The use of systems development methodologies by virtual software development teams

TM Pitso

20949014

Dissertation submitted in partial fulfilment of the requirements for the degree *Magister Commercii* in Computer Science and Info Systems at the Potchefstroom Campus of the North-West University

Supervisor: Prof HM Huisman

Co-supervisor: Prof L Venter

May 2016

It all starts here ™

NORTH-WEST UNIVERSITY
YUNIBESITI YA BOKONE-BOPHIRIMA
NOORDWES-UNIVERSITEIT

Acknowledgements

First of all I would like to thank God for helping me and my supervisors work together to achieve one goal. It has been a very long year for me and very stressful but all that is in the past now as I am knocking on the door to victory.

Secondly I would like to thank my Mother Mats'episo Pitso; my Father Hlalele Pitso and family for all the support that they gave me. I really appreciate every little penny that they sacrificed for me. The words of encouragements by themselves meant a lot to me.

Most importantly would like to thank both my supervisors Prof Magda Huisman and Prof Lucas Venter for being patient with me throughout this journey, and for lifting me up when I was down and directing me to the right path. I really appreciate. To my friend and mentor Dr Thabiso Mokotjomela - I really appreciate your supports and encouragement. Lastly, to all the participants and my friends who took part in this study I really appreciate it and salute you.

ABSTRACT

In this study, we are investigating how systems development methodologies and virtual software development teams work together to produce a better end product. Systems development methodologies have been defined as an approach that helps organisations to build their systems in a well-standardised and comprehensive manner in order to achieve common goals. This includes all the procedures or steps to be followed depending on the specific approach used by the individual organisation. A virtual team has been defined as a group of people, separated locally or internationally, working to achieve a common goal. Achieving a desired goal involve use of various technologies such as telephones, e-mails, video-conferencing, and any other effective communication modes. Human resource could include the most qualified candidates from inside or outside their companies who always form or reform a specific task team continuously. Organisations would also be able to deliver quality products and increase client's satisfaction and build strong reporting relationships as team members report to various authorities. I followed a positivistic approach which entails a survey as a research method using a questionnaire for data collection from a large group of people. Statistical analysis was applied for data analysis. Research findings were very reliable and significant based on the test for validity of data. Techniques that were used to analyse data were descriptive analysis, factor analysis, reliability analysis, correlation analysis, cross-tabulations and regression analysis.

Specific objectives of this study included identifying challenges and benefits faced by virtual software development teams. The results showed that even though there are still challenges; there are more benefits in virtualisation. For instance, challenges such as technology, culture, customs, personal conflict and distance between the teams occurred. Another objective was to evaluate the use (if any) and the suitability of current systems development methodologies for use by virtual software development teams. Descriptive measures using frequency and correlations and cross-tabulations were applied to the data obtained through questionnaire surveys. The results showed that virtual software teams are using a combination of traditional and agile systems development methodologies.

The other objective was to define how many are using SDM; what SDM they are using and if none; and what are the reasons for not using them. The results showed that 35% of virtual teams adapted systems development methodology (SDM) on a project-to-project basis while 33% of participants used a general guideline for all projects, and 32% of participants used a standard procedure which is followed thoroughly for all projects.

Another objective was to determine the success of projects developed by virtual software development teams. The results showed that the participants strongly felt that their projects were very successful in terms of customer satisfaction, product, competencies, process, communication and leadership. The last objective was to investigate the role (if any) of systems development methodologies in the success of those projects. The stepwise regression analysis was used to determine factors that influence the overall success of recent projects developed. The results for this study showed that support provided by SDM as production technology, support provided by SDM as control technology, *gender_m* (males) and team performance have a large influence on the success of projects developed by virtual software development teams.

Keywords

Systems development methodologies, virtual software development teams.

OPSOMMING

In hierdie studie ondersoek ons hoe stelselontwikkelingsmetodologieë en virtuele sagteware ontwikkelingspanne saamwerk om n beter eindproduk te lewer. Stelselontwikkelingsmetodologieë is gedefinieer as n benadering gemik op die bereiking van gemeenskaplike doelwitte. Dit sluit in al die prosedures wat organisasies kan help om goedgestandaardiseerde en omvattende oplossings te vind om gemeenskaplike doelwitte te bereik. Dit sluit al die prosedures of stappe in wat gevolg moet word deur die individuele organisasie. n Virtuele span is gedefinieer as n groep mense wat plaaslik of internasionaal verspreid werk aan „n gesamentlike doelwit. Om die verlangde doel te bereik word gekyk na die gebruik van verskillende tegnologieë soos telefone, e-posse, video-konferensies en ander effektiewe kommunikasiemiddele. Menslike hulpmiddele kan die volgende insluit: die bes-gekwalifiseerde kandidate van binne of buite hulle maatskappye wat deurentyd taakspanne vorm en verander. Organisasies kan ook goeie kwaliteitprodukte aflewer en kliënte se vereistes nakom terwyl hulle ook sterk verslagdoeningsverhoudinge opbou soos wat spanlede aan verskillende hoofde rapporteer. Ons het n positivistiese benadering gevolg wat „n opname behels het en waarvoor „n vraelys as wetenskaplike instrument gebruik is om data te versamel van n groot groep mense. Statistiese ontleding is gedoen om data-analise te doen. Navorsingsbevindings was baie betroubaar en betekenisvol gesien teen die agtergrond van die geldigheidstoets van die data. Tegnieke wat gevolg is in die ontleding van data was beskrywende analise, faktor-analise, betroubaarheids-analise, korrelasie-analise, kruis-tabulering en regressie-analise.

Spesifieke doelwitte van hierdie studie het ingesluit uitdagings en voordele wat ervaar word deur virtuele sagteware-ontwikkelingspanne. Die resultate het aangetoon dat hoewel daar nog uitdagings is, daar meer voordele opgesluit is in virtualisering. Byvoorbeeld het uitdagings hulle voorgedoen soos tegnologie, kultuur, gebruike, persoonlike konflik en afstande tussen spanne.

Nog,n doelwat was om die gebruik (indien enige) van die geskiktheid van huidige stelselsontwikkelingsmetodologieë vir gebruik deur virtuele ontwikkelingspanne te evalueer. Beskrywende metodes soos frekwensie en korrelasies en kruis-tabulerings is toegepas op die data wat verkry is uit vraelysondersoeke. Die resultate het aangetoon dat virtuele

sagtewarespanne gebruik maak van,n kombinasie van tradisionele en buigbare stelselontwikkelingsmetodologieë.

Die ander doelwit was om te definieer hoe baie mense gebruik maak van stelselontwikkelingsmetodologieë (SOM). Watter SOM gebruik hulle, en indien nie, hoekom gebruik hulle dit nie? Die resultate het aangetoon dat 35% van virtuele spanne SOM gebruik op,n projek-tot-projekbasis, terwyl 33% van die deelnemers dit as "n algemene riglyn gebruik vir alle projekte, en 32% van deelnemers dit as „n standaaardprosedure gebruik wat deeglik nagevolg word vir alle projekte.

Nog n doelwit was om die sukses te bepaal van projekte ontwikkel deur sagtewareontwikkelingspanne. Die resultate het aangetoon dat hulle projekte suksesvol was in terme van kliënte-tevredenheid, produk, vaardighede, prosesse, kommunikasie en leierskap. Die laaste doelwit was om die rol (indien enige) van stelselontwikkelingsmetodologieë in die sukses van sulke projekte te bepaal. Die stapsgewyse regressie-analise is gebruik om faktore wat die oorkoepelende sukses van onlangse projekte te bepaal. Die resultate het aangetoon dat ondersteuning wat verskaf is deur SOM as produksietegnologie, ondersteuning gebied deur SOM as kontrole-tegnologie, geslag (mans) en spanprestasie n sterk invloed gehad het op die sukses van projekte ontwikkel deur virtuele sagteware-ontwikkelingspanne.

Sleutelterme

Stelselontwikkelingsmetodologieë, virtuele sagteware-ontwikkelingspanne

TABLE OF CONTENTS

LIST OF TABLES

List of Figures

CHAPTER 1: RESEARCH INTRODUCTION AND PROBLEM STATEMENT

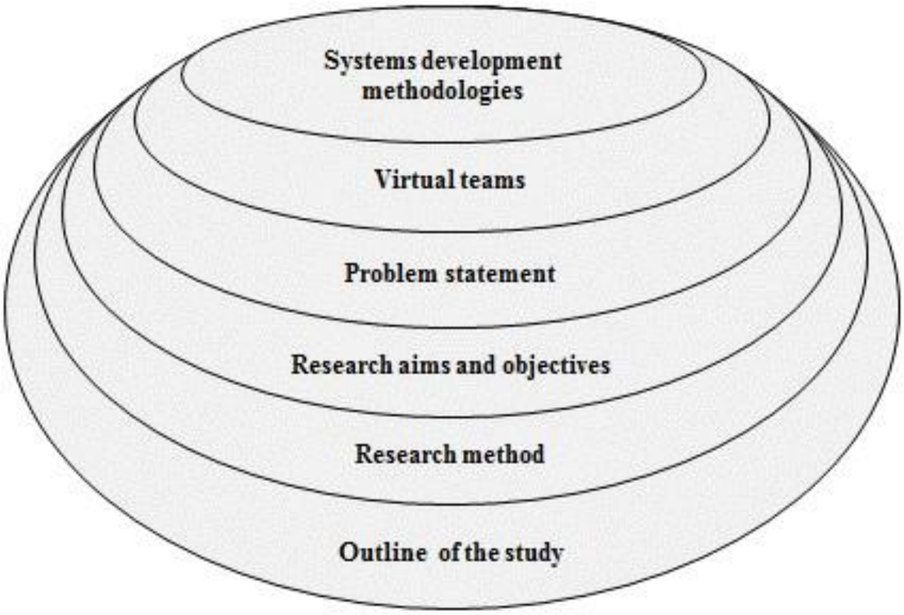


Figure 1-1: Outline of the study

1.1 Introduction

In the past, it was very difficult to communicate with people anytime or anywhere and to meet face to face was a regular mode of communication. There were no cell phones, no computers, no internet, only telephone communication and postal mails which were the most used modes of communication. Then first computers were invented in the form of big machines that no one would ever carry around. It made life better, then the small and portable ones emerged and before I know it everyone owned a laptop or a cell phone. These allowed everyone to communicate with everyone around the world. Then social media followed, and it introduced a whole new dimension or communication.

Distributed teams used to have difficulties when communicating as they had to travel in order to achieve their goals and when developing their systems. Today there is no such complexity as far as virtualization is concerned. Teams are able to communicate using the most advanced technology platforms to deliver quality products.

A virtual software development team may consist of programmers working interdependently between different countries using media such as electronic mail, instant messaging, telephone, shared databases and video conferencing (Cramton & Webber, 2005). This effective communication allows production of products over reasonable time frames because team members can exchange ideas.

For the past decades, systems development methodologies have been designed to assist in producing quality products (Gonzalez-Perez *et al.*, 2005). Systems development methodologies allow teams to create test plans, use case realisations and design models for end products (De Vries, 2012). Although systems development methodologies can assist software developers in various ways, not much is known about its use by virtual teams. In this study, I will investigate this issue.

1.2 Systems development methodologies

Systems development methodology (SDM) can be regarded as an approach that helps organisations to build their systems in a well standardised and comprehensive manner in order to achieve common goals. This includes all the procedures or steps to be followed depending on the specific approach/approaches used by the organisation. In attempt to attain a quality software product, different systems development methodologies have been produced.

According to Avison and Fitzgerald (2006), different organisations adopt a particular systems development methodology in order to improve the systems development process and to establish a standardised process. The adoption of each systems development methodology is influenced by several factors such as availability of a market for the targeted products; access to qualified staff to be employed and business capital (Cumps *et al.*, 2006). Some organisations create their own in-house systems development methodologies by combining formal systems development methodologies, like the Rational Unified Process, with their own systems development practices.

One of earliest methodologies developed for developing new software systems was the Systems Development Life Cycle (SDLC). It was developed in the late 1960s in response to the need for management control and structure in the system development process (Fitzgerald, 1997). The major problem of the SDLC is that each phase had to be strictly accomplished in order to get to the next one, and a change in one phase might affect other phases (Griffin & Brandyberry, 2008). Due to shortcomings of the SDLC, many systems development methodologies such as Structured

Analysis and Structured Design (SASD) and Object Oriented Analysis and Design evolved so as to improve software quality (Griffin & Brandyberry, 2008). This involves the use of tools such as data flow and entity relationship diagrams which allows reuse of the “code”. Another method is the Rapid Application Development (RAD) Griffin (2008). This method is composed of processes such as prototyping in order to promote user involvement in the design process.

However, Griffin (2008) indicated that generally there is no specific systems development methodology that can be regarded as good for a certain company due to some differences in culture, virtual software development teams involved and the ever-changing technology designs. According to Bygstad *et al.* (2008)’s results, respondents were asked whether or not they were using a formal SDM, the majority do not use a formal method, but a number of techniques and tools. In this study, I will focus on virtual software development teams and their use (or not) of systems development methodologies.

1.3 Virtual teams

A team can be regarded as a group of people working together to achieve a common goal. This group of people could be working together in the same company or outside their company. According to Samson and Daft (2003), teams are the main component helping organisations to perform and deliver better products. Organisations have been using traditional teams to deliver their products until virtual teams emerged during the 1990s as a new process (Furst *et al.*, 2004). Today many organisations are building their teams made up of talented and qualified people around the world with the aim to meet customers’ needs and be competitive (Kankanhalli *et al.*, 2007).

A virtual team can be regarded as a group of people, separated locally or internationally, working to achieve a common goal. This goal could be achieved using various technologies such as telephones, emails, video-conferencing, etc. It could be achieved with the help of most qualified candidates from inside or outside their companies who form or reform a team continuously.
Organisations would also be able to deliver quality products and increase client’s satisfaction and build strong reporting relationships as team members have many people to report to.

As a result of globalization, virtual software development teams were formed. Virtual software development teams can be regarded as a global team which develops systems in the hope of meeting customers’ needs either locally or globally. They can be able to work faster than traditional teams at the lower costs and with more talented and qualified teams around the world. Apart from other challenges on virtualisation, Olariu and Aldea (2014) indicated that another challenge is the management of processes in virtual software development teams. They further indicated that the use of BPMN (Business Process Model and Notation) could improve the work process using BPMN methodologies and tools.

Global organisations are increasing the use of virtual software development teams, in which participants apply advanced technologies to interact during business development (Workman, 2007). To my knowledge, virtual software development teams have no specific system development methodology to overcome their challenges. However, da Silva Estacio and Prikladnicki (2015) indicated that geographically distributed teams have adopted agile methodologies as a fit for their projects. This methodology involves distributed pair programming, which involves two developers working remotely on the same code. I will further explain virtual software development teams in chapter 2.

1.4 Problem statement

The literature suggests that there has been limited research done on how SDMs are used in virtual software development teams during global system development. Database for literature searched has been shown in Table 2-4.

In the previous paragraphs, I discussed systems development methodologies, virtual teams, and virtual software development teams. Many organisations are developing systems with distributed teams and not really paying attention to the suitability of SDMs that can fit their projects very well. The question is,

> *Do the virtual software development teams use SDMs when developing their systems for better results? If no, what are the reasons for non-application of SDMs?*

To answer this question, it is necessary to investigate whether and how organisations are using certain SDMs in their virtual software development teams. Challenges faced by the virtual software development team need to be identified and recommendations for best practice be made.

This research is crucial as it will help academics and practitioners: for instance academics could gain more knowledge regarding the use and effectiveness of system development methodologies in virtual software development teams. On the other hand, it will help practitioners who are involved and those who want to be involved in virtual teams to produce better quality systems. It is important to do this research as it will help most organisations involved in virtual teams on how to best approach their projects with methodology/methodologies, which will best suit their environment. Indeed, failures by virtual software development teams could result in severe financial losses to a company because of globalization. Almost all research in literature have focused on the cultural differences, time zone issues, and trust but the aim of this study is to analyse systems development methodologies used by virtual software development teams and the application of system development methodologies so that challenges can be detected.

1.5 Research aims and objectives

The aim of this research is to study the use of systems development methodologies by virtual software development teams. In order to address the research aim, the following objectives will be considered:

- To identify challenges and benefits faced by virtual software development teams.
- To evaluate the use (if any) and the suitability of current systems development methodologies for use by virtual software development teams.
- If no systems development methodologies are used, determine the reason why they are not used.
- To determine the success of projects developed by virtual software development teams.
- To investigate the role (if any) of systems development methodologies in the success of those projects.

1.6 Research method

A positivistic approach was used as research approach, survey as a research method, questionnaire as a means of data collection while statistical analysis was applied for data analysis.

Questionnaires were distributed locally and globally for companies that are using systems development methodologies in their virtual teams. Online and drop-off questionnaires were used in order to collect primary data from affected parties.

Eighty percent of participants were from South Africa and Lesotho while the remaining 20% were shared among participants in Botswana, India, China and the USA (Chicago). Participants who were targeted were software developers, software architects, managers, team leaders, and business systems analysts in order to understand how systems development methodologies are used. Moreover, how organisations apply systems development methodologies towards the successful end product. To also identify barriers that can affect a success in virtual software development teams.

Data was collected from all the designated groups in order to come up with a framework that can minimise the virtual software developments teams' projects failures. Data was analysed using the non-parametric statistics including descriptive analysis, correlation analysis, factor analysis and regression analysis. According to Zar (1996), systematic sampling is a type of probability sampling method in which sample members from a larger population are selected according to a random starting point and a fixed, periodic interval. For this study, I selected companies that are involved in systems development not specifically considering the location.

1.7 Outline of the study

The major aim the study is to investigate the use of systems development methodologies by virtual software development teams. There has been limited research on how the SMDs are in use by virtual software development teams and this leads to a limited understanding of failures reported in virtual IT projects. This study has been broken down into the following chapters:

- Chapter 1: Introduction and research problem statement

This chapter introduces the study background and consists of the research problem statement, the aims, and the objectives.

- Chapter 2: Literature study of software development methodologies and virtual software development teams

The chapter explores the existing knowledge on systems development methodologies and virtual software development teams. The definitions of systems development methodology and virtual teams and system development methodologies are provided. Also, the benefits and problems faced by virtual teams, related work, and the perceptions of other researchers and successes of the projects developed by virtual teams are considered in order to identify the gap intended for investigation in the study.

- Chapter 3: Research method and design

The chapter provides the research method that was used. For this study, a survey in the form of the questionnaire was used. Statistical analysis was used which showed the validity and significance of data.

- Chapter 4: Research results

Chapter 4 provides the analyses of the data and the results of the study that reflect how SDMs are used by the virtual software development teams in various organisations.

- Chapter 5: Research conclusions and recommendation

This chapter finalises the research and puts forward some conclusions and recommendations. Then the conceptual model is developed.

1.8 Summary

This chapter consists of the research background and the problem statement identified in the study, aims and objectives of the research, research method, and outline of the study consisting of literature review, research method, research results and research conclusions or recommendations drawn from the results. The next chapter defines the literature on SDMs from other researchers, virtual teams and some background on the use of SDMs, if any, by virtual software development teams.

CHAPTER 2: SYSTEMS DEVELOPMENT METHODOLOGIES AND VIRTUAL TEAMS

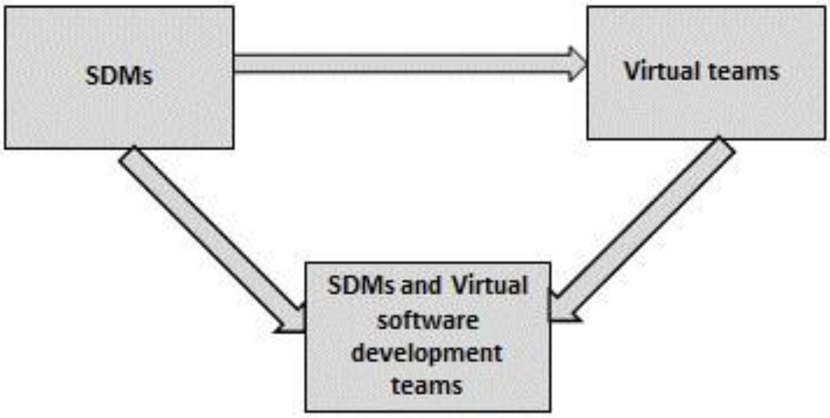


Figure 2-1: Literature review construction

2.1 Introduction

In the previous chapter, I presented an overview of the study. Systems development methodologies have been in use for a long time and are still currently being used by many organisations. Many systems development methodologies (SDMs) are emerging due to the ever-changing technology. Some organisations are developing their systems using SDMs while others are buying such SDMs through outsourcing companies. It is very important to develop software using SDMs as they have guidelines to assist in the success of projects provided they are followed from the beginning until the end of the project. In this chapter, I will focus on the literature and background information and SDMs usage by virtual software development teams. Three topics will be discussed, namely systems development methodologies, virtual teams and a combination of both. In the first section I will discuss systems development methodologies that are used in organizations and the success of IT projects in the below aspects:

- The definitions of systems development methodologies;
- History of SDMs
- The usage and effectiveness of systems development methodologies;
- The advantages and benefits of systems development methodologies;
- The disadvantages and criticisms of systems development methodologies;

- Categorisation and comparison frameworks of different types of systems development methodologies namely; STRADIS, IE, RUP, XP, ETHICS, and SSM.

In the second section I discuss virtual teams in systems development within the aspects outlined below:

- Effectiveness of virtual team
- Benefits of virtual teams
- Challenges faced by virtual teams

In the third section I discuss the combination of both (Systems Development Methodology and virtual software developments teams) in terms of the following aspects:

- Systems development methodologies and virtual teams
- Previous studies on the use of SDMs by virtual teams
- Success of projects developed by virtual teams

2.2 Systems Development Methodologies (SDMs)

Systems Development Methodologies are the guidelines or standards which can be followed when developing a system in any organisational sector. These methodologies are meant to reduce any risks that are associated with projects which can lead to financial loss, loss of clients or even company reputation if the project fails. To avoid these problems, I will discuss the use of SDMs, their benefits, effectiveness, and challenges.

2.2.1 Introduction

In order to have a clear understanding with regard to SDMs, I will provide the reader with a number of definitions from other researchers. The idea is to provide some different perspectives on SDMs and understand how they work in the real world.

According to Avison and Fitzgerald (2006:568): “A systems development methodology is a recommended means to achieve the development, or part of the development, of information systems based on a set of rationales and an underlying philosophy that supports, justifies and makes coherent such a recommendation for a particular context. The recommended means usually include the identification of phases, procedures, tasks, rules, techniques, guidelines, documentation and tools. They might also include recommendations concerning the management and organization of the approach and the identification and training of the participants".

On the other hand, Huisman and Iivari (2006), mentioned systems development methodology as a combination of a systems development approach, process model, method, and technique. Ramsin and Paige (2008: 3) defined methodology as "a means to provide timely and orderly execution of various fine-grained techniques and methods of software engineering".

According to Vavpotic and Bajec (2009: 528-545), "Systems development methodology is defined as a recommend mean to achieve the development of program systems, based on a set of rationales and an underlying philosophy. It involves phases, procedures, tasks, rules, techniques, guidelines, documentation and tools".

According to Iivari, Hirscheim and Klein (1998:165), an information systems development methodology can be defined as "an organized collection of concepts, methods, beliefs, values and normative principles supported by material resources". Moreover, information system development methodology (ISDM) is put together in a set of goal-oriented procedures in order to guide the stakeholders work and involvement among others that are affected by the system.

Moreover, systems development methodology helps the team to find a way or close any gap to the identified problem and propose a solution. This solution is provided in order to produce the production in business to the identified problem, Meso *et al.* (2006).

When examining the definitions of SDMS provided in the literature above, they can be classified into two groups namely;

- A combination of systems development approach, process model, method, techniques.
- Providing a solution to an identified problem.

In this study, I define an SDM as an approach that helps organisations to build their systems in a well standardised and comprehensive manner in order to achieve common goals. This includes all the procedures or steps to be followed depending on the specific approach/approaches used by the organisation.

2.2.2 History of systems development methodologies (SDMs)

Avison and Fitzgerald (2003) stated that in the early 1960s and 1970s, computer applications were developed without any methodology. Methodology emerged in the late 1980s to assist in the development of software. Systems development methodologies are categorised according to the following eras, namely pre-methodology era, early methodology era, methodology era and the era of methodology reassessment (Avison & Fitzgerald, 2006 & Dessie, 2009). Information systems development methodologies are used by various organisations in the sense to structure their development process (Zaied *et al.*, 2013).

2.2.2.1 Pre-methodology era

This was the first era which was known decades back in and around the sixties around European and North American companies. Developers were just building systems without a use of any systems development methodology (Avison & Fitzgerald, 2006). Developers were only focusing on solving specific technical problems and were not aligning the solutions to the organizations" needs due to lack of communication with non-technical people. According to Dessie (2009), there are also some characteristics that linked to developers which are highly technical; this technicality involves programming and maintaining the system with their own understating.

Because the approach to development was depending on individual experiences, the needs of the customers were rarely met and this led to poor control and management of projects. Systems development methodologies were established as more organizations were searching for a standard and controlled approached when building their systems. Moreover, developers were spending most of their time improving or changing the developed systems which led them to be more overworked (Avison & Fitzgerald, 2006). Dessie (2009) also mentioned that problems were caused by a failure to understand business and organisational context, the omission of core user requirements, communication problem and poor project management and control.

2.2.2.2 Early methodology era

Because of the problems described in the pre-methodology era, analysis and design were introduced as part of systems development and the role of systems analyst was introduced. As more organisations were growing in size and complexity, it became more advisable to move towards more integrated information systems. That is how the Systems Development Life Cycle

(SDLC) emerged in order to develop information systems. The SDLC consisted of phases, procedures, tasks, rules, techniques, guidelines, documentation programs and tools. Some of the most popular SDLC models were waterfall model, V-shaped, incremental life cycle model, spiral model.

Waterfall was the most regularly used model. It consisted of stages, these being a feasibility study, systems investigation, analysis, design, implementation, and maintenance. Figure 2-2 below represents the waterfall model.

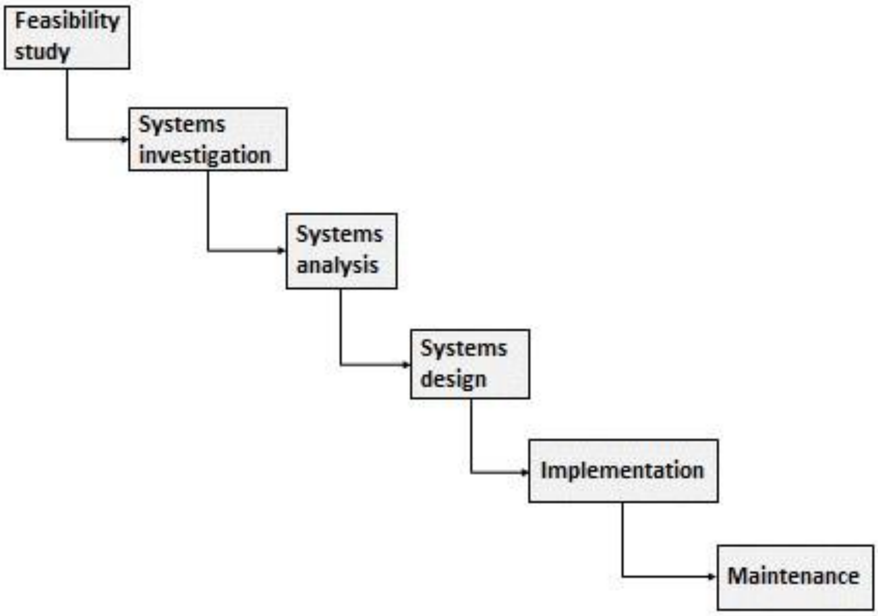


Figure 2-2: Waterfall model (Adapted from Avison *et al.*, 2003)

The waterfall model was used in the 1970s and 1980s and is still in use today together with other systems development methodologies (Avison & Fitzgerald, 2006). According to Dessie (2009), the early methodology is characterised by phase-based development which involved start in finishing. Since there were problems associated with the waterfall model, such as inflexibility, as it is difficult to change requirements, this led to the user dissatisfaction and failures to meet management's needs.

2.2.2.3 Methodology era

Due to the lack of flexibility and other problems of SDLC explained in the early methodology era. These critiques included among others (Avison & Fitzgerald, 2006);

- Failure to meet the management needs because of concentration on single applications at the operational level.
- Unambitious systems design because the emphasis was on computerising the existing system.
- Instability because of the modelling of processes which are unstable due to frequently changing business environments.
- Inflexibility because of the output design that makes changes more costly.
- User dissatisfaction because of documentation problems and inability to provide a simulation of how the system would work.

Various approaches to systems development methodology to overcome these problems emerged during the mid-1980s to late 1990s which led to the new methodology era. In this era, the methodology was used for the first time to define various approaches and methodologies. Those methodologies emerged from one of two sources, these being those developed from practice and those developed from theory. Methodologies derived from practice evolved from the usage in an organization and then into a commercial product while methodologies developed from theory came from universities or research institutions (Avison & Fitzgerald, 2006). What emerged during this era was the classification of the SDMs into categories; we will explain this further in section 2.2.6.

Since the 70s, there have been many developments in techniques and tools and incorporated as a modern version of the waterfall model. These techniques included entity relationship modelling, normalization, data flow diagramming, structured English, management software, data dictionary software, system repositories and drawing tools (Avison & Fitzgerald, 2006). Dessie (2009) on the other had described the methodology era as being characterised by methodologies from practice, theory, diversified techniques and tools and development of methodology as a product. Moreover, there seemed to be difficulties in adopting methodologies and insufficient focus on social and organisational issues.

Agile software methods emerged in a way of helping organisations to develop their systems in a different way as compared to traditional methodologies. Agile methodology success is based on hearsay rather than hard facts so research is still minimal in the academic world (Chow & Cao (2008). In support of this, Darwish and Rizk (2015) found that 60% of the projects failed from 2004 to 2012 which showed a very high rate of failure in IT projects. To overcome these failures, many organisations became more engaged with the agile methodology as its currently booming in the market. It is more flexible in terms of quicker delivery, simple phases, easy to change requirements and strong communication between developers and customers as compared to traditional methodology. Since the word *agile* means ability, it is easy to move and not only that but is also quick.

It was introduced mainly to overcome problems of waterfall methodology in the early 90s. It is an iterative approach using shorter and lightweight development cycles and various deliverables. Highsmith (2010) mentioned agile as a successful methodology that delivers today and adapts tomorrow. Agile methods are based on some principles, project initiation which involves team formation, resource planning, and requirements, iteration process which takes place until all requirements are met by both customers and team members then project closure.

2.2.2.4 Post-methodology era

Here methodologies were being perceived as having moved beyond the pure methodology era. This was the most recent era from the mid-1990s till today which was classified by practicalities of the methodologies of the methodology era. Because of this situation, some organizations were confused as to which methodology was good for their projects. For this reason, some organisations tend to other different methodologies and approaches while others completely stopped using methodologies in their organizations at all. Moreover, some organizations were moving in different directions, for instance, outsourcing (Avison & Fitzgerald, 2006). Dessie (2009) indicated the reaction of the post-methodology era using ad hoc development, outsourcing and contingency and to wait for better methodologies. However, there was a dilemma as to whether:

- Organisations should give up on methodologies,
- There is any need to understand more about methodologies or
- They have enough knowledge of methodologies‟ practical application and impact.

Due to the existing problems and limitations, a step was taken to examine some of these problems. The post-methodology era was characterised by the era of methodology reassessment. The era of methodology reassessment helped developers with six considerations, namely; external development, continuing refinement and improvement, ad hoc development, contingency, agile development and consolidation.

The first consideration was external development. However, if the choice was to develop internally, users might demand their methodology in use to be refined and improved, which is the second consideration.

The third consideration was contingency, meaning users might prefer to adopt a methodology according to their needs or an ad hoc approach (fourth consideration) which might be more informal and risky. Some organisations have turned into rapid and agile approaches (fifth consideration) because of their flexibility and speed in developing systems. These approaches have strengthened user and customer involvement. Finally, as compared to the early days of methodology, IT industry is currently in a more stable environment which predicts a more consolidation future (sixth consideration).

2.2.2.4.1 External development

As mentioned above, organisations are moving in different directions. External development means organizations go out to outsource services for system development or buy in their requirements in the form of packages or bespoke applications developed by specialist software developers. External development has been recommended as a quicker method and a cheaper way of implementing systems since no extra cost would either be incurred for requirements missed or misunderstood (Avison & Fitzgerald, 2006). According to Zaied *et al.*, (2013), it is useful to use expert systems as a tool to automate the process of selecting suitable systems development methodology as compared to a single methodology for all projects.

Because of the problems SDMs were giving, this resulted in offshoring/outsourcing of systems development. Many organizations lost interest in how systems were developed – they rather focused more on end-product and the effectiveness of the system being delivered. Because there had to be a third party when outsourcing, the chosen vendor would be responsible for delivering quality and satisfying systems. This process would reduce time for originations in thinking about SDMs to use. Again, it is the responsibility of every organisation to select the correct vendor when buying or outsourcing for the benefit of good end results. In this case, customers would only be interested in the final product other than the how the systems are developed (Avison & Fitzgerald, 2006).

2.2.2.4.2 Continuing refinement and improvement

The search was still continuing for the right/suitable systems development methodology. Moreover, methodologies are evolving and continuing to be developed from time to time. There are important techniques that might be very beneficial and important in systems development but are excluded or rarely used by some organisations when developing systems. Such techniques include stakeholder analysis, rich pictures, case-based reasoning, cognitive mapping, scenario planning and lateral (Avison & Fitzgerald, 2006).

2.2.2.4.3 Ad hoc development

Developers here were responsible for choosing any approach that they thought would work by just applying their skills and area of expertise. This brought back the approach of pre-methodology where there was no formalized methodology to be used when developing systems. But it also carried the risk of repeating problems that had been encountered before the invention of methodology. The risks could be reduced by providing developers with some support, training, control and maintenance of the development process (Avison & Fitzgerald, 2006).

2.2.2.4.4 Contingency

This is an information systems development approach representing the format to help developers. Contingency helps in development by taking into account the tools and techniques that might be expected to be used and adapted or not used at all depending on the specific situation. For instance, project size, projects type and its objectives, organisation and its environment, users, developers and their skills are considered to be specifically unique. The project might differ in terms of purpose, complexity, importance and projected life or impact. The contingency approach might be regarded as one methodology for all developments of which it is adopted by some organisations. It is normally problematic when the advantages of standardisation are not gained and therefore many skills are required to different types of approaches.

2.2.2.4.5 Agile development

This approach involved users and customers of the system in a joint approach to more than processes and tools. There is less documentation and rather working software, less contract negotiation and responding to change rather more customer collaboration. The software is delivered in chunks unlike in traditional methodologies and in a much shorter time. Changing requirements are accepted as compared to traditional methodologies. These approaches follow more today's information systems development (ISD) needs than many of the ISD methodologies of the "methodology era", for instance, when reacting to internet speed development. These development features are normally found in extreme programming (XP), SCRUM and DSDM.

2.2.2.4.6 Consolidation

Avison stated that since 2006, there has been a decline in numbers as some methodologies and their associated techniques and tools have not increased. However, this does not mean the non-use of methodologies and frameworks for ISD as a whole but methodologies for developing IS. Some methodologies are still being used successfully and effectively together with the agile and contingent approaches to ISD. This might be seen as a consolidation continuing process as there is maturity in the IS field generally.

I attempted to review the history and background of systems development methodologies. Various methodology eras were discussed such as the pre-methodology era and the post-methodology era. The post methodology era seemed to be described as an ideal methodology as it is composed of various approaches and solutions to most of the problems. An ideal methodology is a modern one that falls under agile methodologies. Agile methodologies have improved as compared to the traditional methodologies which had some problems when developing systems. Even though some organisations are resistant to change, they could still use both traditional and modern methodologies as they have borne good results so far.

2.2.3 The usage and effectiveness of systems development methodologies

To date, there are some organisations that are not using any methodology at all; some are using them partially while others are using them rigorously. Methodologies usage are categorised according in terms of commercial to internal. According to Avison 2006, these methodologies" usage was not analysed separately due to the fact that they did not show any significant difference after being analysed. There were only ten out of 23 respondents who mentioned that they were using a commercial methodology rigorously.

Agile projects are successful three times more often than non-Agile projects (42% vs. 14%). Success is achieved when a project that is delivered on time, on a planned budget and includes all planned features. Agile projects are more successful than traditional projects (67% vs. 50%) and are challenged less (27% vs. 36%) as well as fail less often (6% vs. 14%), (Ambler, 2011).

According to Bygstad *et al.* (2008), it was found that 57% of practitioners use approaches and do not use any formalised methodologies while only 8% do follow a methodology. Organisations are more comfortable in using what they know rather than using new methods. This means that people are scared of change or they might not like what they have to deal with. It is normal to be scared but sometimes change is good, it might bring about positive results in the long run which could not have happened had there been no move.

The graph below shows methodology usage in percentages according to years.

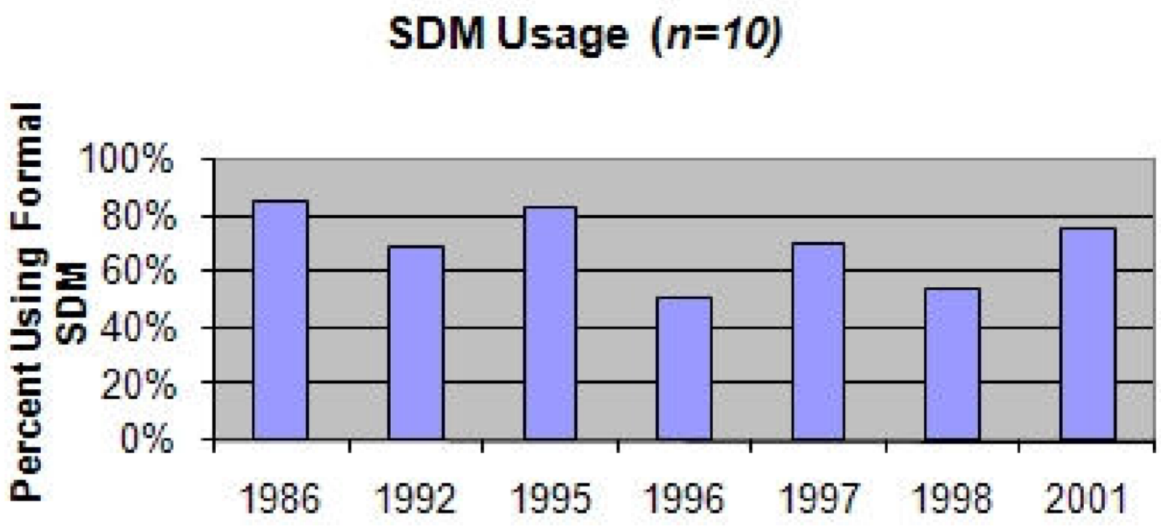


Figure 2-3: Methodology usage (adapted from Griffin & Brandyberry, 2008)

The researcher collected data from one country and one continent on SDM usage in the U.S and Asia. U.S surveys were 86% on average in the mid-1980s while Asia surveys were 68% from 1998 of companies using SDM, which means both surveys were consistent because it was collected in various organisations.

According to Griffin and Brandyberry (2008), a survey by Hardy *et al.* (1995) looked at methodology usage and customization in the United Kingdom of which five hundred and ten companies were selected randomly, based on the graduate jobs and courses. The response rate was 20%, 18% of those were not using methodologies, while 44% were using a formally structured system development methodology. Moreover, most companies used in-house methodology or collection of tools. Other 24% used Structured Systems Analysis and Design Methodology (SSADM), others Yourdon, Jackson Structured Design (JSD), OOD, formal specification and Information Engineering (IE).

According to Griffin and Brandyberry (2008), a survey by Chatzoglou (1997) involved participants in U.K projects. They were categorised as academics, software houses and consultancies, and industry. Seventy-two responses of the 182 surveys were sent out. Only 69% of the projects used methodologies in different stages of the development process. And 31% used no methodology. For projects that used the methodology, 23% of them used SSADM and in-house methodologies, 8% used prototyping. Twenty-seven percent of projects used “other” unspecified methodologies.

According to Griffin and Brandyberry (2008), a survey by Iivari and Maansaari (1998) was sent out to IS managers and eighty-seven organisation identified as using CASE tools. Out of 420 questionnaires that were sent through the mail, 63 (15%) were returned from 44 companies. The majority of respondents used object-oriented approaches (39%), followed by SA/SD at 23%. Some methodologies that were used included IE, JSD, in-house developed methodologies, other and anonymous. In the “anonymous” category, participants listed some techniques rather than formal methods. Twelve participants did not respond to this question. 34% of participants used a commercial methodology and 34% used an in-house methodology adapted from a standard method.

Since systems development methodologies were designed to build quality products, many organisations have adopted the SDMs (Workman, 2007). For instance, advanced organisations have taken the use of agile methodology into consideration in orders to compete in the global marketplace (Blaise *et al.*, 2008).

Other researchers regarded quality in terms of system, application, and information. System quality in terms of operations refers to reliability, availability, accessibility, security, and compliance (Gorla & Lin, 2010; Van Bon, 2007). Application quality relates to effective development and deployment of applications (Arnott & Pervan, 2008); ease of use, and usefulness (Gorla & Lin, 2010). Information quality characteristics relate to accuracy, completeness, currency and format (Nelson *et al.*, 2005). In contrast, Griffin (2008) indicated that users prefer to use defective products as they are more familiar with them than switching altogether to the high-quality product as it is more difficult to associate with inexperienced applications.

Most companies do not follow any methodology or life cycle. Thus, from analysis of failures, if the wrong people do the wrong things, use the wrong methods and techniques, and do not attend to the necessary variety of complexity, application success is unlikely (Conger, 2010). According to Baschob and Piott (2007) information systems (IS) department is a source of tremendous frustration, missed an opportunity, and inefficiency in companies. He further indicated that software development projects have regularly encountered problems and shortcomings that resulted in noteworthy delays and cost overruns, as well as occasional total failures. SDLC is composed of five phases which must be completed sequentially in order to develop software solution; this is time-consuming for projects (Bassil, 2012). That means each phase must strictly be accomplished in order to get to the next one, and a change in one requirement will affect all other phases (Bassil, 2012).

This shows that most organisations may not use SDMs unless they have knowledge of what a specific methodology can deliver. According to Huisman and Iivari (2006), even though there are about 1000 methodologies existing, organisations are still thinking they are under pressure to use them. They further indicated that many organisations were not using any methodologies because of debatable associated value besides the high investment in their development.

2.2.4 The advantages and benefits of systems development methodologies

According to Vavpotic and Bajec (2009), other researchers indicated that the use of formal systems development methodology increases productivity hence the quality. Different motivations took companies to introduce systems development methodologies. Avison and Fitzgerald (2006) showed three main categories of rationale; a better product, a better development process or a standardized process.

2.2.4.1 A better product

Even if there are some ways of comparing results of using various methodologies, the elements that are perceived to constitute measures of quality differ from person to person. A better end product could be achieved by using an SDM throughout the project life cycle (Avison & Fitzgerald, 2008; Huisman & Iivari, 2006). In order to determine the functionality of the system, there are several quality measure components involved. These components are fast development rate, effectiveness, low coupling, efficiency, documentation, reliability, acceptability, availability, cohesiveness, compatibility, ease of learning, flexibility, functionality, implement ability, amenability, portability, robustness, security, simplicity, testability, timeliness and visibility (Avison & Fitzgerald, 2008).

2.2.4.2 A better development process

There were various steps in the systems development life cycle which contributed to a better development process. The use of SDMs combined with SDLC can deliver a better end product.

However is it sometimes argued that “the use of methodology reduces level of skills required of the analyst, which improves the development process by reducing its cost” (Avison & Fitzgerald, 2008: 571).

2.2.4.3 A standardized process

Methodologies consist of standardised processes and this helps the development team to achieve a common goal. Every organisation using methodology follows a certain procedure and this makes it easier to integrate the system. Everyone involved in the development becomes familiar with the standards and becomes more flexible, knowledgeable and gain experience which could help in other projects in future (Avison & Fitzgerald, 2008).

According to Peterson (2013), SDM normally serves as a guideline to deliver systems on time, within budget and without failures. If organisations use SDM, they will be able to provide better estimates, deliver stable systems and always update customers about their product. SDM helps in identifying problems before they can occur and this helps developers to be more proactive in managing their occurrence and developing a plan for when they do occur. Methodologies facilitate project control and increase prominence in the development process (Fitzgerald, 1998). The use of methodologies also allows for fields of specialisation as more people will be recruited such as business analysts, system designers, programmers, and testers.

Benefits developers expect to receive by using a formal methodology include higher quality code, lower costs, shorter time to market, and improved productivity (Ware, 2001). Although methodology usage has been highly recommended in the academic research arena for many years, it has not been universally embraced by business. There is still no consensus on how best to develop software applications (Griffin, 2008).

2.2.5 Some disadvantages and criticisms of systems development methodologies

Avison and Fitzgerald (2006) mentioned that methodologies are not stable but moving targets that are on-going to develop and evolve. Due to its instability; it is difficult to know which version of methodology has been applied in a certain situation.

Avison and Fitzgerald (2006) continued and indicated that documentation of methodology is not always published or available and organizations are not buying methodology. They further stated that the implementation of these methodologies is different from what has been prescribed in the documentation or by methodology authors.

In addition, different developers follow and interpret the use of a methodology according to their own understanding (Avison & Fitzgerald, 2006). Other methodologies do not exactly mention their underlying philosophy and this makes the analysis more difficult as it has to be searched for and interpreted. In most cases, methodologies are not widely used in web-based applications. Developers use the trial-and-error approach, only putting their trust in experienced and skilled people in the organization as compared to the pre-methodology era of the 1960s and 1970s (Avison & Fitzgerald, 2003).

There are many methodologies that exist (Avison & Fitzgerald, 2006), and soft methodologies differ from hard methodologies such as a structured approach, a factor which organisation need to consider before choosing methodology. There is also an assumption that methodologies are applicable to all development situations, and that is not the case as some methodologies cannot work well on virtual teams, that it why there must be a contingency for each development situation. Due to the fact that SDMs do not address factors like individual creativity and intuition projects are likely to fail due to this shortage (Fitzgerald, 1998).

Reasons for other developers for not using methodologies include that they do not increase productivity; they are complex and designed for the largest and complex development projects. They are often above what is required (Avison & Fitzgerald, 2003). However, the use of methodologies also requires highly technical skills that may be difficult and expensive for developers, and for the end users to learn or acquire (Avison & Fitzgerald, 2003). Methodologies are costly and difficult to use and yet still not delivering enough benefit that will satisfy developers and users (Avison & Fitzgerald, 2003).

The methodology does not address the organization's issues or problems well because it only adopts one approach to the development of projects and it is said to be one-dimensional (Avison & Fitzgerald, 2003).

The use of the methodology in organizations can lead to a focus on following the procedures of that methodology rather than on focusing on the real business issues. Some organizations have found it difficult to adopt methodologies in practice, confronting conflicts from developers and users (Avison & Fitzgerald, 2003). Some organizations that have rejected the use of methodologies altogether are returning to the less formal and more flexible approaches (Avison & Fitzgerald, 2003).

Criticisms of all of these life cycles and methodologies are there in large numbers. The most condemning statement is that they appear to make no difference to the resulting quality of an application (Avison & Fitzgerald, 2006). Another thing is that every focus on one aspect of an application results in ignoring, constraining, or assuming other aspects of the application (Boehm, 2006). Even if methodologies are used in some of the organisations, if they are not rigorously followed it will always lead to some errors in the developed systems. However, some methodologies can be improved or redefined by mentioning their underlying philosophy for easy analysis.

The use of a single methodology to guide all project work is failing because there is „no silver bullet" and no one SDLC or methodology that can usefully guide the variety of work done in a typical IT development department (Conger, 2010). As many as 50% of programmers have less than four years of college, are overwhelmed by their work, and do not use good software or design practices (Boehm, 2006). According to Conger (2010) many risks attendant on development projects are normally ignored. Major project practice risks relate to realism of schedule and budgets (Conger, 2010); insufficient attention to functional complexity (Boehm, 2006); inability to learn from past failures and insufficient attention to user interface (Conger, 2010); problem avoidance (Sherman *et al.*, 2006); inability to control project scope (EwusiMensah, 2003); and lack of adequate technical skills (Boehm, 2006; Ewusi-Mensah, 2003).
Development practices and failure to manage risks are not the only failing.

2.2.6 Categorisation and comparison framework of different types of SDMs

There are a large number of systems development methodologies which we need to categorize in order to understand their applications. There are also various frameworks available to classify SDMs. According to Yaghini (2009) on Andersen's framework identifies a checklist which includes criteria relating to values and society, such as the Normative Information Model-based Systems Analysis and Design approach which is based on the models and epistemology of systems thinking and to a large degree one can evaluate and measure a methodology against these criteria. The Avison and Taylor Approach identifies five different classes of situation and appropriate approaches.

As mentioned earlier, an SDM includes all the procedures or steps to be followed depending on the specific approach/approaches used by the organisation. Avison and Fitzgerald (2006) mentioned frameworks for hard and soft system methodologies in a comparison which consists of six main elements, namely philosophy, model, tools and techniques, scope, outputs, practice and product. I will be using the framework by Avison and Fitzgerald (2006), because it is based on the idea of an underlying philosophical paradigm and therefore it has an influence on all other aspects. It may be explicit but in most methodologies, is understood because methodology authors rarely stress their philosophy.

2.2.6.1 Philosophy

This means a set of principles underlying the methodology. It is considered to be one of the most important aspects because this is where the orientation and participation choices are made (Avison & Fitzgerald, 2006; Palacio, 2010).

Philosophy could be defined as a principle or set of principles that are the background of methodology. It consists of paradigm, objectives, domains, and target. Paradigm looks at whether the methodology is scientific or system model before application to the project. Objectives of the project and organization must be clear, and that means a methodology could help in providing guidelines on objectives as well as tasks to be performed.

Domain falls under philosophy and means that organisations could be able to see the suitability of methodology in terms of parameters and thus one can decide whether the methodology is only going to solve a specific problem or is it only for strategic approach. Every organization should know whether a methodology is only targeting a specific size or environment of the organization.

2.2.6.2 Model

This model serves as a communication tool. It takes the core problems between different phases such as from design to implementation. It is categorized into different types, these being verbal, analytical or mathematical, iconic, pictorial or schematic and simulation (Avison & Fitzgerald, 2006; Palacio, 2010).

Modelling is the element of framework that concentrates on the analysis of the model that the methodology must adhere to. Every organization must consider different methodology models because they are the means of communication. For instance, data-flow diagrams, process orientation, integration of both data-flow diagrams and the iterative development phases. These models could aid in the selection of methodologies after considering the project's flexibility. It is also a representation that provides insight into the problem or area of concern (Yaghini *et al.*, 2009).

2.2.6.3 Techniques and tools

Tools are used to support specific parts of the development process. They are designed to support some techniques or complete methodologies (Palacio, 2010). According to Avison and Fitzgerald (2006), techniques support the collection, collation, analysis, representation or communication of information about systems requirements and attributes, techniques such as entity modelling, normalization and data-flow diagram are used.

A key element of the framework is the identification of tools or techniques used in a methodology. Tools and techniques must be identified according to the methodology selected. This is really recommended as some methodologies can only be applied with specific tools and techniques, and others do not even have specific tools or techniques while other methodologies could be applied without the use of tools or techniques as they are not important.

2.2.6.4 Scope

Scope means the information systems development life cycle stages that are covered by the methodology (Palacio, 2010). This framework is recommended to review the level of detail in which stages are covered.

Scope could help organisations to take into consideration the structural elements of the methodology. Scope consists of phases, sub-phases or stages of systems development life cycle which a specific methodology covers. Even though some systems development methodologies do not follow a life cycle, they may still adopt an iterative or spiral model. According to Yaghini *et al.* (2009), it is useful to have an assumption that an examination of methodology's scope in relation to the life cycle.

2.2.6.5 Outputs

Outputs refer to what is delivered at each stage and at the end of the whole process. They are defined to guide analysts (Palacio, 2010).

This element consists of a system and data modelling which serve as deliverable per phase or an activity with allocated responsibility. These outputs can be data-flow diagrams, data structure diagrams, data dictionaries, decision trees, decision tables, structured English, normalization and other outputs that fall under various methodologies

2.2.6.6 Practice

This element describes the number and type of users who currently apply the methodology (in practice) as well as the type of participants responsible for implementing the methodology (Vries 2004). There are three sub-elements according to which a practice can be measured being: background (Commercial or Academic), user base (Type of users) and participants (Users or Specialists) (Palacio, 2010).

Practice consists of background, user base and participants. The background can help organisations to identify in which environment it is created, whether it is academic or commercial. User base might really contribute to systems development methodology suitability. Participants normally determine whether systems development methodology can be undertaken by professional technical developers or users and what skills could be required.

2.2.6.7 Product

The product defines a final result of a certain process, a methodology for instance. This means buying methodology as a product may include any software, training, customer service and consultancy (Palacio, 2010). The product could be a document or training if necessary.

Table 2-1 illustrates the framework for methodology comparison that should be taken into consideration when developing systems. The main aim of Table 2-1 is to summarise different SDMs and determine how they relate to each other. According to this presented framework analysis (Avison 2006), organisations can benefit when selecting a suitable methodology based on features. A proper choice of the methodology by organisations can ensure its successful application. I will then explore how each methodology fits better for virtual teams based on the Avison’s framework provided in Table 2-1.

Table 2-1: An assessment of systems development methodologies using the Avison (2006)'s framework for methodology comparison.

	Structured Analysis, Design and Implementation of Information Systems (STRADIS)	Information Engineering (IE)	Rational Unified Process (RUP)	Extreme Programming (XP)	Effective technical and human implementation of computer-based systems (ETHICS)	Soft Systems Methodology (SSM)
Philosophy						
• Paradigm	Science paradigm	Science paradigm	Science paradigm	Science paradigm	Systems paradigm	Systems paradigm
• Objectives	Clear objectives to develop computerized Information Systems	Clear objectives to develop computerized Information Systems	Clear objectives to develop computerized Information Systems	Clear objectives to develop computerized Information Systems	Improving the quality of working life and enhancing the job satisfaction of users	Improving the quality of working life and enhancing the job satisfaction of users
• Domain	Specific problem solving	Planning, organization and strategy type	Specific problem solving	Specific problem solving	Specific problem solving	Planning, organization and strategy type
• Target	General purpose Large organisations	General purpose Large organisations	General purpose (not very helpful for simple limited systems) Large organisations	General purpose Large organisations	Applicable in human activity situations	Applicable in human activity situations
Model	Data flow diagram	Process dependency diagram	Integration of both data and process orientation	Incremental development Iterative development Phases	Process orientation	Process orientation

Techniques and tools	Described in terms of techniques	Not fundamental part of methodology	Specific techniques fundamental to methodology	Specific techniques fundamental to methodology	Does not advocate any tools	Specific techniques fundamental to methodology
Scope	Feasibility Analysis Logical design Physical design	Feasibility Testing Implementation Evaluation Maintenance	Analysis Logical design Physical design Testing Implementation	Feasibility Analysis Logical design	Feasibility Logical design Physical design Implementation Evaluation	Feasibility Analysis Design
Outputs	Data flow diagrams Data structure diagram Data dictionary Decision trees Decision tables Structured English Normalization	Entity modelling Normalisation Entity life cycle Decision trees Decision tables Structured English Action diagrams Critical success factors	Object oriented UML Class diagram Use case diagram Interaction diagram Activity diagram Sequence diagram State chart diagram Pert diagram Gantt diagram	Functional decomposition Time boxing Pareto principle MoSCoW rules JAD work sessions Prototyping (Architectural Spike) User stories Paired programming	Role player analysis JAD (Joint application development)	Rich pictures Root definitions Conceptual models
Practice						
Background	Commercial	Commercial	Academic	Commercial	Academic	Academic
User base	n.a	n.a	n.a	n.a	n.a	n.a
Participants	Systems development is undertaken by professional technical developers	Systems development is undertaken by professional technical developers	Systems development is undertaken by professional technical developers	Systems development is undertaken by professional technical developers	Systems development is undertaken by users	Systems development is undertaken by professional technical developers
Product	n.a	n.a	n.a	n.a	n.a	n.a

According to Avison's (2006) framework for hard system methodologies, refers to the technical part of the system while soft systems methodologies refer to the theoretical and software of the system. I chose this framework because it consists of methodologies that are popular in both commercial and academic in the industry.

Even though some organizations avoid using them because of negativity, some still finish their projects successfully with the aid of methodologies, and what matters is how they apply these methodologies. Do organisations follow the steps or do they just do what they think is best for their product? The next section will discuss virtual teams and SDMs.

2.2.7 Summary

In this section, I discussed systems development methodologies, their history, their usage and effectiveness, their advantages and disadvantages are involved via some criticisms and finally categorisation and comparison of frameworks. Although there are disadvantages in the use of SDMs, their advantages and effectiveness are more valuable than the criticisms. In this regard, they should still be used continuously. In the next section, I will discuss virtual teams.

2.3 Virtual teams

There are various terms that are used to define virtual teams; namely multi-site, distributed and geographically dispersed team (GDT). Chudoba *et al.* (2005) explained that the literature related to virtual teams revealed a lack of depth in the definitions. Although virtual teamwork is a current topic in the literature on global organizations, it has been problematic to define what virtual means across multiple institutional contexts".

According to (Shin, 2005; Malhotra *et al.*, 2007; Ebrahim *et al*., 2009) a virtual team is defined as a group of people or team members who are scattered geographically, working as employees and using a combination of telecommunication and information technologies in order to achieve organizational goals and tasks. Casey and Richardson (2006) reported that the virtual team is the core building block of the virtual organization while a co-site team is a social group of individuals who are collocated and interdependent in their tasks. Virtual teams are distributed teams whose members are scattered across the globe through electronic information and using different communication technologies such as e-mail, video-conferencing and telephone (Hertel *et al*., 2005).

According to Samson and Daft (2003), teams are the main components that help organisations to perform and deliver better products. Organisations have been using teams to deliver their products until virtual teams emerged during the 1990s as a new process (Furst *et al.*, 2004). Today many organisations are building their teams made up of talented and qualified people around the world with the aim to meet customers' needs and be competitive (Kankanhalli *et al.*, 2007).

According to Lee-Kelley (2008), virtual teams as stated by Alge *et al.* (2003)

"are typified as geographically dispersed team members who communicate with each other using some variant mix of information and communication technologies. The temporary aspect of the team appears less emphasised, although Bal and Teo's (2000) synthesis of other prevailing definitions included temporary existence and cross-functional and inter-organisation collaborations". Global virtual teams are also known as virtual matrix-managed teams which are geographically dispersed and are becoming mutual in technology. They are culturally diverse and electronically communicating through workgroups (Daim *et al.*, 2012). On the other hand, Huang and Trauth (2007) mentioned virtual teams as global information systems development work that involve collaboration between two people or more organisations or one organisation and its subsidiaries across national boundaries.

According to Schlenkrich and Upfold (2009), virtual teams may be categorized into different types of virtual teams according to the characteristics that a particular team has. Teams, which possess more virtual characteristics, are placed higher on the range/scale of virtuality than teams with fewer virtual characteristics. This range is illustrated in the table below.

Table 2-2: Types of virtual teams and degree of virtuality ranging from high level to low level (Schlenkrich & Upfold, 2009).

Team Type	Virtual Characteristics
Action	This is where distance and organizational limits are usually crossed and teams get immediate response
Management	This is where national limits are regularly crossed whereas organizational limits are usually intact, teams are separated by distance and time
Service	Here teams rotate so that there is always one team in operation, distributed across time and distance
Work	Beginning of work where teams cross time and distance limits, teams perform regular and ongoing work
Project Development	Project development exists longer than parallel teams because projects are conducted for customers for a period of time, normally time, distance and organizational limits are often crossed
Parallel	Team members work on a short-term basis to address specific issues, they carry out special tasks that regular organizations are not equipped to perform, time distance and organizational limits are regularly crossed

Networked	Members rotate in relation to their needed expertise, they collaborate to achieve one goal, time, distance and organizational limits are crossed
Community of Practice	Here membership is voluntary and teams are learning as opposed to particular deliverables, they support people who are working on common tasks
Executive	Here teams are semi-permanent and responsible for specific functions in the organization, they consist of managers who are on the team because of their positions in the organization

According to McKinney and Whiteside (2006), "distributed teams face all the traditional challenges that any other team faces, and in addition to this, still have to deal with new communication challenges. In general, virtual teams that are higher on the range/scale of virtuality, possess a greater number of virtual characteristics, and will therefore be more sensitive to the social problems caused by these traits, than more traditional teams".

In this study, I define a *virtual team* as a group of people, separated locally or internationally, working to achieve a common goal. This goal could be achieved using various technologies such as telephones, emails, video conferencing etc. It could be achieved with the help of most qualified candidates from inside or outside of their companies who form or reform a team continuously. Organisations would also be able to deliver quality products and increase client's satisfaction and build a strong reporting relationship as team members have many people to report to.

2.3.1 Characteristics that are common to virtual teams

Below in table 2-3, are the characteristics of modern virtual teams as compared to the traditional teams. I wanted to know how these characteristics could help in today's software development.

Table 2-3: Characteristics which differentiate virtual teams from traditional teams (Schlenkrich & Upfold, 2009).

Traditional Characteristics	Virtual Characteristics
Fixed team membership	Shifting team membership
Members are from within the organization	Members can be from within the organization and outside the organization
Members are only part of one team	Members are part of multiple teams
Members are co-located organizationally and geographically	Members are distributed organizationally and geographically
Teams have a fixed starting and ending point	Teams form and reform continuously
Teams are managed by a single person	Teams have multiple reporting relationships with different people at different times

2.3.2 Effectiveness in virtual teams

According to Rice *et al.* (2007), the effectiveness of virtual teams has been determined by the adoption of formal procedures and structured processes. For a virtual team to be effective, social dimensional factors have to be considered before the teams can start working (Lin *et al.*, 2008). For the effectiveness in virtual teams, Shachaf and Hara (2005) suggested four dimensions of leadership, these being:

- Communication between the leader and team member which provide feedback, engage often in communication and clarifies tasks.
- Understanding each other on the team is also important, the leader must listen to team's members' opinions and suggestions, care about their problems, know the team very well and show interest in them.
- There must also be role clarity where the leader of the team clarifies roles and responsibilities of team members, exercise authority and mentors virtual team members.
- There must also be leadership attitude where the leader must be forward all the time and maintains a steady attitude over the life of the project.

The use of new advanced technology also contributes to the effectiveness of virtual teams (Mikkola *et al.*, 2005).

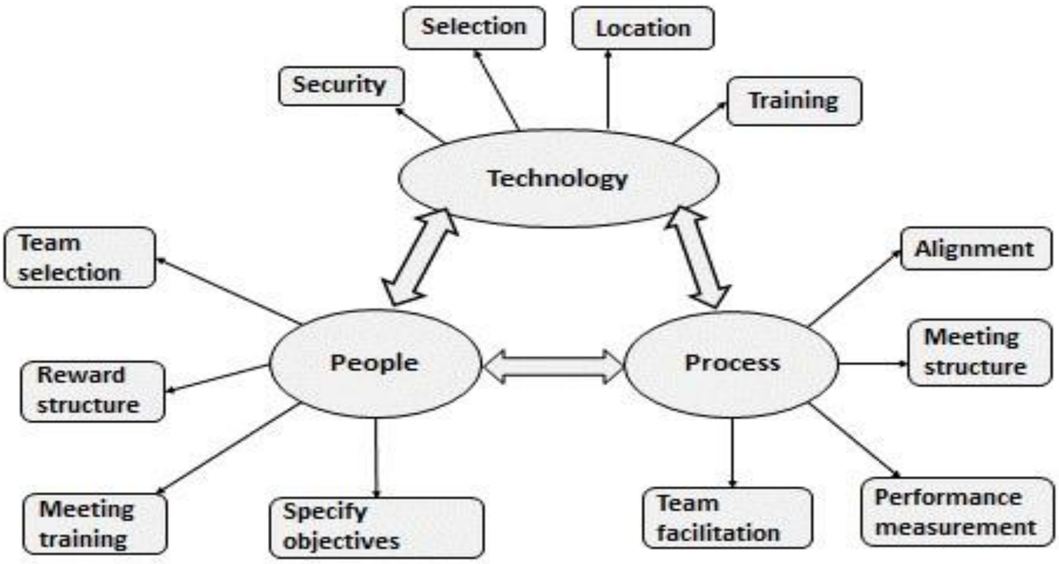


Figure 2-4: Interactive model for effective virtual team-working (adapted from Ebrahim *et al.*, 2009)

The above model was extracted in relation to the three main aspects, namely people, technology and process that can help in the successful of virtual teams when developing systems. Each aspect has its own four processes as shown in figure 2-1. Focus of early research found that inputs, socio-emotional process, outputs and task processes were the main factors that allowed virtual teams to work together until the end product (Powell *et al.*, 2004).

This model can accommodate most of the measures that I wanted to identify in this research. This includes, among others, performance of virtual teams in IS departments, SDMs used in virtual teams, methodology support provided as production technology, methodology support provided as control technology, methodology on the quality of the system developed, methodology support provided as cognitive and cooperative technology, methodology impact on the quality and the productivity of the development, challenges faced by virtual teams, meetings, communication, benefits of working in virtual teams, reliability of a developed system and the overall success of the project.

2.3.2.1 Technology point of view

2.3.2.1.1 Selection

A virtual environment by itself represents major challenges to effective communication (Ebrahim *et al.*, 2009; Walvoord *et al.*, 2008). The greatest obstacle to the effectiveness of virtual teams was to implement technology because information richness seemed to be the most important element for technology selection (Mikkola *et al.*, 2005). Dekker *et al.*, (2008) stated that virtual teams communicate through technology from different discipline to achieve a common goal, so the way that technology is implemented makes the virtual team outcome more or less the same (Anderson *et al.*, 2007).

2.3.2.1.2 Location

Regardless of the location, virtual teams allow organizations to access the most qualified individuals for a particular job around the world and also provide flexibility to those who are working from home or on the road (Ebrahim *et al.*, 2009).

2.3.2.1.3 Training

The results of Anderson *et al.* (2007)‟s systematic lab study confirm many of the observations include explicit preparation and training for virtual teams as a way of working collaboratively. Fuller *et al.* (2006b) results indicated that in the case of computer collective efficacy, computer training related to more advanced skills sets may be useful in building virtual team efficacy. Hertel *et al.* (2005) suggested that the training led to increased cohesiveness and team satisfaction.

2.3.2.1.4 Security

Special technologies and security levels needs of the virtual team should be identified by the team leaders (Hunsaker & Hunsaker, 2008).

2.3.2.2 People point of view

2.3.2.2.1 Team selection

Team selection is one of the key factors that differentiate successful teams from unsuccessful ones (Ebrahim *et al*., 2009). Hunsaker & Hunsaker (2008) stated that it is the responsibility of team leaders to make sure that project goals and objectives are clearly defined and to select members with the necessary skills to meet project targets.

2.3.2.2.2 Reward structure

At the beginning of the virtual teamwork, a fair and motivating reward system must be developed so that team members must put more effort into their tasks knowing that they will be rewarded for a job well done (Hertel *et al.*, 2005).

2.3.2.2.3 Meeting training

According to Ebrahim *et al.* (2009), training is an important aspect that cannot be neglected as it helps in building a team; virtual teams need some different types of training as compared to ordinary teams, such as self-managing skills, communication skills, meeting training, project management skills and technology training.

2.3.2.2.4 Specify objective

Virtual team leaders must identify familiarities of the team members earlier so that the team can achieve the objectives of the project and must also provide a clear context for recognizing team success (Ebrahim *et al*., 2009). More empowerment and delegation of managerial functions to the members are needed, so members of virtual team must be managed more effectively (Hertel *et al.*, 2005).

2.3.2.3 Process point of view

2.3.2.3.1 Alignment

Unlike in face-to-face teams, here processes of the organization must be re-aligned with the capabilities of the virtual teams (Ebrahim *et al.*, 2009). Moreover, Rosen *et al.* (2007) mentioned that hardware and software are not only the key elements in knowledge sharing but also the ability and willingness of team members to participate actively in the knowledge sharing process.

2.3.2.3.2 Meeting structure

Shin (2005) has argued that lack of physical interactions and informal relationships decreases the cohesiveness of virtual teams. It is a team leader's responsibility to implement a formal team structure as virtual teams rarely meet face-to-face (Ebrahim *et al.*, 2009). Synchronous written documents helped virtual teams overcome challenges associated with spoken language, and this enabled teams to overcome challenges associated with asynchronous (Shachaf, 2008). On the other hand Steinfield (2002) found video conferencing to be more difficult in virtual teams because of language differences. Instead, they preferred to use email for dividing tasks and gathering results at the end of the project. In contrast, Binder (2009) stated that the use of telephone and video conferencing are important as teams would not be able to travel more often for physical contact. The communication tool could help to show the high level of effectiveness among the teams. Moreover Chinbat (2010) also reported that communication in virtual teams ranged from telephone calls and faxes to more improved internet-based audio and video conferencing, email, groupware and so forth. Ásólfsdóttir (2012) mentioned that phone calls or Skype calls were the main mode of communication, which implied that some teams are still uninformed or unaware of information technology that must be used in virtual teams.

2.3.2.3.3 Performance measurement

Performance in virtual teams is differentiated by the respect for individual differences, passionate dedication to goals, a balance between unity and identification and emotional bonding among team members (Ebrahim *et al.*, 2009).

2.3.2.3.4 Team facilitation

Each virtual team member must have a clear role and responsibility, as lack of recognition can also cause team members to feel less accountable for the results, and as a result explicit facilitation of the teamwork takes on enhancement as important for virtual teams (Ebrahim *et al.*, 2009). Ebrahim (2015) stated that research on virtual teams is still in an early stage at the moment therefore setting up its infrastructure requires an effort. He further indicated that virtual teams deal effectively with a cultural context by designing and introducing new products for the specific market and collaborating with participants based locally. Communication technology improvements and continued globalisation have increased virtual teams worldwide. Currently the practice has moved away from working with people close by to working with people globally.

2.3.3 Benefits of Virtual Teams in Systems Development

Research on virtual teams is still new or emerging and many areas of research have not been studied (Badrinarayanan & Arnett, 2008). Zhang *et al.* (2008) stated that for the engineering projects to be successful in organizations there must be effective and efficient cooperation across distributed teams. El-Tayeh *et al.* (2008) suggest that more research on virtual teams is needed in order to investigate ways to improve performance of virtual teams. However, Hasab and Faraahi (2014) mentioned that there is an increasing interest in agile global software development as a mainstream in software industry.

Even though there are challenges in global software development, they mentioned that several studies have indicated cloud computing as a way to address challenges in agile global software development by adopting recent advances in virtualisation, software services and distributed computing. Those challenges still exist because virtual software development is not easily applicable in agile development, which might include temporal, geographical and language diversity. For the last two decades, distributed work has become much easier, faster and more efficient with the help of rapid development of electronic information and communication media (Hertel *et al.*, 2005).

Virtual teams are crucial for many organisations and industries to share skills and knowledge from across the world (Munkvold & Zigurs, 2007). Because of the competitive global economy many organizations face new challenges which necessitate some changes to organizational designs (such as team-based structures) which involves virtual teams (Shachaf, 2008). It has been shown that: "organizations are currently facing important and unprecedented challenges in an ever dynamic, constantly changing and complex environment" (Rezgui, 2007: 2653).

Few researchers have analysed the functionality of virtual teams in organisations even though some challenges still remain unsolved (Lira *et al.*, 2008). Holmstrom *et al.* (2006) for instance, considered agile systems development methodologies as being helpful to complex global software development projects, and recommended that agile systems development methodologies may be open to global software development as compared to previously used technologies. Hertel *et al.* (2005) stated that due to rapid development in new technology in terms of communication usage

such as internet, most organizations have deployed and adapted virtual teams. Because technology is improved today, many organizations seem not to be left behind, so they are also in the race to produce good quality products faster.
Virtual teams give organisations some ability to bridge time and space, as compared with the expenses and time travel by face-to-face teams, virtual teams are cost-efficient and provide a means for better utilization of distributed human resources (Berry, 2011). Shea *et al.* (2011) stated that virtual teams share some ideas and thoughts which in return promote competitive advantages in multinational companies.
Gibson and Gibbs (2006) developed characteristics in the form of four dimensions, a psychologically safe communication climate that can help reduce challenges posed. These four dimensions are geographic dispersion (space and time zones), electronic dependence which is a mix of face-to-face or no face-to-face, dynamic structure which is a more permanent teams versus an ad hoc team and finally national diversity which refers to the amount of different nationalities.
According to Horwitz *et al.* (2006), virtual teams are easily adaptable for specific work processes and people with different new skills can be positioned as required because of ever-changing business and customer requirements. Virtual teams can utilize 24 hours‟ work schedules with electronic communication because of different parts of the team they are working with around the world and also maintain an effective relationship with employees and customers at the same time enabling organisations to attain a distant geographic reach (Berry, 2011). Virtual teams are flexible and operate at a lower cost; they use improved resources that are necessary to meet changing needs and task requirements in the global business environment (Horwitz *et al.*, 2006). Hunsaker and Hunsaker (2008) mentioned that the adoption or use of virtual teams in organisations allows them to access the most and talented qualified individuals who can master their jobs regardless of the location or distance, thus enabling the organizations to respond faster to increased competition and therefore allowing flexibility wherever the employee is working from.
Sorli *et al.* (2006) emphasised that success in companies, especially manufacturing concerns, is influenced by lead time or time to market. In virtual teams, outcomes are normally stored electronically and automatically because of the easy review and performance and everyone can talk wherever they want because participation occurs at the same time instead of in a serial way as with face-to-face teams (Berry, 2011).

There are several advantages associated with virtual software development teams. The advantage of having virtual software development teams in organisations when building systems is to share skills and knowledge globally which can lead to some flexibility (Munkvold & Zigurs, 2007). Today, organizations are capable of rapidly creating teams of talented people who can respond to the needs of their customers and they are destined for success in the competitive and complex global economy of today (Kankanhalli *et al.*, 2007). Virtual teams are crucial for many organisations and industries to share the skills and knowledge across the world (Munkvold & Zigurs, 2007). It has been stated that due to rapid developments in new technology in terms of communication usage such as the internet, most organizations have deployed and adapted virtual teams (Hertel *et al.*, 2005). Shea *et al.* (2011) stated that virtual teams share some ideas and thoughts which in return promote competitive advantage in multinational companies. In addition, Hunsaker and Hunsaker (2008) mentioned that the adoption or use of virtual teams in organisations allows them to access the most qualified individuals who can master their jobs regardless of the location or distance. When analysing data, it was realized that most teams communicate through telephone and email which reduces travelling costs. The next section will discuss challenges faced by virtual teams.

2.3.4 Challenges faced by Virtual Teams

Distance causes problems of coordination and control negatively when communicating, which in turn affects software development globally (Carmel & Agarwal, 2001). There was inadequate and informal communication, between team members who work together but are separated by distance and time-zone differences (Sengupta *et al.*, 2006). In support, Rusman *et al.* (2010) stated that communication might not be spread equally in time due to time difference of zones. In addition, the team members often communicate regularly in the initial phase but seldom communicate towards deadlines.

Moreover Horwitz *et al.* (2006) mentioned that there are some risks associated with virtual teams being social dynamic risks which are associated with the building and sustaining of team and organizational commitment, trust, cross-cultural difference, member conflict, role doubt and complex decision-making can cause some difficulties in team relationship. In addition to common challenges encountered by traditional teams, virtual teams still have to deal with communication challenges as the teams operate remotely (McKinney & Whiteside, 2006).

Wrong or irregular information dissemination in virtual teams can cause re-work especially when common spoken language, normally the English language, may have slight differences in meaning (Sengupta *et al.*, 2006). At the beginning of the project, virtual team members tend to share little information as compared to face-to-face team members (Berry, 2011). Binder (2007) mentioned that main challenges include language, time zones, different locations, organizational and personal culture, policies and regulations of the organization, business processes, political climate, management skills and project leadership skills.

Based on a few previous studies, Casey (2010) identified trust as a key factor in the success or failure of virtual teams where cultural differences are even more visible. Kankanhalli *et al*. (2007) also stated that “there is much potential for conflict in virtual teams as members work across cultural, geographical and time-bound environments and this conflict leads to ineffective communication and as soon as team members stop communicating effectively, barriers begin to form between them leading to a decrease in productivity and interaction”. In virtual teams there is not enough trust, there are also communication breakdowns at some points, conflicts and power struggles (Rosen *et al.*, 2007; Baskerville & Nandhakumar, 2007).

Cultural and functional differences lead team members to think differently about processes. Also to develop a trust among virtual team members is very challenging (Jacobsa *et al.*, 2005; Kankanhalli *et al.*, 2006; Munkvold & Zigurs, 2007; Paul *et al.*, 2005; Poehler and Schumacher, 2007; Shachaf, 2005). According to Badrinarayanan and Arnett (2008) and Bergiel *et al. (*2008), sometimes virtual teams require complex technological applications. It is believed that virtual software development teams have to consider technology platforms that can suit all the teams before commencement of the project. Effective communication could reduce a lot of misunderstandings and reworks.

There are however, several challenges faced by virtual software development teams. Even though few researchers have analysed the functionality of virtual teams in organisations, some challenges still remained unresolved (Lira *et al.*, 2008). In addition, there is a cultural difference which makes each organisation unique and this can affect the overall team’s success. Functional differences between the teams can lead to group members with different knowledge bases, reasoning abilities and motivations (Daim *et al.*, 2012). For instance, developers may reason and react differently compared to business analysts. Organizational differences also require a unique culture and behaviour. For instance, Apple and IBM might make similar computers but their employees develop quite different values and behaviours (Daim *et al.*, 2012). An Apple employee would probably be uncomfortable at IBM, and vice versa. National differences are also influenced by the culture of the nationality (Daim *et al.*, 2012).

According to Rosen *et al.* (2007), there are also barriers to information and knowledge sharing which result in constraints on building trusting relationships, time constraints and deadline pressures, technology constraints and cultural constraint on knowledge sharing. It has been suggested that a key element for the success of any team-based project is the development of trust and cooperation (Casey, 2010). Furthermore he stated that technology on its own is not enough without trust and cultural differences can also lead to misunderstandings and misrepresentation of the actions and activities of remote colleagues (Casey, 2010).

In previous studies, temporal, geographical and socio-cultural distances gave rise to a number of global software development challenges (Conchúir *et al.*, 2006). According to studies of Badrinarayanan and Arnett (2008) and Bergiel *et al.* (2008), sometimes virtual teams require complex technological applications to overcome challenges. There is inadequate communication especially of an informal nature, between remote team members and time-zone differences (Sengupta *et al.*, 2006). Rusman *et al.* (2010) stated that communication might not be spread equally in time due to time difference of zones and team members often communicate regularly in the initial phase.

It has been stated that virtual software development increases the likelihood of an end product with defects because of the increased likelihood of defect dispersion, new virtual development related defect causes and causes already existing in co-located development are more likely to occur (Jacobs *et al.*, 2005). Changes in communication patterns and the lack of effective communication channels are recognised as cause of delays in global software development projects (Huang, 2007). Moreover, according to Daim *et al.* (2012) research has indicated that one of the major reasons for the failure of global virtual teams is related to building trust as it is essential to any global virtual teams. Trust allows people to engage in risk-associated activities that they cannot control or monitor. Thomas *et al.* (2007) also mentioned that with better technology facilitation, team members can spend more time enjoying what they do, and less time under stress and working late nights or weekends due to missed deadlines and failed virtual team interaction.

2.3.5 Virtual software development teams

According to Daim *et al.* (2012) virtual projects have become crucial. He further explained that we are living in the global world where various components of projects are handled in various locations. The main problems faced by virtual software development teams are new dimensions of technology and global economy and challenging for managers to manage their teams effectively. Virtual software development teams have become increasingly common in new product development and information systems. This was due to globalisation that was witnessed by different organisations and industries (Sarker & Sahay, 2004).

According to Ågerfalk *et al.* (2005), there are many reasons why an organization should consider adopting a global software development (GSD) model, including access to a larger labour pool and a broader skills base, cost advantage, and round the clock development. GSD is perhaps most evident in the many cases of outsourcing of software development to low-cost countries but is also relevant in the case of utilizing local expertise to satisfy local demands. This is perhaps where virtual software development teams fit in as a general benefit to organisations (Ågerfalk *et al.*, 2005). According to da Siva Estacio and Prikladnicki (2015), both pair programming and geographically distributed teams have created a form of distributed pair programming which works the same way as pair programming in terms of delivering quality product.

Traditionally, literature on GSD has focused on technical aspects (Kotlarsky & Oshri, 2005) and previous research suggests that proper application of technical and operational mechanisms such as collaborative technologies, IS development tools and coordination mechanisms are the key to successful system development projects (Carmel, 1999). A related stream of studies has focused on issues relating to the dispersion of work and the constraints associated with this. In these studies, constraints such as temporal distance, geographical distance and socio-cultural distance are identified, and while they indeed increase the scope of organizational operation (Sahay, 2003) and open up for a broader skill and product knowledge base (Baheti *et al.*, 2002), there is little doubt that these constraints challenge communication, coordination and control mechanisms (Herbsleb & Mockus, 2003; Damian, 2002).

In this study, virtual software development teams are regarded as a global team which develops systems in the hope of meeting customers' needs either locally or globally. They can work faster than traditional teams at the lower costs and with qualified teams around the world.

2.3.6 Summary

In this section, I discussed the literature on virtual teams from various researchers, characteristics that are common in virtual teams, effectiveness in virtual teams‟ challenges and finally virtual software development teams. Virtual teams seemed prone to face many challenges in the past decades, but because of the ever-changing technology and emerging tools, it makes it easier for the teams to deliver what was expected. The next section will deal with SDMs and virtual software development teams.

2.4 Systems development methodologies and virtual software development teams

Sakthivel (2005) mentioned systems development processes in virtual projects for which systems development approaches are categorized into structured, iterative and incremental and integrated ones. Moreover, the selection of the right development approach is necessary to address issues such as lack of a user's knowledge of application, lack of users' involvement, missing, incorrect, evolving requirements and risks associated with new technology and poor quality, the use of inappropriate development approach can have a risk of increasing development time and cost therefore putting the quality of the product at risk, virtual work with inappropriate development approach also can increase risks (Sakthivel, 2005).

According to Chinbat (2010), there are many problems associated with systems development by virtual teams due to lack of a formal planning phase, and he further stated that initial planning is more important towards the success of distributed projects because this is where many strategic decisions are made. Cusumano (2008) also stated that an adapted iterative model is good for distributed projects. It is evident that programming and managing these projects are very hard and even impossible without utilizing a methodology (Yaghini *et al.*, 2009). That means each of these methodologies uses a set of tools, models, and specific concepts, and thus, methodologies may be successful and applicable in some specific backgrounds of each organisation.

According to Avison (2006) and Yaghini *et al.* (2009) methodology comparison is done with the following purposes; academic purpose which refers to the better understanding of the nature of methodologies in order to improve information system development. This is a practical purpose of which selects one or more methodologies for specific applications. This includes philosophy, model used, tools and techniques applied for each methodology, scope of each methodology, outputs, practice applied and expected end product (Avison, 2006). A practical purpose refers to the selection of one or more methodologies for specific projects or a company as a whole taking into consideration the compatibility of the project scope with the methods (Yaghini *et al.*, 2009).

According to Peterson (2013), SDMs normally serve as a guideline to deliver systems on time, within budget and without failures. If organisations use SDMs, they will be able to provide better estimates of deliverables, deliver stable systems and always update customers about their product and progress (Peterson, 2013). The SDMs also help in identifying problems before they can occur and this helps developers with proactive management of their occurrence and developing a contingency plans during the process. According to Fitzgerald (1998), methodologies also facilitate project control and increase prominence into the development process. Holmstrom *et al.* (2006) for instance considered agile systems development methodologies as being helpful to complex global software development projects, and recommended that agile systems development methodologies may be suitable for global software development as compared to traditional technologies. Currently, there is some flexibility on the methodology and teams can work with each in the form of the pair-programming methods. Also, there is no huge amount of documentation as compared to other methodologies which thus safe time.

2.4.1 Previous studies in the use of SDMs by virtual teams

According to Guzman *et al.* (2010), software development methodologies that are normally used in global software development projects are: Rational Unified Process (RUP), Metrica Version 3, Structured Systems Analysis and Design Method (SSADM), Scrum or Project Management Body of Knowledge (PMBOK). In addition, in the past decade agile methodologies such as Extreme Programming (XP, Scrum, Dynamic Systems Development Methodology, Adaptive Programming and Crystal Clear) were also introduced in order to attain high quality software (Griffin, 2008). Even though some researchers have indicated that the above methodologies are not suitable for development if team members are scattered in different geographical areas in terms of trust and communication they can also lead to project success. Success could be achieved when tasks are distributed among different teams and locations. Not only that but also with each team member in the same place so that each member can provide feedback to the other team members where they belong (Guzman *et al.*, 2010).

Jaakkola *et al.* (2010) stated that agile methodology includes new methodologies which encourage teamwork in software development and give people a chance to share their opinions and views. This involves the culture which seems to be the main problem in virtual teams. For this reason, agile introduced new types of leadership culture which help in the flow of communication from the beginning of the project until the end of the project.

This new developments in agile methodologies could help in the sense that projects can be delivered in chunks and with less documentation. This could allow remote teams not to waste time but to deliver quality products based on what has been communicated to and the responsibilities or roles each one is holding.

2.4.2 Success of projects developed by virtual teams

Countries that were developing methodologies included the United Kingdom and Greek projects for Bank Co.'s migration which is how the virtual teams were created (Lee-Kelly & Sankey, 2008). These virtual teams were not exactly full time on this migration because they had other projects which they were involved in. Therefore, they were working on both projects successfully even though there were slight delays.

According to Kotlarsky and Oshri (2005), another project that was successful was that of SAP and LeCroy, where they based themselves on product success or a desired performance of a distributed team of which technical solutions, human-related issues in the form of social ties and knowledge sharing were reported as keys to successful collaboration. Data was collected from India and Germany for SAP, and Switzerland and USA for LeCroy, interviewees were chosen to include counterparts working closely from remote locations and diverse roles such as managers and developers of which five interviews were conducted for each company and made total of ten interviews (Kotlarsky & Oshri, 2005). According to Johnston and Rosin (2011), a global study found that 24% of all IT projects were successful while 53% of them had experienced problems, and 34% of all projects at Fortune 500 firms were successful.

2.5 Conclusion

The purpose of this chapter was to define the systems development methodology, comparison of systems development methodologies and its history, evolution and development. Advantages and disadvantages of systems development methodology were also discussed – both its effectiveness and some criticisms. Most of the participants in previous projects were not using any methodologies and some were using some techniques and tools. There is really a gap in the use of systems development methodologies by virtual teams. There is still a huge chance that projects developed by virtual software development teams can absolutely succeed. However, there must be a lot of knowledge sharing since cultures are different because this could help in building strong relationships with these vibrant teams.

The other part was to define the term virtual team and the combination of both SDMs and virtual software development teams. The summary of the sources that were used is presented in table 24 (Database). The measurement of how developers in virtual software development teams use methodology or not, was determined by statistical analysis in the next chapter. The next chapter will discuss the research methodology and design, thus how the research was conducted.

Table 2-4: Database searched on the background and use of system development methodologies, system development methodologies and Virtual Teams.

Keywords	**Google Scholar**	**EBSCO Host**	**IEEE**	**Scopus**	**Science Direct**	**Internet**
SDMs	1998, 2006, 2012:3, 2013	2006, 2005, 2008	2009		2006:2,2009	2005, 2013
Virtual teams & knowledge sharing	2008, 2007, 2006:2, 2009, 2015	2010:2	2007, 2006, 2010, 2011		2011,2007:2, 2009	2012
SDMs & virtual teams	2004		2011		2005	2009, 2005
Challenges & benefits by virtual teams	2008, 2009				2007,2011	2010:3
Agile methods	2006, 2014	2011			2008	2008:2, 2011
Cultural influence on virtual teams	2006	2011:2	2006		2010,2008	2009, 2007
Conflicts in virtual teams, anger & behaviour		2008, 2012			2005, 2011	
Effects, leadership, trust & communication on virtual teams	2010,2009,2007	2007	2008, 2010, 2006	2012	2010:3, 2009, 2012:2, 2007, 2006	2009:2, 2008
Effectiveness on virtual teams, performance and measurement	2007,2009	2011, 2010	2009	2008	2010	2010, 2011
Factors that affect success in virtual teams	2014	2010	2007	2008	2008	2010

CHAPTER 3: RESEARCH METHOD AND DESIGN

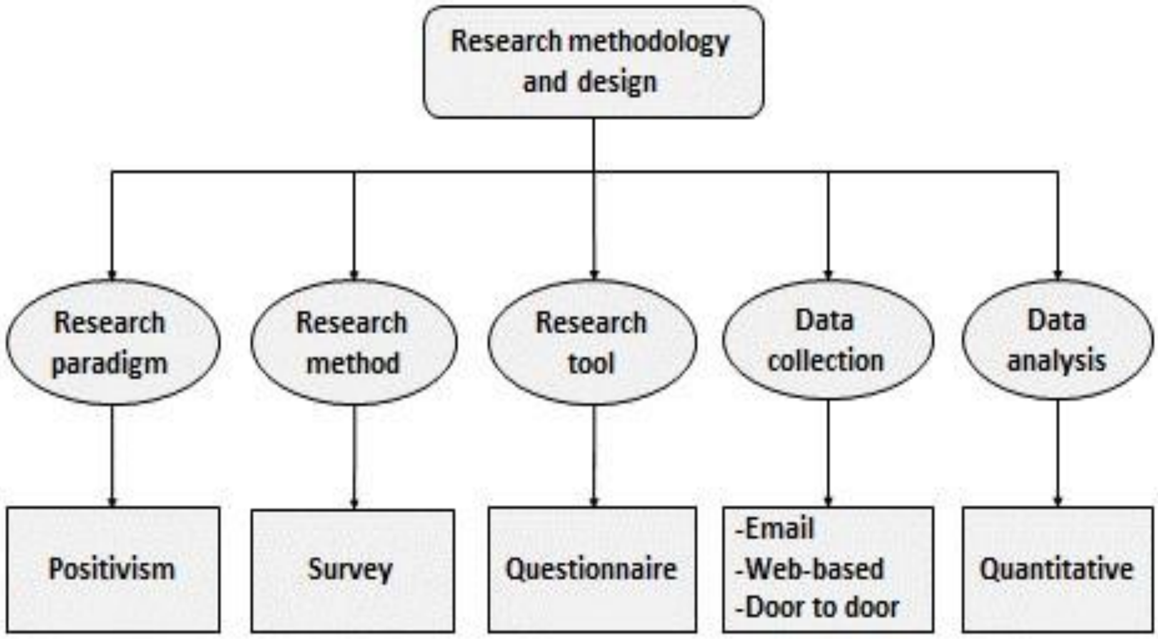


Figure 3-1: Outline of research methodology and design

3.1 Introduction

In the previous chapter, the literature review on systems development methodologies and virtual software development teams and a combination of both was presented. Problems and benefits faced by virtual teams, developers' perceptions and the success of virtual teams were considered. In this chapter, I will explain the theoretical background of the study

The positivistic research paradigm is discussed. I also further discuss research methods where a questionnaire survey was used. I further explain the questionnaire design, including how the questionnaire was developed. The questionnaire addresses three dimensions, namely background information, systems development methodologies and virtual software development teams will be discussed. I discuss the background information in detail in this section. The other two dimensions are discussed further in chapter 4 as they were analysed in the anticipation to meet the objectives stipulated in chapter 4. The data-collection method that was used when collecting data is explained. Lastly I discuss the data-analysis method which is how data was statistically analysed.

3.2 Research paradigm

Most of the sections in this chapter were referenced from Oates (2006).

3.2.1 Choices on research paradigm

There are some choices in research paradigm. Some research strategies are driven by a fixed research philosophy while for other strategies there is a choice. The choice of paradigm depends on the following:

- The nature of the research question.
- Researchers own personal beliefs and values, which shape how the researcher perceives our world and what kind of knowledge the researcher wants to create.
- Whether the researcher wants to do the kind of research that is typically done in his/her discipline.
- Whether the researcher is willing to take a risk and want to challenge the status quo. There are three main types of research paradigms, namely; positivism, interpretive and critical research paradigm. Positivism dominates in scientific research which is carried out in the natural sciences field (Oates, 2006). The interpretive research paradigm seeks to give an overall understanding of social context, beliefs, personal values, principles, preference and historical background (Darren & Lindsey, 2007). Critical research identifies power relations, conflicts, and contradictions, and authorizes people to serve as the main sources of alienation and domination to harmonize state of affairs (Oates, 2006).

In this study, I followed positivism because a survey was used where the research is based on empirical testing of theories. It is regarded as the oldest paradigm as it originates from the scientific method.

There has been a little discussion of the research underlying philosophy in computing literature. This paradigm evolved over the past 400 to 500 years and eventually became established because it took a long time to develop. That is why it is important to know the underlying philosophy if one wants to cite previous research based on strategies rather than design. Philosophy also helps to evaluate literature and assess the quality of the research.

Many researchers in computing do not think about their underlying research philosophy when creating their Information Technology artefacts. Researchers in computing feel comfortable with the positivist paradigm together with their strong scientific and mathematical background useful to them in performing some analyses.

3.2.2 Characteristics of positivism

A paradigm is a world view of which each of them has research methods which can be used to drive scientific investigation, and especially where experiments are not feasible, researchers turn to surveys.

3.2.2.1 The world exists independently of humans

In the real world the physical and social world exists, it is not in people's minds, not to be studied, captured or measured because it is something that is out there.

3.2.2.2 Measurement and modelling

This involves the explanation of any aspects of the world and means that by making observations and measurements one might fit in the model and then after that produce a model to see if it can prove or disprove anything if necessary.

3.2.2.3 Objectivity

Objectivity means a neutral mindset in the researcher as an impartial observer. Here, facts about the world can be found out or determined separately from the researcher's personal values and beliefs.

3.2.2.4 Hypothesis testing

The use of the mathematical approach provides an objective means of analysing observations and results. This is achieved through the use of mathematical modelling and proofs, and statistical analysis. Hypotheses are tested by comparing their predictions with observed data and observations that confirm a prediction. The study will not follow the hypothesis testing which is based on an attempt to prove or disapprove of what the actual results and the assumptions about that study. The use of systems development methodologies requires a clear understanding and procedures for a specific methodology to be followed.

3.2.2.5 Quantitative data analysis

Here the aim is to measure the relationship between an independent variable and a dependent variable in a population. It is often used in the positivist paradigm.

3.2.2.6 Universal laws

This is where one finds universal laws, patterns or some certain facts that seem to be true regardless of the researcher and the occasion. Universal laws are concerned with the patterns or indisputable facts that can be shown to be true.

Positivists argue that to give a causal explanation of an event means to presume a statement which describes it. Using premises of the presumption one or more universal laws, together with certain singular statements (Stable, 2011).

In order to make the theory more precise and clear, the positivists believe that the existing theoretical explorations should be true without any need for proof. The objective knowledge is possible when this knowledge developed by positivist approach (Stable, 2011).

3.2.3 Some criticism about positivism

Positivism is well suited in studying aspects of the natural world and less suited to researching the social world which involves the peoples world.

Sometimes it is difficult to break complex things down into simpler things that need to be studied or even worse one misses the whole picture. For instance, when studying at a certain university, I cannot specifically point to the rector of the university, but instead, I look at the holistic view of the whole university and all the complexities and relationships that affect the university as a whole.

Generalising is not always wanted, thus, the concentration on regular laws and generalizable patterns may miss out the bigger picture or portion of it.

This is a study of how individuals see the world and how they interpret it. Everyone sees the world differently. Some people will interpret any object the way they see it, not the way other people see it.

This is where people must see ways to study how groups, organizations, cultures and societies perceive their world and interpret it because regular laws and patterns might appear to be observable in the social world while they are the construction of people.

3.3 Research method (survey)

The Positivistic approach was used as research approach, the survey as a research method, the questionnaire as a means of data collection while statistical analysis was applied for data analysis.

This study followed a survey format as it collects information from large groups of people in a standardized way (Oates, 2006). According to Rea and Parker (2012), there are also types of survey for which information can be collected, I followed some of those methods being; mail out, web-based, telephone, in-person, and interviews.

According to Glasow (2005: 1), "Survey in research is used to answer questions that have been raised, to solve problems that have been posed or observed, to assess needs and set goals, to determine whether or not specific objectives have been met, to establish baselines against which future comparisons can be made, to analyse trends across time, and generally, to describe what exists, in what amount and in what content".

A survey is a tactical way of collecting information from different people using the population of choice or interest (Anonymous, 1999). A mail-out survey involves the distribution of printed questionnaires to participants where they will be asked to complete the questionnaire and return it to the researcher by mail (Rea & Parker, 2012). Its purpose is to collect information in a quantitative form through the use of a structured and standardized questionnaire (Anonymous, 1999).

3.3.1 Characteristics of a survey in research

- It is used to define or describe certain aspects of a particular population (Glasow, 2005).
- Data collected is subjective due to the fact that it is collected from people (Glasow, 2005).
- Not everyone can be used as part of the population, there is a part or portion of the population that can be used so that findings can be summarized based on that population (Glasow, 2005).

Surveys are strongly associated with positivism because they seek patterns in the world (Oates, 2006). The researcher using a survey has to assume that certain patterns exist. For instance, 80 per cent of government employees earns small salaries as compared to employees in private companies. Surveys usually use quantitative data and statistical analysis to reflect relationships among data (Oates, 2006). A survey cannot confirm the cause and effect like experiments do but can confirm associations even though they seem to be weaker than experiments (Oates, 2006).

3.3.2 Advantages and disadvantages of surveys

3.3.2.1 Advantages

Surveys can be done by mail, telephone, fax or face-to-face (Anonymous, 1999). There is a possible cost saving as communication through the mail is less expensive than telephone and interviews (Rea & Parker, 2012). Huge volumes of information can be collected in a short period of time in a survey and one can spend less time when analysing as compared to qualitative work (Anonymous, 1999).

Questionnaires can be completed at the respondents convenience and there is also enough time for participants to look at the questions and understand them before completing the questionnaire (Rea & Parker, 2012). A structured and standardized questionnaire reduces interviewer bias as the interviewee cannot be open to asking questions (Anonymous, 1999). According to Rea and Parker (2012), mail-out questionnaires can be more useful to participants than telephone survey as it can use photographs and maps for elaboration. A survey can complete structured questions with many people within a short time (Anonymous, 1999).

3.3.2.2 Disadvantages

It takes a long time as questionnaires require a few weeks before they can be returned to the researcher, the researcher has to make follow-ups in order to achieve an appropriate sample size (Rea & Parker, 2012). Sometimes it can be more expensive (Anonymous, 1999). Mail-outs normally have a lower response rate than telephone survey which can lead to incorrectness in the sample (Rea & Parker, 2012).

In order to interpret results, statistical knowledge and sampling are needed (Anonymous, 1999). According to Rea and Parker (2012), lack of involvement is also a problem as the researcher is not there to explain the questions that may not be understandable by the participants. Sometimes it is difficult to collect comprehensive understanding of participants as compared to interviews (Anonymous, 1999).

Data was collected using questionnaires. The survey was conducted through the use of questionnaires where participants were asked questions with multiple choice options and open-ended questions of which the collected data was analysed and interpreted at the later stage.

3.4 Research tool (Questionnaire)

A questionnaire is a set of questions arranged in an order, and which participants are expected to answer in order to provide the researcher with data which can be analysed and interpreted into meaning. They are normally associated with the survey research strategy. A questionnaire can be sent by e-mail, telephone, be web-based or delivered by hand to a sample of people. This group of people was being asked to complete the questions in that questionnaire and return it to the researcher. After collecting all the responses, data was then analysed using a statistical method, identified patterns and generalizations were made about the actions or views of a population.

Questionnaires can also be used in other strategies such as case studies, action research or design, and creation. Moreover, they can be self-administered or researcher-administered. This is where the former indicated that when participants complete the questionnaire without the researchers presence, the latter indicated that participants will be asked questions by the researcher of which the researcher will be jotting down the responses of all questions asked. The latter is a kind of structured interview and can also be face-to-face or telephonic.

In this study, questionnaires were distributed to participants and collected after few days. In some instances they were completed and returned at the same time, while others were distributed through the mail where many people did not respond while others were web-based and the response was much better than the email one.

Questionnaires are best suited to situations where the researcher:

- Wants to obtain data from a large number of people.
- Wants to obtain relatively brief and uncontroversial information from people.
- Needs to obtain standardized data, by posing similar questions to each participant and predefining the range of answers which can be given.
- Can expect the respondents to be able to read and understand the questions and possible answers.
- Has the money to pay for printing, distributing and collecting questionnaires and the time to wait between posing the questions and getting the responses back.

For this study, mail-out format, drop-off questionnaires and web-based ones were used for data collection. Mail-out involves the dissemination of printed questionnaires through the mail to participants. Participants were asked to complete the questionnaire and return it back to the researcher once completed within specific time.

3.4.1 Questionnaire design

There were three dimensions on which the questionnaire was built, that is, participants background information, systems development methodologies and virtual software development teams. In figure 3-2, I describe the three dimensions. Inside these dimensions, the information is clustered into measures in order to achieve the objectives. I show how the theory of chapter 3 was used to design a useful questionnaire that gives valid data. I focus on the virtual team profile, participant profile and organisational profile in this chapter. The rest is discussed in chapter 4.

The experimental approach (i.e. questionnaire) was designed so that the collected data could clarify three different dimensions, as illustrated in Figure 3-2 below. The aim of this process was to analyse the information gained from the data. The statistical analysis confirmed the validity of collected data. Trends were found that led us to achieve our objectives.

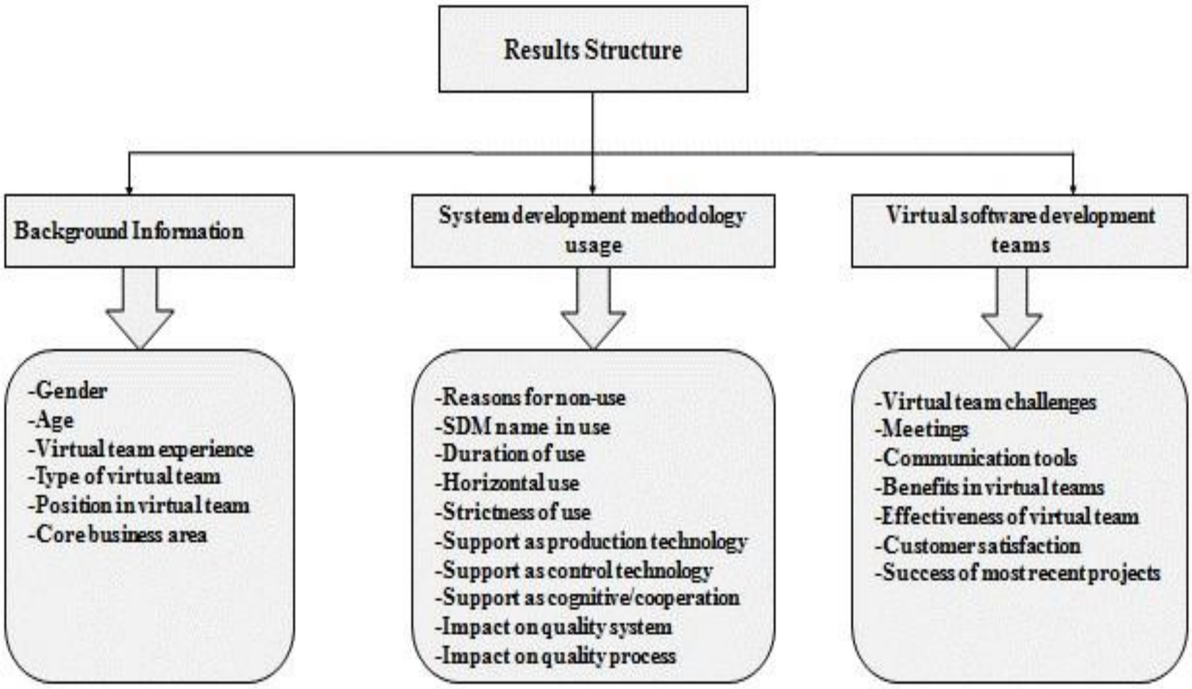


Figure 3-2: The structuring of the information collected during the questionnaire survey

Here we wanted to investigate the relationship between the systems development methodology used and the success of projects developed by virtual software development teams.

The design of the questionnaire included mainly closed-ended questions. This is where participants could select one from the list of possible answers provided in the questionnaire. Relevant questions were organised and presented in the order in which they could be easily followed and understood. The above three dimensions are presented in table 3-1. Literature was also used to assist on the suitable instruments that could be used to capture the data.

Table 3-1: Background information

Variables	Classification	Motivation	Reference
Gender	• Male • Female	Gender has an effect on the success of projects as only more male professionals are IT professionals.	Tesch *et al.* (2009)
Age	• Younger than 20 • 20-29, • 30-39, • 40-49 • 50 and above	Age was grouped in order for participants to select their correct age. This is important as age might play a major role in every project and could have an effect on the success of the project.	Nkone S 2013. Most respondents were males between 25-35 of age with 42% response, and females showed high percentage between the age of 36-45 with 10% response rate.
Virtual team experience	Any number of years was provided by participants	Participants were provided with space to fill in the years they have been working in virtual teams. This could have an influence on the success of the project.	
Virtual team type	• Multiple locations (Inside your country) • Multi-national (outside your country) • Combination (Inside and Outside your country)	It was important to ask participants the type of virtual team they were/are involved in as various teams have their own way of achieving successful projects depending on the type of team	
Position in the virtual team	• Manager • Software developer • Software architect • Team lead • Software engineer • Business analyst • Other roles	Participants were asked to provide us with their roles in order to see the different perspectives with regard to the systems development methodologies	Most respondents were Business analyst/programmer (Huisman & Iivari, 2005)

Core business area	• Administrative services • Finance/banking/insurance • Software house/consulting • Manufacturing • Retail/wholesale • Education • Other business areas	A core business area was asked to be provided as different sectors or industries have their own way of functioning and this could help in gap that has identified in the literature	Majority of respondents were from Manufacturing (Huisman & Iivari, 2005)
Number of employees in the organisation	• 1-4 • 5-50 • 51-200 • More than 200	The total number of employees was also asked as it could have an influence on the success of the projects.	Most respondents were from very large organisation with more than 200 employees (Huisman & Iivari, 2005)
Number of employees in the IS department	• 1-5 • 6-20 • 21-50 • More than 50	Participants were asked about the number of employees in the IS department as it could influence the outcome of the projects.	Mostly there were 6-20 employees in IS departments (Huisman, 2000)
Virtual team performance	• Speed of developing new applications is high • Functionality of new applications is high • Productivity of the application developers is high etc.	The performance of the virtual team was rated on a scale of 1-5 in order to measure the productivity of the team.	Huisman & Iivari (2005)

Table 3-2: Systems development methodology usage

Variables	Classification	Motivation	Reference
SDM name in use	• SDLC • STRADIS • YSM • IE • RAD • OOA&D • RUP • XP • Scrum • Other methodologies	Participants were asked to select more than one methodology they use/used in their virtual team. This could help achieve objective 4.	Huisman & Iivari (2005)
Duration of use	• Less than 1 year • 1 - 2 years • 3 - 5 years • 6 - 10 years • Over 10 years • Don't know	Duration of methodology use has an effect on the project outcome.	Huisman & Iivari (2005)
Horizontal use	• None • 1 – 25% • 26 – 50% • 51 – 75% • Over 75%	Participants were asked in terms of horizontal use in percentages how they use their methodologies. This could have an impact on the project success.	Huisman & Iivari (2005)
Strictness of use	• A general guideline for all projects • A standard which is followed rigorously for all projects • Adapted on a project-to-project basis	Participants were asked to select the strictness use of methodology. This could have an impact on the outcome of the project as a certain procedure has to be followed when developing a system.	Huisman & Iivari (2005)
Methodology support as production technology	• Our systems development methodology helps to align the system to be developed with the business • Our systems development methodology helps to capture requirements for the system to be developed etc.	Participants were asked to select in a scale of 1-5, methodology support as production technology. This measure was asked in order to achieve objective 3.	Huisman & Iivari (2005)
Methodology as support control technology	• Our systems development methodology helps to decompose the system to be developed in workable parts • Our systems development methodology helps to estimate the time and effort required for the development of a planned system etc.	Participants were asked to select in a scale of 1-5, methodology as support control technology. This measure was asked in order to achieve objective 3	Huisman & Iivari (2005)

Methodology as support cognitive and co-operative technology	• Our systems development methodology defines our desired systems development practice • Our systems development methodology describes a sound way of developing systems etc.	Participants were asked to select in a scale of 1-5, methodology as support cognitive and co-operative technology. This measure was asked in order to achieve objective 3	Huisman & Iivari (2005)
Methodology impact on quality system	• Our systems development methodology helps to develop more functional systems • Our systems development methodology helps to develop more reliable systems etc.	Participants were asked to select in a scale of 1-5, methodology impact on quality system. This measure was asked in order to achieve objective 3.	Huisman & Iivari (2005)
Methodology impact on quality and productivity of the development process	• Our systems development methodology helps to develop new applications faster • Our systems development methodology helps to improve the functionality of new applications etc.	Participants were asked to select in a scale of 1-5, methodology impact on quality and productivity of the development process. This measure was asked in order to achieve objective 3.	Huisman & Iivari (2005)
Expectations for the use of SDMs in virtual teams	• Make more use of our systems development methodology • Replace our systems development methodology • Supplement our systems development methodology with other methodologies • Abandon the use of our systems development methodology • No change • Other	Participants were asked to select in a scale of 1-5, Expectations for the use of SDMs in virtual teams. This measure was asked in order to achieve objective 3.	Huisman & Iivari (2005)

Table 3-3: Virtual software development teams

Variables	Classification	Motivation	Reference
Virtual team challenges	• Time zones • Language • Holidays, local laws, customs etc.	Participants were asked to select in a scale of 1-5 or provide with the challenges that were/are faced by their teams. This question was asked in order to achieve objective 1.	There is inadequate communication especially informal one, between remote team members and time-zone differences (Sengupta *et al.*, 2006) The main challenges include language, time zones,

			different locations, organizational and personal culture (Binder, 2007)
Meetings attended	• Never • Once a year • Twice a year • Three times a year • More than three times a year	Participants were asked to select the meetings they had with their teams. This question was asked in order to see how often teams that are engaged meet.	
Communication tools	• E-mail • Any virtual collaboration tool • Groupware • Voice over IP applications etc.	Participants were asked to select in a scale of 1-5. This question was asked in order to identify communication tools used by various virtual teams.	Virtual teams use different communication technologies such as e-mail, video-conferencing and telephone (Hertel *et al.*, 2005)
Virtual team benefits	• Ability to bridge time and space • Cost saving • Flexible to work at anytime and anywhere • Communication is flexible	Participants were asked to select in a scale of 1-5. This question was asked in order to achieve objective 1.	Virtual teams enable organisations to attain a distant geographic reach (Berry, 2011)
Virtual team effectiveness	• The tasks/work that we have already completed have less revisions/reworks • My virtual team is getting more and more work from our satisfied existing customers	Participants were asked to select in scale of 1-5. This question was asked in order to identify the effectiveness of virtual teams.	The effectiveness of virtual teams has been determined by the adoption of formal procedures and structured processes (Rice *et al.,* 2007) For a virtual team to be effective, social dimensional factors have to be considered before the teams can start working (Lin *et al.*, (2008). For the effectiveness in virtual teams, Shachaf and Hara (2005) suggested four dimensions of leadership:
Success of most recent projects	• Product • Process • Customer satisfaction • Leadership • Communication • Competencies • Other	Participants were asked to select on a scale of 1-5. This question was asked in order to achieve objective 3.	Leadership attitude throughout the project life cycle (Shachaf & Hara, 2005)

3.5 Data collection

The data was gathered from companies involved in the use of systems development methodology (SDM). Participants included managers, software developers, software architects, team leads, software engineers and business systems analysts, and other roles that were not mentioned in the questionnaire yet important to consider. A letter of request was sent out to participants with the questionnaire included. A total of 340 questionnaires were distributed with 200 of them through email, 80 questionnaires web-based while 60 questionnaires were door-to-door. Only 82 questionnaires were retrieved after being properly filled in by participants.

There were some difficulties in collecting data as some of the companies were not allowed to participate in the questionnaire survey while other participants just ignored the emails (e.g. by just opening and closing them without filling them in). In an attempt to resolve the problem of low participation, we took an initiative to distribute about 60 questionnaires in person in Lesotho and some in South Africa and waited for them to be filled in at the scene. This improved the success rate as all of them were completed.

Table 3-4: Response rate survey.

Distribution channel	Number distributed	Number returned	Response rate (%)
Online (Email)	200	7	4%
Web-based	80	17	21%
Door to door	60	58	97%

Most responses were from the door to door channel as compared to email and web-based responses.

3.5.1 Background information

The aim of this section was to examine the significance of each variable with regard to the independent variable. This process was done in order to identify the relationship between independent and dependent variables. It was important to present these variables as they might have an impact on the success of projects. Below I will discuss the participant's background information, virtual team background information and organisational background information.

3.5.1.1 Participants' background information

Gender

In this section, participants were asked to indicate their gender. I wanted to find out how many females or males are involved in virtual teams and use SDMs.

Table 3-5: Gender for participants working in virtual teams.

Gender		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	Male	57	69.5	70.4	70.4
	Female	24	29.3	29.6	100.0
	Total	81	98.8	100.0	
Missing	System	1	1.2		
Total		82	100.0		

In table 3-5, 57 respondents were male while 24 of them were female. The valid percentage of males who responded was 70% while female was 30%. This shows that the majority of respondents were males with 70.4% as compared to females, and furthermore I determined that there is a lack of IT female professionals in IT industry.

Age category

I wanted to know the age of participants who are involved in virtual teams: i.e. whether the majority are a young generation or an old one, which can help in decision-making on the use of SDMs.

Table 3-6: Age category of participants working in virtual teams.

Age Category	Frequency	Percent	Valid Percent	Cumulative Percent
20-29	2	37	45.1	45.1
30-39	3	36	43.9	89.0
40-49	4	9	11.0	100.0
	Total	82	100.0	

In table 3-6, 37 respondents were aged between 20-29 with a higher valid percentage of 45.1%, while 36 respondents were aged 30-39, with an average valid percentage of 43.9% and nine respondents were between the ages of 40-49 with a lower percentage of 11%. The majority of respondents (89%) were younger than 40.

3.5.1.2 Participants' level of experience in years

Here I wanted to find the level of experience of the participants in virtual teams.

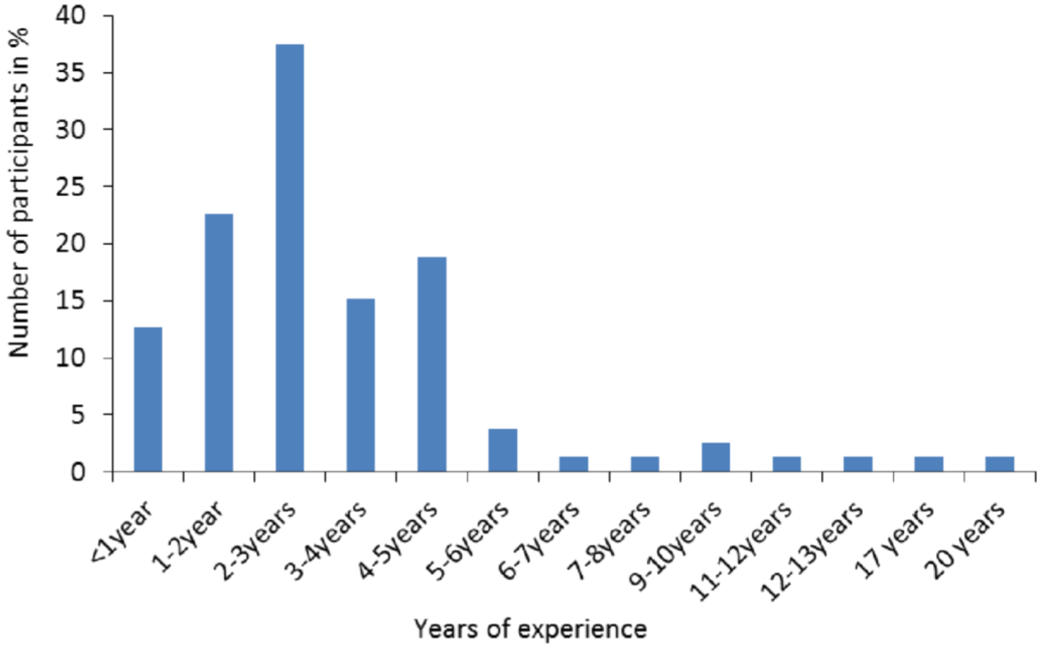


Figure 3-3: Years of experience in virtual teams

In figure 4-1, the majority of respondents had an experience of 2-3 years, with a valid percentage of 20%, followed by five years of experience with a valid percent of 16.3, followed by three years with a valid percentage of 15%, and finally, some had an experience of four years in virtual teams.

3.5.1.3 Participants' roles in their organisations

We wanted to know what role the participants fulfilled in the contribution to systems development.

Table 3-7: Role in the organisation

Roles in the Organisation	Frequency	Percentage	Valid Percentage	Cumulative Percentage
1 – Manager	9	11.0	11.0	11.0
2 - Software developer	30	36.6	36.6	47.6

3 - Software architect	4	4.9	4.9	52.4
4 - Team leader	5	6.1	6.1	58.5
5 - Software engineer	2	2.4	2.4	61.0
6 - Business Systems Analyst	25	30.5	30.5	91.5
7 – Other	7	8.5	8.5	100.0
Total	82	100.0	100.0	

Table 3-7 above, indicates roles played by people involved in systems development. Most participants were software developers with 36.6%, followed by business systems analysts with 30.5%, followed by managers with 11%, followed by other roles that were not mentioned in the questionnaire with 8%, followed by team leaders with 6.1%, followed by software architects with 4.9% and finally software engineers with 2.4%.

However, there are other positions that were mentioned by participants that were not mentioned in the questionnaire. These positions include Backup & Recovery Operator, Graphic & Usability Designer, IT Auditor, IT Help Desk, IT Help Desk Support, MIS Analyst and Software Specialist.

3.5.1.4 Virtual team background information

3.5.1.4.1 Virtual team type

I wanted to know the type of virtual team participants is involved in, whether multiple locations, multi-national or combination of both.

Table 3-8: Type of virtual team

Virtual Team Type	Frequency	Percentage	Valid Percentage	Cumulative Percentage
1 - Multiple locations (Inside your country)	66	80.5	80.5	80.5
2 - Multi-national (Outside your country)	2	2.4	2.4	82.9
3 - Combination (Inside and Outside your country)	14	17.1	17.1	100.0
Total	82	100.0	100.0	

In table 3-8 above, the location has been classified according to type, lots of respondents are involved in virtual teams inside their countries which contribute to 80.5% while only 2.4% are in multinationals and 17.1% work inside and outside their countries. Based on the results, most participants involved in virtual teams were working within their countries.

3.5.1.4.2 Virtual Team Performance

I wanted to measure the performance. Below are the items used to measure the perceived performance of IS department. In this section, participants were asked to agree or disagree on the listed items that contributed to the performance of the system build within their organisations.

Table 3-9: Virtual team performance

	Totally Disagree	Disagree	Neutral	Agree	Totally Agree	Mean	STD DEV.
Scale	**1**	**2**	**3**	**4**	**5**		
Speed of developing new applications is high.	-	11%	45%	28%	16%	3.49	0.892
Functionality of new applications is high.	-	5%	43%	37%	16%	3.63	0.809
Productivity of the application developers is high.	-	9%	39%	29%	23%	3.67	0.930
Cost of systems development is high.	6%	12%	29%	35%	17%	3.45	1.102
Quality of systems development products is high.	-	5%	32%	44%	20%	3.78	0.817
Cost of systems maintenance is high.	1%	13%	27%	40%	18%	3.61	0.978
Documentation of systems development products is of high quality.	1%	12%	26%	42%	20%	3.66	0.971
The morale in my virtual team is high.	-	4%	29%	50%	17%	3.80	0.761
My virtual team achieves its goals well.	-	2%	34%	43%	21%	3.82	0.788
My virtual team has a reputation of excellent work.	1%	1%	28%	51%	18%	3.84	0.777

3.5.1.5 Organisational background information

3.5.1.5.1 Business area of the organisation

I wanted to identify the types of organisations or business areas in which participants were/are involved.

Table 3-10: Core business area of the organisation

Business area of the Organisation	Frequency	Percentage e	Valid percentage	Cumulative percentage
1 - Administrative services	2	2.4	2.4	2.4
2 - Finance/Banking/Insurance	27	32.9	32.9	35.4
3 - Software house/Software consulting	16	19.5	19.5	54.9
4 - Manufacturing	-	-	-	-
5 - Retail/Wholesale	1	1.2	1.2	56.1
6 - Education	1	1.2	1.2	57.3
7 - Other	35	42.7	42.7	100.0
Total	82	100.0	100.0	

Table 3-10 shows the different business areas/sectors of the participants‟ organizations. The Education sector is the area with the higher valid percentage of 42.7 followed by Finance/Banking/Insurance with 32.9%, followed by Software house/Software consulting with 19.5% while the other two sectors; Retail/Wholesale, Administrative services, and Administrative services were slightly represented in the sample.

However, there were other business areas that were mentioned but were not in the questionnaire but were also involved in virtual teams and development. These business areas include Design & Print (1.2%), Health and Wellness (1.2%), Public transport and works (1.2%), Lesotho Revenue Authority (8.5%), Telecommunications South Africa (25.6%), Travel & Leisure and Vodacom Lesotho (3.7%).

3.5.1.5.2 Total number of employees in the organisation

The researcher wanted to know the number of employees in the organisation, as it is an indication of the size of the organization.

Table 3-11: Size of the organisation

Size of the Organisation	Frequency	Percentage	Valid percentage	Cumulative percentage
1 – (1-4)	3	3.7	3.7	3.7
2 – (5-50)	18	22.0	22.0	25.6
3 – (50-200)	5	6.1	6.1	31.7
4 – (More than 200)	56	68.3	68.3	100.0
Total	82	100.0	100.0	

Table 3-11 above shows that 68.3% of participants were from large companies, 22% from small companies and 6.1% from medium companies while 3.7% very small companies. This shows that large companies are more involved in virtual teams.

3.5.1.5.3 Total number of employees in IS department

We wanted to know the total number of employees in the Information Systems department.

Table 3-12: Number of employees in IS department

Number of employees in IS department	Frequency	Percent	Valid percent	Cumulative percent
1 -1-5	8	9.8	9.8	9.8
2 -6-20	16	19.5	19.5	29.3
3 -20-50	18	22.0	22.0	51.2
4 -More than 50	40	48.8	48.8	100.0
Total	82	100.0	100.0	

Table 3-12 above, shows the number of employees in IS departments with 48.8% of IS employees in large companies, followed by medium companies with 22.0% of IS employees, followed by small companies with 19.5% of IS employees and finally the very small companies with 9.8% of IS employees.

3.6 Data analysis

This section entails how data collected through experiments and surveys was analysed. It is normally based on numbers, it is primarily used and analysed by positivist researchers and in some rare cases, used by interpretive and critical researchers too (Oates, 2006). Quantitative data means data or evidence, based on numbers (Oates, 2006). The main idea of data analysis is to look for patterns in the data so that conclusions can be drawn from that data (Oates, 2006). There are different kinds of techniques used for analysing data in quantitative, namely; tables, charts or graphs of which they make it easier for the researcher or the reader to see some patterns (Oates, 2006).

The researchers have to be clear about what statistical test is he/she going to use, and what kinds of quantitative data they require before they can even start data generation (Oates, 2006).

3.6.1 Statistical data analysis

There are various types of statistical analysis that are used to analyse data being: Descriptive, reliability, correlations, regression and factor analysis.

Descriptive analysis

Descriptive statistics is where I have numbers of participants, minimum, maximum, mean and standard deviations for each question. It is always important to provide one with the maximum and minimum effect size. The most important index of variability of effect sizes is their standard deviation (Rosenthall, 1991). Normally frequency is used here for the discrete variable.

Reliability analysis

Here I used Cronbach Alpha which is the measure of scale reliability. I looked at the highest number which is nearest to 1 which shows whether it is significant or not. The value of 0.7-0.8 is the accepted Cronbach's alpha. Here the data set is split into two and the average of these two values is Cronbach Alpha (Field, 2005).

Correlations analyses

Correlations normally refer to relationships between the datasets for different variables. That means more than two variable can have a relationship while other not. All the correlations are statistically significant at the 5% level of significance (Steyn, 2002). Furthermore, in order to have a clear understanding of the relationship guideline values below:

NB: W = effect size

W = 0.1 small effect

W = 0.3 medium effect

W = 0.5 large effect

Regression analysis

This also involves investigating relationships among/between variables. This means to discover the effect of certain variables upon others (Sykes, 1999). Factor analysis

Factor analysis is the way of taking data and shrinking it to a smaller data set so that it can be manageable and understandable. It is a way of finding hidden patterns, show how those patterns overlap and what characteristics are seen in multiple patterns (Andale, 2014).

3.7 Conclusions

This chapter covered the research paradigm, research method, questionnaire survey, questionnaire design, data collection and data analysis. I also discussed the three dimensions being the background information, systems development methodologies usage and virtual software development teams. It is important to know the research paradigm and method in order to use the correct tool when collecting data for the better analysis. In the next chapter, I discuss data analysis and results gathered from the questionnaires.

CHAPTER 4: RESEARCH RESULTS AND ANALYSIS

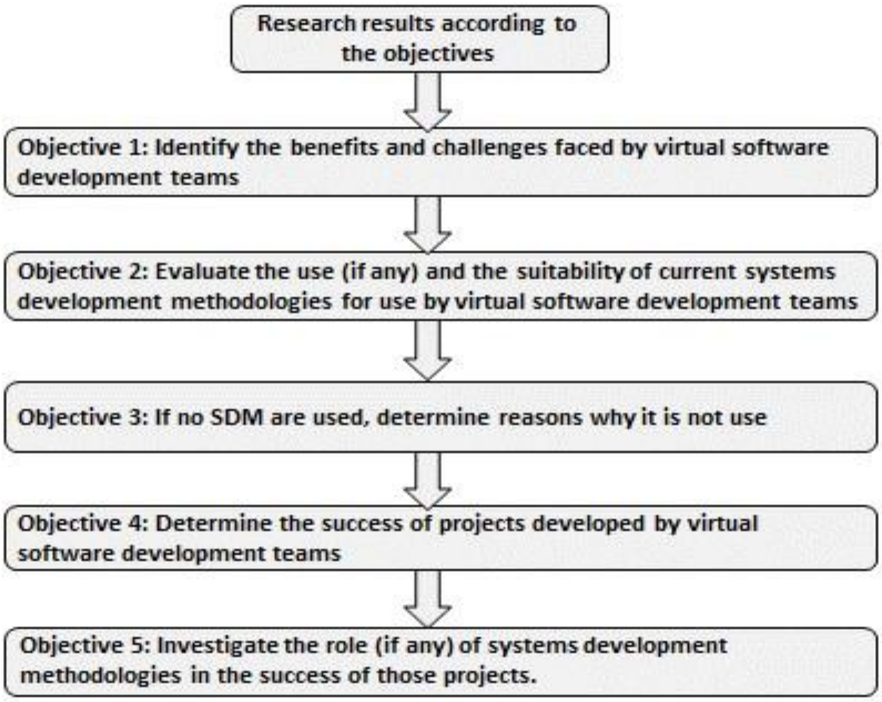


Figure 4-1: The structure of research findings according to study objectives

4.1 Introduction

The aim of this study was to investigate whether and how organisations are using systems development methodologies (SDMs) in their virtual software development teams. To address this aim, the following objectives were formulated in chapter 1.

1. To identify challenges and benefits faced by virtual software development teams.
2. To evaluate the use (if any) and the suitability of current systems development methodologies for use by virtual software development teams.
3. If no systems development methodologies are used, determine the reason why it is not used.
4. To determine the success of projects developed by virtual software development teams.
5. To investigate the role (if any) of systems development methodologies in the success of those projects.

In order to reach these objectives, data was gathered using a questionnaire as stipulated in chapter 3. The data was analysed and the results will be presented in this chapter.

4.2 Objective 1: Identify the benefits and challenges faced by virtual software development teams.

The first objective of this study was to identify both the benefits and challenges faced by virtual software development teams. To reach this objective I performed frequency analysis, factor analysis, and reliability analysis to analyse the data. The frequencies above the average value were considered to be important while the ones below the average were considered less important.

4.2.1 Virtual software development team benefits

To measure the benefits of working in a virtual software development team, ten items that were derived in the literature study were included in the questionnaire. Respondents were asked to evaluate these items based on their experience while working on a virtual team on a scale from 1 (totally disagree) to 5 (totally agree). The results are presented in Table 4.1. These results are sorted according to the mean value for each Item.

Table 4-1: Benefits experienced while working in a virtual software development team.

	Totally Disagree	Disagree	Neutral	Agree	Totally Agree	Mean	STD DEV.
Item in questionnaire	**1**	**2**	**3**	**4**	**5**		
Outcomes are normally stored electronically	-	4%	18%	33%	45%	4.20	0.867
Flexible to work at anytime and anywhere	2%	2%	16%	46%	33%	4.05	0.901
Ability to bridge time and space	1%	6%	21%	35%	37%	4.00	0.969
Easy review and performance	1%	6%	20%	38%	35%	4.00	0.956
Enables organisations to attain a distant geographic reach	1%	4%	29%	33%	33%	3.93	0.940
Maintains an effective relationship with employees and customers	1%	6%	22%	46%	24%	3.87	0.899
Communication is flexible	2%	6%	22%	43%	27%	3.85	0.970
Cost saving	2%	7%	17%	50%	23%	3.84	0.949
Bring under control the talent and knowledge to solve problems	-	10%	31%	35%	24%	3.74	0.940
Tasks are completed on a 24/7 schedule	4%	11%	31%	29%	26%	3.62	1.096

Table 4-1 shows that the highest mean is 4.20 for *"Outcomes are normally stored electronically"* and the lowest mean is 3.62 for *"Tasks are completed on a 24/7 schedule".* All the means are greater than 3, which indicates that the respondents agree that they experience these benefits while working in a virtual team.

In order to investigate the perceived benefits of working in a virtual software development team further, I performed a factor analysis and reliability analysis. The results are depicted in Table 4-2.

Table 4-2: Factor loading items used to measure benefits of working in a virtual software development team.

Cronbach's Alpha	Factor Name	Items in questionnaire	Factor Loading
F1 **0.910**	Perceived benefits of working in a virtual software development team	Maintains an effective relationship with employees and customers	0.805
		Easy review and performance	0.798
		Bring under control the talent and knowledge to solve problems	0.793
		Tasks are completed on a 24/7 schedule	0.778
		Flexible to work at anytime and anywhere	0.768
		Enables organisations to attain a distant geographic reach	0.768
		Cost saving	0.723
		Ability to bridge time and space	0.693
		Outcomes are normally stored electronically	0.692
		Communication is flexible	0.635

Only one component was extracted. The highest factor loading was for the item "*Maintains an effective relationship with employees and customers"* with a factor loading of 0.805. To check the reliability of the items included in the component, Cronbach Alpha was performed. This resulted in a value of 0.910 which proves the validity of the items grouped in the component.

4.2.2 Virtual software development teams challenges

I also wanted to determine the challenges faced by virtual teams. To measure the challenges of working in a virtual software development teams, eight items that were compiled in the literature study were included in the questionnaire. Respondents were asked to evaluate these items based on their experience while working on a virtual team on a scale from 1 (totally disagree) to 5 (totally agree). The results are presented in Table 4.3. These results are sorted according to the mean value for each item. The results are shown table 4.3.

Table 4-3: Challenges faced by virtual software development teams.

	Totally Disagree	**Disagree**	**Neutral**	**Agree**	**Totally Agree**	**Mean**	**STD DEV.**
Scale	**1**	**2**	**3**	**4**	**5**		
Technology	16%	12%	22%	29%	21%	3.27	1.352
Culture: Organizational difference	12%	18%	28%	15%	27%	3.26	1.359
Holidays, local laws, customs	13%	11%	32%	29%	15%	3.21	1.225
Personal conflicts and misunderstanding	10%	15%	33%	31%	12%	3.21	1.141
Distance	16%	20%	17%	28%	20%	3.16	1.374
Culture: National differences	21%	21%	15%	17%	27%	3.09	1.517
Language	21%	13%	22%	29%	15%	3.04	1.365
Time zones	48%	11%	11%	17%	13%	2.38	1.537

The highest mean is 3.27 for *"Technology"*, closely followed by Culture: Organisational difference with a mean of 3.26. The lowest mean is 2.38 *"Time zones"* with the standard deviation of 1.537. Most of the means are greater than 3, which indicates that the respondents agreed that they experienced challenges while working in a virtual team. The only mean less than 3 was for the item "Time zones", which means that respondents don't perceive this item as a challenge while working on a virtual software development team.

In order to investigate the perceived challenges of working in a virtual software development team further, we performed a factor analysis and reliability analysis. The results are depicted in Table 4-4.

Table 4-4: Factor loading items used to measure challenges of working in a virtual software development teams.

Cronbach Alpha	Factor Name	Variables	Factor Loading 1	Factor Loading 2
F1 0.845	Culture and values	Culture: National differences	0.821	
		Culture: Organizational differences	0.816	
		Personal conflicts and misunderstanding	0.722	
		Technology	0.722	-0.258
		Language	0.687	0.432
		Holidays, local laws, customs	0.603	0.421
		Distance	0.592	-0.237
		Time zones		0.881

The above factor analysis resulted in two factors: Factor 2 consists of only one item. To check the reliability of the items included in Factor 1, Cronbach Alpha was performed. This resulted in a value of 0.845 which proves the validity of the items measured.

Objective 2: Evaluate the use (if any) and the suitability of current systems development methodologies for use by virtual software development teams.

The second objective of this study was to evaluate the use (if any) and the suitability of current systems development methodologies for use by virtual software development teams. To reach this objective I performed frequency analysis, factor analysis, and reliability analysis to analyse the data.

To investigate the use of systems development methodologies I focused on the following dimensions of use:

- Type of SDM used
- Vertical use (intensity of use) of SDM
- Time of use
- Horizontal use of SDM (in terms of number of people who apply SDM regularly and the number of projects that are developed by applying SDM knowledge)
- Strictness of use

To investigate the suitability of systems development methodologies I focused on the following dimensions:

- The support provided by SDM as production technology in virtual teams.
- The support provided by SDM as control technology in virtual teams.
- The support provided by SDM as cognitive/co-operative technology in virtual teams.
- The impact of SDM on the quality of systems development by virtual teams.
- The impact of SDM on the quality of the development process followed by virtual teams.

These aspects of use will be discussed next.

Systems development methodology usage (vertical use)

Vertical use is concerned with how intensively systems development methodologies are used. I defined systems development methodologies as the totality of systems development approaches, process models, specific methods and specific techniques. To measure vertical use, I focus on the systems development methodologies and participants were provided with a list of nine commercial systems development methodologies, and they were asked to indicate to what extent does the IS department is using these methods (if any) on a scale from 1 (nominally) to 5 (extensively). The results are presented in Table 4-5. These results are sorted according to the mean value for each of the SDM.

Table 4-5: Type of systems development methodology used, and the intensity of use.

	Nominally	Not Nominal	Average	Regularly	Intensively	Mean	STD DEV.
Scale	**1**	**2**	**3**	**4**	**5**		
SDLC (Systems development life cycle)	3%	-	15%	42%	40%	4.17	0.874
RAD (Rapid Application Development)	19%	5%	11%	24%	41%	3.63	1.527
Scrum	12%	4%	19%	42%	23%	3.60	1.237
XP (Extreme Programming)	15%	10%	12%	41%	23%	3.48	1.349
OOA&D (Object Oriented Analysis and Design by Coad and Yourdon)	39%	7%	10%	31%	13%	2.73	1.550
STRADIS (Structured Analysis, Design and Implementation of Information Systems)	44%	8%	13%	22%	13%	2.51	1.544
RUP (Rational Unified Process)	53%	15%	10%	17%	5%	2.05	1.333
IE (Information Engineering)	56%	14%	10%	14%	5%	1.98	1.316
YSM (Yourdon Systems Method)	69%	9%	18%	2%	2%	1.58	0.975
Other, please specify	-	-	-	-	-	4.13	0.354

The highest mean is 4.17 *"SDLC"*, followed by "*RAD*", "*Scrum*" and "*XP*". The lowest mean is 1.58 for *"YSM"*. However, there are other methodologies that were mentioned by participants which were not included in the questionnaire, namely Kanban and Magic. Only 7% of participants used Kanban and 2% used Magic.

Table 4-6: Cross-tabulation on modern and traditional methodologies to measure what combination of SDMs are used per company.

			Modern		Total
			.00	1.00	
Traditional	.00	Count	7	5	12
		% within Traditional	58,3%	41,7%	100,0%
		% within Modern	41,2%	7,7%	14,6%
		% of Total	8,5%	6,1%	14,6%
	1.00	Count	10	60	70
		% within Traditional	14,3%	85,7%	100,0%
		% within Modern	58,8%	92,3%	85,4%
		% of Total	12,2%	73,2%	85,4%
Total		Count	17	65	82
		% within Traditional	20,7%	79,3%	100,0%
		% within Modern	100,0%	100,0%	100,0%
		% of Total	20,7%	79,3%	100,0%

In table 4-6 above, there are 58.3% of organisations that do not use either traditional or modern methodologies while 41.7% of companies are using modern methodologies. There are only 14.3% of organisations not using methodologies and 85.7% of organisations using both modern and traditional methodologies.

Some organisations used different types of SDMs; others used only one while some did not follow any methodology at all. The aim was to identify the importance and advantages of primitive, new methodologies and non-use of SDMs.

Table 4-7: Correlations of the participants from one company in terms of Gender.

Variables	Male	Female	STD. Dev.	P-Value	Effect size
Product	4.05	4.46	0.466	0.02	0.59

Process	3.98	4.21	0.518	0.20	0.31
Customer satisfaction	3.98	4.36	0.725	0.10	0.59
Leadership	3.79	4.33	0.744	0.11	0.63
Communication	3.97	4.21	0.758	0.25	0.28
Competencies	4.05	4.33	0.458	0.09	0.41

Small effect: d = 0.2, (b) Medium effect: d = 0.5 and (c) Large effect: d = 0.8

The results indicated that virtual teams use both the traditional SDLC and agile systems development methodologies. The product has a large effect of 0.6 with regard to gender. The process has a medium effect of 0.3 with regard to gender, Leadership has a large effect of 0.6 with regard to gender and communication has a medium effect of 0.3 with regard to gender.

4.3 Objective 2: Evaluate the use (if any) and the suitability of current systems development methodologies for use by virtual software development teams.

The second objective was to evaluate the suitability of systems development methodologies currently used by virtual software development teams. In order to achieve these objectives, I performed frequency analysis and descriptive analysis. The frequencies above the average value were considered to be important while the ones below the average were considered less important.

4.3.1 SDMs usage in years by virtual software development teams.

Here I wanted to know how long the virtual team has been using SDM.

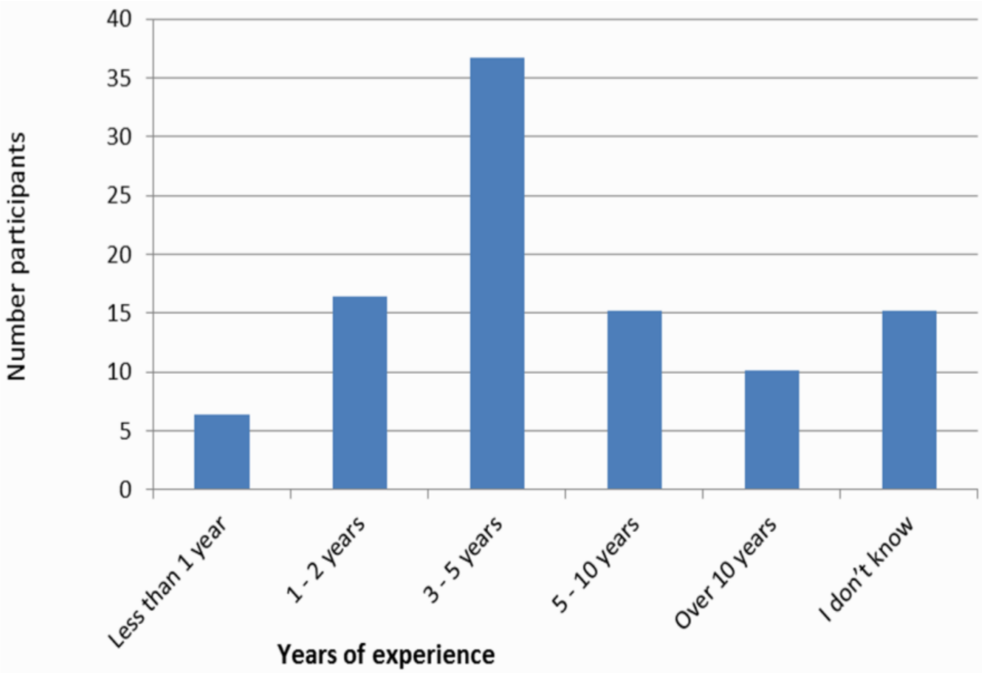


Figure 4-2: Methodology usage in years by virtual software development teams.

The figure 4-2 showed the number of years that virtual teams have been involved in systems development methodologies, 37% of participants stated that their teams have been using systems methodologies for 3-5 years, followed by 17% for 1-2 years, followed by 15% of participants for 5-10 years, followed by 10% of participants for over 10 years, followed by 65% of participant for less than a year and the other 15% of participants don't know for how long their methodology have been in use.

4.3.2 Methodology usage by virtual teams in proportion of projects developed (horizontal use)

This defines how widely systems development methodologies are being used by virtual software development teams. This usage was divided into two categories, namely the proportion of projects developed in the IS department by applying systems development methodology knowledge, and the proportion of people in the IS department that apply systems development knowledge regularly.

Table 4-8: Methodology usage by virtual software development teams in the proportion of projects developed as part of horizontal use.

Methodology usage	Frequency	Percent	Valid percent	Cumulative percent
1 –None	-	-	-	-

2 -1-25%	11	13.4	13.8	13.8
3 -26-50%	15	18.3	18.8	32.5
4 -51-75%	39	47.6	48.8	81.3
5 -Over 75%	15	18.3	18.8	100.0
Total	80	97.6	100.0	
Missing	2	2.4		
Total	82	100.0		

Table 4-8 shows the proportion of projects that have been developed by virtual teams applying systems development methodology knowledge, 48.8% of the respondents indicated that between 51 -75% of their projects are being developed by applying SDM knowledge, 18.8% of participants applied systems development methodology in 75% of their projects, another 18.8% of participants applied their methodology in 26%-50% of their projects while 13.8% of participants only applied it in 1% -25% of their projects.

4.3.3 Methodology usage by virtual teams in proportion of people who apply systems development methodology knowledge regularly (horizontal use)

Table 4-9: Methodology usage by virtual software development teams in the proportion of people who apply systems development methodology knowledge regularly as part of horizontal use.

Methodology usage	Frequency	Percent	Valid percent	Cumulative e percent
1 –None	1	1.2	1.3	1.3
2 -1-25%	11	13.4	13.8	15.0
3 -26-50%	20	24.4	25.0	40.0
4 -51-75%	24	29.3	30.0	70.0
5 -Over 75%	24	29.3	30.0	100.0
Total	80	97.6	100.0	
Missing (System)	2	2.4		
Total	82	100.0		

The above table 4-7 showed that 1.3% were not using any methodology, 14% of participants used them in a proportion of 1% -25%, 25% of participants used them in a proportion of 26%50%, 30% of virtual team applied systems development regularly in a proportion of 51-75% and another 30% in a proportion over 75%.

4.3.4 Strictness of SDMs usage

In order to determine how strictly SDMs are used by virtual teams, three options were presented to the participants: a general guideline, a rigorous standard, or adapted on a project-to-project basis

Table 4-10: Methodology in strictness of use.

Procedure		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	A general guideline for all projects	26	31.7	32.9	32.9
	A standard which is followed rigorously for all projects	25	30.5	31.6	64.6
	Adapted on a project-to-project basis	28	34.1	35.4	100.0
	Total	79	96.3	100.0	
Missing	System	3	3.7		
Total		82	100.0		

Table 4-10 shows that 35% of virtual teams adopted SDM on a project-to-project basis while 33% of participants used a general guideline for all projects and 32% of participants used a standard procedure which is followed thoroughly for all projects.

4.3.5 Support provided by systems development methodology as a production technology in virtual teams

To measure the support provided by SDM as production technology in a virtual software development teams, eleven items that were included in the questionnaire. Respondents were asked to evaluate these items based on their experience while working on a virtual team on a scale from 1 (totally disagree) to 5 (totally agree). The results are presented in Table 4-11.

Table 4-11: Support provided by system development methodology as production technology in virtual software development teams.

	Totally Disagree	Disagree	Neutral	Agree	Totally Agree	Mean	STD DEV.
Scale	**1**	**2**	**3**	**4**	**5**		
Our systems development methodology helps to get the systems accepted.	1%	-	21%	39%	39%	4.14	0.838
Our systems development methodology helps to involve end users in systems development projects.	-	5%	18%	41%	36%	4.09	0.860

Our systems development methodology helps in system design.	-	3%	15%	55%	28%	4.08	0.725
Our systems development methodology helps to build management commitment in our systems development projects.	-	1%	25%	44%	30%	4.03	0.779
Our systems development methodology helps to align the system to be developed with the business.	-	3%	18%	56%	24%	4.01	0.720
Our systems development methodology helps to capture requirements for the system to be developed.	1%	-	24%	53%	23%	3.95	0.761
Our systems development methodology helps in implementing developed systems.	3%	4%	20%	46%	28%	3.93	0.925
Our systems development methodology helps to reuse earlier requirements, designs and code during systems development.	3%	3%	26%	41%	28%	3.89	0.928
Our systems development methodology helps to design the architecture of the system to be developed.	1%	1%	28%	50%	20%	3.86	0.791
Our systems development methodology helps in testing developed systems.	3%	4%	33%	36%	25%	3.78	0.954
Our systems development methodology helps in reviewing developed systems.	1%	5%	34%	38%	23%	3.75	0.907

Table 4-11 shows that all the means for the individual items were greater than 3, which means that the participants agree that SDM provides support as a production technology in their virtual teams. The highest mean is 4.14: *"Our systems development methodology helps to get the systems accepted"* And the lowest mean is 3.75 *"Our systems development methodology helps in reviewing developed systems"*.

In order to investigate this further, factor analysis was performed. The results are presented in Table 4-12.

Table 4-12: Results of factor analysis for support provided by system development methodology as support production technology in virtual software development teams.

Cronbach Alpha	Factor Name	Variables	Factor Loading
F1 **0.921**	SDM support as production technology	Our systems development methodology helps to capture requirements for the system to be developed.	0.859
		Our systems development methodology helps to align the system to be developed with the business.	0.813
		Our systems development methodology helps to get the systems accepted.	0.804
		Our systems development methodology helps to design the architecture of the system to be developed.	0.800
		Our systems development methodology helps in system design.	0.781
		Our systems development methodology helps to build management commitment in our systems development projects.	0.753
		Our systems development methodology helps to reuse earlier requirements, designs and code during systems development.	0.733
		Our systems development methodology helps in reviewing developed systems.	0.723
		Our systems development methodology helps in testing developed systems.	0.717
		Our systems development methodology helps in implementing developed systems.	0.670
		Our systems development methodology helps to involve end-users in systems development projects.	0.640

Only 1 component was extracted. To check the reliability of the items, Cronbach Alpha was used, which was 0.921 and which then proves the validity of the items measured.

4.3.6 Support provided by SDM as control technology in virtual teams

To measure the support provided by SDM as control technology in a virtual software development teams, nine items that were included in the questionnaire. Respondents were asked to evaluate these items based on their experience while working on a virtual team on a scale from 1 (totally disagree) to 5 (totally agree). The results are presented in Table 4-13.

Table 4-13: Support provided by system development methodology as control technology in virtual software development teams.

	Totally Disagree	Disagree	Neutral	Agree	Totally Agree	Mean	STD DEV.
Scale	**1**	**2**	**3**	**4**	**5**		
Overall, our systems development methodology helps us to manage our systems development projects.	-	5%	15%	51%	29%	4.04	0.803
Our systems development methodology helps to organize systems development projects.	-	5%	18%	55%	23%	3.95	0.778
Our systems development methodology helps to keep our systems development projects under control in different locations.	1%	6%	16%	58%	19%	3.86	0.838
Our systems development methodology helps to estimate the project risks.	-	6%	21%	53%	20%	3.86	0.807
Our systems development methodology helps to estimate the time and effort required for the development of a planned system.	1%	3%	29%	48%	20%	3.83	0.823
Our systems development methodology helps in defining useful milestones for our systems development projects.	-	1%	38%	39%	23%	3.83	0.792
Our systems development methodology helps to estimate the size of the system to be developed.	-	6%	21%	58%%	15%	3.81	0.765
Our systems development methodology helps to plan systems development projects.	1%	4%	35%	38%	23%	3.76	0.889
Our systems development methodology helps to decompose the system to be developed in workable parts.	3%	4%	21%	61%	11%	3.75	0.803

Table 4-13 shows that all the means for the individual items were greater than 3, which means that the participants agree that SDM provides support as control technology in their virtual work. The highest mean is 4.04 *"Overall, our systems development methodology helps us to manage our systems development projects* and the lowest mean is 3.75 *"Our systems development methodology helps to decompose the system to be developed in workable parts".*

In order to investigate this further, factor analysis was performed. The results are presented in Table 4-14.

Table 4-14: Results of factor analysis for support provided by system development methodology as control technology in virtual software development teams.

Cronbach Alpha	**Factor Name**	**Variables**	**Factor Loading**
F1 **0.922**	Support, planning and measuring,	Our systems development methodology helps to organize systems development projects.	0.835
		Overall, our systems development methodology helps us to manage our systems development projects.	0.824
		Our systems development methodology helps to keep our systems development projects under control in different locations.	0.818
		Our systems development methodology helps to plan systems development projects.	0.806
		Our systems development methodology helps to estimate the time and effort required for the development of a planned system.	0.804
		Our systems development methodology helps to estimate the project risks.	0.784
		Our systems development methodology helps to estimate the size of the system to be developed	0.772
		Our systems development methodology helps in defining useful milestones for our systems development projects.	0.726

		Our systems development methodology helps to decompose the system to be developed in workable parts.	0.693

Only one component was extracted. To check the reliability of the items, Cronbach Alpha was used, which resulted in a value of 0.922 which proves the validity of the items measured.

4.3.7 Perceived SD methodology impact on the quality of the system developed by virtual teams.

To measure the perceived impact of SDM on the quality of the system developed by virtual software development teams, eight items were included in the questionnaire. Respondents were asked to evaluate these items based on their experience while working on virtual teams on a scale from 1 (totally disagree) to 5 (totally agree). The results are presented in Table 4-15.

Table 4-15: Impact on the quality of the developed system.

	Totally Disagree	**Disagree**	**Neutral**	**Agree**	**Totally Agree**	**Mean**	**STD DEV.**
Scale	**1**	**2**	**3**	**4**	**5**		
Overall, our systems development methodology helps to develop better systems	-	4%	14%	56%	26%	4.05	0.745
Our systems development methodology helps to develop more usable systems	1%	4%	20%	49%	26%	3.95	0.855
Overall, our systems development methodology helps to make users more satisfied with our systems	-	3%	33%	33%	33%	3.95	0.870
Our systems development methodology helps to develop more functional systems.	-	3%	26%	54%	18%	3.86	0.725
Our systems development methodology helps to develop more reliable systems.	-	3%	33%	48%	18%	3.80	0.753
Our systems development methodology helps to develop more maintainable systems.	-	6%	26%	50%	18%	3.79	0.807

Our systems development methodology helps to develop more efficient systems	1%	4%	26%	54%	15%	3.78	0.795
Our systems development methodology helps to develop more portable systems.	-	10%	29%	49%	13%	3.64	0.830

Table 4-15 shows that all the means for the individual items were greater than 3, which means that the participants agree that SDM has a positive impact on the quality of the systems developed by virtual teams. The highest mean is 4.05 *"Overall, our systems development methodology helps to develop better systems"* And the lowest mean is 3.64 *"Our systems development methodology helps to develop more portable systems"*.

To analyse this further, factor analysis was performed in order to identify patterns that are hidden in the data set. The results are presented in Table 4-16.

Table 4-16: Results of factor analysis for the impact of SDMs on the quality of the systems developed by virtual software development teams.

Cronbach Alpha	Factor Name	Variables	Factor Loading
F1 **0.901**	Methodology impacts on the quality of the developed system	Our systems development methodology helps to develop more reliable systems.	0.840
		Overall, our systems development methodology helps to develop better systems.	0.827
		Our systems development methodology helps to develop more functional systems.	0.820
		Our systems development methodology helps to develop more efficient systems.	0.781
		Our systems development methodology helps to develop more maintainable systems.	0.773
		Overall, our systems development methodology helps to make users more satisfied with our systems.	0.770

		Our systems development methodology helps to develop more usable systems.	0.718
		Our systems development methodology helps to develop more portable systems.	0.646

Only one component was extracted. To check the reliability of the items, Cronbach Alpha was used, of which it was 0.901 which proves the validity of the items measured.

4.3.8 Support provided by SDM as cognitive and co-operative technology in virtual teams.

To measure the perceived support provided by SDM as cognitive and co-operative technology in virtual software development teams, eleven items were included in the questionnaire. Respondents were asked to evaluate these items based on their experience while working on a virtual team on a scale from 1 (totally disagree) to 5 (totally agree). The results are presented in Table 4-17.

Table 4-17: Support provided by SDMs as cognitive and cooperative technology in virtual software development teams.

	Totally Disagree	Disagree	Neutral	Agree	Totally Agree	Mean	STD DEV.
Scale	**1**	**2**	**3**	**4**	**5**		
Our systems development methodology allows us to learn from our systems development experience	1%	1%	14%	59%	25%	4.05	0.745
Our systems development methodology provides useful guidelines for conducting systems development	1%	3%	16%	59%	21%	3.96	0.770
Our systems development methodology describes a sound way of developing systems.	1%	4%	18%	60%	18%	3.89	0.779
Our systems development methodology forms a useful standard for our systems development	1%	-	29%	51%	19%	3.86	0.759
Our systems development methodology defines our desired systems development practice.	3%	1%	25%	53%	19%	3.84	0.834
Our systems development methodology reminds me about activities/tasks of systems development	-	4%	28%	53%	16%	3.81	0.748
Our systems development methodology provides a useful list of possible systems development activities	-	4%	29%	53%	15%	3.79	0.741
Our systems development methodology provides a useful toolbox of techniques to be applied	3%	5%	38%	26%	29%	3.74	1.016
Without a systems development methodology one cannot estimate how systems development should be conducted	1%	8%	30%	45%	16%	3.68	0.883

Our systems development methodology defines an ideal process of systems development that is useful, even though it is not followed	1%	8%	30%	46%	15%	3.66	0.875
in practice							
Without a systems development methodology it is impossible to evaluate our systems development practice	4%	10%	36%	26%	24%	3.56	1.077

Table 4-17 shows that all the means for the individual items were greater than 3, which means that the participants agreed that SDM provides support as cognitive and co-operation technology in virtual teams. The highest mean is 4.05 *"Our systems development methodology allows us to learn from our systems development experience"* The lowest mean is 3.56 *"Without a systems development methodology it is impossible to evaluate our systems development practice".*

In order to investigate this further, factor analysis was performed. The results are presented in Table 4-18.

Table 4-18: Results of factor analysis for support provided by SDMs as cognitive and cooperative technology in virtual software development teams.

Cronbach Alpha	Factor Name	Variables	Factor Loading 1	Factor Loading 2
F1 **0.903**	Tools, techniques, guidelines, standards and tasks/activities	Our systems development methodology provides a useful tool-box of techniques to be applied	0.863	
		Our systems development methodology describes a sound way of developing systems.	0.813	
		Our systems development methodology defines our desired systems development practice.	0.794	
		Our systems development methodology forms a useful standard for our systems development	0.784	
		Our systems development methodology provides useful guidelines for conducting systems development	0.780	
		Our systems development methodology reminds me about activities/tasks of systems development	0.754	
		Our systems development methodology provides a useful list of possible systems development activities	0.739	
F2 **0.744**	Schedule, evaluation, experience, and process	Without a systems development methodology, one cannot estimate how systems development should be conducted		0.917
		Without a systems development methodology, it is impossible to evaluate our systems development practice		0.834
		Our systems development methodology allows us to learn from our systems development experience	0.260	0.547
		Our systems development methodology defines an ideal process of systems development that is useful, even though it is not followed in practice	0.286	0.471

Two components were extracted. To check the reliability of the construct items, Cronbach Alpha was used, which resulted in a value of 0.903 for factor 1 and 0.744 for factor 2, which proves the validity of the items measured.

4.3.9 Impact of SDMs on the quality and the productivity of the development process

To measure the perceived impact of SDM on the quality and the productivity of the development process followed by virtual software development teams, ten items were included in the questionnaire. Respondents were asked to evaluate these items based on their experience while working on a virtual team on a scale from 1 (totally disagree) to 5 (totally agree). The results are presented in Table 4-19.

Table 4-19: Impact on the quality and the productivity of the development process.

	Totally Disagree	Disagree	Neutral	Agree	Totally Agree	Mean	STD DEV.
Scale	**1**	**2**	**3**	**4**	**5**		
Our systems development methodology helps to improve the quality of the systems	-	3%	29%	44%	25%	3.91	0.799
Our systems development methodology helps to achieve the goals of my virtual team	1%	4%	20%	53%	23%	3.91	0.830
Our systems development methodology helps to improve the functionality of new applications	1%	3%	20%	59%	18%	3.89	0.763
Our systems development methodology helps to improve my virtual team‟s reputation of excellent work.	-	4%	25%	53%	19%	3.86	0.759
Our systems development methodology improves the morale in my virtual team	1%	4%	24%	51%	20%	3.85	0.828
Our systems development methodology helps to increase the productivity of the application developers.	1%	5%	24%	49%	21%	3.84	0.863
Our systems development methodology helps to develop new applications faster.	-	4%	29%	49%	19%	3.83	0.776
Our systems development methodology helps to improve the documentation of the systems	1%	6%	28%	41%	24%	3.80	0.920
Our systems development methodology helps to decrease the cost of systems maintenance	1%	8%	24%	50%	18%	3.75	0.879
Our systems development methodology helps to decrease the cost of systems development	3%	8%	30%	48%	13%	3.60	0.894

Table 4-19 shows that all the means for the individual items were greater than 3, which means that the participants agree that SDM has a positive impact on the quality and the productivity of the development process followed by virtual software development teams. The highest mean is

3.91 for *"Our systems development methodology helps to improve the quality of the systems"* and "*Our systems development methodology helps to achieve the goals of my virtual team"* The lowest mean is 3.60 *"Our systems development methodology helps to decrease the cost of systems development".*

In order to investigate this further, factor analysis was performed. The results are presented in Table 4-20.

Table 4-20: Results of factor analysis for the impact of SDMs on the quality and productivity of the development process followed by virtual software development teams.

Cronbach Alpha	Factor Name	Variables	Factor Loading
F1 **0.929**	Methodology impacts on the quality and the productivity of the development process	Our systems development methodology improves the morale in my virtual team.	0.836
		Our systems development methodology helps to increase the productivity of the application developers.	0.817
		Our systems development methodology helps to achieve the goals of my virtual team.	0.796
		Our systems development methodology helps to improve my virtual team"s reputation of excellent work.	0.795
		Our systems development methodology helps to improve the quality of the systems.	0.792
		Our systems development methodology helps to improve the functionality of new applications.	0.777
		Our systems development methodology helps to develop new applications faster.	0.774
		Our systems development methodology helps to improve the documentation of the systems.	0.759
		Our systems development methodology helps to decrease the cost of systems maintenance.	0.752
		Our systems development methodology helps to decrease the cost of systems development	0.742

Only one component was extracted. To check the reliability of the items, Cronbach Alpha was used, which resulted in a value of 0.929 which proves the validity of the items measured.

4.4 Objective 3: If no SDM are used, determine reasons why it is not used

Here only 1% of participants responded, indicating that they support an application that is built while the other said they outsource services, that means they buy methodology from vendors instead of developing it themselves.

4.5 Objective 4: Determine the success of projects developed by virtual software development teams.

The fourth objective of this study was to determine the success of projects developed by virtual software development teams. To achieve this objective we performed frequency analysis, factor analysis, and reliability analysis to analyse the data.

To measure the success of projects developed by virtual teams, six items were included in the questionnaire. Respondents were asked to evaluate these items based on their experience while working on a virtual team on a scale from 1 (totally disagree) to 5 (totally agree). The results are presented in Table 4-21.

Table 4-21: Success of projects developed by virtual software development teams.

	Totally Disagree	**Disagree**	**Neutral**	**Agree**	**Totally Agree**	**Mean**	**STD DEV.**
Scale	**1**	**2**	**3**	**4**	**5**		
Customer satisfaction	-	1%	15%	43%	42%	4.24	0.746
Product	-	-	17%	49%	34%	4.17	0.699
Competencies	-	-	17%	51%	32%	4.15	0.687
Process	-	4%	12%	60%	24%	4.05	0.718
Communication	-	6%	17%	44%	33%	4.04	0.867
Leadership	1%	6%	16%	50%	27%	3.95	0.888

Table 4-21 shows that all the means (except for one) for the individual items were greater than 4, which means that the participants feel strongly that their projects are very successful. The highest mean is 4.24 for *“Customer satisfaction”*. The lowest mean is 3.95 for *“Leadership”.*

To analyse this further, factor analysis was performed. The results are presented in Table 4-22.

Table 4-22: Results of factor analysis for the success of projects developed by virtual software development teams.

Cronbach Alpha	Factor Name	Variables	Factor Loading
F1 **0.898**	Success of projects developed by virtual teams	Customer satisfaction	0.877
		Product	0.874
		Competencies	0.826
		Process	0.799
		Communication	0.770
		Leadership	0.743

Only 1 component was extracted. To check the reliability of the items, Cronbach Alpha was used, which resulted in a value of 0.898 which proves the validity of the items measured.

4.6 Objective 5: Investigate the role (if any) of systems development methodologies in the success of those projects

The fifth objective of this study was to investigate the role of systems development methodologies in the success of those projects. The regression analysis was used to see the significance of the independent variables towards a dependent variable.

4.6.1 Regression analysis

Table 4-23: Results of the regression analysis predicting the success of the product.

Variables	Models			Effect size
Factor name	**1**	**2**	**3**	
Virtual team performance		0.279**	0.296***	
Support-production technology				

Support provided control technology	0.571***	0.420***	0.393***	0.44
Impact quality of the developed system				
Support provided cognitive/cooperation technology				
Impact on the quality and the productivity of the development process				
Gender			-0.239**	
Strictness of use				
R	0.571	0.618	0.662	
R square	0.326	0.381	0.438	
Adjusted R square	0.317	0.365	0.415	
F	37.261***	23.426***	19.461***	

$p \leq 0.1$, $*p \leq 0.05$, $**p \leq 0.01$, $***p \leq 0.001$

Stepwise regression analysis was used to determine factors that influence the overall success of recent projects developed. The first column shows a list of all possible variables that might have an effect on the product as the dependent variable. The second column contains the results of the first step performed during the stepwise regression; the third column contains the second step performed during the stepwise regression in the order in which the results were entered. The last four rows of terms of regression consist of the significance of the independent variables towards a dependent variable (R, R-square, adjusted R square and F). A large effect size of 0.44 on support provided as control technology by systems development methodologies indicated that it may influence the success of the product.

Table 4-24: The regression results for the factor representing the success of the project developed by the virtual team using coefficients.

Model		Unstandardized Coefficients		Standardized Coefficients	t	Sig.
		B	Std. Error	Beta		
1	Perceived methodology support as control technology	0.627	0.103	0.571	6.104	0.000
2	Virtual team performance	0.318	0.122	0.279	2.606	0.011
3	Gender_m	-0.365	0.133	-0.239	-2.741	0.008

Three models were extracted. Table 4-24 shows that the perceived methodology support as control technology is significant because the p-value is 0.000. The p-value for virtual team performance is significant as 0.011 as it is less than 0.05. The p-value for gender m is also significant as 0.008 because it is less than 0.05.

Table 4-25: Results of the regression analysis predicting the success of the process.

Variables	Models		Effect sizes
Factor name	1	2	
Virtual team performance			
Support-production technology		0.312*	
Support provided control technology	0.557***	0.343**	0.36
Impact quality of the developed system			
Support provided cognitive/cooperation technology			
Impact on the quality and the productivity of the development process			
Number of employees			
Strictness of use			
R	0.557	0.602	
R square	0.311	0.362	
Adjusted R square	0.302	0.345	
F	34.680***	21.548***	

$p \leq 0.1$, $*p \leq 0.05$, $**p \leq 0.01$, $***p \leq 0.001$

Stepwise regression analysis was used to determine factors that influence the overall success of recent projects developed. The first column shows a list of all possible variables that might have an effect on the process as the dependent variable. The second column contains the results of the first step performed during the stepwise regression; the third column contains the second step performed during the stepwise regression in the order in which the results were entered. The last four rows of stepwise regression consist of the significance of the variables towards the dependent variable (R, R-square, adjusted R-square and F). A large effect size of 0.36 on the support was provided as control technology by systems development methodologies indicated that it may influence the success of the process.

:

Table 4-26: The regression results for the factor representing the success of the project developed by the virtual team using coefficients.

Model		Unstandardized coefficients		Standardized coefficients	t	Sig.
		B	Std. Error	Beta		
1	Perceived methodology support as control technology	0.640	0.109	0.557	5.889	0.000
2	Perceived methodology support as production technology	0.362	0.146	0.312	2.473	0.016

Two models were extracted. Table 4-26 shows that the perceived methodology support as control technology is significant because the p-value is 0.000. The p-value for Perceived methodology support as production technology is significant as 0.016 as it is less than 0.05.

Table 4-27: Table: Results of the regression analysis predicting the success of the customer satisfaction.

Variables	Model	Effect size
Factor name	**1**	
Virtual team performance		
Support-production technology	0.555***	0.31
Support provided control technology		
Impact quality of the developed system		
Support provided cognitive/cooperation technology		
Impact on the quality and the productivity of the development process		
R	0.555	
R square	0.308	
Adjusted R square	0.299	
F	34.636***	

p ≤ 0.1, *p ≤ 0.05, **p ≤ 0.01, ***p ≤ 0.001

Stepwise regression analysis was used to determine factors that influence the overall success of recent projects developed. The first column shows a list of all possible variables that might have an effect on customer satisfaction as the dependent variable. The second column contains the results of the first step performed during the stepwise regression. The last four rows of stepwise regression consist of the significance of the independent variables towards a dependent variable (R, R-square, adjusted R square and F). A large effect size of 0.31 on support-production technology by systems development methodologies indicated that it may influence the success of the customer satisfaction.

Table 4-28: The regression results for the factor representing the success of the project developed by the virtual team using coefficients.

Model		Unstandardized Coefficients		Standardized Coefficients	t	Sig.
		B	Std. Error	Beta		
1	Perceived methodology support as production technology	0.664	0.113	0.555	5.885	0.000

Only one model was extracted. Table 4-28 shows that the perceived methodology support as production technology is significant because the p-value is 0.000.

Table 4-29: Results of the regression analysis predicting the success of the leadership.

Variables	Models			Effect size
Factor name	1	2	3	
Virtual team performance				
Support-production technology	0.496***	0.473***	0.278	0.34
Support provided control technology			0.284	
Impact quality of the developed system				
Support provided cognitive/cooperation technology				

:

Impact on the quality and the productivity of the development process				
Gender		-0.237	-0.235	
Strictness of use				
R	0.496	0.549	0.587	
R square	0.246	0.302	0.344	
Adjusted R square	0.236	0.284	0.318	
F	25.157***	16.435***	13.138 ***	

$p \leq 0.1$, $*p \leq 0.05$, $**p \leq 0.01$, $***p \leq 0.001$

Stepwise regression analysis was used to determine factors that influence the overall success of recent projects developed. The first column shows a list of all possible variables that might have an effect on leadership as the dependent variable. The second column contains the results of the first step performed during the stepwise regression; the third column contains the second step performed during the stepwise regression in the order in which the results were entered. The last four rows of terms of regression consist of the significance of the independent variables towards a dependent variable (R, R-square, adjusted R-square and F). A large effect size of 0.34 on support-production technology by systems development methodologies indicated that it may influence the success of leadership.

Table 4-30: The regression results for the factor representing the success of the project developed by the virtual team using coefficients.

Model		**Unstandardized Coefficients**		**Standardized Coefficients**	**t**	**Sig.**
		B	**Std. Error**	**Beta**		
1	Perceived methodology supports as production technology	0.711	0.142	0.496	5.016	0.000
2	Gender_m	-0.469	0.191	-0.237	-2.462	0.016

3	Perceived methodology support as control technology	0.403	0.183	0.284	2.207	0.030

Three models were extracted. Table 4-30 shows that the perceived methodology support as production technology is significant because the p-value is 0.000. The p-value for gender_m is significant as 0.016 as it is less than 0.05. The p-value for Perceived methodology support as control technology is also significant as 0.030 because it is less than 0.05.

Table 4-31: Results of the regression analysis predicting the success of communication.

Variables	**Model**	**Effect size**
Factor name	**1**	
Support-production technology	0.468***	0.22
Support provided control technology		
Impact quality of the developed system		
Support provided cognitive/cooperation technology		
Impact on the quality and the productivity of the development process		
R	0.468	
R square	0.219	
Adjusted R square	0.209	
F	21.847***	

$p \leq 0.1$, $*p \leq 0.05$, $**p \leq 0.01$, $***p \leq 0.001$

Stepwise regression analysis was used to determine factors that influence the overall success of recent projects developed. The first column shows a list of all possible variables that might have an effect on communication as the dependent variable. The second column contains the results of the first step performed during the stepwise regression. The last four rows of stepwise regression consist of the significance of the independent variables towards a dependent variable (R, R-square, adjusted R-square and F). A medium effect size of 0.22 on support-production technology by systems development methodologies indicated that it may influence the success of communication.

:

Table 4-32: The regression results for the factor representing the success of the project developed by the virtual team using coefficients.

Model		Unstandardized coefficients		Standardized coefficients	t	Sig.
		B	Std. Error	Beta		
1	Perceived methodology supports as production technology	0.650	0.139	0.468	4.674	0.000

Only one model was extracted. Table 4-32 shows that the perceived methodology support as production technology is significant because the p-value is 0.000.

Table 4-33: Table: Results of the regression analysis predicting the success of the competencies.

Variables	Model	Effect size
Factor name	**1**	
Support-production technology	0.507***	0.26
Support provided control technology		
Impact quality of the developed system		
Support provided cognitive/cooperation technology		
Impact on the quality and the productivity of the development process		
Strictness of use		
Virtual team performance		
R	0.507	
R square	0.257	
Adjusted R square	0.248	
F	26.695***	

$p \leq 0.1$, $^*p \leq 0.05$, $^{**}p \leq 0.01$, $^{***}p \leq 0.001$

Stepwise regression analysis was used to determine factors that influence the overall success of recent projects developed. The first column shows a list of all possible variables that might have an effect on competencies as the dependent variable. The second column contains the results of the first step performed during the stepwise regression. The last four rows of stepwise regression consist of the significance of the independent variables towards a dependent variable (R, R-square, adjusted R-square and F). A large effect size of 0.26 on support-production technology by systems development methodologies indicated that it may influence the success of competencies.

Table 4-34: The regression results for the factor representing the success of the project developed by the virtual team using coefficients.

Model		Unstandardized Coefficients		Standardized Coefficients	t	Sig.
		B	Std. Error	Beta		
1	Perceived methodology supports as production technology	0.548	0.106	0.507	5.167	0.000

Only one model was extracted. Table 4-34 shows that the perceived methodology support as production technology is significant because the p-value is 0.000.

4.7 Conclusion

The overall objective was to provide the findings based on the objectives of this research. Findings were reliable and significant based on the validity of data. Those findings can help in the development of systems in future. The research method and survey questionnaire were also discussed. Techniques that were used to analyse data were descriptive analysis, factor analysis, reliability analysis, correlation analysis and regression analysis.

CHAPTER 5: DISCUSSION AND COMMENTS

5.1 Introduction

The aim of this chapter is to answer the questions that were raised in this study. This will be done in order to achieve the objectives of this study. All the objectives will be summarised, interpreted and some recommendations offered. A conceptual model will also be presented and discussion provided on future work in this field. I will start with the background information, benefits and challenges faced by virtual software development teams, systems development methodology usage, the success of projects developed by virtual software development teams and the role of systems development methodologies in the success of those projects.

In chapter 1, I mentioned the following objectives that need to be achieved based on this study. In this chapter, I will determine whether those objectives were achieved or not based on the results I presented in chapter 4.

1) To identify challenges and benefits faced by virtual software development teams.
2) To evaluate the use (if any) and the suitability of current systems development methodologies for use by virtual software development teams.
3) If no systems development methodologies are used, determine the reason why it is not used.
4) To determine the success of projects developed by virtual software development teams.
5) To investigate the role (if any) of systems development methodologies in the success of those projects.

5.2 Identify challenges and benefits faced by virtual software development teams

The first objective of this study was to identify both the benefits (Table 4-1) and challenges (Table 4-2) faced by virtual software development teams in order to assess best ways or approaches that can be used to address these challenges. To investigate the benefits and challenges faced by virtual software development teams, the descriptive measures using frequency and factor analysis were applied to the data obtained through questionnaire surveys. The frequencies of responses were converted into percentage proportions of the total sample size. Factor analysis was also used to determine the validity among correlated variables in order to create one construct variable that measures similar things. Based on the results presented in the table above (Table 4-1), there are several benefits that are associated with virtual teams. The most important benefit that was identified in this study with the higher mean was "*Outcomes are normally stored electronically*" with 4.20 mean, followed by "*Flexible to work at anytime and anywhere*" with 4.05 mean, followed by "*Ability to bridge time and space*" with 4.00 mean, followed by "*Easy review and performance*" with 4.00 mean, followed by "*Enables organisations to attain a distant geographic reach*" with 3.93 mean, followed by "*Maintains an effective relationship with employees and customers*" with 3.87 mean, followed by "*Communication is flexible*" with 3.85 mean, followed by "*Cost saving*" with 3.84 mean followed by "*Bring under control the talent and knowledge to solve problems*" with 3.74 mean and finally "*Tasks are completed on a 24/7 schedule*" with 3.62 mean. The results are consistent with other previous findings reported in the literature by Berry (2011); Hunsaker and Hunsaker (2008) and Horwitz *et al.* (2006).

Berry (2011) indicated that virtual teams give organisations some ability to bridge time and space, as compared with the expenses and time needed to travel by face-to-face teams. They are also cost-efficient and provide a means for better utilization of distributed human resources. Consistently, the results of this study also showed that cost was the important driving success factor in virtual software development teams with a mean of 3.84 which shows a positive response from participants. This reflects that globalisation allows virtual software development teams to save expenses they could have incurred while using traditional system. Horwitz *et al.* (2006) reported that virtual software development operates at a low cost, they use improved resources that are necessary to meet changing needs and task requirements in the global business environment.

Hunsaker and Hunsaker (2008) mentioned that the adoption or use of virtual teams in organisations allows them to access the most and talented qualified individuals. That means individuals who can master their jobs regardless of the location or distance, thus enabling organizations to respond faster to increased competition. Moreover, allowing flexibility wherever the employee is working from. Horwitz *et al.* (2006) also mentioned that virtual teams are flexible as they use improved resources that are necessary to meet changing technological business needs. Berry (2011) also indicated that in virtual teams, outcomes are normally stored electronically and automatically because of the easy review and performance. He also indicated that everyone can talk wherever they want because participation occurs at the same time instead of in a serial way as with face-to-face teams. Therefore, the results of this study also confirmed that results of the projects are always stored electronically with the highest mean of 4.20 which shows a very positive response from participants. These allowed virtual software development teams to access and view the outcome of their hard work and make some recommendations for future projects.

5.2.1 Challenges that were faced by virtual software development teams

In table 4-3, the highest mean was 3.27 for *"Technology"*, closely followed by "*Culture: Organisational difference*" with a mean of 3.26, followed by "*Holidays, local laws, customs*" with the mean of 3.21, followed by "*Personal conflicts and misunderstanding*" with the mean of 3.21, followed by "*Distance*" with the mean of 3.16, followed by "*Culture: National difference*" with the mean of 3.09 and the lowest mean was 2.38 *"Time zones"* with the standard deviation of 1.537. Most of the means are greater than 3, which indicates the respondents who agree that they experience challenges while working in a virtual team. The only mean less than 3, was for item "Time zones", which means that the respondents don"t perceive this item as a challenge while working on a virtual software development team. Rusman *et al.* (2010) stated that communication might not be spread equally in time due to the time difference of zones. Moreover, Binder (2007) mentioned that main challenges include language, time zones, different locations and other challenges. In addition, Sengupta *et al.* (2006) indicated that poor communication among virtual teams is caused by separated distance and time-zone differences.

Factor analysis was also used to determine the validity among correlated variables in order to create an index with a variable that measures similar things. This resulted in two factors, called "Culture and values" and "Timezone". Binder (2007) mentioned that main challenges include language, time zones, different locations, organizational and personal culture, policies and regulations of the organization, business processes, political climate, management skills and project leadership skills. Based on the on a few studies, Casey (2010) identified trust as a key factor in the success or failure of virtual teams where cultural differences are even more stated.

Kankanhalli *et al.* (2007) also stated that: "there is much potential for conflict in virtual teams as members work across cultural, geographical and time-bound environments and this conflict leads to ineffective communication and as soon as team members stop communicating effectively, barriers begin to form between them leading to a decrease in productivity and interaction". In virtual teams, there is not enough trust, there are also communication breakdowns at some point, as well as conflicts and power struggles (Rosen, *et al.*, 2007; Baskerville & Nandhakumar, 2007). Cultural and functional differences lead team members to think differently about processes. Also to develop a trust among virtual team members is very challenging (Jacobs *et al.,* 2005; Kankanhalli *et al.*, 2006; Munkvold & Zigurs, 2007; Paul *et al.*, 2005; Poehler & Schumacher, 2007; Shachaf, 2005).

In literature (e.g. Ebrahim *et al.*, 2009; Walvoord *et al.*, 2008), a virtual environment by itself represents major challenges to effective communication. The greatest obstacle to the effectiveness of virtual teams was to implement technology because information richness seemed to be the most important element for technology selection (Mikkola *et al.*, 2005). Dekker *et al.* (2008) stated that virtual teams communicate through technology from a different discipline to achieve a common goal, so the way that technology is implemented makes the virtual team outcome more or less the same (Anderson *et al.*, 2007).

In support of literature, the most difficult experience that might affect virtual software development teams was technology selection. The participants also agreed that technology might have affected their virtual software development teams negatively if implemented incorrectly. The ultimate result of this problem is a failure to achieve the desired end product or a high-quality product as it is believed that in order for the teams to understand each other they have to also understand the technology that they are using. In the previous literature (Sengupta *et al.* (2006); Binder, 2007) time zone is still a challenge for virtual software development teams. However, in this study, I found that time zones are not a challenge anymore. This could be because of the ever-changing and improving technology which makes it easier for the teams to communicate using more sophisticated tools. Moreover, organisations are benefiting from virtualization to some extend that time zone is no longer a challenge at all.

5.3 Evaluate the suitability of current systems development methodologies for use by virtual software development teams

The second objective of this study was to evaluate the suitability of current systems development methodologies for use by virtual software development teams. To evaluate the suitability of current systems development methodologies, the descriptive measures using frequency and correlations and cross-tabulations were applied to the data obtained through questionnaire surveys. The frequencies of responses were converted into percentage proportions of the total sample size. Correlations were also applied in order to see systems development methodology that has been used or still in use by virtual software development teams and the relationships among those methodologies if any.

The figure 3-3 showed the number of years in which participants have been working on virtual software development teams. Only 10% of participants have been in virtual software development teams for more than ten years; 65% of participants for less than a year and 37% of participants have been involved for 3-5 years. This shows that virtual software development is still not regularly used by organisations. Table 4-5 showed the highest methodology was *"SDLC"* with the mean of 4.17, followed by "*RAD*" with the mean of 3.63, followed by Scrum with the mean of 3.60 and "*XP*" with the mean of 3.48. The lowest mean is 1.58 for *"YSM"*. Other means were below 3.00, therefore, they were considered insignificant. However, there are other methodologies that were mentioned by participants which were not included in the questionnaire, namely Kanban and Magic. Only 7% of participants used Kanban and 2% used Magic.

Cross-tabulation was done on modern and traditional methodologies to measure what combination of SDMs is used per company in table 4.6. This indicated that organisations are really moving into consolidation by joining both traditional and modern methodologies for best results. Holmstrom *et al.* (2006), considered agile systems development methodologies as being helpful in complex global software development projects, and recommended that agile systems development methodologies may be open to global software development as compared to previously used technologies. In table 4-5, SDLC, RAD, Scrum, and XP are used by virtual software development teams as they also fall under agile methodologies which are not fully utilised by many organisations except for SDLC with falls under traditional methodologies. Based on the results, virtual software development teams seemed to be engaging in modern methodologies which showed a good sign of methodology adoption.

In addition, Prikladnicki (2015) indicated that geographically distributed teams have adopted agile methodologies as a fit for their projects. This methodology involves distributed pair programming which involves two developers working remotely on the same code. Darwish and Rizk (2015) also found that 60% of the projects failed from 2004 to 2012 which showed a very high rate of failure in IT projects. These failures were overcome in many organisations by becoming more engaged with the agile methodology as it is currently booming in the market. It is more flexible in terms of quicker delivery, simple phases, easy to change requirements and strong communication between developers and customers as compared to traditional methodology. Since the word agile means ability, it is easy to move and not only that but also quickly.

In table 4-6, there are 58.3% of organisations that do not use either traditional or modern methodology while 41.7% of companies are using modern methodologies. There are only 14.3% of organisations not using methodologies and 85.7% of organisations using both modern and traditional methodologies. According to Guzman *et al.* (2010), software development methodologies that are normally used in global software development projects are Rational

Unified Process (RUP), Metrica Version 3, Structured Systems Analysis and Design Method (SSADM) and Scrum. There were a lot of systems development methodologies that were being used by virtual teams as shown above. Our results are consistent with the literature because they showed that virtual software teams are using a combination of traditional and agile systems development methodologies.

5.4 If no systems development methodologies are used, determine the reason why it is not used

The third objective was to determine the reason for not using systems development methodology. There were two reasons that were mentioned by virtual software development teams for not using systems development methodologies namely: they outsource and support application that was developed by other companies. This does not reflect the good results in future as organisations will spent more money on outsourcing and support instead of applying methodology themselves.

5.5 Determine the success of projects developed by virtual software development teams

The fourth objective was to determine the success of projects developed by virtual software development teams. Table 4-8 to 4-20 shows that all the means (for the individual items were greater than 3, which means that the participants feel strongly that their projects were very successful regarding the following measures:

Horizontal use, strictness of use, support provided by system development methodology as production technology in virtual software development teams, support provided by system development methodology as control technology in virtual software development teams, impact on the quality of the developed system, support provided by SDMs as cognitive and cooperative technology in virtual software development teams and the impact on the quality and the productivity of the development process. All the means were significant, that means all variables that were used to measure the impact on the quality and the productivity of the development process

5.6 Investigate the role (if any) of systems development methodologies in the success of those projects

The fifth objective of the study was to investigate the role of systems development methodologies in the success of those projects. The success of projects was measured in tables 4.20, 4.21, 4.22, 4.23, 4.24. Stepwise regression analysis was used to determine factors that influence the overall success of recent projects developed. Below in table 5-1 is the list of all possible variables that might have an effect on product, process, customer satisfaction, leadership, communication and competencies in relation to the size effect.

Table 5-1: Factors that influence the success of projects

Variables	Factors influencing the success of the projects					
	Product	**Process**	**Customer Satisfaction**	**Leadership**	**Communication**	**Competencies**
Support provided by SDM as production technology	-	✓	✓	✓	✓	✓
Support provided by SDM as control technology	✓	✓	-	✓	-	-
Gender_m	✓	-		✓	-	-
Team performance	✓					
Effect size	Large effect (0.44)	Large effect (0.36)	Large effect (0.31)	Large effect (0.34)	Medium effect (0.22)	Large effect (0.26)

< 0.13 = non-significant, 0.13-0.25 = significant and lager than 0.25 = practically important

The overall success of the projects was influenced by the support provided by SDM as production technology, the support provided by SDM as control technology, team performance and gender_m (Males). *Support provided by SDM as control technology, team performance*, and *gender_m* contributed towards the success of the product with the effect size of 0.44 which is highly significant. *Support provided by SDM as production technology* and *support provided by SDM as control technology* contributed towards the success of the process, with the effect size of 0.36 which is highly significant.

Support provided by SDM as production technology contributed towards customer satisfaction with the effect size of 0.31 which is highly significant. *Support provided by SDM as production technology, Support provided by SDM as control technology and Gender_m* contributed to the good leadership with the effect size of 0.34 which is highly significant. *Support provided by SDM as production technology* contributed towards good communication with the effect size of 0.22

which is significant. Lastly, *Support provided by SDM as production technology* contributed towards competencies of the teams with the effect size of 0.26 which is highly significant.

The results showed that support provided by SDM as production technology, the support provided by SDM as control technology, gender_m (males) and team performance have a large influence on the success of projects developed by virtual software development teams.

The overall research contribution

The results of my study showed that even though organisations are moving into agile, waterfall methodology will still be used as a backup. Organisations must take into consideration the importance of SDM especially in virtual teams before starting the project. This will help virtual teams to choose the best methodology that will suit their project from the beginning until the end. Most virtual software developments teams do not follow a specific methodology but rather use any method that suits their projects at the point in time. It has been identified that the following systems developments have been contributed to the research: The use and the suitability of current systems development methodologies for use by virtual software development teams. The study also shows some organisations can effectively use both traditional and modern systems development methodologies, depending on the suitability of their project and the size of such project.

5.7 Limitations and future work

5.7.1 Limitations

I looked at the previous literature in relation to virtual software development teams and this current study made a contribution to the existing literature. However, there are some limitations in this study. There is not enough literature that could be used with regard to the use of systems development methodology by virtual software development teams.

It was very difficult to collect data from participants as most of them were unwilling to complete the questionnaire. Despite the low response rate on email and web-based, data that was collected was valid and significant enough to be analysed. The data was collected in Lesotho, South Africa, partly India and Chicago and China.

5.7.2 Future work

More work must be done with regard to virtual software development teams especially in South Africa and other countries where the English language is not their first language. This can help in the effort to overcome language as a barrier to virtual software development teams. This might also open an opportunity for other students or researchers who would like to see how other continents or countries apply systems development methodologies when developing their software/systems.

BIBLIOGRAPHY:

Ågerfalk PJ, Fitzgerald B, Holmström Olsson H, Ó Conchúir E (2008) Benefits of global software development: the known and unknown. Presented at international conference on software process, ICSP 2008, Leipzig, Germany.

Ågerfalk, P.J., Fitzgerald, B., Holmström, H., Lings, B., Lundell, B., & Ó Conchúir, E. (2005) *A framework for considering opportunities and threats in distributed software development.* Presented at international workshop on distributed software development (DiSD), Paris, France.

Alge, B. J., Wiethoff, C., & Klein, H. J. (2003). When does the medium matter? Knowledge-building experiences and opportunities in decision-making teams. Organizational Behavior and Human Decision Processes, 91, 26–37.

Ambler, S. (2011). Standish Group CHAOS Manifesto. Project Success Survey. Agile Impact Report. [Online] United States: PricewaterhouseCoopers LLP, pp.1-12. Available at: https://www.pwc.com/us/en/insurance/publications/assets/pwc-using-agilemethodology.pdf [Accessed 6 Dec. 2015].

Anderson, A., Mcewan, R., Bal, J. & Carletta, J. (2007). Virtual team meetings: An analysis of communication and context. *Computers in Human Behavior*, 23(5), pp. 2558-2580.

Arnott, D. & Pervan, G. (2008). Eight key issues for the decision support system discipline.

Ásólfsdóttir, A. (2012). *Leading a virtual team: globalisation and it project management*. Masters dissertation, School of Science and Engineering, Reykjavík University.

Avison, D. & Fitzgerald, G. (2006). *Information systems development: methodologies, techniques and tools* 4th ed. Berkshire UK: The McGraw-Hill Companies.

Bachelor of Software Engineering & Management Thesis. University of Gothenburg.

Badrinarayanan, V. & Arnett, D. (2008). Effective virtual new product development teams: an integrated framework. *Journal of Business & Industrial Marketing*, 23(4), pp. 242-248.

Baheti, P., Williams, L., Gehringer, E. & Stotts, D., (2002). Exploring Pair Programming in Distributed Object-Oriented Team Projects, In Proceedings of XP/Agile Universe Chicago, August 4-7, 2002; Lecture Notes in Computer Science 2418: Springer Verlag, pp. 208-220

Bal J, Teo PK. Implementing virtual teamworking, part 1: a literature review of best practice. Logist Inform Manage 2000,13(6):346–52.

Baschab, J. and & Piott, J. (2007) The Executive‟s Guide to Information Technology, 2nd Edition, NY: Wiley.

Baskerville, R. & Nandhakumar, J. (2007). Activating and Perpetuating Virtual Teams: Now

Bassil, Y. (2012). A Simulation Model for the Waterfall Software Development Life Cycle. *Behaviour & Information Technology*, 27(2), pp. 107-114.

Bergiel, J. B., Bergiel, E. B., & Balsmeier, P. W. 2008. Nature of virtual teams: a summary of their advantages and disadvantages. *Management Research News, 31*(2), pp. 99-110.

Berry, G. (2011). Enhancing Effectiveness on virtual Teams: Understanding Why Traditional Team Skills Are Insufficient. *Journal of Business communication*, 48(2), pp. 186-206.

Binder, J. (2007). *Global Project Management: Communication, Collaboration and Management Across Borders.* Hampshire: Gower Publishing Limited.

Binder, J. (2009). Global Project Management Communication, Collaboration and Management across Borders. Strategic Direction, 25(9).

Binder, J. (2009). Global Project Management: Communication, Collaboration and Management across Borders. *Strategic Direction*, 25(9), pp. 2-18.

Blaise, B., Bergiel, E. & Balsmeier, P (2008). Nature of virtual teams: a summary of their advantages and disadvantages. *Management Research News*, Vol. 31(2), pp. 99–110

Boehm, B. (2006). A View of 20th and 21st Century Software Engineering. University of Southern California. *ICSE.*

Bygstad, B., Ghinea, G. & Brevik, E. (2008). Software development methods and usability: Perspectives from a survey in the software industry in Norway. *Interacting with Computers*, 20(3), pp. 375-385.

Carmel, E. & Agarwal, R. (2001). Tactical approaches for alleviating distance in global software development. *IEEE Software*, 18(2), pp. 22-29.

Carmel, E. (1999). *Global Software Teams: collaborating across borders and time Zones.* Prentice Hall PTR.

Casey, V. & Richardson, I. (2006). Project management within virtual software teams. *Global*

Casey, V. (2010). Developing Trust in Virtual Software Development Teams. *Journal of Theoretical and Applied Electronic Commerce Research*, 5(2), pp. 41-55.

Chinbat, S. (2010). Lessons learned in virtual teams from global software development.

Chow, T. & Cao, D. (2008). A survey study of critical success factors in agile software projects. *Journal of Systems and Software*, 81(6), pp. 961-971.

Chudoba, K. M., Wynn, E., Lu, M., & Watson-Manheim, M. B. (2005). How virtual areChudoba, K., Wynn, E., Lu, M. & Watson-Manheim, M. (2005). How virtual are we? Measuring virtuality and understanding its impact in a global organization.
Communication of the ACM, 50(11), pp. 85-90.

Conger, S. (2010). Software Development Life Cycles and Methodologies: Fixing the old and adopting the new. *Sprouts: Working Papers on Information Systems*, 10(172). http://sprouts.aisnet.org/10-172.

Costello, A. B. & J. Osborne (2005). Best practices in exploratory factor analysis: four recommendations for getting the most from your analysis. *Practical Assessment Research & Evaluation*, 10(7).

Cramton, C. & Webber, S. (2005). Relationships among geographic dispersion, team processes, and effectiveness in software development work teams. *Journal of Business Research*, 58(6), pp. 758-765.

Cumps, B., Viaene, S. & Dedene, G. (2006). An Empirical Study on Business/ICT Alignment in European Organisations. Proceedings of the 39th Hawaii International Conference on System Sciences.

Cusumano, M. (2008). Managing Software Development in Globally Distributed Teams. *Communications of the ACM*, 51(2).

Da Silva Estacio, B. & Prikladnicki, R. (2015). Distributed Pair Programming: A Systematic Literature Review. *Information and Software Technology*, 63, pp. 1-10.

Daim, T., ha, A., Reutiman, S., Hughes, B., Pathak, U., Bynum, W. & Bhatla, A. (2012). Exploring the communication breakdown in global virtual teams. *International Journal of Project Management* 30(2), pp. 199-212.

Dalcher, D. & Brodie, L. (2007). Success IT projects. London: Thomas Learning. Doi, R. 2006. Why IT projects fail. (Paper for the Fifth Audit Seminar of the INTOSAI Standing Committee on IT Audit, November 2006).

De Vries, M. (2012).Evaluating or selecting a suitable information system development methodology: A case study. *The South African Journal of Industrial Engineering*, 15(2), pp. 15-23.
Decision Support Systems, 44(3), pp. 657-672.

Dekker, D, Rutte, C. & Van den Berg, P. (2008). Cultural differences in the perception of critical interaction behaviors in global virtual teams. *International Journal of Intercultural Relations*, 32(5), pp. 441–452.

Dessie, M. (2009). *The use and effectiveness of systems development methodologies in small, micro and medium enterprises*. Masters dissertation, Northwest University South Africa.

Earnhardt, M.P. (2009). Identifying the key factors in the effectiveness and failure of virtual teams Leadership Advance Online, Issue XVI, *School of Global Leadership & Entrepreneurship.*

Ebrahim, A., Shamsuddin, A. & Taha, Z. (2009). Virtual Teams: A Literature Review. *Australian Journal of Basic and Applied Science*, 3(3), pp. 2653-2669.

Ebrahim, N. (2015). Virtual R&D Teams: A new Model for Product Development. *International Journal of Innovation*, 3(2), pp. 1-27.

Ebrahim, N., Ahmed, S. & Taha, Z. (2009). Virtual teams for new product development – An innovative experience for R & D engineers. *European Journal of Educational Studies,* 1(3).

El-Tayeh, A., Gil, N. & Freeman, J. (2008). A methodology to evaluate the usability of digital

Ewusi-Mensah, K. (2003). *Software development failures: anatomy of abandoned projects, Boston, MA*: MIT Press.

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

Fitzgerald, B. (1997). The use of systems development methodologies in practice: a field study. *Information Systems Journal*, 7(3), pp. 201-212.

Fitzgerald, B. (1998). An empirical investigation into the adoption of systems development methodologies. *Information & Management*, 34(6), pp. 317-328.

Fuller, M., Hardin, A. & Davison, R. (2007). Efficacy in Technology-Mediated Distributed Teams. *Journal of Management Information Systems*, 23(3), pp. 209-235.

Fuller, M.A., Hardin, A.M. & Davison, R.M. (2006b). Efficacy in technology-mediated distributed team. *Journal of Management Information Systems*, 23, pp. 209-235.

Furst, S., Reeves, M., Rosen, B. & Blackburn, R. (2004). Managing the life cycle of virtual teams. *Academy of Management Executive*, 18(2), pp. 6-20.

Garcia Guzman, J., Saldana Ramos, J., Amescua Seco, A. & Sanz Esteban, A. (2010). How to get mature global virtual teams: a framework to improve team process management in distributed software teams. *Software Quality Journal*, 18(4), pp. 409-435.

Gibson, C. & Gibbs, J. (2006). Unpacking the concept of virtuality: The effects of geographic dispersion, electronic dependence, dynamic structure, and national diversity on team innovation, *Administrative Science Quarterly,* 51, pp. 451–495.

Glasow, P. A. (2005). Fundamentals of survey research methodology. Washington C3 Center. McLean, Virginia.

Gonzalez-Perez, C., McBride, T. & Henderson-Sellers, B. (2005). A meta-model for Assessable Software Development Methodologies. Software Quality Journal. 13(2),195-214

Gorla, N. and & Lin, S.-C. (2010). Determinants of software quality: A survey of information systems project managers. Information and Software Technology. Amsterdam, 52(6), pp. 602.

Griffin, A. & Brandyberry, A. (2008). *Systems Development Methodology Usage in Industry: A Review and Analysis*. Department of business and Economics, Pennsylvania State University.

Griffin, A.S. 2008, *Examining the decision process and outcomes of system development methodology adoption*. PhD thesis, Kent State University.

Griffin, S. & Brandyberry, A. 2008. System development methodology usage in industry: a review and analysis. *Journal of information system applied research (JISAR)*, 3(19), pp. 1-18.

Hasab, S. A & Faraahi, A. (2014). An Overview of Applying Cloud Computing in Addressing Agile Global Software Development Challenges Reef Resources. *Assessment and Management Technical Paper*, 40 (5), pp. 184-191.

Helena Holmström, H., Brian Fitzgerald, B., Pär J. Ågerfalk P.J. & Eoin Ó. Conchúir, E.O. (2006): Agile Practices Reduce Distance in Global Software Development. *Information Systems Management*, 23(3), pp. 7-18.

Hertel, G., Geister, S. & Konradt, U. (2005). Managing virtual teams: A review of current empirical research. *Human Resource Management Review,* 15(1), pp. 69–95.

Highsmith, J. (2009). *Agile projects management: creating innovative products* 2nd Edition. Addison-Wesley NJ.

Holmstrom, H., Fitzgerald, B., Agerfalk, P. & Chonchuir, E. (2006). Agile Practices Reduce Distance in Global Software Development. *Information Systems Management*, 23(3), pp. 7-18.

Holmström, H., Fitzgerald, B., Ågerfalk, P. J., & Conchúir, E. Ó. (2006). Agile practices reduce distance in global software development. Information Systems Management, 23(3), 7-

Horwitz, F., Bravington D. & Silvis U. (2006). The promise of virtual teams: identifying key factors in effectiveness and failure. *Journal of European Industrial Training*, 30(6), pp. 472 – 494.

Huang, H. & Trauth, E. M (2009). *Cultural Influences and Globally Distributed Information Systems Development: Experiences from Chinese IT Professionals*. PhD thesis, Information Sciences and Technology, The Pennsylvania State University.

Huisman, M. & Iivari, J. (2006). Deployment of systems development methodologies: Perceptual congruence between IS managers and systems developers. *Information & Management*, 43(1), pp. 29-49.

Huisman, M. & Iivari, J. (2006). The individual deployment of systems development methodologies. *Advanced Information Systems Engineering*, 2348, pp. 134-150.

Hunsaker P. & Hunsaker J. (2008). Virtual teams: a leader"s guide. *Team Performance Management*, 14(1/2), pp. 86-101.

Iivari, J., Hirschheim, R. & Klein, H. (1998). A Paradigmatic Analysis Contrasting Information Systems Development Approaches and Methodologies. *Information Systems Research*, 9(2), pp. 164-193.

Information System Journal, 15, 279–306.

Information Systems Journal, 15(4), pp. 279-306.
International Journal of Engineering & Technology, 2(5), pp. 2049-3444.

Islam, Z., Doshi, J., Mahtab, H. and Ariffin Ahmad, Z. (2009). Team learning, top management support and new product development success. Int J Managing Projects in Bus, 2(2), pp.238-260.

Jaakkola, H., Heimburger, A. & Linna, P. (2009). Knowledge-Oriented software engineering process in a multi-cultural context. *Software Quality Journal*, 18(2), pp. 299-319.

Jacobsa, J., van Moll, J., Krause, P., Kusters, R., Trienekens, J. & Brombacher, A. (2005). Exploring defect causes in products developed by virtual teams. *Information and Software Technology*, 47(6), pp. 399-410.

Johnston, KA. & Rosin, K. (2011). *Global Virtual Teams: How to manage them. IEEE*, Department of Information Systems, University of cape Town, South Africa.

Kankanhalli, A., Tan, B. & Wei, K. (2007). Conflict and Performance in Global Virtual Teams. *Journal of Management Information Systems*, 23(3), pp. 237-274.

Kotlarsky, J. & Oshri, I. (2005). Social ties, knowledge sharing and successful collaboration in globally distributed system development projects. *European Journal of Information Systems*, 14(1), pp. 37–48.

Lee-Kelly, L. & Sankey, T. (2008). Global virtual teams for value creation and project success: A case study. *International Journal of Project Management, European Academy of Management*, 26(1), pp. 51-62.

Lin, C. P. (2008). Examination of gender differences in modeling OCBs and their antecedents in business organizations in Taiwan. *Journal of Business and Psychology*, 22, pp. 261– 273.

Lin, C., Standing, C. & Liu, Y. (2008). A model to develop effective virtual teams. *Decision Support Systems*, 45(4), pp. 1031-1045.

Lira. E., Ripoll. P., Peiro, J. & Orengo. V. (2008). How do different types of intragroup conflict affect group potency in virtual compared with face-to-face teams? A longitudinal study.

Malhotra, A., Majchrzak, A. & Rosen, B. (2007). Leading virtual teams. *Academy of Management Perspective*, 21(1), pp. 60-70.

McKinney, V. & Whiteside, M. (2006). Maintaining Distributed Relationships. *Communications of the ACM*, 49(3), pp. 83 - 86.

Meso, P., Madey, G., Troutt, M. & Liegle, J. (2006).The knowledge management efficacy of matching information systems development methodologies with application characteristics-an experimental study. *Journal of systems and software*, 79(1), pp. 15-28.

Mikkola, J.H., Maclaran, P. & Right, S. W. (2005). Book reviews. *Research & Development Management*, 35, pp. 104-109.

Motivation and Emotion, 30(2), pp. 165–169.

Munkvold, B. & Zigurs, I. (2007). Process and technology challenges in swift-starting virtual teams. *Information & Management*, 44(3), pp. 287–299.

Nelson, R.R., Todd, P.A., and & Wixom, B.H. (2005). Antecedents of information and system quality: an empirical examination within the context of data warehousing. *Journal of Management Information Systems*, 21 (4), pp. 199–235.

Oates, B.J. (2006). *Researching Information Systems and Computing*. Sage Publications: India.

Olariu, C. & Aldea, C. (2014). Managing Processes for Virtual Teams-A BPM Approach. *Procedia-Social and Behavioral Sciences*, 109, pp. 380-3-84.

Palacio, J. (2010). *Information systems development methodologies for data-driven decision support systems*. Masters dissertation, University of Manchester.

Paul, S., Seetharaman, P., Samarah, I., & Mykytyn, J. (2005). *Understanding Conflict in Virtual Teams: An Experimental Investigation using Content Analysis.* Paper presented at the 38th Hawaii International Conference on System Sciences, Hawaii.

Paul, S., Seetharaman, P., Samarah, I., & Mykytyn, P.P., (2004). Impact of heterogeneity and collaborative conflict management style on the performance of synchronous global virtual teams. *Information & Management*, 41(3), pp. 303–321.

Peterson, J. (2013). The benefits of adhering to software development methodology concepts. [Blog] Segue blog. Available at http://www.seguetech.com/blog/topic/sdm [Accesses 5 Dec. 2015].

Poehler, L. and & T. Schumacher, T. (2007). *The Virtual Team Challenge: Is It Time for Training?* PICMET 2007. Portland, Oregon - USA.

Powell, A., Piccoli, G. & Ives, B. (2004). Virtual Teams. *SIGMIS Database,* 35(1), pp. 6-36.

RamadanDarwish, N. & M. Rizk, N. (2015). Multi-Dimensional Success Factors of Agile Software Development Projects. *International Journal of Computer Applications*, 118(15), pp. 23-30.

Ramsin, R. & Paige, R. (2008). Process-centred review of object oriented software development methodologies. *ACM Computing Survey*, 40(1), pp. 1-89.

Rezgui, Y. (2007). Exploring virtual team-working effectiveness in the construction sector. *Interacting with Computers*, 19(1), pp. 96-112.

Rice, D., Davidson, B., Dannenhoffer, J. & Gay, G. (2007). Improving the Effectiveness of Virtual Teams by Adapting Team Processes. *Computer Supported Cooperative Work*, 16(6), pp. 567-594.

Rosen, B., Furst, S., & Blackburn, R. (2007). Overcoming Barriers to Knowledge Sharing in Virtual Teams. *Organizational Dynamics,* 36(3), pp. 259–273.

Rosen, B., Furst, S., & Blackburn, R. (2007). Overcoming barriers to knowledge sharing in virtual teams. *Organizational Dynamics*, 36(3), pp. 259–273.

Rosenthall, R. (1991). Meta-analytic procedures for social research. Newbury Park: Calif. Sage Publications.

Rusman, E., Bruggen, J., Sloep, P. & Koper, R. (2010). Fostering trust in virtual project teams: Towards a design framework grounded in a TrustWorthiness Antecedents (TWAN) schema. *International Journal of Human Computer Studies,* 68(11), pp. 834-850.

Sahay, S. (2003). Global software alliances: the challenge of "standardization." *Scandinavian Journal of Information Systems*, (15), pp. 3–21.

Sakthivel, S. (2005). Virtual workgroups in offshore systems development. *Information and Software Technology*, 47(5), pp. 305–318.

Samson, A. & Daft, L. (2003). *Fundamentals of Management*. Melbourne, Australia, Thomson Learning Australia.

Samson, D. & Daft, R. (2003). *Fundamentals of Management*, Pacific Rim ed., Thomson Publishing, Melbourne.

Sarker, S. & Sahay, S. (2004). Implications of space and time for distributed work: an interpretive study of US- Norwegian systems development teams. *European Journal of Information Systems*, 13(1), pp. 3-20.

Schlenkrich, L. & Upfold, C. (2009). A Guideline for Virtual Team Managers: The Key to Effective Social Interaction and Communication. *The Electronic Journal Information Systems Evaluation,* 12(1), pp. 109 – 118.

Sengupta, B., Chandra, S. & Sinha, V. (2006). A research agenda for distributed software development. IBM Research and Indian Institute of Technology, New Delhi, India.

Shachaf, P. & Hara, N (2005). Team Effectiveness in Virtual Environments: An Ecological Approach. INFERRIS, P.A.G., S., (Ed.) Teaching and Learning with Virtual Teams. Idea Group Publishing.

Shachaf, P. (2005). Bridging cultural diversity through E-mail. *Journal of Global Information Technology Management,* 8(2), pp. 46-60.

Shachaf, P. (2008). Cultural diversity and information and communication technology impacts on global virtual teams: An exploratory study. *Information & Management*, 45(2), pp. 131-142.

Shea, T., Sherer, P., Quilling, R. & Blewett, C. (2011). Managing Global Virtual Teams Across Classrooms, Students and Faculty. *Journal of Teaching in International Business*, 22(4), pp. 300-313.

Sherman, D.K., Mann, T., and & Updegraff, J.A. (2006). Approach/Avoidance Motivation, Message Framing, and Health Behavior: Understanding the Congruency Effect.

Shin, Y. (2005). Conflict Resolution in Virtual Teams. *Organizational Dynamics,* 34(4), pp. 331345.

Sorli, M., Stokic, D., Gorostiza, A. & Campos, A. (2006). Managing product/process knowledge in the concurrent/simultaneous enterprise environment. *Robotics and Computer Integrated Manufacturing*, 22(5-6), pp. 399–408.

Steinfield, C. (1998).Virtual Teams and Group Collaboration Technologies: Challenges in Supporting Distributed Groups. http://cscw.msu.edu/papers/virtualteams_prepub.pdf.

Steinfield, C. (2002) Realizing the benefits of virtual teams. *IEEE Computer*, 35(3), pp. 104-106.

Steyn, S. (Jr.). (2002). Practically significant relationships between two variables, *SA Journal of Industrial Psychology,* 28(3), pp. 10-15.

That We‟re Mobile, Where Do We Go? *IEEE Transactions on Professional Communication,* 50(1), pp. 17– 34.

Thomas, D., Bostrom, R. & Gouge, M. (2007). Making knowledge work in virtual teams.

Van Bon, J. (2007). IT Service Management: An Introduction, London: itSMF International.

Van Ho, P. (2011). *Total quality management approach to the information systems development processes: an empirical study*. PhD dissertation. Virginia Polytechnic Institute and State University.

Vavpotic, D. & Bajec, M. (2009). An approach for concurrent evaluation of technical and social aspects of software development methodologies. *Information and Software Technology,* 51(2), pp. 528-545.

Walvoord, A., Redden E., Elliott, L. & Coovert, M. (2008). Empowering followers in virtual teams: Guiding principles from theory and practice. *Computers in Human Behavior,* 24(5), pp. 1884–1906.

Ware, L. C. (2001). *Improving software development*. CIO Research Reports.

we? Measuring virtuality and understanding its impact in a global organization.

Workman, M. (2007). The effects from technology-mediated interaction and openness in virtual team performance measures. *Behaviour & Information Technology,* 26(5), pp. 355-365.

Yaghini, M., Bourouni, A & Hesam Amiri, R. (2009). A framework for Selection of Information Systems Development Methodologies. *CIS*, 2(1).

Yaghini, M., Bourouni, A. & Amiri, R.H. (2009). A framework for selection of information systems development methodologies. *Computer and information science,* 2(1), pp. 3.

Yaghini, M., Bourouni, A. & Hesam Amiri, R. (2009). A Framework for Selection of Information Systems Development Methodologies. *CIS*, 2(1).

Zaied, H. A., Abdel Aal S. & Hassan, M. (2013). Rule-based Expert Systems for Selecting Information Systems Development Methodologies. *IJISA*, 5(9),19-26.

Zar, J. H. (1996). Biostatistical Analysis (3rd ed.). Upper Saddle River, NJ: Prentice-Hall

Zhang, Y., Gregory, M. & Shi Y. (2008). Global engineering networks (GEN). *Journal of Manufacturing Technology Management,* 19(3), pp. 299 - 314.

APPENDIX A (GLOSSARY OF TERMS)

Term	**Definition**

SDLC	Systems development life cycle
STRADIS	Structured Analysis, Design and Implementation of Information Systems
YSM	Yourdon Systems Method
IE	Information Engineering
RAD	Rapid Application Development
OOA&D	Object Oriented Analysis and Design by Coad and Yourdon
RUP	Rational Unified Process
XP	Extreme Programming
GSD	Global Software Development
SASD	Structured Analysis and Structure d Design
ISDM	Information System Development methodology
BPMN	Business Process Model and Notation
SSADM	Structured Systems Analysis and Design Methodology
JSD	Jackson Structured Design
GDT	Geographically Dispersed Team

APPENDIX B (PARTICIPANTS QUESTIONNAIRE)

Section 1: Background information

1. **Gender**: Male ☐ Female ☐

2. **Age category:** Younger than 20☐**,** 20-29 Years☐, 30-39 Years Benefits of global software development,
40- 49 Years☐, 50 years and above ☐

3. **How long have you been involved in virtual teams?**

..

Please, read each phrase on the left carefully, and answer the questions by putting an "**X**" on the right.

4. What type of virtual team were you involved in?	
Multiple locations (Inside your country)	
Multi-national (Outside your country)	
Combination (Inside and Outside your country)	

5. What is/was your position in the virtual team?	
Manager	
Software developer	
Software Architect	
Team lead	
Software Engineer	
Business Systems Analyst	
Other, please specify	

6. What is the core business area of your organization?	
Administrative services	1
Finance/Banking/Insurance	2
Software house/Software consulting	3
Manufacturing	4
Retail/Wholesale	5
Education	6
Other, please specify ……………………………………………........................	7

7. What is the total number of people employed in your organization (at all locations)?	
1 – 4	1
5 – 50	2
51 – 200	3
More than 200	4

8. What is the total number of people employed in your organization's information system (IS) department (at all locations)?	
1 – 5	1
6 – 20	2
20 – 50	3
More than 50	4

NB!! 1=Totally disagree, 2=Disagree, 3= Neutral, 4= Agree, 5= Totally agree

9. To what extent do you agree with the following statements about your virtual team's performance?

		Totally disagree				Totally agree
9.1	Speed of developing new applications is high.	1	2	3	4	5
9.2	Functionality of new applications is high.	1	2	3	4	5
9.3	Productivity of the application developers is high.	1	2	3	4	5
9.4	Cost of systems development is high.	1	2	3	4	5
9.5	Quality of systems development products is high.	1	2	3	4	5
9.6	Cost of systems maintenance is high.	1	2	3	4	5
9.7	Documentation of systems development products is of high quality.	1	2	3	4	5
9.8	The morale in my virtual team is high.	1	2	3	4	5
9.9	My virtual team achieves its goals well.	1	2	3	4	5
9.10	My virtual team has a reputation of excellent work.	1	2	3	4	5

Section 2: Systems development methodology used

For the purpose of this questionnaire, a systems development methodology is defined as a combination of the following:

- **Systems development approach/approaches, e.g. Processes oriented, data oriented, object oriented, agility**
- **Systems development process model/models, e.g. Phased process, iterative, incremental**
- **Systems development technique/techniques e.g. Data dictionary, Flowcharts**
- **Systems development method/methods, commercially or in-house developed which is used to develop systems in your virtual team, e.g. STRADIS, XP, IE, Scrum**

NB: If your virtual team is not using any systems development methodology, state the reason why and skip to section 3.

…………………………………………………………………………………………………

…………………………………………………………………………………………………

1. To what extend is your virtual team using the following standard *(commercial)* systems development methods (Vertical use)? You may mark more than one item.

		Nominally				Intensively
1.1	SDLC (Systems development life cycle)	1	2	3	4	5
1.2	STRADIS (Structured Analysis, Design and Implementation of Information Systems)	1	2	3	4	5
1.3	YSM (Yourdon Systems Method)	1	2	3	4	5
1.4	IE (Information Engineering)	1	2	3	4	5
1.5	RAD (Rapid Application Development)	1	2	3	4	5

1.6	OOA&D (Object Oriented Analysis and Design by Coad and Yourdon)	1	2	3	4	5
1.7	RUP (Rational Unified Process)	1	2	3	4	5
1.8	XP (Extreme Programming)	1	2	3	4	5
1.9	Scrum	1	2	3	4	5
1.10	Other, please specify ……………………………………………......	1	2	3	4	5

2. How long has your systems development methodology been in use in your virtual team?	
Less than 1 year	1
1 - 2 years	2
3 - 5 years	3
5 - 10 years	4
Over 10 years	5
I don"t know	6

3. What is the proportion of projects that are developed in your virtual team by applying systems development methodology knowledge (Horizontal use)?	
None	1
1 – 25 %	2
26 – 50 %	3
51 – 75 %	4
Over 75 %	5

4. What is the proportion of people in your virtual team that apply systems development methodology knowledge regularly?	
None	1
1 – 25 %	2
26 – 50 %	3
51 – 75 %	4
Over 75 %	5

5. Which of the following best describes how your virtual team makes use of its systems development methodology?	
A general guideline for all projects.	1
A standard which is followed rigorously for all projects.	2
Adapted on a project-to-project basis.	3

6. To what extent do you agree/disagree with the following statements (Perceived methodology support as production technology)?

		Totally disagree				Totally agree
6.1	Our systems development methodology helps to align the system to be developed with the business.	1	2	3	4	5
6.2	Our systems development methodology helps to capture requirements for the system to be developed.	1	2	3	4	5

6.3	Our systems development methodology helps to design the architecture of the system to be developed.	1	2	3	4	5
6.4	Our systems development methodology helps in system design.	1	2	3	4	5
6.5	Our systems development methodology helps in implementing developed systems.	1	2	3	4	5
6.6	Our systems development methodology helps in reviewing developed systems.	1	2	3	4	5
6.7	Our systems development methodology helps in testing developed systems.	1	2	3	4	5
6.8	Our systems development methodology helps to reuse earlier requirements, designs and code during systems development.	1	2	3	4	5
6.9	Our systems development methodology helps to involve end-users in systems development projects.	1	2	3	4	5
6.10	Our systems development methodology helps to build management commitment in our systems development projects.	1	2	3	4	5
6.11	Our systems development methodology helps to get the systems accepted.	1	2	3	4	5

7. To what extent do you agree/disagree with the following statements (Perceived methodology support as control technology)?

		Totally disagree				Totally agree
7.1	Our systems development methodology helps to decompose the system to be developed in workable parts.	1	2	3	4	5
7.2	Our systems development methodology helps to estimate the size of the system to be developed	1	2	3	4	5
7.3	Our systems development methodology helps to estimate the time and effort required for the development of a planned system.	1	2	3	4	5
7.4	Our systems development methodology helps to plan systems development projects.	1	2	3	4	5
7.5	Our systems development methodology helps in defining useful milestones for our systems development projects.	1	2	3	4	5
7.6	Our systems development methodology helps to organize systems development projects.	1	2	3	4	5
7.7	Our systems development methodology helps to keep our systems development projects under control in different locations	1	2	3	4	5
7.8	Our systems development methodology helps to estimate the project risks.	1	2	3	4	5
7.9	Overall, our systems development methodology helps us to manage our systems development projects.	1	2	3	4	5

8. To what extent do you agree/disagree with the following statements (Perceived methodology impact on the quality of the developed systems)?

		Totally disagree			Totally agree	
8.1	Our systems development methodology helps to develop more functional systems.	1	2	3	4	5
8.2	Our systems development methodology helps to develop more reliable systems.	1	2	3	4	5
8.3	Our systems development methodology helps to develop more maintainable systems.	1	2	3	4	5
8.4	Our systems development methodology helps to develop more portable systems.	1	2	3	4	5
8.5	Our systems development methodology helps to develop more efficient systems.	1	2	3	4	5
8.6	Our systems development methodology helps to develop more usable systems.	1	2	3	4	5
8.7	Overall, our systems development methodology helps to develop better systems.	1	2	3	4	5
8.8	Overall, our systems development methodology helps to make users more satisfied with our systems.	1	2	3	4	5

9. To what extent do you agree/disagree with the following statements (Perceived methodology support as cognitive and co-operation technology)?

		Totally disagree			Totally agree	
9.1	Our systems development methodology defines our desired systems development practice.	1	2	3	4	5
9.2	Our systems development methodology describes a sound way of developing systems.	1	2	3	4	5
9.3	Our systems development methodology forms a useful standard for our systems development.	1	2	3	4	5
9.4	Our systems development methodology reminds me about activities/tasks of systems development.	1	2	3	4	5
9.5	Our systems development methodology provides a useful list of possible systems development activities.	1	2	3	4	5
9.6	Our systems development methodology provides useful guidelines for conducting systems development.	1	2	3	4	5
9.7	Our systems development methodology provides a useful tool-box of techniques to be applied.	1	2	3	4	5
9.8	Our systems development methodology defines an ideal process of systems development that is useful, even though it is not followed in practice.	1	2	3	4	5
9.9	Without a systems development methodology one cannot estimate how systems development should be conducted.	1	2	3	4	5
9.10	Our systems development methodology allows us to learn from our systems development experience.	1	2	3	4	5
9.11	Without a systems development methodology it is impossible to evaluate our systems development practice.	1	2	3	4	5

10. To what extent do you agree with the following statements? (Perceived methodology impact on the quality and the productivity of the development process).

Totally disagree   Totally agree

10.1	Our systems development methodology helps to develop new applications faster.	1	2	3	4	5
10.2	Our systems development methodology helps to improve the functionality of new applications.	1	2	3	4	5
10.3	Our systems development methodology helps to increase the productivity of the application developers.	1	2	3	4	5
10.4	Our systems development methodology helps to decrease the cost of systems development.	1	2	3	4	5
10.5	Our systems development methodology helps to improve the quality of the systems.	1	2	3	4	5
10.6	Our systems development methodology helps to decrease the cost of systems maintenance.	1	2	3	4	5
10.7	Our systems development methodology helps to improve the documentation of the systems.	1	2	3	4	5
10.8	Our systems development methodology improves the morale in my virtual team.	1	2	3	4	5
10.9	Our systems development methodology helps to achieve the goals of my virtual team.	1	2	3	4	5
10.10	Our systems development methodology helps to improve my virtual teams reputation of excellent work.	1	2	3	4	5

6. What are your expectations for the use of systems development methodologies in virtual team?	
Make more use of our systems development methodology.	1
Replace our systems development methodology.	2
Supplement our systems development methodology with other methodologies.	3
Abandon the use of our systems development methodology.	4
No change.	5
Other, please specify ………………………………………………..…… ……………………………….......	6

Section 3: Virtual software development teams

Virtual teams refer to a group of people or team members who are scattered geographically, working as employees and using combination of telecommunication and information technologies in order to achieve organisational goals.

The following questions should be related to your **most recent work** in a virtual team. Please, read each phrase on the left carefully, and answer the questions by putting an “**X**” on the right.

1. To what extent do these challenges affect your virtual team?

	Totally disagree				Totally agree
Time zones	1	2	3	4	5
Language	1	2	3	4	5
Holidays, local laws, customs	1	2	3	4	5
Technology	1	2	3	4	5
Distance	1	2	3	4	5
Personal conflicts and misunderstanding	1	2	3	4	5
Culture: Organizational difference	1	2	3	4	5
Culture: National difference	1	2	3	4	5
Other, please specify ……………………………………………… ………………………………………………	1	2	3	4	5

2. Meetings					
	Never	**Once a year**	**Twice a year**	**Three times a year**	**More than three times a year**
How often does your virtual team meet in person?					

3. How often do you use the following communication tools for your virtual team?					
	Never	Rarely	Sometimes	Often	Always
E-mail					
Any virtual collaboration tool					
Groupware					
Voice over IP applications					
Special websites for collaboration					
Industry-specific collaborations tools					
Video conferencing					
Whiteboard applications					
Social networking services					
Wikis or knowledge blocks					
Land or mobile phone					
Other, please specify ……………………………………………… ………………………………………………					

4. What benefits have you experienced while working on virtual team?

	Totally disagree				Totally agree
Ability to bridge time and space	1	2	3	4	5
Cost saving	1	2	3	4	5
Flexible to work at anytime and anywhere	1	2	3	4	5
Communication is flexible	1	2	3	4	5
Bring under control the talent and knowledge to solve problems	1	2	3	4	5
Tasks are completed on a 24/7 schedule	1	2	3	4	5
Maintains an effective relationship with employees and customers	1	2	3	4	5
Enables organisations to attain a distant geographic reach	1	2	3	4	5
Outcomes are normally stored electronically	1	2	3	4	5
Easy review and performance	1	2	3	4	5
Other, please specify **..** **..**	1	2	3	4	5

5. **Please state your level of agreement with the following statements about the effectiveness of virtual teams by considering the virtual team that you are currently working/has worked with**.

		Totally disagree			Totally agree	
5.1	Customer satisfaction for the tasks/work that my virtual team contributed is high	1	2	3	4	5
5.2	Customer complaints for the tasks/work that my virtual team contributed is low	1	2	3	4	5
5.3	The tasks/work that we have already completed have less revisions/reworks	1	2	3	4	5
5.4	My virtual team is getting more and more work from our satisfied existing customers	1	2	3	4	5
5.5	My virtual team has a very good recognition from all the stakeholders because of completed work/tasks	1	2	3	4	5
5.6	My virtual team is able to increase the customer base through superior task/work performance	1	2	3	4	5

6. **Please state your level of agreement with the following statements on virtual teams based on the virtual team working experiences that you have in your current workplace.**

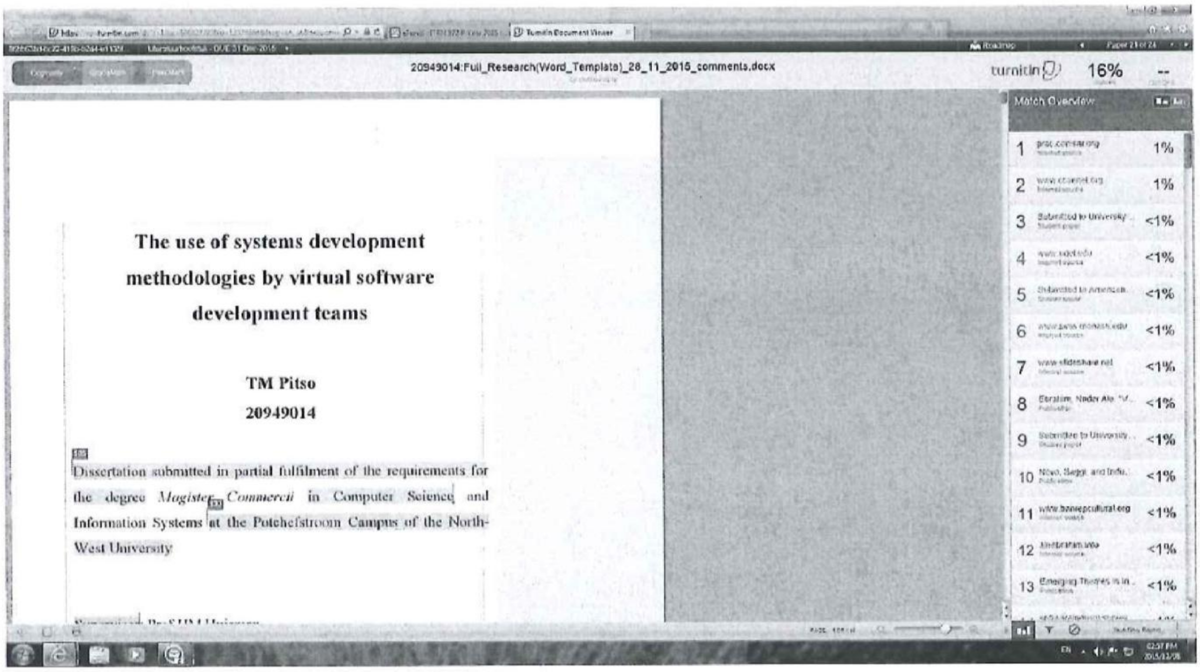
		Totally disagree			Totally agree	
6.1	Software development used in my organization encourages virtual team members to perform better in work that they are doing	1	2	3	4	5
6.2	Compensation systems of my organization encourage virtual team members to perform better in their jobs	1	2	3	4	5

6.3	Clear understanding of roles and responsibilities by each virtual team member improves the effectiveness of the virtual team performance	1	2	3	4	5
6.4	Providing clear goals and objectives to the team help to	1	2	3	4	5
	establish the commitment to the virtual team work					

7. How would you rate the overall success of the most resent project developed by your virtual team according to the following categories?

		Very low			Very high	
7.1	Product	1	2	3	4	5
7.2	Process	1	2	3	4	5
7.3	Customer satisfaction	1	2	3	4	5
7.4	Leadership	1	2	3	4	5
7.5	Communication	1	2	3	4	5
7.6	Competencies	1	2	3	4	5
7.7	Other, please specify	1	2	3	4	5

Thank you very much for your time and cooperation!


20949014:Full_Research(Word_Template)_26_11_2015_comments.docx
turnitin
16%
Match Overview
1 proc.conisar.org 1%
2 www.ccsenet.org 1%
3 Submitted to University... <1%
4 www.sdsu.edu <1%
5 Submitted to American... <1%
6 www.pass.monash.edu <1%
7 www.slideshare.net <1%
8 Ebrahim, Nader Ale "V... <1%
9 Submitted to University... <1%
10 Niazi, Saggi, and Indu... <1%
11 www.bonneproudural.org <1%
12 Alrebraham.edu <1%
13 Emerging Themes in In... <1%
The use of systems development methodologies by virtual software development teams
TM Pitso
20949014
Dissertation submitted in partial fulfilment of the requirements for the degree Magister Commercii in Computer Science and Information Systems at the Potchefstroom Campus of the North-West University

Declaration

This is to declare that I, Annette L Combrink, accredited language editor and translator of the South African Translators' Institute, have language-edited the dissertation

by

TM Pitso (20949014)

with the title

The use of systems development methodologies by virtual software development teams

Prof Annette L Combrink

Accredited translator and language editor

South African Translators' Institute

Membership No. 1000356

Date: 7 December 2015

MOOIRIVIEROEWERTRUST

IT3 152/2009

PLOT 25 VYFHOEK, PO BOX 19124, NOORDBRUG, 2522

TEL: 082 551 9840, FAX 086 295 4164

(ANNETTE COMBRINK, ACCREDITED TRANSLATOR AND LANGUAGE EDITOR, SOUTH AFRICAN TRANSLATORS' INSTITUTE)

Invoice

Invoice number: 2015/224

Date: 7 December 2015

Client: Thato Pitso

Title of document	Service (language editing, translation, certification)	Number of words	Rate	Total
Dissertation	Language editing	33648	R10 per 100 words (student rate)	R3364.80
	Translation summary	509	R35/100 words	R178.15
			Minus discount	R542.95
Total				**R3000.00**

Banking details

Mooirivieroewertrust, Nedbank, Current account
Branch number: 171338
Branch name: Potchefstroom, Account number: 1005163383