool. As mentioned earlier, Chrome's developer's tool allows for run-time performance analysis of an application. The data collected on this experiment is based on the data that the chromes developers' tool provides. The results are reviewed by averaging the iterations' outcomes, and the data from all the iterations for each task is supplied in a table. The survey gathered both quantitative and qualitative information.

3.2.1 *Target Population*

The target population is defined as the collection of the subject that will be used in data analysis (Nguyen, Ebnesajjad, Cole & Stuart, 2017). The full collection of groups of understudies with the same features or observations is sometimes referred to as the study's target population. The study will target 85 respondents from different organizations dealing Agile methodology in their software.

3.2.2 *Research Philosophy*

Research philosophy refers to a system of beliefs and assumptions about the development of knowledge and phenomenon on data how it is used gathered, analyzed, and used. Every stage of the research process should, in theory, be founded on assumptions about

the sources and nature of the knowledge. The research philosophy reflects the researcher's important assumptions based on a research strategy that contains several branches related to a wide range of disciplines. The selection of a particular viewpoint is impacted through the practical applications of the research. The concept of existing different paradigms on research plays a paramount role to view how the research is conducted within the project. The research will adopt positivism philosophy, where deductions will be elucidated from the data that will be collected from the respondents. Based on positivism philosophy the research is assumed to be independent from the investigation, where there are no changes of biasness during the study. In this philosophy, it is the belief of the researcher to remain neutral. This research will be justified by the results that will be obtained in this study. Moreover, this philosophy the researcher needs to concentrate on facts, whereas phenomenology concentrates on the meaning and provision for human interest. Accessing the work of Saunders et al. (2009), this will help the researcher to make values and assumption more explicit and explaining them in using language of research philosophy between the researcher and beliefs. The philosophy it gives the researcher a role that limits data collected and interpretation through objective approach and the research findings based on observation and quantify. Basically, positivism paradigm is based on pure facts that are considered as external and objective to the project. This philosophy summarizes five principles that a researcher has to follow while undergoing the investigation as follows:

1. There are no differences in the logic of inquiry across sciences.
2. The research should aim to explain and predict.
3. Research should be empirically observable via human senses. Inductive reasoning should be used to develop statements (hypotheses) to be tested during the research process.

4. Science is not the same as the common sense. The common sense should not be allowed to bias the research findings.
5. Science must be value-free, and it should be judged only by logic.

In general, positivism philosophy is based on quantifiable observations that lead to statistical analyses. It is important to note that this philosophy has an atomistic, ontological, and empiricist view of the world as consisting of discrete, observable elements and events that interact in a predictable, predictable, and regular manner. Furthermore, in positivist studies, the researcher is an autonomous type of study, and there are no provisions in the study for human interests. It also has to do with the fact that researchers must focus on facts, but phenomenology focuses on meaning and allows for human interest.

3.3 Empirical Model

Empirical models focus on describing the data with the specifications of very few assumptions on analyzed data. It offers simplistic solutions for quantitative comparisons between different operation conditions and are supported by experimental data. The data obtained from the experiments are correlated to obtain some derived quantities. This it models uses Artificial Neural Network (ANN) and designs of experiments. Empirical model plays a paramount role of providing modeled experiment under operation condition. It becomes difficult for this model to predict particular designed experiment. However, in specified system, empirical model can be used to calibrate and predict the experimental data and abilities to achieve process optimization. Empirical modeling (EM) has proven to be a useful tool for analyzing a variety of problems in a variety of fields. As is well known, this sort of modeling is particularly useful when parametric models cannot be built for a variety of reasons. EM helps the developer to gain an early understanding of the relationships that exist

among the different variables that belong to a specific system or process using various procedures and approaches (e.g., Least Squares Method, LSM). An empirical model, as is well known, can assist researchers in gaining an early understanding of the relationship between two or more variables that describe a certain system or process. Despite their inherent limitations, empirical model results can occasionally assist researchers when making decisions about factors that intervene in the system/process under investigation. However, when there is insufficient data to suggest a parametric model, it is obvious that an exploratory analysis based on empirical knowledge to obtain an initial model and solution can be warranted. Following this phase, the developer can determine whether the first model is a sufficient representation of the relationships that exist among the many variables that are part of the problem under investigation based on his or her experience. Generally, nonparametric data analysis is used by EM to investigate trends or behaviors in the provided data. Due to inadequate knowledge, it is expected that models based on well-defined parameters and distribution functions cannot be developed. Additionally, empirical model is mostly used to represent real-life problems that needs non-analytical methods.

3.3.1 Organization and Measurement of Variables

In this research on Integration of Agile Methodology in Software Projects, the measurement of variables included both dependent variables and independent variables, as deduced from the research questions. The measurement of the variable is defined as unknown attribute that measures a particular entity and can take one or more values. It is used in scientific research purposes for quantitative and qualitative values in statistics. In this research, integration of Agile methodology is considered to be independent variable, while development of software projects on the other hand is dependent variable. Variable is a term mostly used in research projects that identifies and designates variables while designing quantitative research

projects. In this study, independent and dependent variables will be used to measure factors through process of operation. There are two types of variables that are measured in a project. In investigations, independent parameters are thought to be the cause, while dependent factors are considered the consequence. Contextual factors and topic parameters are two types of independent factors. Contextual factors are those that the experimenter can directly regulate and manipulate. The factors that are naturally present in participants are referred to as subject characteristics. The degree to which variables are measured, to begin with, aids the researcher in determining how to interpret the data from that variable. You may use numerical values to represent the lengthier names with shortcodes when you know a nominal measure. Second, understanding the degree of measurement aids in determining what statistical analysis is appropriate for the provided results. If a measure is nominal, then you know that you would never average the data values or do a test on the data. The organization of variables supports the data file structure. Variable names and labels aid to structure the data file by allowing part of the documentation to be integrated into the data file and assisting researchers in understanding the structure of the data sets. Variable names, on the other hand, should be short and adhere to conventional software criteria, as they are utilized as calling codes in software operations. Additionally, the researcher set the variable width for each variable in the data file, which is the number of characters or the length of the integer and fractional parts of a number. Even if they are left blank, the set amount of characters or digits for each variable is reserved for every scenario. Supplementary variables are included in data files to help with orientation and management, to verify data integrity, and to run various analyses. As a general rule, each case (individual responder) should have its own unique identification (or collection of identifiers) in the file. Data resulting from overrepresentation sampling tactics, distinct waves

of study, and so on must be separated in an analysis, especially if groups of instances characterized by them are to be analyzed in different ways.

3.4 Data Collection Instruments

In this research that relates to integration of Agile methodology in software projects, the researcher heavily relied on online surveys and available literature materials. Data collection is the process of gathering and measuring information on variables of interest, in an established systematic fashion that enables researcher to answer stated research questions, test hypotheses, and evaluate outcomes. It is described as a physical or nonphysical method of generating quantitative or qualitative data for analysis and interpretation. This procedure necessitates some sort of proof that the data is sufficiently connected to the construct of interest. The data must be examined and interpreted, and the interpretation must be supported by evidence. The gathering of data relating to Agile in software projects was crucial part of the research process to deduce a conclusion on their applicability. The data collection instrument chosen by the researcher is determined by the type of data to be collected if it is qualitative or quantitative and how it will be collected. Different methodologies use data collection and analysis depending on research type that includes interviews, focus group discussions, observation, surveys, questionnaires, and case studies. Data collection instrument involves accuracy and systematic data collection that is critical to conduct scientific research. It allows the researcher to collect information that they want to collect based on the study objects. Questionnaire as a method of collecting data instrument consists of series of questions and other prompts for the purpose of gathering information from respondents. All respondents are given an opportunity to give feedback on the research project. A well- designed questionnaire has to meet the research goals and objectives and minimizing unanswered questions. This method allows researcher to collect the most accurate data. The researcher clearly defines the

target, study population from which the data is collected. Additionally, Questionnaires can provide quick responses, but adequate care must be taken when developing questionnaires, to ensure you don't influence the response you receive. The design of your questionnaire should reflect your research aims and objectives.

The development of the survey is based on the critical components relating to the implementation of agile methodology in the software projects.

3.4.1 Online Survey Sample

QUESTIONNAIRES

1. Does your organization apply Agile Methodology in the implementation of Software Projects?

- ☐ Yes
- ☐ No

If yes Explain how

2. How do you rate the success of integration of agile methodology in comparison with other methodologies?

- ☐ Excellent
- ☐ Good
- ☐ Fair
- ☐ Poor

Explain

3. What benefits can you pinpoint in the application of Agile Methodology in Software Projects in your organization. Explain

4. What are the Challenges that you have experienced while applying Agile Methodology in software Projects in your Organization? Explain

5. Do you think that Agile fosters both individual and group responsibility?

- ☐ Yes
- ☐ No

Explain your Answer

6. Can you recommend the application of Agile Methodology in Software Projects?

Explain

- ☐ Yes
- ☐ No

7. Which industry are you in and how do you implement Agile Methodology in your Software Projects?

3.5 Ethical Consideration

It comprises the collection of elements and concepts aimed at resolving the issues of what is good and wrong in human events. Ethics look for justifications for behaving and demonstrating things. It's crucial to remember that ethical concerns are meant to help people learn more. With the help of Arifin's (2018)'s research, ethical considerations may be handled at both the individual and social levels. The honest assessment of how other people's actions most influence people. In terms of the language used to articulate ethical claims, there are two

categories. The term "ethics" refers to a set of claims regarding what is good and evil, what should be done and what should not be done, as well as the grounds behind such assertions. For instance, investigators must obtain their agreement before actually utilizing people's genetic data in a project since doing so respects their autonomy and freedom of choice. In ethics, claims defining the types of reasons used in prescriptive assertions may also be found. For example, one of the "principles ethics" principles is to respect a person's autonomy. Acting to get the best outcome, on the other hand, is a justification based on results rather than ideals. The researcher examines several ethical considerations that are likely to be considered when making and defending genetic information judgments in this chapter. The legislative options presented in this Report to preserve genetic information find a balance among these considerations. All scientists have faced obstacles in adhering to ethical norms that underpin their study. Participants should, in particular, offer their full consent to engage. Educated consent refers to researchers providing enough knowledge and guarantees about involvement so that subjects may fully understand the implications of involvement and make a more informed, intentional, and voluntarily chose whether or not to participate.

Chapter 4: Discussion, Findings, and Results

4.1 Introduction

In this chapter, the main conclusions of a research project, or what the project suggested, revealed, or indicated, will be discussed. Presentation findings for this study will be followed by the results and discussion of the findings. In agile methodology, software development is divided into distinct phases that contain activities with the intention of better planning and management. The process includes pre-defined deliverables and artifacts that the project team must generate and complete to develop a software project. Most software organizations have implemented the agile methodology process to ease the development process. For the last decade, the long goal is a repeatable and predictable process that can improve productivity and quality. Some programmers have tried to standardize software design work, while others have adopted project management techniques. Software projects can easily be delivered late or over budget if project management is ineffective. Effective project management seems to lack in a huge number of software applications that do not their goals in terms of functionality, cost, or delivery timeline. Following the study, the findings were discussed simultaneously with the results that were realized based on the development of software projects using agile methodology (Albayrak, 2017). Additionally, Agile software development has become a major influence on how software development is conducted. It has become a catch-all word for several improvements in how software engineers organize and coordinate their work, as well as how they communicate with consumers. The agile development method is evaluated as a reaction to plan-based or traditional methods that emphasize a rationale engineering-based approach for extensive planning and addressing challenges of unpredictable values towards software development.

4.1.1 Response Rate

The response rate of the survey directed to different respondents from different organizations using Agile methodology in their software projects were as shown in the table below

Table 2

Response Rate of the Survey Directed to Different Respondents from Different Organizations

Sample size	Frequency	Percentage
Correctly filled and submitted	85	100
Not submitted	0	0
Total	85	100

Note. Researcher (2021)

The findings were collected from the questionnaires that were accurately filled up and given back. The trial size was 85 respondents, out of which the reply percentage was 100%. The reaction percentage is outstanding given the proposal; a response percentage of 50% is enough for examination and reporting. A rate of 60% is usually good quality, while a response percentage higher than 70% is outstanding (Lomulen, 2016). Generally, when we base on this affirmation, it means that the response rate for this research was sufficient.

4.1.2 Age of the Respondents

The table below analyzes the ages of the respondents who were interviewed using online survey tool.

Table 3

Ages of the respondents who were Interviewed Using Online Survey Tool

Age Range	Frequency	Percentage
21-25	6	6.4

26-35	34	41.0
36-45	27	33.3
46 and above	17	19.3
Total	85	100

Note. Researcher (2021)

Based on the above, it shows that the highest population interviewed / those who filled online survey were between the age of 26-35 years (41%), followed by those between 36-45 years (33.3%), then those with the age of 46 years and above (19.3%), and lastly those between 21-25 years, who formed 6.4%

4.1.3 Agile Project Management Industry

Table 4

Agile Project Management Industry

Industry	Frequency	Percentage
Extractive	10	11.8
Manufacturing	15	17.6
Information Technology	59	69.4
Assembly	1	1.2
Total	85	100

Note. Researcher (2021)

Based on this the respondents interviewed, they came from diverse industries dealing with different project management projects. Out of these projects, 10 (11.8%) of the respondents came from extractive industry, 15 of the respondents (17.6%), 59 out of 85

respondents came from Information Technology Industry dealing with software projects, while 1 (1.2%) came from Assembly Industry.

4.2 Analysis Approach

When it came to data gathering, the investigator relied on MS Excel, that was invaluable in assessing information from diverse source materials. The methodology was experimental since the bulk of the other parameters in this investigation were either unique or changed from their inputs. Qualitative component evaluation was used to evaluate the instrument initially. The structure of the links among the derived structures was next investigated. Finally, the complete model was tested using a multiple linear regression model. Three elements were included in the facts and figures: ecological input, technical input, and a reduction in forward preparation. The use of agile methods was initially planned to include four components. The expected component of iterative distribution did not show in this information collection, thus it was eliminated from the analysis; nevertheless, one of the elements that should have registered on iterative distribution rather registered on environmental input. Despite the fact that the importance of well-drafted agreements develops in lockstep with the size of the software system, the programmers' and customers' connections and collaboration take precedence over strict contracts. Minimal additions and brief iterations are used to build software, allowing for rapid detection of dangers and issues and appropriate response. Every iteration adds new techniques to the mix while also tweaking the ones that have already been implemented. The hypotheses of continuity in the association connecting the independent parameters, uniformity for every variable's allocation, and independence of the two parameters are all used in correlation test.

homoscedasticity across the relationship's range. while running universal retrogressions between the quality primary variable & the other items in the model, after

performing numerous assessments of the residuals. The assumption of linear relationships between the conjectures and the primary variables was unlikely to hold, according to the homoscedasticity of residua and a review of enhanced component-plus-residua plot.

4.3 Results

Throughout the last century, agile methodologies approaches have developed as a key fresh perspective of planning and implementing project activities. Agile proponents have produced a number of assertions regarding the advantages of agile approaches, such as increased group productivity and effectiveness, improved software reliability, and business effectiveness. Although latest evidence on the usage of agile methods has yielded promising findings, a variety of other agile process proponents' prescriptive assertions have yet to be validated.

It has long been believed that software development processes are influenced by ecological and other variables. This implies that the effectiveness of a software production methodology is determined by relationships among the atmosphere's characteristics and the production product's characteristics. Conventional software techniques have been impossible to generate high standard of quality of implementation, especially with a relatively large starting expenditure, according to agile proponents, since the software development life cycle is fundamentally unpredictable and experimental. However, agile proponents say that the software creation methodology is much closer like to that of a fresh technology design approach than it is to that of a design phase. Agile approaches, as a result, have incorporated methods, approaches, and accompanying technology that are supposed to improve production performance. Nevertheless, as discussed in Chapter 2, a large number of agile methodologies' prescriptive claims have not yet been analyzed, and the approaches themselves depict only partially homogeneous results.

The findings reveal that the amount to which agile methods are used has a beneficial impact on project performance. Project management measures, product attributes, and anticipated economic implications all had substantial implications on expected successful project views. On the other side, the influence of the amount to which the agile technique is used on project management success characteristics was demonstrated to be nonlinear. Whereas the benefits of agile usage were typically favorable, the gradient of the impact had previously been significantly reduced at the average. drastically growing once more The use of the agile technique has proven to be beneficial in the development of a software project. The findings revealed that, in most models, the implementation of agile methods has a positive influence on project success; nevertheless, the benefits were diverse. The use of agile impact has a regular favorable direct influence on the Quality element. In many scenarios, greater quality was associated to higher levels of agile approach adoption, however the effects were not linear. While the use of agile methodologies was consistently high, the consequences were complex, with some good and negative effects happening within each paradigm. In the architectural difficulty assessments, all impacts were significant and in the predicted pattern. In the methodological difficulties scenarios, the impact on expenditure was non-significant, the impact on schedule was varied in orientation, and the result on breadth was as expected. Eventually, the dynamic analysis is shown, had a considerable, but reversed, impact on the budget outcome, while the implication on time and scope results was as anticipated.

4.4 Methods

Agile development methodologies place a strong emphasis on quality standards for integrating the design process. It's a methodology for managing software development initiatives with professionalism. Despite widespread and profitable in software design, agility proposes an alternate and gradual technique for developing software initiatives.

Development begins with agile methodologies when the needs are properly specified. Agile development's brief cycles are designed to promote flexibility and programmer concentration. A reasonably accurate estimation of personnel and schedule needs is practically impossible absent a rigorous design step. Throughout the development phase, the programmer has the ability to modify or delete functionality. "Scum, Extreme Programming (XP), Crystal, Lear Development, Agile UP, and Dynamic Systems Development Method" are examples of agile software development approaches (DSDM). Extreme programming, which adheres to the SCRUM paradigm, emphasizes on rapid development and client fulfillment. FDD is an agile strategy for managing small iterative cycles that result in usable software. It is centered on a highly flexible software development process that allows for unexpected software modifications. prerequisite! to coordinate software development depending on the functionality description in the corporate demands prerequisite. SCRUM is yet another agile software development strategy that emphasizes on generating the most benefit in the quickest amount of timeframe. It is the greatest widely utilized agile technique in software development. It is generally relevant owing to its accessibility, which concentrates on software management challenges instead of technological program design processes. Scrum, on the other hand, is an agile method structure that provides for the monitoring and administration of objectives and also software engineering. It's a cyclical and progressive software development paradigm with a specified methodology, like as a software component that can be reused.

Little components were produced gradually. Furthermore, the aggressive coding style is extremely handy when clients' expectations or requirements are continually altering, or when they are unsure regarding the program's operation. It encourages rapid business deployments with short developmental cycles, increasing system performance and providing a stage in which any customer requests may be immediately implemented. The "waterfall"

paradigm of serial software development can be distinguished from agile software. Agile methodologies workflow technology, and also monitoring and monitoring technology for agile development, may enable programmers completely exploit the advantages of these approaches, such as adaptability and short lifecycle durations. As a corollary, user-friendly agile methodologies software suite is in high need. Agile approach distinguishes from conventional technique in regards of the structure used. The agile model is continuous and evolutionary, for each phase possessing its very own discrete technology developmental lifetime span. Client happiness is a top concern, that may be achieved by deploying applications that is just correct, not overactive or undeveloped, and delivered early and often. Additionally, since operational software is a crucial metric of performance in agile approach, every iteration's software should be operational if this approach is to be applied. The operational specifications for each iteration must be functional if this technique is to be used. The functional requirements for each iteration typically grew over time while remaining as simple as possible. As previously said, the agile technique's internal practices are highly varied and have a wide range of focuses. The requirement for continual input, on the other hand, is universally recognized in agile techniques. Except for Lean/Kanban, every agile method advocates for an iterative delivery cycle. Each development action is considered an iteration in Lean/Kanban, which brings the iterative cycle to its logical conclusion.

4.5 Findings

Based on the findings study on the integration of agile method in developing software projects, the comparison among a long-standing release developed with classical methods to a new release developed using agile methods, the developers discovered that 66% rise in terms productivity for the recent agile release as compared to the old one. Nevertheless, in such a case, the agile project teams had more experienced staff in software engineering and project

management experience than the classical project team member. A common brief made by the promoters of agile approach is, that agile processes amount to better quality in various ways. The arguments in favor of this are frequently related to continuous code testing and integration, rather than doing so after the project as in plan-driven approaches. The data was conducted using a questionnaire and more than 100 respondents on the agile method on the development of software projects as an effective method. This method provides successful refinement and creates a product that is extremely suitable for software development in the software industry due to agile project management iteration and continual feedback. It also decomposes features into smaller increments that require less amount of planning for a short time. Each team member in the project works in every software development for planning, analyzing, designing, developing, testing, and implementation of the application development process. Nevertheless, the findings show a considerable essence for future studies on the effects of agile approaches and environmental conditions on the effective delivery of software projects.

4.6 Discussion

This sections clearly discusses how the research objectives have been well achieved. Based on the foregoing, agile methodology has been used in the creation of software projects, requiring developers to collaborate continuously. The selection of this method shows that software development methodologies have the greatest impact when there is a high fit between surrounding factors and the practices of the method. Due to the rapid changes requested by software developers, traditional development methodologies such as the waterfall method are no longer used in software project development. As a result, agile methodology is proposed as the best method for how tests are performed, and developers adopt this methodology to overcome issues raised by the majority of team members. In an agile environment, the

majority of choices are made by the development team, which includes software developers and the client. While the approaches have different facilitation practices and distinct areas of focus, this dissertation explains that they all have a concept of the usefulness of feedback. We conceived agile method utilization at a measurable level across the many agile methodologies in use today, particularly the various feedback mechanisms that are founded on the shared viewpoint of the agile project manifesto. Therefore, discussion based on approaches of software development such as waterfall, and agile method have some advantages in a real-world work atmosphere of the developers that involves them to equip skills on software project development. The software industry discovered that agile processes are better suited to small, stand-alone projects. Scaling up and integrating agile principles into a company with well-defined traditional processes is difficult for developers and managers. As a result, the industry is looking for a way to combine agile and traditional approaches so that their benefits can be combined.

4.7 Conclusions

Agile methodologies advance from the viewpoint that the software development effort must unswervingly and efficiently handle the change. These techniques prescribe intricate networks of progressions, practices, and procedures, and supporting technology, all of which are intended to improve software delivery in unpredictable settings. Whereas such methodologies are extensively unrelated in their respective practices, such practices that are universal to all of the approached are the ones that are premediated to obtain feedback hints from the environment. Ideally, it is only via the processes that these environmental hints that software development teams can establish adaptive capacity and agility. Several recommendations have been made regarding the effects of the agile approach application on and the environmental conditions for which the methodologies are best suited. Agile

methodology application is anticipated to have the highest effects on environments that are greatly dynamic. Additionally, software development is the backbone of today's digital economy. Additionally, the findings of this study emphasize the need for feedback in dealing with the unpredictable nature of software development. However, the findings suggest that more research is needed into the effects of agile approaches and surrounding factors on the effective delivery of software development projects. In this research, many types and methodologies for agile software development have been explained whereby each methodology has its merits and demerits, hence, there is no excellent methodology for different types of projects, each project has its requirements, characteristics, and needs to be done. Therefore, selecting the best agile methodology to be used in the project development must be done with due diligence, based on these distinctions, or sometimes there is no agile methodology that can be used on some projects development so the traditional methods can be optimal for these cases, such as the organizations with many teams and employees, and projects with a critical huge budget.

By concluding, we can allude that the agile approach is a s to plan-driven methodology (eg: UML or Waterfall Model) in the development of quality software. Because developers do not frequently focus on models as the main component of the final software, it is also ideal for small to medium-sized projects. Agile approaches feature a distinct approach to each software engineering phase that focuses on feedback and change. The research has shown that traditional plan-driven software development methodologies are not used in practice. It has been argued that the traditional methodologies are too mechanistic to be used in detail. As a result, industrial software developers have become skeptical about "new" solutions that are difficult to grasp and thus remain unused. Agile software development methods, "officially" started with the publication of the agile manifesto, attempt to bring about a paradigm shift in

the field of software engineering. Agile methods claim to place more emphasis on people, interaction, working software, customer collaboration, and change, rather than on processes, tools, contracts, and plans.

4.8 Recommendations

It is worth noting that before applying the agile methodology to software project development, developers need to undergo training to help them to acquire key agile project management competencies. This project recommends agile methodology as the effective method in the development of software projects. Traditional techniques, such as the waterfall methodology, were created to help in delivering software after a project. As a result, developers did not engage with the program until after the project was completed. Due to its focus on the early delivery of working software, the agile methodology allowed mismatches between created software and developers to be discovered initially in the project. When a result of this, agile methodologies may be better equipped to make adjustments in meeting the requirements that emerge only as developers associate with the real system. For this reason, agile methodology has been proposed by developers as an effective method that can provide a positive impact on software project performance. Due to software iterative delivery paradigm, agile methodology practitioners have given a positive recommendation that agile method provides better results and more useful software. This enhanced delivery success is considered to minimize risks and offer a good return on investment. Agile methods proponents argue that the approach should be implemented in stages, with the major pain points of a team being addressed first. As a result, early in the usage cycle, agile techniques are likely to yield quick results, as the practices that are highly likely to improve performance are deployed first. This result is instinctive when reflecting on the commendations of agile developers. Additionally, the Agile method application was initially predicted to encompass four perspectives,

nevertheless, the data comprised into 3 aspects, environmental feedback, technical feedback, and minimized upfront planning. The theoretical aspect of repetitive delivery did not exist in this data set and was declined from the analysis, even though one of the items that were expected to load on iterative delivery that is well composed rather than environmental feedback. Agile developers, on the other hand, emphasize that, due to the conjointly reinforcing perspective of agile practices, the method's full performance benefit is only realized when the majority of the practices are implemented. Because the interaction of these techniques is complicated, teams must learn how to use them in order to be successful. This means that, after experiencing the immediate initial performance effects teams will continue using agile principles in the hopes of further performance improvements.

Chapter 5: Research Implications, Recommendations, Limitations, And Lessons Learned

5.1 Introduction

In this chapter, it offers an overview of the research implications, recommendations, lessons learned, challenges encountered among others. Additionally, this section provides reiteration of the problem and purpose statements, research questions with reference to results and findings. The main goal of this research was to perform a comprehensive analysis. Based on the problem statement relating to integration of Agile Methodology in the Development of Software Project, Traditional software development methodologies have been coupled with several problems. Ideally, most corporations dwell on offering quality and obtaining customer product quality, handling complications and improving production output and performance. It is worthy to observe that the technological developments in the automotive field are in line with application of agile methodology in the assembling of vehicles particularly on matters of programming and execution of software modules. Digital operation and the innovation has led automotive industries to have improved stakes particularly in their processes and value chain. The industry is redefining stratagems on what a car and fulfillment with the aim of implementing their projects. The key problem with the traditional project management and software development methodologies are associated the complexity of implementation process. The issue lies on selecting between the traditional methodologies and agile development methodologies. However, both approaches have positive and negative sides. Making the right choices play a crucial role while starting a new project. The key issues to put into consideration while selecting the methodology has been a major challenge while choosing the software development methodology. Most of the organizations have been found themselves in challenges especially mapping their business needs.

The application of agile methodology requires the problem to be well defined in advance. The solution to be offered by agile also needs to be well determined in advance. Though with agile methodology there is flexibility, the problem lies on the integration of agile methodology in the development of software project. Agile projects come with several challenges that are distinct from those faced by projects following traditional methodology. Based on the research done, there are various issues affecting the implementation of agile projects in the software development process. Such problems include issues with communicating, management of day today operational problems, customers and team members, changing culture and mindset, among other challenges. Notably, selecting the right methodology and matching the methodology and project needs has been one of the key challenges as far as project management is concerned.

Agile methodology is considered ideal when the product details cannot be defined or agreed in advance with any degree of accuracy. This issue requires a collaborative environment existing between the user or clients and the developer. One of the crucial challenges related with agile methodology is based on the availability and popularity of the methodology is critical, customer availability, and organizational culture. In line with Agile manifesto affirms that project managers must deliver working software frequently. Based on the above problems connected with the development of software projects, the proposed solution is based on the integration of agile methodology in the development of the software project. With Agile projects, the idea is to implement software projects incrementally and through iterative approach to development. Integration of agile in the software projects comprises of various stages which include project inception, planning, requirements refinement, product backlog, testing, and transition. The solution realized in the implementation process of agile approach focus on time, cost, quality, and features, which are

variable in nature unlike in the traditional methodology that are used in software development. Lastly, the solution derived from the integration of agile methodology in the software development projects is based on identifying factors which makes the transition from traditional to agile methodology necessary. Additionally, this solution can be realized whether the software project is small or big.

5.2 Effectiveness of Agile Methodologies in Software Projects

Notably, agile software development methodologies have emerged to be of great significance in the past few decades. Agile professionals have made several claims relating to the impacts on the application of Agile approaches including the related enhancements in team effectiveness as well as performance and software qualities. Where recent research on the application of agile methodology have shown considerable results, various prescriptive claims by agile methodology practitioners have not yet been tested vividly. Additionally, software development practices have been hypothesized for quite long due to the contingent and surrounding factors. This implies that the efficiency of software development methodology is as a result of function between surrounding factors and other factors. Agile software development professionals have affirmed that since software development processes is fundamentally uncertain and empirical, classical software development methodologies have not been able to develop plans of project execution in a successful manner. In this dissertation, we elaborated that, while agile methodologies have adopted practices and practices that have been significant in software development, it has become necessary to have different point of focus in the realization of the required results. Various studies have shown agile teams executing performing specific agile methods. Implementation of agile methods in software development is mostly based on the application of agile manifesto. Agile methodology, projects are broken down into small portions, which are finished in work- related phases that

operates from the design phase, testing to quality-assurance phase (Špundak 2014). The agile development methodology is also referred to as Scrum. Application of Agile method is mostly where rapid feedback is required. The agile method mostly involves four phases, which include Scanning the emerging trends as well as the risks. Analyzing the information led decisions and solutions. The third is responding to the opportunities and risks. The fourth phase is change whereby it involves shaping the future environments.

The results shows that the degree of agile methodology application positively affects the success of project. Significant implications were noted to be on the projected perspectives of the project success, which include project management measurements, product qualities, and organizational effects. Nevertheless, the effects of the impact of the application of agile method on the project success was found to be non-linear. While the application agile in this research was found to have a positive correlation on software projects. Agile methodologies promote on the application should be adopted. The subduing implication of the technical complexity on the implication to the extent of agile methodology application. As information technologies become pervasive, most of the business have shift to agile project management method where the 7implementation of their Information Technology infrastructure. Other industries have greatly expanded in a tremendous manner, thus reducing the cost of promoting the efficiency as they continue to make business profits or gains. Datrika (2018) argues that the features that have greatly changed chin various industry in product development are based on the customers' demands for improvement. With this, companies have continuously faced many challenges in managing the above-mentioned issues or features. Such challenges include dealing with the interconnected and the complex businesses. Large number of people purchasing various organizations products need such products to be equipped with modern technologies.

Agile methodology is a good approach that is crucial for helping define the processes in a project. This method is well regarded for due to its flexibility to project management and it helps in breaking huge tasks into short manageable tasks. The reasoning behind breaking down the project into small ones is to permit the management of the project to swiftly adapt to change as well as fast delivery. The idea of applying agile was at first developed by automotive and manufacturing industry which is a software that is primarily developed for manufacturing and automotive industry. Nayak, Padhye (2017) notes that over the past few years, the automotive industry has been developing an initiative that has brought change. The main focus of the customers is to obtain products that are standardised and meet the personal requirements. The application of agile project methodology in the automotive industry has contributed to enhanced production capacities. Nonetheless, automotive manufacturers are facing tough competition from competitors to offer them modern technologies in manufacturing their products to their worthy customers. According to Vasylykov (2019) through the adoption of agile methodology, automotive manufacturers minimise the time to market their products, enhancing cost efficiency, manufacturing of the product's base on the customer's needs. Other advantages comprise enhanced product quality, handling complexity and enhancement of production optimality together with the performance. It is worthy to note that the technological progressions in the automotive industry are in line with the application of agile methodology in the making of vehicles particularly when it comes to programming and execution of software programs. Digital operation and the innovation driven in automotive industries have improved stakes, mainly in their processes and value chain. The industry is redefining stratagems on what a car can and should do (Gauger et al, 2019). The transition from cars with combustion engine to electric vehicles is sufficient to rock the field through the innovation, whereby software has become the game changer. Due to this, there has been an

enhanced production capabilities by the companies in the automotive industry. The argument behind deploying agile approach is due to the fact that the automotive sector is increasingly becoming driven by innovation and software. In this case, agile offers numerous advantages as indicated elsewhere in this paper.

One of the means of is by delivering similar to a software company. The field has captured excellence to formulate high quality product as cheaper costs and within reasonable short periods. The technology inspired revolution in this industry has been transforming the rules where the automakers are anticipated to be more industrious. The application of the Agile methodology in the manufacturing of vehicles comprises of relying on three pillars. The initial one is launching three pilots at the corporate level to enhance swift decision making (Gaughan et al, 2019). Such pilots dwell on maximising the customer journey from the purchase time to end of purchase, mastering the data driven marketing and reducing the dependence on external providers and lastly concluding the issues of customers swiftly. The second pillar involves restructuring the overall institution to facilitate agility. This mainly comprises organizing teams to lay emphasis on product development. The third pillar in the application of agile methodology in the automotive industry is making prior planning and building momentum with agile ways of working.

5.3 Research Implications

Development of Agile Information Systems In the last decade, methodologies have evolved as a new way of managing work and delivering information systems development teams, with a substantial number of companies reporting the results. Agile approaches are being adopted and used. Only a handful of these assertions, however, have been tested in the academic literature to yet. Extreme programming, Scrum, and other agile approaches prescribe a wide range of practices, some of which are incompatible. Furthermore, the application of

agile methods techniques is not limited to agile development projects; it has been used in non-agile methodologies situations. Nonetheless, prior research has tended to focus on procedures that have been prescribed. The biggest distinguishing feature of agile techniques, according to this dissertation, is their considerable emphasis on gathering and processing feedback from the outside world. The influence of agile approaches as evidenced by the utilization of these feedback processes is the subject of this dissertation. The importance of a team's ability to process repetitive and continuous feedback from the environment is explained using the theoretical lenses of team adaptation, organizational learning, and earlier literature on new product development in this study. Future studies in information systems should focus on the effects of agile techniques. While this research was mostly exploratory in nature, it did open up a number of new avenues for future investigation. This study contributes to the research stream by providing evidence that more development and pragmatic research is needed.

In terms of software development approaches, practitioners are still looking for "silver bullets." However, as previous research and this study reveal, the employment of techniques labeled as agile methodologies has a variety of consequences is conditional on the environment's suitability. As previously stated, software development methodologies have the utmost effect when there is high with reference to the surrounding factors and practice of methods. This was found to be the case in this research. This study backs up previous findings that show that structural complexity has a negative influence on teams in general. When teams function in the midst of increasing degrees of structural complexity, the consequences of the extent of agile method use have a considerable detrimental influence, according to this research. When implementing agile teams, IT professionals should be cautious to keep the structural complexity to a minimum which may have an implication on the teams. Where it is possible, project teams ought to be collocated and the number of stakeholders as well as their

reporting relationships needs to be minimized. This study also discovered that the impact of technological complexity on success is very non-linear. There were unfavorable relationships between agile application and success at both low and high levels of technological complexity. This runs counter to earlier practitioner comments that agile approaches are best suited for less technically challenging tasks.

One key reason for this finding could be because if a project is tremendously basic, people's expectations of its success and consequences are likely to be low. Contrarywise, the technical feedback mechanisms may be overwhelmed by a high level of technological implications and complexity a collection of agile teams. If the technological effect is hampered by high system complexity, technical feedback, the negative effects of technical feedback would be reduced. For example, if high levels of integration of agile methodology in software projects is implemented with external systems, an agile development team development team may not be able to write and complete tests that are meant to fully tests the system. Based on the findings, the existence of high levels of vitality is the single most important component of uncertainty that would imply a strong fit for the adoption of agile approaches. The capacity to recognize and respond to change has been demonstrated to improve performance in various situations (Tripp, 2016). The effects of agile method ado in situations with lesser degrees of dynamism, on the other hand, are less noticeable. As a crucial pointer of possible high performance for agile methodology utilization, organizations need to assess the level to which the team should be able to adjust to change.

Based on the implications of future research, more empirical and research is needed especially on the integration of agile methodologies to software projects. Further, further empirical and theoretical research on agile methodology needs to be undertaken in a comprehensive manner. The rationale behind this is to adopt longitudinal research designed to

facilitate the researchers investigate the impacts of agile methodologies in software projects over time. Additionally, this would be paramount in having an understanding on the effects of agile methodologies and their uncertainties, Moreover, the non-linear nature of the data obtained in this research over time may mean the existence of a recursive process. The implication of the application of experimental research would be of great essence on a cyclical impact on agile method over time. Secondly, with the application of nomological network with the aim of motivating the hypothesis, was notable that network on its own wasn't tested in this research. With the clear understanding of this network, it would be of great significance and interest when furthering this research. Furthermore, the proposed nomological network might be applied to any software development methods. This concept of a nomological network of software development methods components has never been tried before. With proper testing of this concept of agile methods in software projects, it would be of great significance in the field of information systems.

Thirdly, the correlations between different constructs would be contrary to the prior research that has made a proposal on linear negative relationships existing between project complexity and its success and linear positive correlation that exist between complexity and project success. With these findings, they show an indication on the essence of the establishment of new theory. A considerable theoretical has recognized the application of non-linearity in the literature of organization behavior. Nevertheless, information systems theories have though with a few exceptions supported the proposal of establishments of linear relationships and different studies have designed measurements as well as tests that makes assumption on the linear correlation. With such results, they add to previous call for the development of theories relating to non-linear effects.

5.4 Limitations of this Study

As a result of conceptualization of a comprehensive extent relating to integration of agile methodology application construct, its impact of agile on diverse dimensions of the success of the project has not yet been tested comprehensively. The first identified limitation in this study on theorizing impacts of application of agile methods in software projects which was motivated by team adaptability perspective and organization learning. Such theories are integrally reinforcing due to their cyclical nature. However, this research study was undertaken based on cross-sectional research design that show the state of conceptual model based on particular point in time. Secondly, based on the sample assessed in this study, it comprised of agile teams and was a purposive sample. With this, it created intentional sampling bias through inclusion in the research in the research where only the teams that self-identified as application of agile methodologies. Nevertheless, such bias required as the concepts and constructs relating to agile teams' process and practices that would not be possible to be interpreted by non-agile professionals. With this, it appears that there is a limitation in terms of the generalization of the study and does not show the applicability of the model of the study or conclusions made outside agile teams.

Another limitation relates to the study, which sought to include the constructs and variables that were identified through the review of literature. Ideally, those variables implemented were half the number of variables identifies. Hence, imposing a limitation of this study. This further imposed limitation on the respondents interviewed who were not subjected to all research variables identified in this research. The specific nature of interactions with the client and the frequency as well as the quality of interactions with them was identified to be of extreme importance by the 2 out of the 5 focus groups. Ideally, to the extent, that the research

model did not consider such variables, we must remain cautious in the application in the research findings.

5.5 Conclusion

Integration of Agile Methodologies in software projects advance from the premise that software formulation effort that should efficiently and consistently handle changes in the software projects. Ideally, such approaches are prescribed to complex network of procedures, practices, and procedures. And supportive technologies that are in a similar way are meant to bring a positive implication in the delivery of the software projects within uncertain environments. Such technologies are known to be heterogeneous in nature based on their defined practices. Various claims and assumptions have been made about the application of agile methodology in software projects and their success rate. Agile methodology application in software projects is anticipated to be the most effective in case of low structural complexity barriers. Notably, the results of this research and investigation outlines the nature and significance of feedback for further research on the integration of agile methodologies in software projects and environmental conditions of for the successful completion and delivery of software projects. Additionally, the application of agile project methodologies in software development is critical and can never be undermined. Lastly, with the application of agile methodology, organization can minimize the time need to market and advertise their products, increased cost efficiency, development of the products based on the customer's requirements, among other advantages.

References

- Abrahamsson, P., Oza, N., & Siponen, M. T. (2010). Agile software development methods: A comparative review. *Agile software development*, 31-59.
- Adolph, S., Kruchten, P., & Hall, W. (2012). Reconciling perspectives: A grounded theory of how people manage the process of software development. *Journal of Systems and Software*, 85(6), 1269-1286.
- Alaidaros, H., Omar, M., & Romli, R. (2018, June). A theoretical framework for improving software project monitoring task of Agile Kanban method. In *International Conference of Reliable Information and Communication Technology* (pp. 1091-1099). Springer, Cham.
- Albayrak, A. (2017). Restructuring of System Analysis and Design Course with Agile Approach for Computer Engineering/Programming Departments. *INTERNATIONAL JOURNAL OF ADVANCED COMPUTER SCIENCE AND APPLICATIONS*, 8(11), 84-89.
- Alqudah, M., & Razali, R. (2018). An Empirical Study of Scrumban Formation based on the Selection of Scrum and Kanban Practices. *International Journal on Advanced Science, Engineering and Information Technology*, 8(6), 2315-2322.
- Al-Saqqa, S., Sawalha, S., & AbdelNabi, H. (2020). Agile Software Development: Methodologies and Trends. *International Journal of Interactive Mobile Technologies*, 14(11).
- Arifin, S. R. M. (2018). Ethical considerations in qualitative study. *International Journal of Care Scholars*, 1(2), 30-33.
- Beck, K., Beedle, M., van Bennekum, A., Cockburn, A., Cunningham, W., Fowler, M., ... & Thomas, D. (2001). *Agile Development Methods: Philosophy and Practice*.

- Beerbaum, D. (2021). Applying Agile Methodology to regulatory compliance projects in the financial industry: A case study research. *Applying Agile Methodology to Regulatory Compliance Projects in the Financial Industry: A Case Study Research* (April 26, 2021).
- Begel, A., & Nagappan, N. (2007, September). Usage and perceptions of agile software development in an industrial context: An exploratory study. In *First International Symposium on Empirical Software Engineering and Measurement (ESEM 2007)* (pp. 255-264). IEEE.
- Beux, J., Bellei, E., Brock, L., Bertoletti, A. C., & Hölbig, C. (2018). Agile design process with user-centered design and user experience in web interfaces: a systematic literature review. *Latin American Journal of Computing Faculty of Systems Engineering Escuela Politécnica Nacional Quito-Ecuador*, 5(2), 53-60.
- Boehm, B., & Turner, R. (2005). Management challenges to implementing agile processes in traditional development organizations. *IEEE software*, 22(5), 30-39.
- Christou, I., Ponis, S., & Palaiologou, E. (2009). Using the agile unified process in banking. *Ieee Software*, 27(3), 72-79.
- Cobb, C. (2019). What Are the Advantages and Disadvantages of Agile and Scrum?. Retrieved from <https://managedagile.com/what-are-the-advantages-and-disadvantages-of-agile-scrum/>.
- Datrika, N. K. (2018). Implementing Agile Methodology Techniques in Automobile Industry.
- Dingsøyr, T., Nerur, S., Balijepally, V., & Moe, N. B. (2012). A decade of agile methodologies: Towards explaining agile software development.
- Evaldas, B. (2015). White Paper: Scrum-ban for Project Management

- Felix, E. A. (2015). Bank Customers Management System. *International Journal of Scientific & Technology Research*, 4(08), 326-343.
- Fitsilis, P. (2008). Comparing PMBOK and Agile Project Management software development processes. In *Advances in Computer and Information Sciences and Engineering* (pp. 378-383). Springer, Dordrecht.
- Flora, H. K., Chande, S. V., & Wang, X. (2014). Adopting an agile approach for the development of mobile applications. *International Journal of Computer Applications*, 94(17), 43-50.
- Gandomani, T. J., Tavakoli, Z., Zulzalil, H., & Farsani, H. K. (2020). The role of project manager in agile software teams: A systematic literature review. *IEEE Access*, 8, 117109-117121.
- Gauger, C., Heller, K., Tordjman, K., Loh, A., & Rehberg, B. (2019). An Agile Game Plan for Automakers. Retrieved from <https://www.bcg.com/publications/2019/agile-game-plan-automakers.aspx>
- Gregory, P., Plonka, L., Sharp, H., & Taylor, K. (2014, June). Bridging the gap between research and practice: the agile research network. In *ECRM2014-Proceedings of the 13th European Conference on Research Methodology for Business and Management Studies: ECRM 2014* (p. 161). Academic Conferences Limited.
- 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.

- Hu, Z. G., Yuan, Q., & Zhang, X. (2009, July). Research on agile project management with scrum method. In *2009 IITA international conference on services science, management and engineering* (pp. 26-29). IEEE.
- Kaleel, S. B., & Harishankar, S. (2013). Applying agile methodology in mobile software engineering: Android application development and its challenges. *Computer Science Technical Reports*, 1-11.
- Katumba, B., & Knauss, E. (2014, December). Agile development in automotive software development: Challenges and opportunities. In *International Conference on Product-Focused Software Process Improvement* (pp. 33-47). Springer, Cham.
- Kumar, G., & Bhatia, P. K. (2012). Impact of agile methodology on software development process. *International Journal of Computer Technology and Electronics Engineering (IJCTEE)*, 2(4), 46-50.
- Loiro, C., Castro, H., Ávila, P., Cruz-Cunha, M. M., Putnik, G. D., & Ferreira, L. (2019). Agile project management: A communicational workflow proposal. *Procedia Computer Science*, 164, 485-490
- Lufthansa Industry Solutions. (n.d). Using Agile Methods to Become More Competitive. Retrieved from <https://www.lufthansa-industry-solutions.com/de-en/solutions-products/automotive/automotive-more-competitive-with-agile-methods/>
- MacGregor, J. N., Ormerod, T. C., & Chronicle, E. P. (2001). Information processing and insight: a process model of performance on the nine-dot and related problems. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, 27(1), 176.
- Moniruzzaman, A. B. M., & Hossain, D. S. A. (2013). Comparative Study on Agile software development methodologies. *arXiv preprint arXiv:1307.3356*.

- Nayak, R., & Padhye, R. (Eds.). (2017). *Automation in garment manufacturing*. Woodhead Publishing.
- Nuottila, J., Aaltonen, K., & Kujala, J. (2016). Challenges of adopting agile methods in a public organization. *International Journal of Information Systems and Project Management*, 4(3), 65-85.
- Oivo, M., & Liukkunen, K. (2016). An Empirical Study on the Impact of Scrumban on Geographically
- Opelt, A., Gloger, B., Pfarl, W., & Mittermayr, R. (2013). *Agile contracts: creating and managing successful projects with Scrum*. Wiley Publishing.
- Papadopoulos, G. (2015). Moving from traditional to agile software development methodologies also on large, distributed projects. *Procedia-Social and Behavioral Sciences*, 175, 455-463.
- Permana, P. A. G. (2015). Scrum method implementation in a software development project management. *International Journal of Advanced Computer Science and Applications*, 6(9), 198-204.
- Projects: The Project Managers' Perspective.
- Rahimian, V., & Ramsin, R. (2008, June). Designing an agile methodology for mobile software development: A hybrid method engineering approach. In *2008 Second International Conference on Research Challenges in Information Science* (pp. 337-342). IEEE.
- Rasnacis, A., & Berzisa, S. (2017). Method for adaptation and implementation of agile project management methodology. *Procedia Computer Science*, 104, 43-50.
- Saunders, M., Lewis, P., & Thornhill, A. (2009). Understanding research philosophies and approaches. *Research methods for business students*, 4(1), 106-135.

- Sharma, S., Sarkar, D., & Gupta, D. (2012). Agile processes and methodologies: A conceptual study. *International journal on computer science and Engineering*, 4(5), 892.
- Sharp, H., Plonka, L., Taylor, K., & Gregory, P. (2014, June). Overcoming challenges in collaboration between research and practice: The agile research network. In *Proceedings of the 1st International Workshop on Software Engineering Research and Industrial Practices* (pp. 10-13).
- Shastri, Y., Hoda, R., & Amor, R. (2021). The role of the project manager in agile software development projects. *Journal of Systems and Software*, 173, 110871.
- Siddique, L., & Hussein, B. A. (2016). Grounded Theory Study of Conflicts in Norwegian Agile Software Projects: The Project Managers' Perspective.
- Šmite, D., Moe, N. B., & Ågerfalk, P. J. (Eds.). (2010). *Agility across time and space: Implementing agile methods in global software projects*. Springer Science & Business Media.
- Socha, D., Folsom, T. C., & Justice, J. (2013). Applying agile software principles and practices for fast automotive development. In *Proceedings of the FISITA 2012 World Automotive Congress* (pp. 1033-1045). Springer, Berlin, Heidelberg.
- Špundak, M. (2014). Mixed agile/traditional project management methodology—reality or illusion? *Procedia-Social and Behavioural Sciences*, 119, 939-948.
- Stamelos, I. G., & Sfetsos, P. (Eds.). (2007). *Agile software development quality assurance*. Igi Global.
- Tipnis, P. (2018). Common Challenges Faced by First-Time Agile Organizations. Retrieved from <https://www.knowledgehut.com/blog/agile/common-challenges-faced-by-first-time-agile-organizations>

- Tripp, J. F. (2012). *The impacts of agile development methodology use on project success: A contingency view* (p. 217). Michigan State University. Business Information Systems.
- Turk, D., France, R., & Rumpe, B. (2002, May). Limitations of agile software processes. In *Proceedings of the Third International Conference on eXtreme Programming and Agile Processes in Software Engineering* (pp. 43-46).
- Vaishnavi, V., Kuechler, B., & Petter, S. (2004). Design science research in information systems. Association for Information Systems.
- Vasylov, O. (2019). How the Automotive Industry Can Benefit from Agile Software Development. Retrieved from <https://clutch.co/developers/resources/how-automotive-industry-benefit-agile-software-development>.
- Warburton, R. D. H., & Cioffi, D. F. (2014). Project management theory: deriving a project's cost and schedule for its network structure. Project Management Institute.
- Wysocki, R. K. (2011). *Effective project management: traditional, agile, extreme*. John Wiley & Sons.