A Framework for a Successful Collaboration Culture in Software Development and Operations (DevOps) Environments

by

KOOS THEMBA MASOMBUKA

submitted in accordance with the requirements for

the degree of

DOCTOR OF PHILOSOPHY

in the subject

Computer Science

at the

UNIVERSITY OF SOUTH AFRICA

SUPERVISOR: Prof. E. Mnkandla

February, 2020

Abstract

Traditional software development methodologies are historically used for the creation of software products in separate departments, namely development and operations departments. The development department typically codes and tests the software, whilst the operations department is responsible for its deployment. This siloed arrangement is not aligned to modern practices, which require a timeous response to changes without necessarily delaying the product release. DevOps culture addresses this silos problem by creating an enabling environment for the two departments to collaborate throughout the software development life cycle. The successful implementation of the DevOps culture should give an organisation a competitive advantage over its rivals by responding to changes much faster than when traditional methodologies are employed. However, there is no coherent framework on how organisations should implement DevOps culture. Hence, this study was aimed at developing a framework for the implementation of DevOps culture by identifying important factors that should be included in the framework.

The literature survey revealed that open communication, roles and responsibility alignment, respect and trust are the main factors that constitute DevOps collaboration culture. The proposed framework was underpinned by the Information System Development Model which suggests that the acceptance of a new technology by software developers is influenced by social norm, organisational usefulness and perceived behavioural control.

A sequential mixed method was used to survey and interview respondents from South Africa, which were selected using convenience and purposive sampling. Statistical analysis of the quantitative data acquired through the questionnaire followed by a qualitative analysis of interviews were undertaken. The results showed that open communication, respect and trust are the key success factors to be included in the framework. The role and responsibility factor was found not to be statistically significant.

This study contributes towards the understanding of factors necessary for the acceptance of DevOps culture in a software development organisation. DevOps managers can use the results of this study to successfully adopt and implement DevOps culture. This study also contributes to the theoretical literature on software development by identifying factors that are important in the acceptance of DevOps collaboration culture.

Declaration

Name: Koos Themba Masombuka

_

Student number: 32637004

_

Degree: Doctor of Philosophy in Computer Science

_

Exact wording of the title of the thesis as appearing on the electronic copy submitted for examination:

A Framework for a Successful Collaboration Culture in Software Development and Operations (DevOps) Environments.

I declare that the above thesis is my own work and that all the sources that I have used or quoted have been indicated and acknowledged by means of complete references.

I further declare that I submitted the thesis to originality checking software and that it falls within the accepted requirements for originality.

I further declare that I have not previously submitted this work, or part of it, for examination at Unisa for another qualification or at any other higher education institution.

SIGNATURE

14/11/2020
DATE

Acknowledgment

I wish to thank the following people who offered assistance to me to make this thesis possible:

1. Prof. E Mnkandla, the Director of School of Computing who is also my supervisor. Thank you for all the effort and time you set aside to make this thesis a success. I would also like to thank you for your patience and understanding when I was down. Lastly, thank you for the encouragement and timeous feedback you gave to me.

2. Dr. Chiyangwa for assisting with the statistical analysis.

3. Dr. Tshepo for language editing of this thesis.

Dedication

I dedicate this thesis to my mother Nomvula Masombuka and my late father Paulos Masombuka. Thank you for encountering me to keep going. I also dedicate it to my wife Nana, my daughter Sinethemba and my son Siyamthemba.

This is for all of you.

Publications from this Thesis

Masombuka, T. and Mnkandla, E., 2018, September. A DevOps collaboration culture acceptance model. In Proceedings of the Annual Conference of the South African Institute of Computer Scientists and Information Technologists (pp. 279-285). ACM.

Contents

List of Figures

List of Tables

List of Acronyms

CAMS	Culture, Automation, Measurement and Sharing
DCAM	DevOps Culture Acceptance Model
DevOps	Development and Operations
DOI	Diffusion of Innovation
EFA	Exploratory Factor Analysis
INT	Intentions to follow DevOps processe
ISD	Information System Development
ISDAM	Information Systems Development Acceptance Model
MPCU	Model of Personal Computer Utilisation
OC	Open Communication
OCM	Organizational Change Management
OU	Organisational Usefulness
PBC-I	Perceived Behavioural Control-Internal
RE	Respect
RR	Roles and Responsibilities alignment

SDLC	Software Development Life Cycle
SMD	Software Development Methodologies
SN	Social Norm
TAM	Technology Acceptance Model
TPB	Theory of Planned Behaviour
TR	Trust
TRA	Theory of Reasoned Action
UTAUT	Unified Theory of Acceptance and use of Technology

Chapter 1

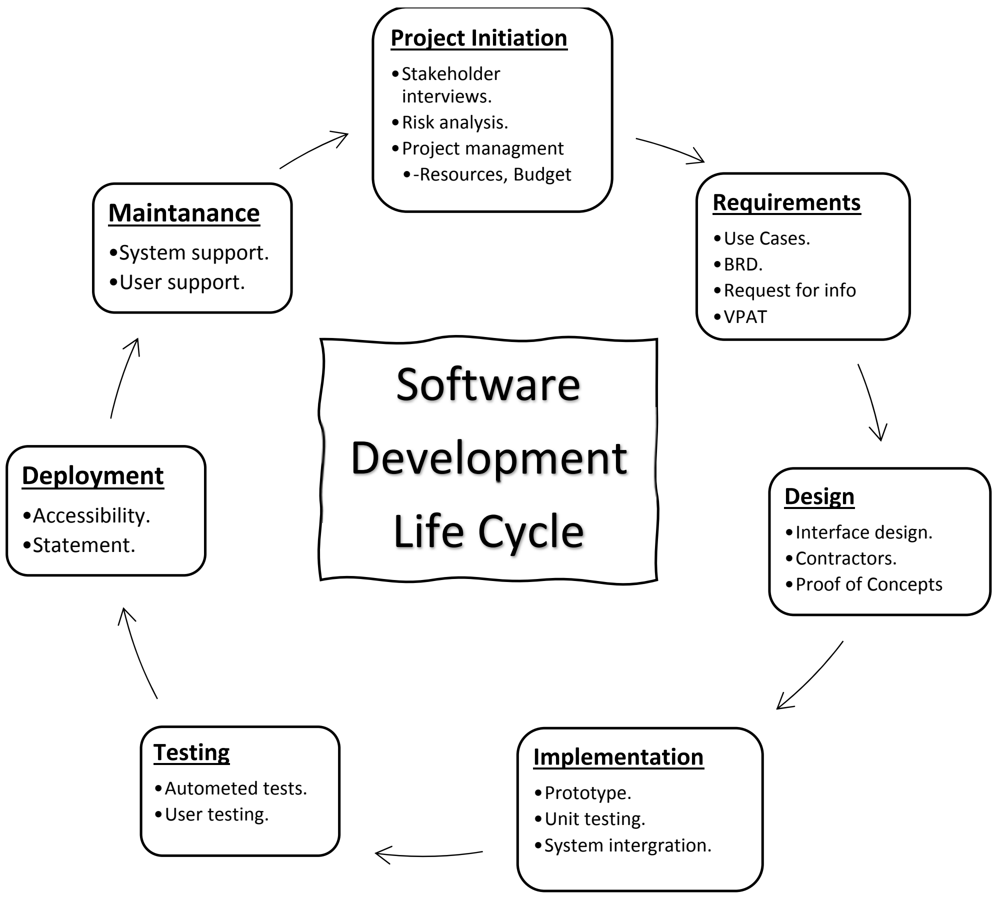
Background

1.1 Introduction and Background

The general goal of software development projects is to deliver software of specific previously agreed upon functionality. In addition, the project should deliver on schedule and inside the permitted spending plan. Ahmad et al. (2015, p. 1) has reported that only 16.2% of software projects are successfully completed. One of the reasons behind the late and over budget delivery of software is the constant changing of user requirements (Ahmad et al., 2015, p. 1). Some software methodologies allow user requirements to change during the entire phases of software development life cycle.

Figure 1.1 shows a typical Software Development Life Cycle (SDLC). The model is iterative in nature, which means a phase can be repeated as many times as necessary. This allows new features of the software to be added in an incremental fashion and thus improving the old features in the process. Changing client requirements are catered for by the iterations. However, changing requirements during the life cycle may influence the cost, planned schedule and product quality. In an investigation led by Javed et al. (2004), it was discovered that insufficient communication with the customer is one of the potential reasons behind late changes to the requirements.

According to Guardado (2012, pp. 15-17), some software project failures are induced by the culture of an organisation. That is, with the exception for late user requirements changes that may lead to a project being delayed or even resulting in failure, the culture of traditional software development may be a contributory factor. Traditionally, two separate divisions, namely the development and the operations departments, are tasked with the development of software. The responsibility of the development department includes the



BRD is Business Requirements Report.
VPAT is Voluntary Product Accessibility Template.

Figure 1.1 – Software Development Life Cycle (University of Melbourne, 2003, n.p).

gathering of user requirements and the development (actual coding) of a fully functional software. Put differently, this development department is responsible for phases 1 to 5 of the SDLC (see Figure 1.1). On the other hand, the operations department is responsible for the deployment and maintenance of the software, that is, phases 6 and 7 of the SDLC (see Figure 1.1). When the software product is sent for deployment, all the necessary exercises that will prepare the product ready for use such as, but not limited to, running the software product under the client's system settings, setting up the software, and ensuring that the installed version is up to date are conducted (Schach, 2002, p. 515).

The development team builds up the software product and utilise testing tools which are available for use by this department (Hüttermann, 2012*a*, p. 6). At the point when the team is happy with the usefulness of the product, the software is moved to the operations department for deployment. Since the operations team has to make sure that the system is ready for client use, the operations personnel have to test the software and configure it on the system representing the real user system. Hüttermann (2012*a*, p. 6) warns that testing done in the development department may miss some of the bugs, only to be discovered in the deployment department. Subsequently, finding bugs at this phase of advancement may cause delays to have them fixed in light of the fact that the product may have to come back to the development department. Requirements constantly change (Hüttermann, 2012*a*, p. 5), and the separation of the two departments into silos causes delays when the need to react to problematic issues arises.

Addressing the problems that are found when the software product is already in the hands of the operations department may result in rescheduling of the project plan as the product is sent back to the development department. This may be coupled with the budget increases. The same is applicable to requirements changes when the software is in the later phase of development.

This department separation culture does not promote free communication between the client, the development and operations teams. In addition, the development and operations teams are not encouraged to collaborate. Development and Operations (DevOps) addresses these challenges by removing barriers between the development and operations departments (Debois, 2012; Elbayadi, 2014; Hüttermann, 2012*b*; Willis, 2010). In this manner, a bridge between the two departments is created. Walls (2013, pp.5-7) defines the DevOps culture in terms of aspects of open communication amongst team members, the alignment of incentive and responsibility, respect for each other and trusting one another. In a culture created by DevOps, the software developing teams, which includes the development and operations teams, collaborate with each other during the entire software development period. In that way, everyone involved is responsible for delivering a working software product at the end of the deployment phase (Barnhart et al., 2009, p. 1). Developing software products in a collaborative DevOps culture means that the development and operations teams should be involved from the start to the end of the project. Thus, the operations teams should be present when software requirements are gathered. Similarly, the development team should also be involved in the release of the software product to the client. A client representative is also encouraged to participate throughout the lifespan of the project.

According to a worldwide survey conducted in 2013 by CA Technology (2013, p. 303) on information technology (IT) organisations, 39% of organisations indicated to have completed with the process of DevOps adoption whilst 27%

were still planning to adopt. However, not everyone was familiar with what DevOps is as 16% of the respondents have indicated. This survey revealed that the biggest driving force behind DevOps was a collaboration between the two departments. Nugent (2012, p. 12) describes collaboration as working together jointly to achieve a common goal. Several authors, (Edwards, 2010; Hüttermann, 2012*b*; Pant, 2009; Willis, 2010) agree that one of the critical aspects of DevOps is a collaboration culture. In short, DevOps can be viewed as a "set of processes, methods and system communications, collaboration and technology operations" (Pant, 2009, p. 12).

A review of the literature on DevOps characteristics has identified the culture of collaboration as one of the challenges being faced by DevOps (Erich et al., 2014, p. 9; Khan & Shameem, 2020, p. 12). In Erich et al. (2014, p. 9)'s view, the handling of cooperation between the development and operations staff is not adequate and needs interventions. As suggested by Tessem and Iden (2008, pp. 107-108), a need therefore exists for an improvement of the software process methodologies to include cooperation. Unlike collaboration, cooperation is an informal relationship between organisations without explicitly defined goals (Nugent, 2012, p. 12).

Several authors (Edwards, 2010; Erich et al., 2014; Hüttermann, 2012*b*; Pant, 2009) acknowledges that collaboration is one of the core elements of DevOps. Research on employees' perceptions of what constitutes effective teamwork and collaboration has been undertaken (Mattessich & Monsey, 1992; LaFasto & Larson, 2001;Rosen, 2007). Several models are known on important keys to creating successful teams that collaborate effectively (Foltz, 2018; Richards et al., 2016). Attempts have also been made to demonstrate how to instil a culture that promotes collaboration in an organisation (Winer & Ray, 1994;

Gratton & Erickson, 2007).

Walls (2013, p. 3) explains how culture is formed by a group of people sharing values and behaviours. Moving from a culture that favours individuals to a culture that promotes collaboration and teamwork may not be straightforward as people would like it to be. Common values and beliefs shared by individuals within an organisation define organisational culture (Dasgupta & Gupta, 2010, p. 4). Organisational culture continues to be studied because it forms the most general form of organisational change (Nugent, 2012, p. 11). Albeit organisational culture is continually changing, Trompenaars and Woolliams (2004, p. 46) contend that organisational culture is hard to change. This is because it speaks to accumulated learning that occurred over a period of time and patterns of related assumptions of a group within an organisational structure. Organisational policies and processes are a formal indication of organisational culture, whilst employees' behaviour is informal indicators. (Rosen, 2007, p. 23).

McLagan (2002, p. 76) and Attaran (2004, p. 43) have acknowledged that the introduction of a new technology sometimes requires a modification of the requisite processes, which may involve changing the entire organisational culture. In fact, the leading challenge reported by Saugatuck Technology (2014) survey on DevOps implementation is overcoming habits associated with culture within organisations. The need for Organisational Change Management (OCM) strategies become relevant in addressing challenges that come as a result of introducing new technology Markus (2004, p. 32). Guardado (2012, p. 17) confirms that the tools provided by OCM can be used to study organisational changes.

Besides preparing users for organisational culture change using OCM tools, factors that influence user acceptance of new technology should be studied (Jones & McCarthy, 2009; Guardado, 2012). There is a number of Technology Acceptance Models (TAMs) that have been proposed over the past years. These models were developed to understand how users behave when faced with a new technology. Based on Fisher and Howell (2004, p. 247), TAMs enable factors that influence the intentions of the user to accept new technologies to be studied. DevOps adoption and implementation may be more than just adopting a new technology. In some cases, it may mean giving up old processes and adopting new ones. Therefore, to assist the organisation and its employees to cope with such type of transitions, OCMs and TAMs need to be well understood.

Although work on collaboration in general terms has been extensively undertaken, no work in the literature was found, by the author of this thesis, that has been carried out in the DevOps settings. In this research study, an attempt was therefore made to understand collaboration in a DevOps environment. Factors that are necessary for successful collaboration between DevOps teams in a DevOps environment were determined. To increase the level of acceptance of the new technology by all stakeholders and thus make the transition to the new technology as smooth as possible requires that TAMs be considered when introducing a new technology.

Many studies appearing in the literature shows a move taken by organisations is towards a collaborative DevOps culture approach. Therefore, it was very important that a study like this one was undertaken with a view to develop a framework that will benefit organisations in their transition journey towards a collaborative DevOps culture.

1.2 Statement of the Research Problem

In an enterprise environment, traditional software methodologies separate software development teams from the operations teams. The development team develops the software until it meets the requirements of the user or client. Once the software has been developed, it is transferred to the operations department where it is deployed. During the deployment phase, the operations team tests the software using the system settings of the real user. Software problems that were previously not obvious may be discovered during these tests. Dijkstra (2013, p. 24) points out that different methods or tools and release cycles of the two departments may not be aligned. The problems occurring in one department may be blamed on the other department down the software delivery pipeline. The problems and issues arising from the fact that the units operate in silo need to be dealt with before the software is released to the client. However, procedures that may need to be followed before these issues can be fixed may delay the software delivery process. Ultimately, such a delay may lead to the project running beyond the stipulated time and being over budget or the software being released missing important features. In worse cases, this may lead to a project failure.

DevOps addresses these problems by removing the barrier between the development and operations departments. DevOps requires that the two departments collaborate during the entire software development process (Edwards, 2010; Erich et al., 2014; Hüttermann, 2012*b*; Pant, 2009). This collaborative culture promotes product ownership whereby staff work together to resolve problems. Hence, problems are discovered very early in the development phase and are thereafter dealt with accordingly, resulting in a continuous delivery of the final software product. Changing customer requirements can be managed through DevOps on a continuous delivery basis, thus giving a DevOps organisation a competitive advantage over traditional software development organisations (Elbayadi, 2014; Erich et al., 2014; Wettinger et al., 2014).

Even though DevOps can be seen a the solution to the above problems, organisations are still facing challenges in adopting it. Leite et al. (2019, pp. 27–28) maintains that DevOps collaboration culture is still one of the challenges facing DevOps deployment within organisations. Kamuto and Langerman (2017, pp. 48–51) mentions the lack of education around DevOps, change resistance and silo mentality as some of the challenges hindering the adoption. Senapathi et al. (2018, p. 65) also indicated unclear responsibilities as an additional challenge. In their study on optimizing DevOps challenges, Khan and Shameem (2020) have shown that people challenges—one of them being collaboration—should be given priority in an organisation in order to successfully adopt DevOps. This work is corroborated by Akbar et al. (2020) that DevOps organisations should focus on culture rather than tools. It is this case that made this study worthwhile.

There was therefore a lack of a coherent framework for the acceptance of a collaborative culture to present a clear approach to the processes that support changes in the organisational culture; encouraging a collaborative DevOps culture. Bringing the development and operations departments together presents new challenges because of different tools, documentations, standards and other norms that are used by these departments (Dijkstra, 2013; Erich et al., 2014). The process of adoption of new technology in an organisation should be managed well in order to ensure that the new technology is well received and accepted.

In summary, the framework made the following contributions:

- Identified the necessary factors for a collaborative DevOps culture.
- Determined effective relationships amongst factors that contribute to successful acceptance; and
- Provided guidelines on how to effectively adopt a collaborative culture in DevOps environments.

The goal of this research study was to use the general collaboration characteristics, the collaboration characteristics of DevOps, and successful technology acceptance factors to propose a framework as a solution to this problem.

1.3 Research Questions

This study was aimed at answering the main research question which was stated as follows:

RQ *What are the success factors that must be included in a coherent framework to support a successful collaborative culture in a DevOps environment?*

To answer the above main question, the following sub-questions were formulated:

SQ1: *What are the characteristics of DevOps collaborative culture?*

SQ_2: *What factors encourage collaborative culture in a DevOps environment?*

SQ_3: *How do the contributing factors interact with each other?*

SQ_4: *How can a collaborative culture be implemented in a DevOps environment?*

SQ_5: *Does the proposed framework support the collaborative culture in a DevOps environment?*

1.4 Research Study Objectives

Two research objectives that this study achieved, namely the primary and secondary objectives, are discussed below.

1.4.1 Primary objective

The main objective of this research study was to propose a framework that will allow the acceptance of the DevOps collaboration culture within DevOps environments.

1.4.2 Secondary objectives

The identified secondary objectives and the respective descriptions of how each of the secondary objectives was going to be achieved are as follows:

Obj_1: To investigate DevOps collaboration requirements.

- Using the available literature, a search investigation was conducted to determine the generic and DevOps collaboration characteristics, processes and approaches to software development.

  Objective aim: To identify the requirements for the definition of the framework.

Obj2: To determine the necessary factors that promote a successful collaborative culture.

- Using the available literature, a search investigation was conducted to identify these key factors.
- In addition, surveys were conducted with a view to capture these factors from people who are already involved in DevOps.

  Objective aim: To identify the important key factors that were included in the framework.

Obj3: To evaluate the key success factors for their interaction with one another.

- Using the literature that was available, a search was conducted to determine the relationships between these factors.
- In addition, surveys were conducted to measure these interactions.

  Objective aim: To determine what relationships exist between these factors.

Obj4: To determine how the interaction of the contributing factors contributes to successful collaboration in a DevOps environment.

- Using the literature that was available, a search was conducted to compare the interactions as measured in this study.
- In addition, analysis of the surveys was conducted.

  Objective aim: To understand which of the relationships between the factors add value to a successful collaboration within a DevOps environment.

Obj5: To propose a comprehensive framework for a collaborative DevOps acceptance.

- A framework was proposed from the synthesis of recommendations and guidelines gleaned from the current body of knowledge.

  Objective aim: To propose a framework for the successful adoption of a collaborative culture within a DevOps environment.

1.5 Purpose and Significance of the Study

The purpose of this research was to develop a coherent framework to support a successful collaborative culture in a DevOps environment. The developed comprehensive framework made the following contributions:

- The key factors that are necessary for a successful collaborative DevOps culture were presented.

- Relationships between these factors were identified.
- Created theoretical knowledge and perceptions of the successful acceptance of DevOps culture.
- Provided guidelines on how organisations intending to adopt DevOps culture should proceed.

1.6 Research Design and Methodology

This research study was based on a pragmatist philosophy that is underpinned by assumptions that result in acceptable research standards. As already mentioned, the main research question for this study was: *What are the success factors that must be included in a coherent framework to support a successful collaborative culture in DevOps environment?* In order to answer this question, research processes and guidelines were followed.

According to Creswell (2003, pp. 4-5), research design requires three questions to be addressed. These questions are about the following:

- The knowledge claims of the investigating researcher.
- The strategies that will be used. These strategies determine which procedures the researcher will use.
- The methods that will be used for the collection and analysis of the data.

Pragmatism is the knowledge claim or a research philosophy that has been adopted for this research study. When this philosophy is used, a researcher is given the freedom to combine different methodologies to address the needs of the research question. This research study therefore combined the quantitative and qualitative methodologies in its enquiry. In this research study, surveys and interviews were used to collect quantitative and qualitative data respectively.

Another issue that needs to be addressed when choosing a research approach is the research methods to be used for the collection and analysis of the data of a research study. The instrument that was used in this research study for the collection of the quantitative data is a questionnaire whereby close ended questions were asked. The questionnaire was followed in a sequential manner by online interviews, which were geared towards the collection of qualitative data.

The collected quantitative data was analysed using relevant statistical techniques with the help of Statistical Package for Social Sciences (SPSS) v25.0 tool. Content analysis was used for the analysis of the data collected from the qualitative interviews using the ATLAS.ti tool.

A comprehensive discussion of the methodological approach of this research study is presented in Chapter 4.

1.7 Reliability and Validity

As described by Payne and Payne (2004, p. 196), reliability and validity are important aspects of conducting research that should be pursued by researchers to show the credibility and trustworthiness of the research results. Various techniques are used to measure the reliability and validity of quantitative and qualitative studies. Details of how this study has adhered to reliability and validity principles are outlined in Section 4.5 of this thesis.

1.8 Research Ethics

Ethical issues are important when conducting research, especially where humans are involved. Fowler (2009, p. 163) explains that all research involving human subjects need to follow ethical guidelines to ensure that no subjects are subjected to some form of suffering induced by the survey. This research study was carried out by adhering to the ethics policy of the University of South Africa (UNISA). The policy requires that all research undertaken under the auspices of the University should be reviewed by the Ethics Committee of UNISA prior to the commencement of the research study. This study complied with this requirement and the requisite approval to continue with this research study was granted by the Committee in the form of an Ethical Clearance Certificate (see Appendix A).

The purpose of the survey was explained to both the subjects that are going to fill-in the questionnaire and/or being interviewed. The study adhered to the following guidelines regarding the participants:

- Their participation and opinions were solely for the purpose of the study and were treated with the strictest confidentiality.
- The participation was voluntary and their right to discontinue at any point in time during the process was recognised.

- The consent of the participants to participate in the research study was obtained and consent forms were signed by all willing participants (see Appendix C).
- The time it would take to complete the survey was made known to the participants in advance prior to the commencement of the survey.
- Permission to record participants in the case of interviews was sought and acquired in advance prior to the commencement of the interviews.

A detailed discussion on ethical considerations is presented in Section 4.7.

1.9 Delimitations

This research study focused exclusively on the human collaboration aspect of DevOps. The research study only involved DevOps practitioners based in South Africa who develops software by following the DevOps collaboration practices.

This research did not explore technologies that are used for collaboration. Other DevOps elements (i.e. Automation, Measurement, and Sharing) did not form part of this research study.

1.10 Terminology

In this research study, the following words and/or phrases shall have the following meanings:

Collaboration is the "working together jointly with others or together especially in an intellectual endeavour"(Nugent, 2012, p. 12).

Cooperation is a "short-term, informal relationship between organizations without explicitly defined goals, objectives or joint structure" (Nugent, 2012, p. 12).

Organisational culture are "common values and beliefs shared by individuals within an organisation" (Dasgupta & Gupta, 2010, p. 4).

DevOps developer represents "software developers, including programmers, testers, and quality assurance personnel" (Hüttermann, 2012*a*, p. 4).

DevOps operator represents the "experts who put software into production and manage the production infrastructure, including system administrators, database administrators, and network technicians" (Hüttermann, 2012*a*, p. 4).

DevOps practitioner is a person carrying the developer or operator role in a DevOps environment.

1.11 Chapter Outline

This chapter presented the background information on the research topic. The research problem, which this study intended to investigate, was thereafter introduced. The chapter identified research questions that needed to be studied during the investigation of the research problem. The purpose and the significance of investigating the research problem were mentioned together with the objectives of the study. The scope of the research project was confirmed and the research approach that was followed was also revealed.

Figure 1.2 shows an outline of this research study.

Chapter 2 provides a detailed review of the literature to show the current work in the relevant fields of collaboration.

Chapter 3 presents the technology acceptance models from which the theoretical framework of this study was built.

Chapter 4 discusses the research methodology this study followed for answering the research questions.

Chapters 5 and 6 present the results obtained as well as their analysis. Whereas

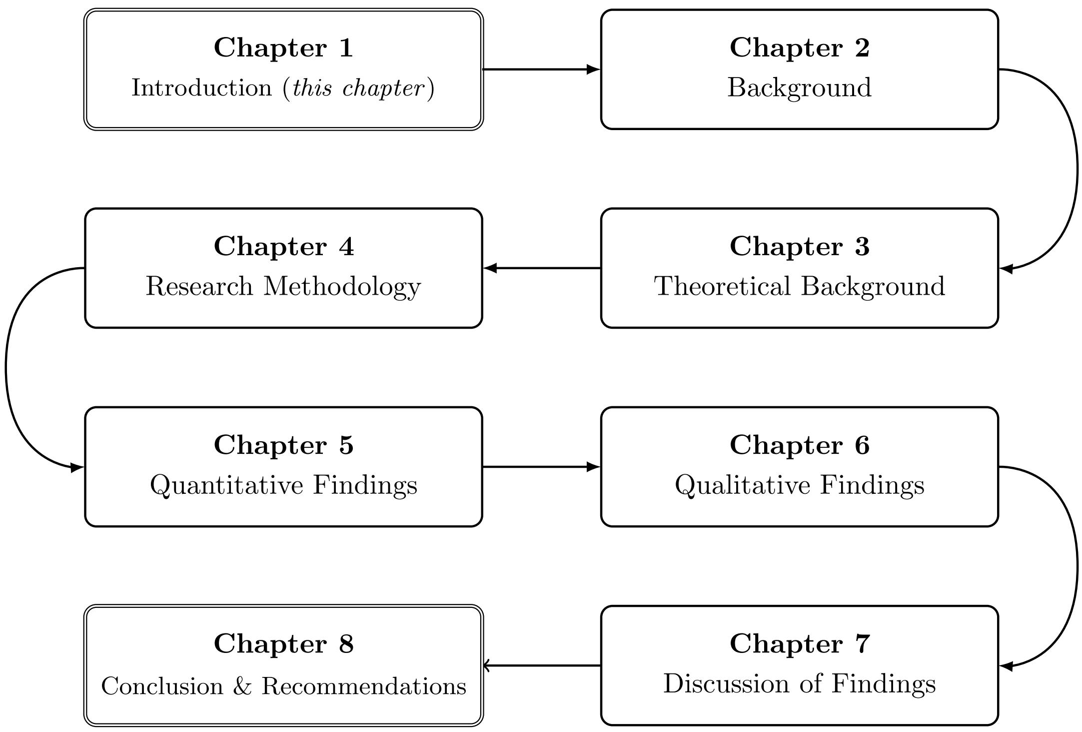


Figure 1.2 – Chapter outline.

Chapter 5 presents the results from the quantitative analysis, Chapter 6 discusses the results from the qualitative analysis perspective.

Chapter 7 provides an interpretation of the results presented in the previous two chapters.

Chapter 8 demonstrates how this research study has addressed the research question and highlights the contribution of this research study to the knowledge base. This chapter constitutes the conclusion of this research study.

Chapter 2

Literature Review

2.1 Introduction

In the previous chapter, a brief background of this research study was presented. To this end, the problem that this research study seek to address was explained and the significance of undertaking this research study was outlined together with the research questions. This literature review gives an introduction to the key areas, namely traditional software development practices and collaboration and the DevOps collaboration culture, which are necessary for the research problem of this research study to be addressed. Other key theories including the technology acceptance theories, which are important for the formulation of the theoretical framework for this research study, are highlighted. Technology acceptance theories are discussed in greater detail in Chapter 3.

2.2 Software Development

Software development organisations follow a specific software development model when developing software within their organisations. Some of these models are custom-made for the organisation; alternatively an organisation can use the pre-existing models. Komma (2010, p. 230) mentions four of these models, namely: Software Development Life Cycle (SDLC), Prototype Model, Rapid Application Development Model and Component Assembly models.

Schach (2002, pp. 8-9) describes a classical SDLC as having the following six phases: Requirements, Specification, Design, Implementation, Maintenance and Retirement. Variations of SDLC exist, however, all of them follow a similar pattern. Figure 1.1 shows one of such variation (University of Melbourne, 2003, n.p).

SDLC is an iterative process in which every phase is repeated as many times as necessary until the final functional software is delivered. Sudhakar (2005, p. 2) acknowledges that software development is a dynamic process characterized by change. Changing requirements during the life cycle may affect the budget, schedule and quality of the product, as mentioned in Section 1.1. According to Ferreira et al. (2003, p. 28) requirements changes occur anytime during the SDLC. In a study conducted by Javed et al. (2004), it was found that inadequate communication with the client is one of the reasons for late requirements changes. Several requirements engineering practices have been adopted to address the challenges that come with this traditional way of software development (Debbiche et al., 2019, p. 5).

Guardado (2012, pp.15-17) reckons organisational culture has an effect on software project failures. Departmental silos brought about by traditional software development practices do not encourage collaboration and open communication between and within teams. Section 1.1 of this thesis gave a broader discussion on challenges brought by these traditional practices.

Adler (2003, p. 16) has added that traditional system development organisations tend to adopt a managerial style that:

- Is command and control based;
- Has high formalisation and standardisations; and
- Has customer participation that is only high during the specification and implementation phases.

This managerial style does not encourage collaboration, free communication and product ownership as per the recommendations of DevOps. The proposal by Tessem and Iden (2008, p. 108) that software process methodologies should be improved to include cooperation has resulted in the development of an SDLC Model 2010 by Ragunath et al. (2010). The model encourages user-developer cooperation during development. Rütz (2019, p. 8) acknowledges that although people may want to stick to their old habits, with the right leadership style, it is possible to change.

The culture of an organisation that allows teams to collaborate, communicate openly with each other and share resources and knowledge is envisaged by DevOps. Research has shown that collaboration, open communication and sharing leads to high project success rate, and this is evident in emergent of methodologies such as Agile and DevOps. These methodologies or practices have shifted focus towards human soft skills that are geared towards encouraging working towards a common goal and having the same vision.

2.3 DevOps Movement

Figure 2.1 shows that DevOps is increasingly gaining popularity. This may be ascribed to the fact that the enterprise software industry is shifting towards DevOps capabilities (Liu et al., 2014, p. 41). In traditional software development methodologies, some software problems are unearthed or discovered during the deployment phase of the software development process. This may cause delays in the release of the software since bugs and performance issues that could not have been detected and fixed during the development phase are discovered later during the deployment phase. One reason such bugs may go undetected in the development department is the different tools used between the two departments to meet that department's objectives or goals. Dijkstra (2013, p. 24) mentions that the operations department may be using different methods such as ITIL[1], ASL[2], and BiSL[3], and their release cycles may not be aligned with that of the development department as mentioned in Section 1.2. As a result, departmental procedures may need to be followed by the operations department personnel to get the development department

[1] Information Technology Infrastructure Library
[2] Application Services Library
[3] Business Information Services Library

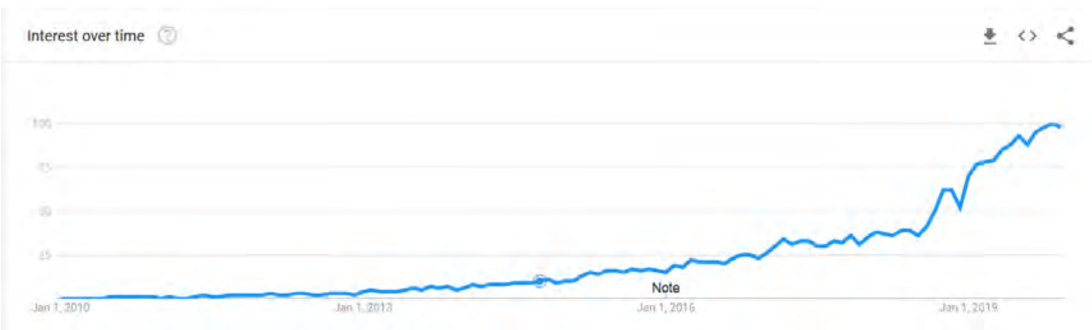


Figure 2.1 – DevOps trends statistics between 2010 and 2019. (Google.com (2019, n.p))

staff to fix the problem, which may further delay the production. Such delays will affect the project as explained in Section 1.1. Communications between the two departments may not be open, and lengthy departmental protocols may have to be followed to have these problems communicated and sorted.

The DevOps approach addresses the disunity found in departmental practices by encouraging a cross–functional collaborative culture between these departments (Bento et al., 2020; Erich et al., 2014; Hüttermann, 2012*b*; Pant, 2009). The collaborative culture of DevOps requires an integration of work performed by the staff of the developing and operations departments. Whereas the DevOps development team is made up of software developers, testers and quality assurance staff, the operations teams include system administrators, database administrators, network technicians and other functions (Liu et al., 2014, p. 42). With all these functions, every staff member is entrusted with and is responsible for developing software that performs well in operations (Barnhart et al., 2009, p. 54).

DevOps advocates such as (Elbayadi, 2014; Erich et al., 2014; Wettinger et al., 2014) are of the view that the successful adoption of DevOps culture would give an organisation a competitive advantage over traditional organisations. Through continuous delivery, the DevOps methodology is able to respond to changing customer requirements much faster than traditional methodologies would. By continuous delivery means that the functionality is only considered ready when it is being used in practice. In fact, Rütz (2019, p. 1) alleges that DevOps is a term that is often discussed when faced with fast delivery of software solutions. According to Dijkstra (2013, p. 24), the definition of done is extended by DevOps in that the process does not stop at the end of development, but continues right until the release of the software. Teams collaborate on tasks from the first day to the last (Luz et al., 2019, p. 4). Continuous delivery can be viewed in two perspectives (Neely & Stolt, 2013, p. 122), namely engineering and business perspectives. The engineering perspective focuses on the development of software and the automation tools used. The business perspective focuses on how various departments play a role in the software development pipeline.

2.3.1 DevOps definition

Several authors have tried to advance an acceptable definition of DevOps . Hüttermann (2012*a*, pp.1-2) defined it as a blend of development (includes programmers, testers and quality assurance staff) and operations (includes administrators and network technicians) patterns intended to improve collaboration. DevOps is also viewed as a set of processes, methods and system communications, collaboration and technology operations (Pant, 2009; Swartout, 2014; Maroukian & Gulliver, 2020*b*). However, Penners and Dyck (2015, p. 3) proposed what they called a scientific definition of DevOps based on the analysis of available descriptions and definitions. In this proposition, DevOps is defined as follows:

> *"DevOps is a mindset, encouraging cross—functional collaboration between teams—especially development and IT operations—within a software development organization, in order to operate resilient systems and accelerate delivery of changes."*(Penners & Dyck, 2015, p. 3).

In this definition, DevOps is viewed as a mindset. OxfordDisctionaries.com (2016, n.p) defines a mindset as an established set of attitudes held by someone.

Penners and Dyck (2015, p. 3) do not support the view of DevOps as a method because they believe that DevOps does not define any processes or techniques. Dyck et al. (2015, p.3) continued to define DevOps as "an organisational approach that stresses empathy and cross-functional collaboration within and between teams—especially development and IT operations—in software development organisations, in order to operate resilient systems and accelerate delivery of changes. ". This definition illustrates how DevOps culture is forms part of organisational culture.

With all that said, this study viewed DevOps as a culture that is practised by a team in non siloed environments. The DevOps culture emphasises team collaboration and it encourages team members to communicate freely with each other and agree on practices rather than prescribing practices of how teams should operate. Walls (2013, pp. 5-7) defined the DevOps culture by characteristics of open communication, incentive and responsibility alignment, respect and trust.

2.3.2 What constitutes a DevOps team approach?

Nybom et al. (2016, p. 132) has chronicled three approaches that can be used to make the Development department (Dev) and Operations department (Ops) to work together:

- A mixed responsibility approach in which both the development and operation responsibilities are assigned to all engineers. In other words, a person performs development and operations roles.

- A mixed personnel approach consists of DevOps developers and DevOps operators working together as a single DevOps team. DevOps developers, including programmers, testers, and quality assurance personnel, whilst DevOps operators are experts who put software into production and manage the production infrastructure, including system administrators, database administrators, and network technicians (Hüttermann, 2012*a*, p. 4).

- Bridge team approach which requires a team to bridge between Devs and Ops. However, the nature of this approach does not remove the silos phenomenon but provides a mechanism to connect the two departments, and is therefore not encouraged (Nybom et al., 2016, p. 133).

Mixed responsibility and mixed personnel approaches are more likely to be adopted in practice to realise the benefits of DevOps. A mixed responsibility approach can be also seen as the 'No–Ops' initiative within the organisation. No–Ops means that no personnel is dedicated to the Ops role (Farroha & Farroha, 2014, p. 293), these roles are performed by the developers. Automation and cloud computation drives this idea by allowing developers to code and deploy a service and manage and scale their code (Farroha & Farroha, 2014, p. 293). In this thesis, the DevOps collaboration culture is used in reference to these two approaches (i.e. mixed responsibility and mixed personnel approaches).

The DevOps development team is made up of software developers, testers and quality assurance staff, whilst the operations teams include system administrators, database administrators, network technicians and other roles (Liu et al., 2014, p. 42). These DevOps team roles are roles from traditional development practices. However, Techbeacon (2015, n.p) has proposed seven new professional roles:

1. The DevOps evangelist—this is a change agent who leads the organisation during the transition period to DevOps practices to ensure its success.

2. The release manager—oversees the coordination, integration and flow of development, testing and deployment to support continuous delivery.

3. The automation architect—analyses, designs, and implements strategies for continuous deployments while ensuring that production and pre-production systems are available.

4. The software developer/tester—this is the heart of the DevOps organisation. The scope of responsibility of these developers has increased when comparing it with that of traditional developers. Not only are they responsible for turning new requirements into code, their function also covers unit testing, deployment, and ongoing monitoring. Operators, that is, system administrators in traditional terms, falls within this category. Farroha and Farroha (2014, 288) acknowledge that this new role, in DevOps' terms, is becoming that of a programmer in that software programming skills and methodologies are increasingly being required for the undertaking of this role.

5. The experience assurance professional—this is a quality assurance role, which has also increased in scope. In this case, not only does the quality assurance team test for functionality, it also includes user experience testing.

6. The security engineer—security engineers need to work side-by-side with developers by embedding their recommendations much earlier during the process, as opposed to leaving it until the deployment phase.
7. The utility technology player—DevOps requires utility team members who can operate effectively across development platforms, tools, networks, servers and databases, and even across development and support. Unlike in traditional software development practices were IT operators focus on keeping the servers running and developers have been only coders with no involvement in post-production systems.

This study only focussed on roles that are directly linked to the actual production of the software products. These roles can easily fall under the developer and operator roles. Roles such as the DevOps evangelist are only deemed valuable during DevOps transition periods and, therefore, do not form part of this study.

2.3.3 DevOps characteristics

Although some researchers have attempted to devise a scientific definition of DevOps, others have looked at what characteristics DevOps should possess. Attempts aimed at defining DevOps have revealed the different views on the definition of DevOps. Different organisations may implement DevOps in different ways, or at different levels. Therefore, there is a no one–size–fit–all solution to the implementation of DevOps.

Before exploring the characteristics DevOps, it is important to consider what drives organisations to adopt DevOps. Humble and Molesky (2011, p. 6) mentions several problems that are inherited from traditional software development methodologies. These problems include:

- The goal of the final product from the perspectives of the development and operations team is not the same. For example, developers do not test their final product for reliability, scalability, performance and high quality.
- Since product success is measured differently within these departments, blame-shifting often occurs when neither of the team wants to take responsibility for the failure of the project.
- Disincentives for releasing the product early in their life cycle results in teams holding back on the product until close to the release date.

Humble and Molesky (2011, p. 7) has acknowledged that these problems can be addressed by using a DevOps approach through culture, automation, measurement and sharing (see Figure 2.2). Four key areas that are relevant to DevOps were identified by Debois (2012, under heading DevOps areas) as:

- Extended delivery to production: The whole delivery process (from the start of the project to production) is improved by allowing the development and operations department to collaborate.
- Extended operations feedback to project: Information flow from operations to development department is broadened for better feedback.
- Embed project knowledge into operations: Development staff members are also responsible for production.

- Embed production knowledge into a project: Involve operation teams from the start of the project.

Within the key above-mentioned areas, three layers are identifiable: Tools, Processes and People or culture (Hüttermann, 2012*a*, p. 4). Culture is the central component of DevOps because it forms the basis of the other cores. Tools are needed for technical support, processes determine how things should be done and people perform these processes using the tools. People utilise tools to perform business processes; that practice is a culture after all. Without people, there would be no culture and it would not be possible to perform those business processes. It is important to understand DevOps culture in

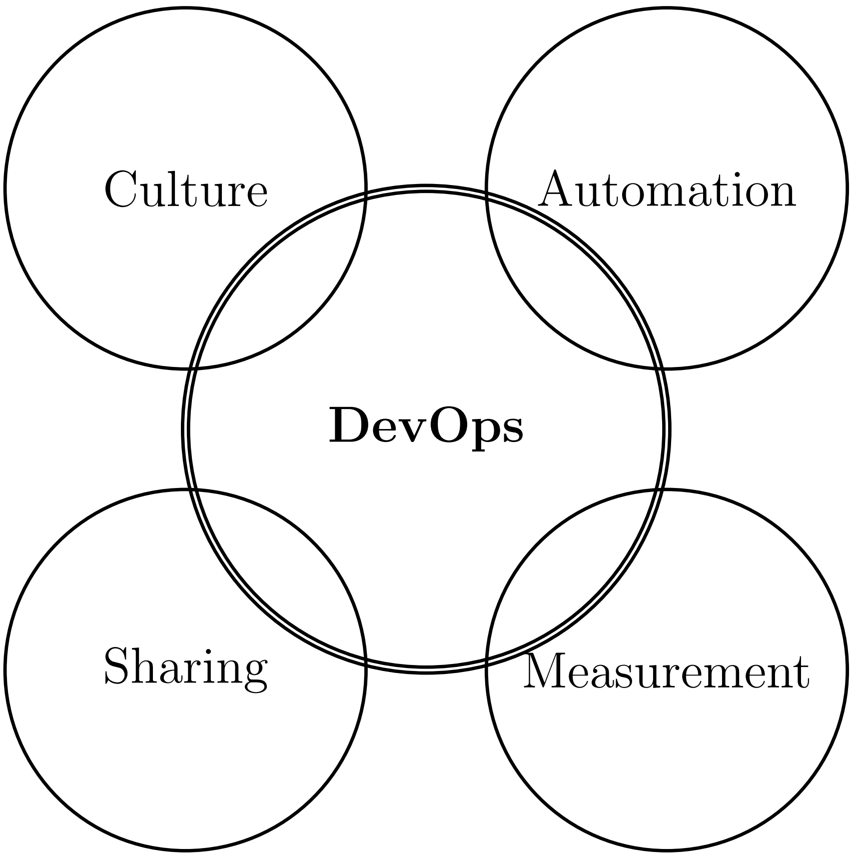


Figure 2.2 – Four core elements that make up DevOps.

order to successfully address the other core values.

Willis (2010, p. 3) has presented another view of DevOps that has four core values: Culture, Automation, Measurement and Sharing (CAMS). Figure 2.2 shows the four core values of DevOps. By scrutinising these views, it becomes apparent that culture seems to dominate. It is for that reason that the DevOps culture was investigated. However, before focusing on this culture, it seems appropriate to pay attention to the challenges that come with the adoption of DevOps.

2.3.4 DevOps challenges

When analysing the characteristics of DevOps, Erich et al. (2014) noted the following challenges of DevOps, which are summarised below:

- Collaborative culture: Cooperation between the development and operations staff is not optimally handled. This affects productivity, software quality and service quality. Therefore, there is a need to improve the software process methodologies to include cooperation as suggested by Tessem and Iden (2008, p. 108). Diaz et al. (2020, pp.25-26) reasons that DevOps culture is a silo remover which improves the effectiveness and efficiency of teams, processes, and project management. With that said, it is convincing to admit that DevOps culture is not just a team culture but an organisational culture. Section 1.2 of the thesis gave a detailed discussion around this challenge as it is the main focus of this study.
- Automation: Various tools can be used to support automation to improve scalability and testability while reducing the work for operations staff.

- Measurement: Employee performance should be measured accordingly. The traditional way of measuring employee performance for development and operations staff may not be sufficient. A metrics-driven framework to act as a common language for collaboration between the two departments can be used. This framework defines shared and actionable metrics for adopting DevOps (Lê-Quôc, 2011, p. 43).

- Sharing: The documentation from both departments should be understandable by both parties. The above metrics-driven framework would be applicable here.

Erich et al. (2014, pp. 35-41) has listed other challenges resulting from the adoption of DevOps. These include quality assurance, structures and standards and services. Khan and Shameem (2020) categorised DevOps challenges as people, business and change challenges. In their prioritisation of the challenges, they concluded that people challenges should be given priority.

There are commonalities in the above DevOps views in terms of what it is or what it should comprise of. These views suggest that an organisation should consider adopting new technologies when implementing DevOps. The adopted technologies may lead to a change of organisational processes, and this may mean changing the organisational structure. It is also important to understand that DevOps encourages a collaborative culture and changing organisational structure is often preceded by cultural changes. Although technology can be used to facilitate the collaboration, technology itself does not drive organisational change; it suffices to say that people are the actual drivers of organisation change (Guardado, 2012, p. 54). In fact, Rosen (2007, p. 89) adds that tools may remove barriers amongst departments and functions if the organisational processes encourage a collaborative culture.

It was therefore important for this thesis to scrutinise technology acceptance models to determine important factors that would lead to effective collaborative culture implementation in DevOps as questioned in Section 1.3. It is thus important to understand what constitutes a successful DevOps collaboration at a deeper level. Technology acceptance models will be explored in detail in Chapter 3.

2.3.5 DevOps collaborative culture

It is evident from the definitions of DevOps that collaboration is the key to DevOps culture. Gottesheim et al. (2015) believe that culture is the most important aspect of DevOps because it changes how teams work together and it also allows teams to learn from one another. In addition, Luz et al. (2019, p. 4) maintain that collaborative culture is core to DevOps adoption. As far as Dijkstra (2013, p. 25) is concerned, culture is the most crucial and hardest part of DevOps and forms the basis for the other core values. As already alluded to, people take preference over process and tools. Since people use tools on processes, it is therefore important to understand the people culture of DevOps in order to successfully implement the other core values. Walls (2013, pp. 5-7) views DevOps culture as possessing the following characteristics:

- Open communication—a team should communicate about the product during its life cycle.

- Roles and responsibility alignment—the whole team should be rewarded for their efforts and take responsibility for product failure. Understanding these roles and their responsibilities is important in a DevOps team (Farroha & Farroha, 2014, p. 288), especially in mixed personnel approach were one person can be assigned to different roles.

- Respect–Walls (2013, p. 6) points out that team members need not have to like each other, but they should recognise the contribution of every member to the team.

- Trust—team members should trust each other and believe that they are pursuing a common goal. Farroha and Farroha (2014, p. 288) believe that breaking down the silos mentality is an enabler of trust within the DevOps team.

In order to realise the DevOps culture, it is important to understand the skills requirements of DevOps practitioners. Apart from technical skills, it is important that software practitioners possess what is termed 'soft skills'. These soft skills include social attitudes, ability to work independently, open and adaptable to changes, being a team player, problem-solving skills, organisational skills, interaction with other people—communication, conflict resolution, cohesion and cooperation (Acuña et al., 2009, p. 629). Matturro et al. (2015, p. 101) acknowledge that these soft skills play an important role in team members of software engineering projects. After studying the important soft skills in software development teams, Matturro et al. (2015, p. 101) have found that communication is an important soft skill from the perspectives of both the team leaders and team members.

Team leaders need to be able to communicate effectively with both the customer and the development teams. In addition, members of the team need to interact with each other in order for the team to function optimally. Bjarnason et al. (2011, p. 39) have observed that communication skills are vital in all stages of software development life cycle.

Not only should members of the DevOps team be able to communicate, they should also be able to communicate freely without fear and at any level. However, Dreesen et al. (2016, p. 4933) state that different cultural backgrounds of team members may affect team communication negatively. For example, employees used to the traditional software development culture may not communicate directly with a team member occupying a higher level.

In order for team members to communicate effectively, it is important that team members trust each other. Vangen and Huxham (2008, p. 493) acknowledge the significance and importance of trust when nurturing a collaborative culture. Furthermore, it was pointed out that once trust has been initiated it needs to be managed effectively; otherwise it can cease developing or lost (Vangen & Huxham, 2008, p. 496). Since trust cannot be bought but earned, Van Gelder (2011, p. 18) claims that trust can be earned by being reliable and staying constant. Trust and openness among team members can be built using high interaction group activities (Van Gelder, 2011, p.22). However, Vangen and Huxham (2008, p. 493) have warned that trust-building and management can be problematic as it may imply coping in situations where trust is lacking, and building it where it is impossible. DeGrandis (2011, p. 34) points out that moving to DevOps system requires trust, as it influences change, and warns that it takes time but can be destroyed by a single act of bad faith. Trust in DevOps is not only relevant to a team member; it also needs to be built amongst managers and team members (Humble & Molesky, 2011, p. 31).

Fiampolis and Groll (2016, p. 15) advises that DevOps training that provides support on the DevOps approach which insist on breaking down silos, improve communication and encourage collaboration should be the starting point of every organisation. This training should initially be targeted towards breaking the barriers and building trust. Operations should trust developers so that they have the same goal of delivering a functional product. On the other hand, the developers need to trust that the operations personnel will not try to sabotage their work.

Walls (2013, p. 17) stresses out that every team member should be treated with respect. Discussions between team members should be conducted in such a way that no member may feel threatened when raising specific issues. Members of the team should listen to the opinions of other team-members with a view to encourage the culture of learning.

In their study involving the impact of mixing responsibilities between developer and operators, Nybom et al. (2016, p. 137) opined that the mixing improves both trust and collaboration between both teams. Breaking the silos mentality and allowing developers and operators to work together towards a common goal addresses the problem of incentives. Since everyone is working on one goal of creating a product that will satisfy the customer, everyone is incentivised after the product is released. This is in direct contrast to rewarding developers for writing a code and punishing operations if the code does not run during production (Walls, 2013, p.6).

2.4 Key Concepts of this Study

In the previous section, various definitions of DevOps from the perspectives of various researchers were presented. From these definitions, it is clear that DevOps is not a one size fit all solution for every organisation. The common theme in all these definitions was the collaborative culture that people should adopt when using DevOps. Furthermore, it was demonstrated that DevOps

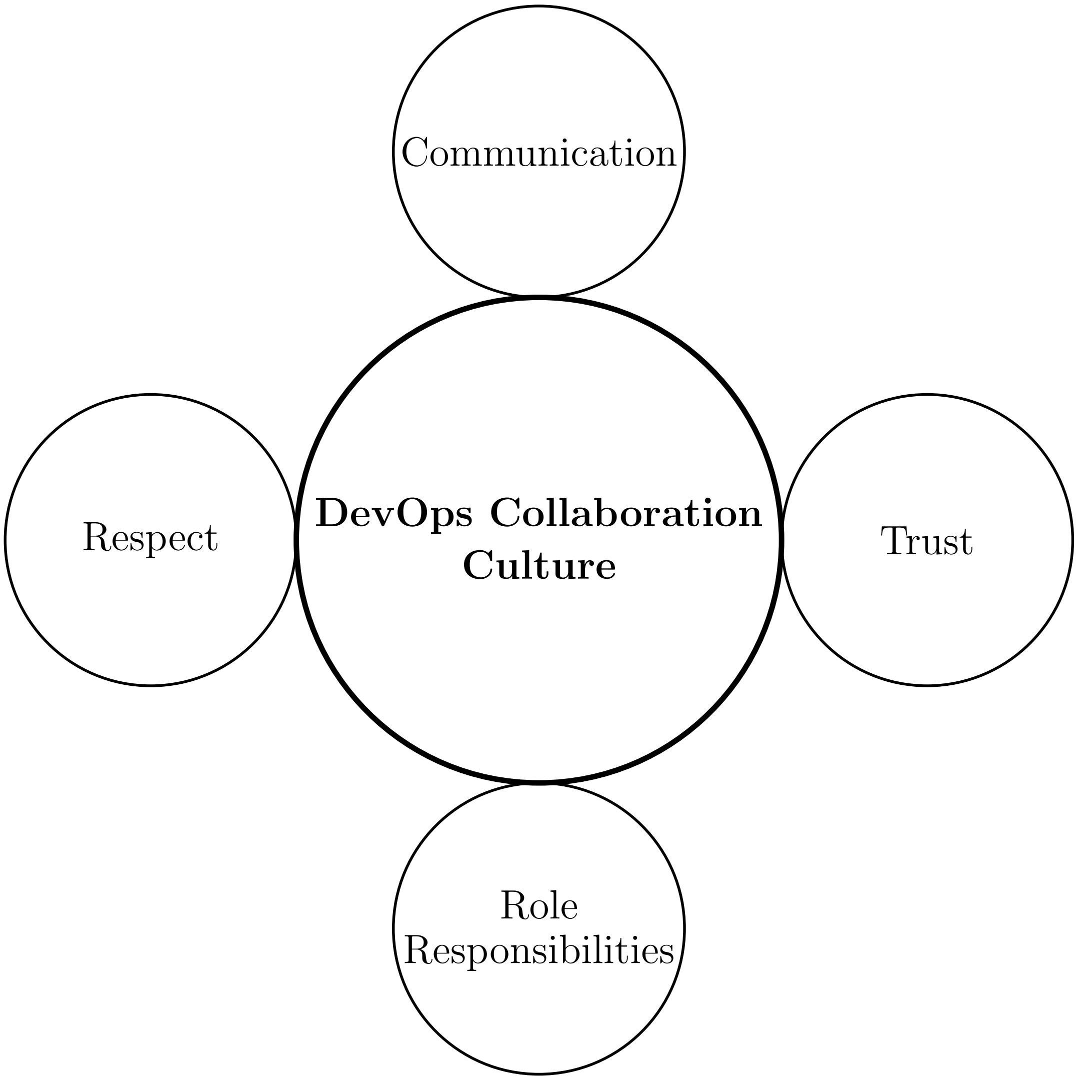


Figure 2.3 – DevOps collaboration culture elements.

involves the automation of processes to speed up the delivery of the desired software product. However, DevOps places more emphasis on human aspects more than tools and processes. A justification of this is that humans are tasked with manipulating the requisite processes using available tools.

A collaborative culture is deemed the hardest part of DevOps to implement. This is because collaborative culture involves human beings who have their own values and belief systems that make it hard to predict their behavioural intentions and actions. Although organisations may adopt new processes that could potentially give these organisations a competitive advantage, it is incumbent upon people who work on these organisations to successfully adopt and implement these processes. Therefore, this study was premised on the identification and understanding of factors that drive these people to accept new technologies.

Four elements that constitute a DevOps culture are: open communication; alignment of roles and responsibilities; respect; and trust. In open communication, levels or structures that restrict communications do not exist. This means each member of the team has a direct communication channel with all the other members of that team. On the other hand, open communication requires that all the team members should be kept informed about the project at all times. In order to communicate efficiently, a communication medium needs to be established for easy interactions. To effectively communicate, team members need to respect one another at all times. Members of a team are more likely to contribute their views if they feel that their input is respected by others. In addition, trust is a key element of the DevOps culture since every member of the team is expected to work towards a common goal. Every team member needs to be assured that the other team members will do their allocated work and would not sabotage other team members. Lastly, the team members should understand their respective and other members' roles in the project and how such roles can be incentivised. This study investigated how the four above-mentioned elements interact and influence each other.

2.5 Collaboration

Collaboration is a key concept that is at the centre of this research study. Before focussing on DevOps collaboration, it is therefore important to understand the characteristics of general collaboration. In this section, a brief explanation of what constitutes collaboration is presented.

2.5.1 What is collaboration

A lot of research has been conducted in an attempt to understand what entails effective teamwork and collaboration based on the perceptions of employees (Mattessich & Monsey, 1992; LaFasto & Larson, 2001; Rosen, 2007). Mattessich and Monsey (1992, p. 11) define collaboration as a mutually beneficial and well-defined relationship entered into by two or more organisations to achieve common goals. According to Nugent (2012, p. 26), collaboration is made up of the following:

- Relationship commitment;
- A mutually evolved organisational structure;
- Responsibility sharing;

- Jurisdiction and accountability; and
- Resource sharing

In addition, Nugent (2012, p. 25) is of the view that not every joint effort between two parties is collaboration. Collaboration should not be seen as the same as cooperation. Cooperation does not need to have clearly defined goals and objectives or a joint structure (Nugent, 2012, p. 25), as opposed to collaboration. Collaboration requires commitment and interdependence in solving challenges. The reason why two parties collaborate is not for their own sake, but for creating value (Nugent, 2012, p. 28). Organisations must create environments that will encourage collaboration and create value for them. Rosen (2007, p. 47) has listed cultural elements that culminate in value for an organisation and common attributes of collaborative cultures (see Table 2.1). The elements that need to nourish in order to increase value or collaboration are quite evident in Table 2.1. The collaboration attributes listed in Table 2.1 are concerned with the interaction between the parties, whilst the value elements focus on what is to be achieved by this interaction.

It is evident that the themes of the DevOps culture are embedded in the above elements. Openness, which can be viewed from different perspectives, is key in DevOps communication. Openness can mean free access to other team members in terms of interaction and communication. In addition, it refers to the frequency of unrestricted interactions. Respect reduces fear and can also be noted in these elements. Respect goes hand in hand with trust. In other words, if there is trust there should be respect and *vice versa*. Tools fit work styles emphasises tools as an important element. Although DevOps collaborative culture separates automation tools from the culture, automation

Table 2.1 – Cultural elements and collaboration attributes (Rosen, 2007, p. 9).

Value elements	Collaboration attributes
Trust	Frequent, cross-functional interaction.
Sharing	Leadership and power spread across the organization.
Goals	People are accessible regardless of their level
Innovation	Reduce fear of failure.
Environment	Broad input into decisions.
Collaborative chaos	Cross-pollination of people.
Constructive confrontation	Spontaneous or unscheduled interaction.
Communication	Less structured interaction.
Value	Formal or informal mentoring.
	Tools fit work styles.

can be viewed as part of the culture. Though this may be the case, roles responsibilities are part of the DevOps culture and they should be clearly defined. These roles can mimic tools fit work styles (Rosen, 2007, p. 49).

2.5.2 Why collaborate?

Schein (2009, p. 57) is of the opinion that productivity, growth and successful execution of organisational processes could be improved by enabling employee collaboration and information sharing. It is believed that when people collaborate, productivity is significantly improved. This means that effective and efficient implementation of collaboration within an organisation would potentially bring about production success. On an empirical study about factors that affect project outcomes of software development (see Figure 2.4), McLeod and MacDonell (2011, p. 23) quoted several studies that found that the environmental, process and human factors play a major role in software project outcomes, amongst others. The environmental factor relates to the organisational culture or policy, structure, planning, accountability, skill limitations and evaluation. The process factor is user-developer relationships, influence, power and commination. The human factor includes commitment, willingness, attitudes and abilities.

McLeod and MacDonell (2011, p. 30) have devised a framework based on these studies. In this framework, people factors are given priority over processes because it is the people who use these process on projects and not vice versa (processes using people) within a certain environment. All of these

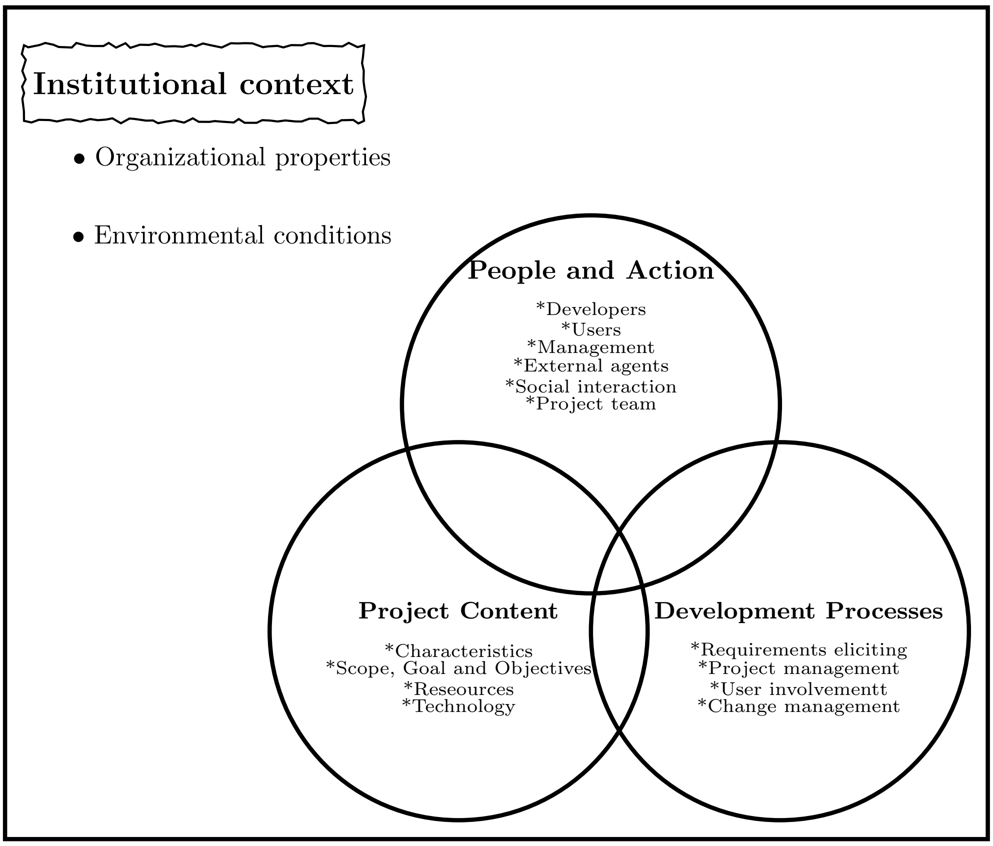


Figure 2.4 – Factors affecting software project outcomes (Mcleod & MacDonell, 2011, p. 5).

factors (i.e environmental, process and human factors) affect one another. That is, users and developers within a project team may need to interact with one another and/or with top management for the successful delivery of the project. Their social interaction is important for the realisation of project goals. However, development processes may limit or enhance team interaction. These development processes may be guided by the project and the environment in which the development processes are being implemented.

Collaboration and teamwork bring benefits to the organisation such as gaining competitive advantage knowledge monopoly. When resources and other skills are shared, information is easily passed from experts to novices, thus bringing about the much needed productivity improvement that is driven by collaborative problem solving. Despite all the potential successes associated with collaboration and teamwork, some factors exist that need to be considered that can potentially affect collaborating in a negative way. Equal sharing of collaboration incentives when other team members have made very little contribution to the work can discourage other members. Some organisational cultures encourage competition as people are rewarded on the basis of individual effort (Dipboye, 2018, p. 288). In such environments where people are incentivised by the knowledge or skill they have, sharing is discouraged.

2.5.3 Building a collaborative environment

A number of models on keys to successful collaborative teams are known. The model by Page (2008, p. 19) includes the following three keys: work design, team composition and environmental context. The model by Frank Lafasto (2001) is derived from the experience of team members. As far as the model of Frank Lafasto (2001) is concerned, the following conditions should be created:

- Expect collaborative behaviour from team members.
- Require people to establish collaborative work relationships.
- Practice collaborative problem-solving.
- Demonstrate collaborative leadership.
- Build a collaborative work environment.

Although the above features are important in creating a collaborative culture, Winer and Ray (1994, p. 45) believe that factors such as leadership, competition, resource, and power can make or break the effort to bring out the culture. Maroukian and Gulliver (2020*a*, pp7-8) suggest that a leadership style that is relevant to DevOps should be transactional, transformational, authentic, servant and Ad Hoc. Leadership should encourage knowledge and resource sharing by employees instead of holding this as a sign of power (Winer & Ray,1994, p. 34). In addition, employees should learn from each other and competitive behaviour should be discouraged (Winer & Ray,1994, p. 35).

In order to instil a collaborative culture within the organisation, Rosen (2007, pp. 50-52) identified ten actions that can be used as a guide:

- Establish a support program to promote the knowledge sharing culture.
- Encourage constructive confrontation by a team member and thus allow open communication.

- Adopt collaborative tools for easy communication.
- Facilitate cross-functional brainstorming in an informal environment to encourage diversity and creativity.
- Support and reward information sharing.
- Incentivise team innovation with recognition and rewards.
- Team leaders who demonstrate that teamwork leads to better decisions and products should be encouraged.
- Use a language that promotes and support collaboration.
- Competition amongst teams should not be encouraged.

In addition to the above, Gratton and Erickson (2007, p. 45) suggest the following to help overcome obstacles to creating a collaborative culture:

- Provide executive support on how to model collaborative behaviour, coaching and mentoring.
- Provide skills training on collaborative practices.
- Relationship-oriented team leaders.
- Team structure and role clarity on challenging assignments that demand creativity.

Whilst attempts have been made by other researchers to identify important key elements that constitute collaborative culture in an organisation, others focused on how it could be implemented. Winer and Ray (1994, p. 39) devised a strategy on how the collaborative culture could be instilled. According to Winer and Ray (1994, pp. 39-40), the four staged path to a collaborative culture is as follows:

Stage 1: Envision results (individual-to-individual).

- Bring people together.
- Improve trust.
- Confirm shared vision.
- Specify desired results.

Stage 2: Empower (individual-to-organisation).

- Clear authority from home organisations.
- Confirm organizational roles.
- Organise the effort.
- Support team members.
- Resolve conflicts.

Stage 3: Ensure success (organization-to-organization).

- Build relationships by finding formal ways to work together.
- Manage the work.

- Develop joint systems.
- Evaluate results. Renew the effort.

Stage 4: Endow continuity (collaboration-to-community).

- Institutionalise success.
- Seek support from more people and organisations.
- Create visibility.
- Involve the entire community.
- Replace out-dated systems.
- Conclude this collaborative effort and start new projects.

Research has shown that collaboration could bring success to an organisation. Collaboration is a work commitment relationship between two parties that share the same vision or goal. It requires the parties to share resources, responsibilities and accountability. Collaboration shares similar attributes as those identified to be DevOps attributes such as open frequent communication, respect and trust amongst the participants, and lastly the clearly defined roles and responsibilities.

Great emphasis is placed on sharing and the interactions between the collaborators. Ways to improve collaboration have been suggested such as rewarding teamwork, providing necessary support and training where needed.

Collaboration has been described in terms of different stages, namely stage 1—from individual to individual, stage 2—individual to organisation, stage

3—organisation to organisation and stage 4—organisations to the community. This study sat at stages 1 and 2 since it focused on the attitudes of individual developers towards other developers and the organisation. This study therefore, looked at identifying key factors that contribute to successful collaboration in a DevOps environment. With that being said, keys that are necessary to raise the level of successful collaboration were identified and, if used correctly, should make DevOps more advantageous and acceptable by team members.

2.6 Chapter Summary

In this chapter, challenges brought about by traditional software development methodologies were discussed. Traditional software development methodologies generally follow a non-collaborative culture during software development. This culture separates the development team from the operations team with restricted communication occurring between the two departments. This restricted communication is often accompanied by problems such as delays when releasing the final product and difficulties in situations where changing requirements are high. The separation of departments also encourages competition between the departments instead of working together.

DevOps, on the other hand, brings solutions to these challenges by removing the silos mentality that is prevalent in departments and encouraging collaboration between the departments. Various definitions of DevOps, as well as DevOps participants, were presented. The following four attributes that are important for DevOps collaborative culture were identified and listed accordingly. These are: open communication, responsibilities and incentives alignment, respect and trust.

Introducing a new technology to an organisation may not be welcomed by all staff members, even if it is for the benefit of the organisation and the majority of staff members. Changing the way people do things can lead to resistance to the change. Studies on how people get to accept new technology have been conducted. In the next chapter, technology acceptance models will be discussed in greater detail. Furthermore, a framework for the acceptance of DevOps collaborative culture will be proposed. Using the proposed framework, the attitudes of DevOps practitioners were measured.

Chapter 3

Theoretical Background and Conceptual Framework

3.1 Introduction

DevOps practitioners believe that in order for an organisation to successfully adopt DevOps, an organisation needs to undergo cultural changes and adopt a DevOps culture (Erich et al., 2014; Hüttermann, 2012*b*; Shamow, 2011; Willis, 2010). In a survey conducted by Saugatuck Technology (2014), it was indicated that the leading challenge hindering the implementation of DevOps is overcoming cultural habits resident within organisations.A recent study by Khan and Shameem (2020) showed that people challenges should be given priority. Guardado (2012, p. 16) indicated that organisational changes may lead to some of the software project failures in an organisation. Introducing new technology to an organisation may mean changing the entire organisational culture. Such technology could be resisted by its intended users thus leading to unnecessary project failures (Hardgrave & Johnson, 2003, p. 322).

For cases where resistance gives rise to project failures, Organizational Change Management (OCM) strategies could be used to mitigate the effects of the challenges that arise from the adoption of new technology (Guardado, 2012, p. 17). Organisational culture can be defined as the way things are done in an organisation (Brown & Harvey, 2011, p. 27). It is a system of shared values and beliefs that interact within an organisation to produce behavioural norms (Nugent, 2012, p. 17). Although organisational culture is constantly changing, Trompenaars and Woolliams (2004, p. 23) and Schein (2009, p. 63) argue that it is albeit difficult to change because it represents a group's accumulated learning and pattern of inter-connected assumptions. That is, culture represents a property of a group. Organisational culture can be formally observed through the organisation's policies and processes and informally observed through employee behaviour (Rosen, 2007;Schein, 2009). OCM provides tools for studying organisational change (Guardado, 2012, p. 17). In addition to employing OCM strategies to prepare users for the cultural change brought about by the adoption of new technology in an organisation, it is important to understand what influences users to accept new technologies (Guardado, 2012; Jones et al., 2010).

Similarly, successful implementation of DevOps culture depends on the acceptance of its development strategies by the DevOps practitioners (Developers and Operators). The need to study the effects of culture when it comes to the acceptance of the new technology is strongly supported by Kashada et al. (2020, p. 35). Research shows that people are prone to resist change to their normal ways of doing things (Khatib, 1997; Waddell & Sohal, 1998;Laurageorgeta, 2008). Reasons for change resistance may range include not understanding why there is a need to change—this could be due to poor communication of the need to change, fear of the unknown and/or lack of competency and trust (Blom & Viljoen, 2016, p. 2). Whilst the need to change may create a competitive advantage for an organisation by cutting costs and increasing productivity, its resistance may have an overall negative impact on an organisation (Durodolu, 2016, p. 7). In an attempt to explain what informs human behavioural intension and actions, theories such as the Theory of Reasoned Action (TRA) and Theory of Planned Behaviour (TPB) have been developed and tested (Bagozzi, 1992, pp. 179-181). TRA and TPB theories were later extended and applied in various settings including by Information Systems (IS) to investigate human behavioural intentions on adoption and acceptance of new technologies (Lai, 2017; Taherdoost, 2018). The adoption of these theories in various settings has resulted in the development of additional theories such as, but not limited to, Technology Acceptance Model (TAM), Unified Theory of Acceptance and Use of Technology (UTAUT).

The purpose of this chapter is to propose a framework for the acceptance of DevOps culture. The chapter starts by providing some background on existing models from which the proposed framework was based. The sections are laid out as follows: Section 3.2 provides a brief discussion on behavioural theories TRA and TPB. This is followed by a discussion of IS theories (TAM and its variants, UTAUT and Diffusion of Innovation (DOI) on technology acceptance and diffusion in Sections 3.3 and 3.4, respectively. In Section 3.5, the theory that was used to study the adoption and acceptance of Software Development Methodologies (SMD) by IT practitioners is discussed. The framework that has been adopted for this study is outlined in Section 3.6. The chosen framework was used to identify the factors that will encourage the acceptance of DevOps as the asked in Section 1.3.

3.2 Behavioural and Innovation Theories

The theories in this category of theories originate in fields such as behavioural social sciences and psychology. Two theories, which are most-cited and frequently applied, are TRA and TPB (Laumer & Weitzel, 2009*a*, p. 12). A brief description of these two theories follows below.

3.2.1 Theory of Reasoned Action (TRA)

In 1975, Fishbein and Ajzen (1975) proposed a TRA theory to explain the relationship between human attitudes and their behavioural intention when carrying out an action. The TRA is used to understand the motivations behind the human's action or intentions to carry out an action. According to TRA, the intent to perform an action is influenced by two factors, namely: (i) the attitude towards performing the behaviour; and (ii) subjective norm related to performing the behaviour (Fishbein & Ajzen, 1975, p. 7). In this context, attitudes are human's beliefs about the outcome of the behaviour (i.e. how likely is the outcome) and evaluation of the potential outcome as good or bad (Fishbein & Ajzen, 1975, p. 8). A subjective norm, on the other hand, is influenced by normative beliefs (i.e. beliefs of those around us) of an individual and motivation to comply with these norms.

3.2.2 Theory of Planned Behaviour (TPB)

TRA is based on experiencing a limitation since it does not recognise behaviours that people have very little control of. TPB connects the perceived behavioural control to the original TRA model as an extension (Ajzen, 1991). Perceived behavioural control refers to resources, and opportunities at the disposal of a person, and to some extent, dictates the possibility of behavioural achievement (Hardgrave et al., 2003, p. 125). Therefore, a behavioural intention in TPB is influenced by attitude, social norm and perceived behavioural control.

Both TRA and TPB have been widely used for the prediction of human behaviour in various contexts, including technology use (Knabe, 2012, p. 36). These models have been extended and used in various fields including the behaviour towards technology.

These two theories form the basis of relevant acceptance models to this research study, hence the reason for their inclusion in this research study. It is important to understand the motivations behind human behavioural action to predict the likelihood of adoption or acceptance of new behaviour. Although it is a person who chooses to perform a specific action, the influence of people around that person cannot be ignored. The views and beliefs of other people are important in a DevOps culture since they influence the behavioural intentions of the person adopting or accepting the behaviour. The behavioural intention of DevOps practitioners can be predicted by employing the two theories. TPB may be more relevant in this regard, as it incorporates control as one of the determinants. This control may be in the form of skills and other resources a practitioner possess, that may facilitate or impede job performance.

3.3 Technology Acceptance Theories

TRA and TPB are not context specific in terms of their application environment. In other words, they could be adapted to fit various research study contexts. The need to study the behavioural intentions of human beings when faced with new technology has led to the development of more technology specific models. These models are discussed in the subsections that follow.

3.3.1 Technology Acceptance Model (TAM)

Technology Acceptance Model (TAM) is a direct adaptation of the TRA model (Fishbein & Ajzen, 1975) to a technology environment. TAM posits that the determining factors of the Behavioural Intention (BI) to adopt technology are influenced by the Perceived Usefulness (PU) of the technology and the Perceived Ease Of Use (PEOU) of the technology (Durodolu, 2016, p. 12). In this context, whereas BI of a person is to adopt a particular technology, PU is the measure to which a person believes that using a particular technology will enhance their job performance. On the other hand, PEOU is the measure of

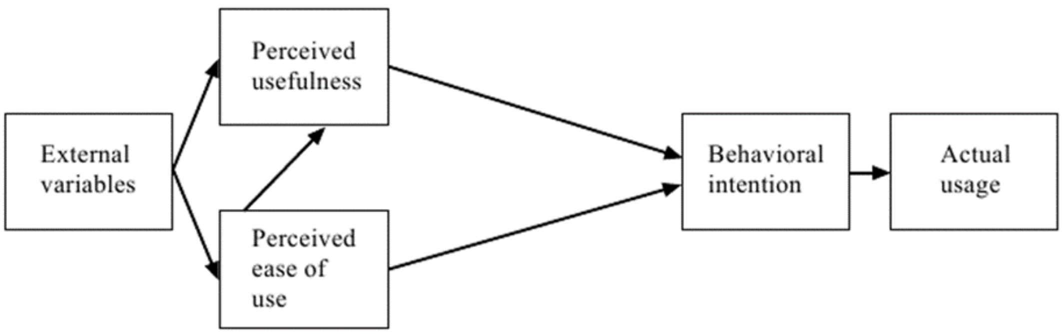


Figure 3.1 – TAM (Davis and Venkatesh, 1996, p. 20).

the extent to which a person believes that using a particular technology will be effortless (Durodolu, 2016, p. 13).

The combination of PU and PEOU inform an intention to use the technology, and the attitude towards using the technology. In addition, TAM posits that PU is influenced by PEOU. This means that technology that requires more effort to use may not be perceived to be useful. However, the technology perceived as useful may receive a positive attitude even if it requires more effort to operate. Figure fig:TAM shows the graphical presentation of TAM.

The original scale for measuring TAM constructs has demonstrated high reliability and validity in several studies (Davis & Venkatesh, 1996, p.11). As such, research has been conducted to assess user acceptance in various domains using TAM. Senarath et al. (2019, p. 6) made an observation that TAM is widely used when user intentions to adopt a new technology are investigated. This has frequently resulted in new constructs and relationships being added to the original TAM to describe BI. Examples of these extensions have been reported (Ghazizadeh et al., 2012; Karahanna et al., 2006; Venkatesh, 2000).

As already mentioned, TAM is an exceptional case of TRA (Taylor & Todd, 1995, p. 148) focusing merely on attitude. Social influence and control factors were not included in the original TAM. This shortcoming was identified by Taylor and Todd (1995, p. 149) and TAM was therefore augmented by adding Social Norm (SN) and Perceived Behavioural Control (PBC) constructs from the TPB. Other factors, which may influence the PU of a technology, were explored by Taylor and Todd (1995, pp. 186-204) in TAM2, which resulted from the extension of the original TAM model. TAM2 (Figure 3.2) describes PU and usage intentions in terms of social influence and cognitive instrumental processes.

These social influence elements are (Venkatesh, 2000, pp.186-204):

Subjective norm this can be viewed as "peer pressure". If a person is surrounded by people who believe that someone should use a given technology, that person's willingness to use the technology at hand will consequently be influenced.

Voluntariness this refers to the degree to which a person chooses to use the system or is mandated to use it.

Image this is the image of the person as seen by others. The technology which is good for a certain social image may be perceived as useful by people in that image.

Experience this refers to an ongoing use of the technology at hand. More experienced users are more likely to continue to use technology than novice users.

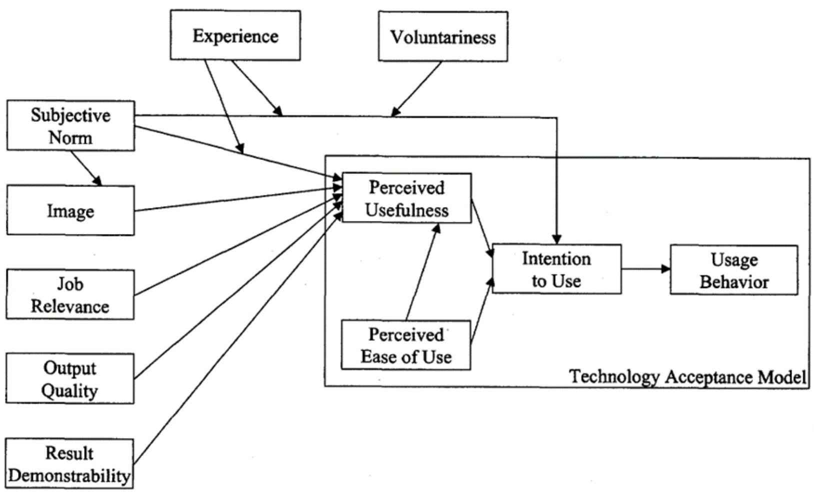


Figure 3.2 – TAM2 (Venkatesh & Davis, 2000, p. 188).

The cognitive instrumental processes elements are (Venkatesh, 2000, pp. 186-204):

Job relevance this is the degree of relevance of the technology to one's job. This can be seen as the definition of PU of TAM. If the technology is relevant to the job's task, it may be perceived as useful.

Output quality this refers to the measure of quality produced by using the technology. If the use of technology improves performance, the technology may be perceived as useful.

Results demonstrability this is the degree by which using the system produces beneficial results.

TAM2 posits that a person's perception of regarding a particular technology as useful is influenced by that person's mental assessment of the match between important goals and the consequences of performing job tasks using that technology (Lai, 2017). SN has an influence of the PU and overall the intention to use the technology.

TAM3 (Venkatesh & Bala, 2008, pp. 273-315) combines TAM2 and (Venkatesh & Bala, 2008, pp. 342-365) model of determinants of PEOU. These determinants are (Venkatesh & Bala, 2008, p. 279):

Computer Self-Efficacy this is the degree to which an individual believes that he or she has the ability to perform a specific task or job using the computer

Perception of External Control this is the degree to which a user believes that required resources exist to support the use of the system.

Computer Anxiety this is is the degree of uneasiness or fear perceived by a person faced with the possibility of using computers.

Computer Playfulness this is "...the degree of cognitive spontaneity in microcomputer interactions" (Webster & Martocchio, 1992, p. 204) as reported in (Venkatesh & Bala, 2008, p. 279).

Perceived Enjoyment this is the level of enjoyment perceived when using the system.

Objective Usability refers to a comparison of the actual level of effort of a system that is required to accomplish a task. This comparison is not based on perceptions.

Three types of relationships are posited in TAM3, namely the relationship between experience and (i) PU and PEOU; (ii) computer anxiety and PEOU; and (iii) PEOU and BI (Venkatesh & Bala, 2008, p. 281). According to Venkatesh and Bala (2008, p. 279), TAM3 provides a complete representation of the determinants of users' IT adoption and use. The strength of TAM3 apparently lies in its comprehensiveness, which ensures that all relevant factors are included.

As an adaptation of TRA, TAM lacks external control over the behavioural intention. The TPB has shown that this construct is important in determining intentions. In addition, social influence is not considered by the TAM, although TRA emphasises its importance. Usefulness and ease of use are the main constructs that influence intentions. TAM2, on the other hand, brings back the social influence and some control in terms of experience and involuntariness. Usefulness variables and ease of use are explored in TAM2 and

TAM3, respectively. The suitability of these variables in software development context where the technology has already been adopted and its use is mandatory has not been determined. Hence, it is important to investigate these variables under these settings.

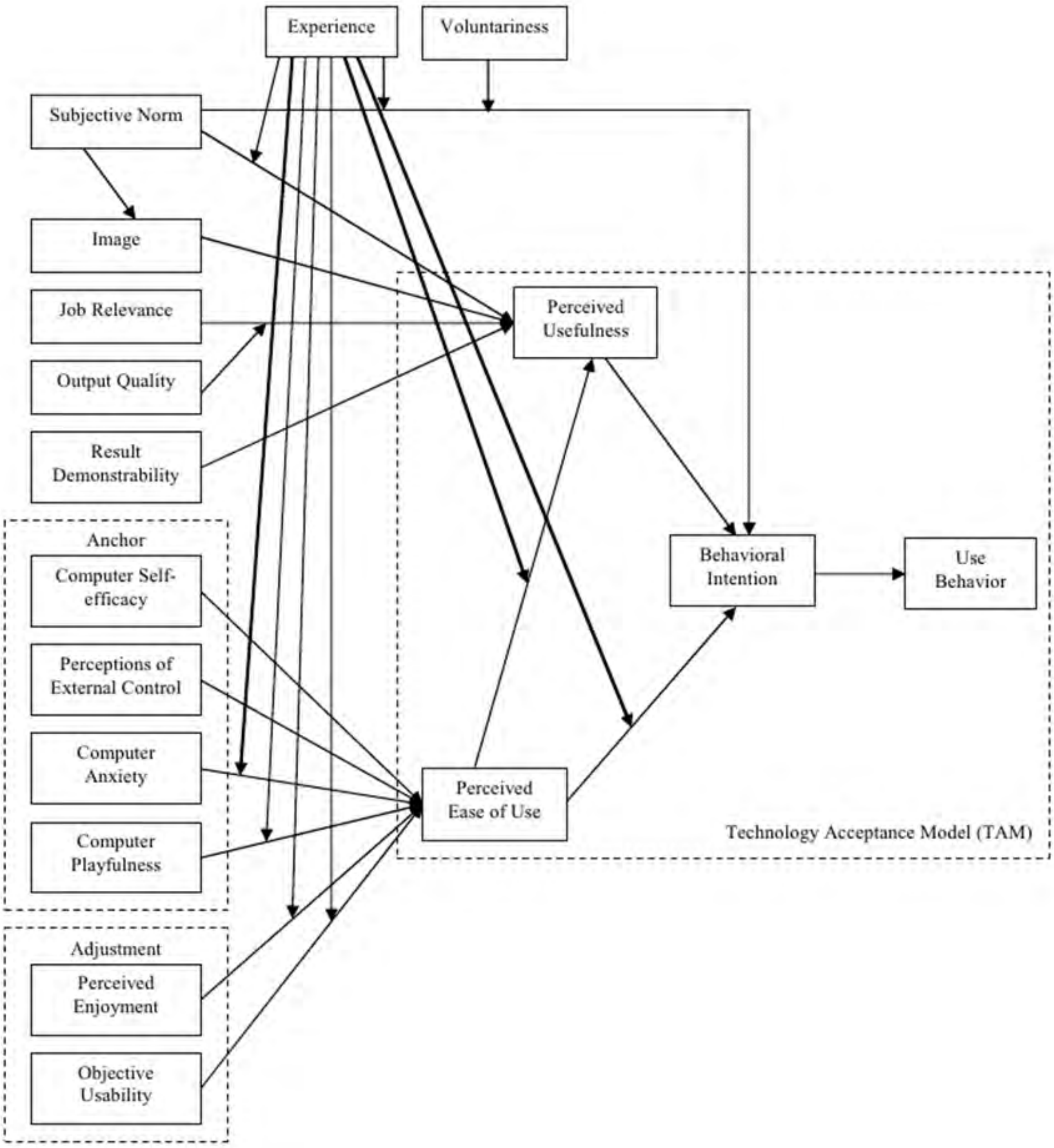


Figure 3.3 – TAM3 (Venkatesh & Bala, 2008, p. 280).

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

Venkatesh et al. (2016) has proposed a new model to explain users' intentions to use information technology and subsequent behaviour. The UTAUT model provides an analysis of individual adoption of technology in the organisational context (Laumer & Weitzel, 2009*a*, p. 12). The UTAUT model has the following four main constructs (Venkatesh et al., 2016, p. 330):

Performance Expectancy (PE) this is the degree to which using the system will improve job performance.

Effort Expectancy (EE) refers to the amount of effort needed to use the system. This can be considered as EU in TAM2 context.

Social Influence (SI) this is similar to subjective norms of TAM2 and it is basically the influence of people surrounding the person.

Facilitating Conditions (FC) refers to the degree to which it is believed that organisational and technical infrastructure is available to support the use of the information technology.

The impact of these constructs on the usage intentions and behavioural actions of the user is moderated by gender, age, experience and involuntariness of use. UTAUT has been used, integrated and extended to study the individual acceptance of technology in a variety of environments (Venkatesh et al., 2016, p. 331). These environments include different users, organisations and technologies. An example of a study that used UTAUT was conducted by Alrawashdeh et al. (2019) to investigate the acceptance of open source software by software developers. A study similar to the current research study has utilised UTAUT to investigate process acceptance by IT practitioners (Guardado, 2012). Other studies, (Al-Gahtani et al., 2007; Im et al., 2011; Rajapakse, 2011; Yuen et al., 2010) have extended the UTAUT model and analysed the moderation effects of national culture on the UTAUT relationships.

An organisational culture study based on the UTAUT model was conducted to investigate the effects of organisational culture on acceptance of Internet technology in a government agency of a developing country (Dasgupta & Gupta, 2010). It was found that organisational culture has an impact on individual acceptance and use of Internet technology and should therefore be carefully managed for successful adoption and diffusion (Dasgupta & Gupta, 2010).

UTAUT is an ideal candidate for the conceptual model being developed in this research study because it presents an analysis of individual adoption of technology in an organisational context, where the environment is mandatory as opposed to voluntary. This study aims to understand the Developers and Operators' perspectives towards the adoption of DevOps collaborative culture. In this regard, the organisational context is a mandatory DevOps setting. UTAUT has been used in IS settings (Guardado, 2012), however, the effects of organisational culture in these settings was unclear. On the other hand, Dasgupta and Gupta (2010)'s study based on UTAUT revealed that organisational culture does influence people on their acceptance of technology.

3.4 Diffusion of Innovation Model (DOI)

After understanding approaches that examine the ways in which individuals or a group of people decide to adopt innovations, it is would be beneficial to consider how these innovations are passed by from one individual or group to the next within and between organisations. The conveyance of innovations will influence the success of the innovation. Roger's DOI theory, which unpacks the processes involved in the spreading of innovation, is discussed in the following section.

3.4.1 Diffusion of Innovation (DOI)

The DOI theory explains how, why, and at what rate does innovation spreads. It is believed that innovation is not simultaneously adopted by everyone in the system but it is passed in stages. DOI is a framework for describing the adoption and non-adoption of new technology (Al-Mamary et al., 2016, p. 153). The process by which innovation is communicated over time amongst participants in the social system is referred to as diffusion by Rogers (2003). Diffusion can be observed when a social system of users share information and their opinions about new technology are communicated over some media, and this occurs progressively in the market (Al-Mamary et al., 2016, p. 153). The diffusion of innovation—a new idea, behaviour, or product- does not reach everyone simultaneously; some people take longer to adopt (late adopters) than others (early adopters) to adopt the innovation. According to the DOI model, there are five factors that influence the adoption of innovation and these are discussed briefly as follows :

Relative advantage this is the degree to which an innovation is seen as better than the previous idea that is being replaced.

Compatibility this is how consistent the innovation is with regards to values, experiences, and needs of the potential adopters.

Complexity this is how difficult the innovation is for use or understanding by potential adopters.

Triability this is the extent by which an innovation can be tested or experimented with, before a potential adopter can commit to it.

Observability this is the degree to which the new innovation can produce tangible results.

Al-Mamary et al. (2016, p. 154) mentions a shortcoming of DOI in that it only focuses on a product or innovation and does not consider the complexities brought by societal, cultural and economic factors that may influence how this innovation is adopted by the society. This model has a weakness in predicting the behaviour of individuals and organisations (Ward, 2013, p. 223). The model is also at the organisational level and it therefore not targeted at individuals (Oliveira & Martins, 2011, p. 111).

Since a DevOps culture encourages collaboration amongst team members, DevOps participants are more likely to influence one another. In this case, DevOps culture can be viewed as an innovation that needs to be communicated and spread amongst team members within the entire DevOps organisation. However, if poorly introduced, the innovation could have a negative impact on its acceptance and its proliferation. The extent to which and how the De-

Table 3.1 – Summary of model constructs.

Model	Constructs
Theory of Reasoned Action (TRA)	Attitude toward behaviour Subjective norm
Theory of Planned Behaviour (TPB)	Attitude toward behaviour Subjective norm Perceived behavioural control
Technology Acceptance Model (TAM), TAM2 and TAM3	Perceived usefulness Perceived ease of use Subjective norm Image Job relevance Output quality Result demonstrability Computer Self efficacy Expectations of external control Computer Anxiety Computer Playfulness Perceived enjoyment Objective Usability
UTAUT	Performance expectancy Effort expectancy Social influence Facilitating conditions.
Diffusion of innovation Theory (DOI)	Relative advantage Ease of use Image Visibility Compatibility Results demonstrability Voluntariness of use

vOps culture influences the adoption of DevOps in an organisation needs to be investigated.

The models and constructs that have been explored thus far in this research study are summarised in Table 3.1. As already mentioned, some of the models result from an extension of other models (e.g. TPB is an extension of the TRA model), and other models are a combination of two or more models. For this reason, similar constructs appear to common to these models.

3.5 Technology Acceptance Theories in Software Development Settings

Whilst the models discussed in the preceding sections have been used to explain the acceptance of developed products, more research is still needed to explain the acceptance of the development processes or methodologies by software developers. Studies on the acceptance of software development innovations such as programming languages (Agarwal & Prasad, 2000), tools (Chau 1996;Iivari J. 2011), and design and analysis techniques (Leonard-Barton 1987;Kozar 1989) have already been conducted. Hardgrave et al. (2003, p. 125) point out the importance of distinguishing between the adoption of technology tools and the adoption of the entire methodology. Although similarities are found in the adoption of technology tools and the adoption of methodologies, some differences are nevertheless still prevalent. Adoption of technology tools tends to be voluntary and incremental as opposed to mandatory and radical (Riemenschneider et al., 2002, p. 1140). When a technology tool is adopting, nothing forces any individual to use that particular tool. Senior management of an organisation would already have adopted a methodology and all that is left is the implementation of that particular methodology by the software developers within the organisation. Time to transform from old methodology to new is urgent. In addition, social pressure has more relevance in methodology adoption than in individual technology tools adoption because greater emphasis is placed on new technology use by project teams during software development (Hardgrave et al., 2003, p. 126). For example, when investigating the determinants (using usefulness, complexity, social pressure, organisational mandate and complexity as constructs) of intentions to adopt SMD, Hardgrave et al. (2003, p. 134) found that the usefulness construct possessed weak significance when compared to that of technology tools adoption. In contrast, the complexity construct was found not to be significant.

The fact that IS managers have adopted a particular SDM does not guarantee that developers will follow it to its maximum potential. Huisman (2004, p. 1) points out the relevance of distinguishing between the adoption and acquisition of technology at the organisational level and its adoption and implementation at the individual level. As far as Rogers (2003, 23) is concerned, SDMs are contingent innovations which have organisations as primary adopting units and individuals as secondary adopting units.

Technology acceptance models have been used in diverse settings, including the acceptance of SDM. As an example, Guardado (2012, pp. 1–177) used the UTAUT model to investigate the acceptance and the adoption of processes by IT practitioners. In addition, the use of the DOI model by Huisman and Iivari (2002) in the study of deployment of SDM at individual level, are some of the examples. Furthermore, Riemenschneider et al. (2002) carried out an investigation to explain developer acceptance of methodologies using constructs from five theoretical models, namely TAM, TAM2, Perceived Characteristic of Innovation—PCI, TPB and Model of Personal Computer Utilisation (MPCU). The study by Riemenschneider et al. (2002) revealed the following determinants as being significant:

- Usefulness (TAM, TAM2, PCI, TPB and MPCU);
- Subjective norm (TAM2, TPB and MPCU);
- Voluntariness (TAM2 and PCI); and
- Compatibility (PCI).

In another study, Hardgrave et al. (2003) investigated the determinants for the intentions of software developers to adopt a methodology, drawing on constructs from TAM and DOI. This study revealed that developer intentions are influenced by the following determinants (Hardgrave et al., 2003):

- Usefulness;
- Social pressure;
- Organisational mandate and
- Compatibility.

The findings of this study (Hardgrave et al., 2003) based on TAM and DOI models confirm the findings of the previous study conducted by Riemenschneider et al. (2002) based on the five models. Both studies suggest the mandate from the organisation to use a methodology, the opinions of co-workers and the compatibility of the methodology as being key drivers of developer intentions. In corroboration of the study on DOI model by Huisman and Iivari (2002), compatibility was also supported as one of the determinants of developer intentions. On the basis of the UTAUT model, Guardado (2012) similarly found social influence, and PE (which is equivalent to TAM's usefulness) to be amongst the significant constructs that inform intentions. Although important, the usefulness of the methodology is significantly weak. When an organisation adopts a methodology, whether developers view the adoption of a methodology by an organisation as being useful or does not exert much influence on the developers' intention to use the methodology as it would have if this was a voluntary decision.

3.5.1 Information Systems Development Acceptance Model (ISDAM)

After investigating the acceptance of information system by software developers, Hardgrave and Johnson (2003) proposed a model based on TPB from which attitude, subjective norm and perceived behavioural control provide the model with some of the foundational elements. This model, the Information Systems Development Acceptance Model (ISDAM), uses other elements from TAM/TAM2 (usefulness, subjective norm and EOU) and goal setting theory (personal and situational factors).

The model (Figure 3.4) posits that intentions to use a new Information System Development (ISD) process has the following determinants:

Organisational Usefulness (OU) it is the usefulness of the ISD process or methodology as seen by the developer. OU is defined as the evaluation of the usefulness of the ISD process to the organisation by the developer (Hardgrave & Johnson, 2003, p. 326).

Social Norm (SN) this influences the developers' intentions to accept the system in two ways—a direct influence from peers and managers, and indirectly through Organisational Usefulness (OU) (Hardgrave & Johnson, 2003, pp. 325-327). SN is the social influence of prominent individuals on a developer's acceptance of an ISD process.

Perceived Behavioural Control-Internal (PBC-I) this influences intentions directly and indirectly through OU (Hardgrave & Johnson, 2003, p. 327). PBC-I is the personal impediments perceived when performing a behaviour (Hardgrave & Johnson, 2003, p. 328). In the context of TPB, PBC is the ease or difficulty of performing a certain behaviour as per-

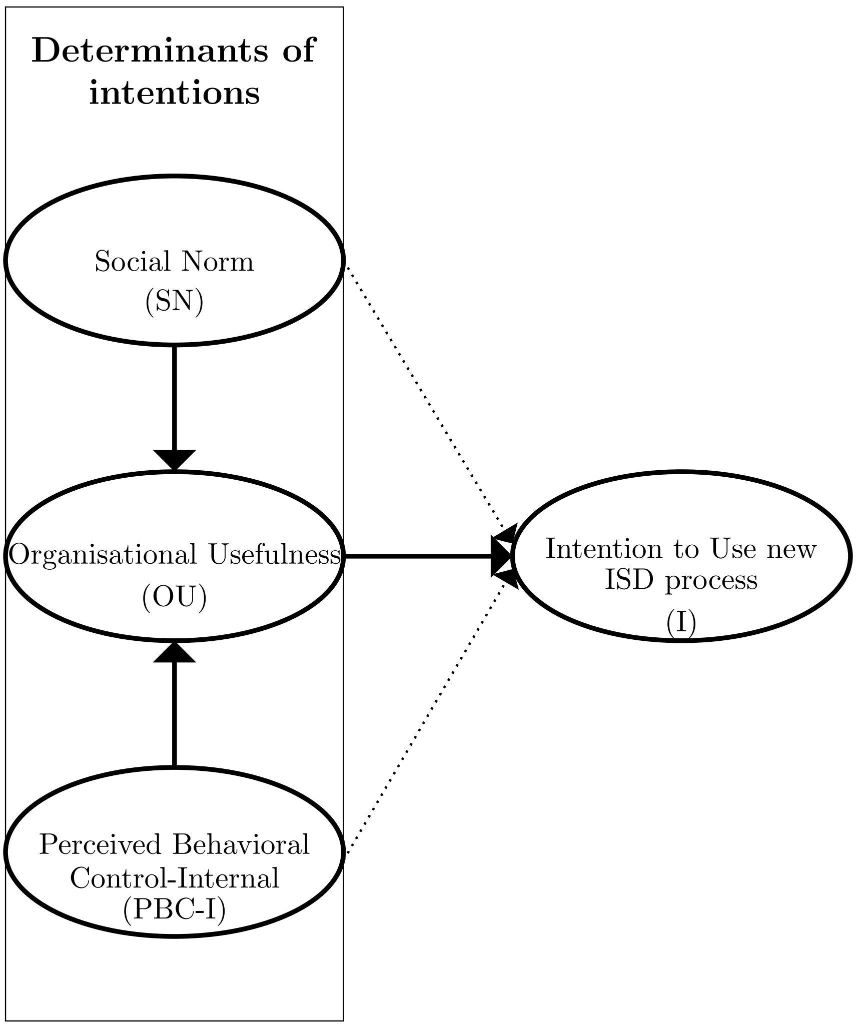


Figure 3.4 – Information System Development Acceptance Model (ISDAM), (Hardgrave and Johnson, 2003, p.331).

ceived by the person with the intent to implement the behaviour. The factors that contribute or control this behavioural choice may be external—those things outside that may affect behaviour, hence PBC-E—or internal—those that are personal, and hence PBC-I. This determinant is equivalent to the TAM personal EOU construct.

The above model was evaluated on 150 system developers and two constructs (Perceived Behavioural Control-External (PBC-E) and Personal Usefulness (PU)), which were part of the initial proposed model, were not found to be significant and were later removed from the final model. PBC-E is the perceived situational inhibitor to performing the behaviour and PU is the evaluation of the usefulness of the ISD process by the developer (Hardgrave & Johnson, 2003, p. 328). In addition, SN and PBC-I (Dotted lines in Figure 3.4) were not found to be significant and were also removed from the final model. Therefore, the only determinant for INT that remained in the final model is OU.

It is believed that the model fulfils its intended purpose—to explain the acceptance of ISD processes at an individual level in an organisation that has already adopted the ISD process (Hardgrave & Johnson, 2003, p. 334). The models discussed above lay a foundation for satisfying the primary goal of this study. This is because the model includes organisational culture, which is in the centre of DevOps philosophy. In addition, the model was developed for software development settings. For the purpose of this research study, SN and PBC-I was retained so that it can be evaluated using data from this research study. Such a retention will allow this research study to corroborate or refute the findings of (Hardgrave & Johnson, 2003, p.333).

The main goal of this research study was outlined in Chapter 1. It is important at this point to refresh the reader's memory about the research questions of this study. To this end, the main research question that this thesis was trying to answer is as follows:

What are the success factors that must be included in a coherent framework to support a successful collaborative culture in a DevOps environment?

The following sub-question has already been answered:

SQ1: *What are the characteristics of DevOps collaborative culture?*

The research questions that this thesis now attempted to answer were as follows:

SQ2: *What factors encourage a collaborative culture in a DevOps environment?*

SQ3: *How do the factors interact with each other?*

SQ4: *How can a collaborative culture be implemented in a DevOps environment?*

In an attempt to resolve the research questions SQ2–SQ4, a framework is proposed in the next section (Section 3.6). The identified characteristics of DevOps derived from SQ2 was used as input to the framework. The factors that promote a collaborative culture amongst Developers and Operators will be revealed and thus provide an answer to SQ2. As required for SQ3, the interactions of these factors were explored using statistical analysis methods. Ultimately, the framework was fine-tuned where necessary to reflect the factors and relationships between the factors in RQ4. The final output of this conceptual framework showed important collaboration factors and the relationships between these factors that is intended for DevOps settings.

3.6 DevOps Culture Acceptance Model (DCAM): The Proposed framework

The conceptual framework adopted in this study is based on the Information Systems Development Acceptance Model (ISDAM) by (Hardgrave & Johnson, 2003, pp. 322-336), which is described in the preceding sub-section. The reason for selecting this model is that it is simple and easy to understand—with only three determinants of intentions. Furthermore, Hardgrave and Johnson (2003, p. 325) argued that, on their own, these theories (TPB, goal-setting theory, and TAM/TAM2) do not answer the question of accepting information system development processes or general question of accepting a complex technological process or products. Hence, a combination of concepts from these theories makes ISDAM a suitable candidate model for utilisation in this research study. Some studies have used models such as DOI and UTAUT (Huisman & Iivari, 2002;Guardado, 2012), however, neither of the studies suggest that one model is better than the other. Likewise, the study by Mathieson (1991, p. 330) combined TPB and TAM without prioritising one model over the other. Additionally, Hardgrave and Johnson (2003, pp. 330) claimed that the ISDAM model had a better explanatory power when compared with previous studies conducted on TPB. Lastly, the ISDAM model was developed in relevant settings—acceptance of ISDM by developers—which is in line with this study.

In this research study, a comprehensive framework that extends the ISDAM model by including DevOps culture as a determinant of intentions is proposed. The proposed framework, DevOps Culture Acceptance Model (DCAM), is illustrated in Figure 3.5. In the DCAM, the DevOps culture has a direct influence on SN. In the context of DCAM, the intentions of the developers may be influenced by their peers and managers. On the other hand, the behaviour and beliefs of these peers and managers may be shaped by the organisational culture they operate in. The perception of the developer towards the usefulness of the ISD process for an organisation is directly influenced by the culture of the organisation, which in turn influences the intentions of the

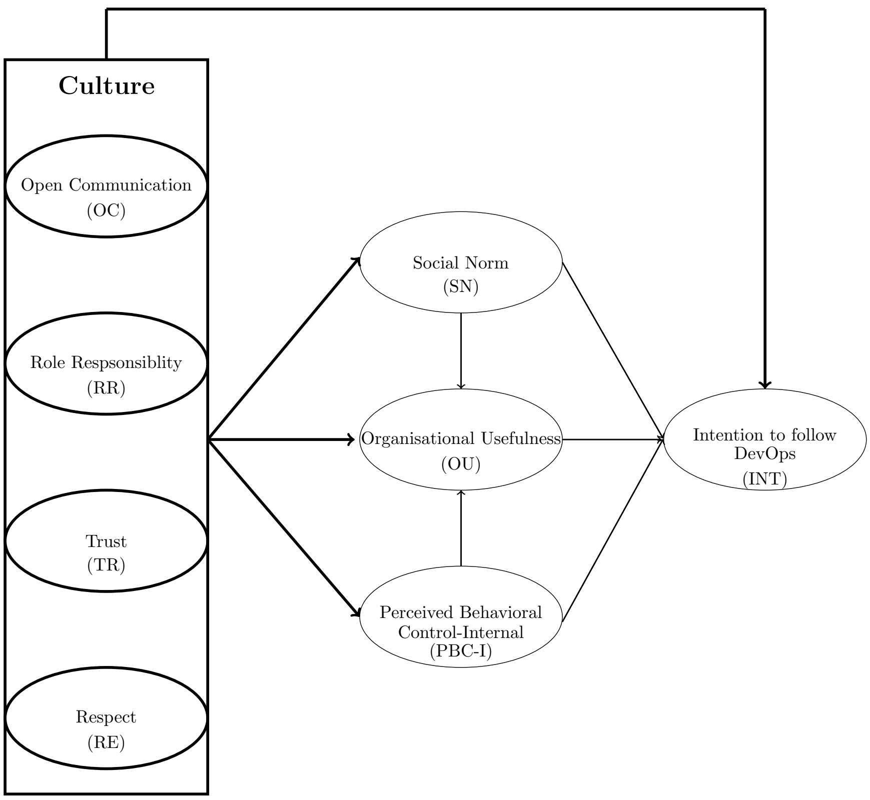


Figure 3.5 – Prosed DevOps Culture Acceptance Model (DCAM) (Masombuka & Mnkandla, 2018, p. 283).

developer. Therefore, organisational culture is a determinant of OU. Lastly, organisational culture may have an influence on the PBC-I.

It was mentioned that DevOps follows agile development methodologies. Agile methodologies are better accepted in originations that adopt group development cultures (Ben Othman et al., 2016, p. 22). The group cultures are human-centric and allow flexibility within the group Chan and Thong (2007, p. 22), which promotes group dynamics. This group development culture also promotes flexibility over stability. A detailed discussion on DevOps collaborative culture was presented in Section 2.3.5. Four core elements of DevOps culture were identified by Walls (2013, pp. 5-7). The elements are open communication, roles and responsibilities alignment, respect and trust and they form part of constructs of the proposed framework.

3.6.1 Proposed framework variables

For the proposed framework (Figure 2.3), nine variables need to be examined. The intention of the developer to follow DevOps as the ISD process in the future is the dependent variable of this research study. These following variables are independent variables:

- Open Communication (OC);
- Roles and Responsibilities alignment (RR);
- Trust (TR);
- Respect (RE);

- Social Norm (SN);
- Organisational Usefulness (OU), and PBC-I.

  The last variable is the intention variable, which is:

- Intentions to follow DevOps processes (INT).

A summary of these variables as determinants of intentions constructs and their source of origin is presented in Table 3.2.

3.6.2 Proposed hypothesis

The variables of the proposed framework have been identified and the relationship between these variables is now explained. There are four relationships that need to be explored. The first relationship is between the independent variables (OC, RR, TR, and RE) and the dependent variable (INT). It is proposed that DevOps culture has a direct influence on the intention to follow DevOps processes. In terms of the independent variables, this signifies that when OC is high within the organisation, it is predicted that the more likely DevOps practitioners will be willing to follow DevOps as an ISD process. Similarly, with the other independent variables, it is predicted that high RR, TR, or RE will result in high intentions to follow DevOps processes. This led to the formulation of the following hypothesis:

H1: ***There is a positive relationship between DevOps culture and intention to use DevOps.***

H1a: *There is a positive relationship between open communication and intention to follow DevOps processes.*

Table 3.2 – DCAM Determinants of Intention Constructs.

Source of origin	**Determinants of Intention constructs**	**Definition/Description**
DevOps Culture	Open Communication (OC)	The team should communicate about the product through its life cycle.
	Roles and Responsibilities alignment (RR)	The whole team should be rewarded for their efforts and take responsibility for product failure.
	Trust (TR)	The team members should trust each other and that they are working towards a common goal.
	Respect (RE)	The team members need not have to like each other but they should recognise every member's contribution to the team.
Information System Development Acceptance Model (ISDAM)	Organisational Usefulness (OU)	The evaluation of the usefulness of the ISD process to the organisation by the developer
	Subjective Norm (SN)	The social influence of important individuals on a developer's acceptance of an ISD process.
	Perceived Behavioural Control-Internal (PBC-I)	The personal impediments perceived when performing a behaviour.

H1b: *There is a positive relationship between roles and responsibilities alignment and intention to follow DevOps processes.*

H1c: *There is a positive relationship between trust and intention to follow DevOps processes.*

H1d: *There is a positive relationship between respect and intention to follow DevOps processes.*

The second set of relationships that requires investigation involve INT in relation to SN, OU and PBC-I. The study conducted by Hardgrave and Johnson (2003) revealed that intentions to use an information system development processes are influenced by SN, OU and PBC-I.

H2: ***There is a positive relationship between SN and INT.***

H3: ***There is a positive relationship between OU and INT.***

H4: ***There is a positive relationship between PBC-I and INT.***

The third set of relationships is between DevOps culture and SN. The framework suggests that SN is influenced by DevOps, culture which in turn influences an individual's intentions. In other words, it is the culture that shapes the beliefs and values of important individuals such as peers and manager who influence the intentions of an individual DevOps practitioner. Therefore, the following hypothesis was put forth:

H5: ***There is a positive relationship between culture and individual subjective norm.***

H5a: *There is a positive relationship between open communication and SN.*

H5b: *There is a positive relationship between roles and responsibilities alignment and SN.*

H5c: *There is a positive relationship between trust and SN.*

H5d: *There is a positive relationship between respect and SN.*

The fourth set of relationships suggested by the framework is between DevOps culture and OU. As stated previously, OU is the evaluation of the usefulness of the ISD process of the organisation by the developer (Hardgrave & Johnson, 2003, p. 325). The framework suggests that culture has an influence on how the developer evaluates a process—DevOps in the case— as being useful to an organisation. This means that the presence of a good value in the culture increases the chances of an individual evaluating that culture as being useful to an organisation. In light of this information, it was demonstrated by Hardgrave and Johnson (2003, p. 325) that OU directly influences the intentions of developers. This study hypothesises the following:

H6: ***There is a positive relationship between culture and OU.***

H6a: *There is a positive relationship between open communication and OU.*

H6b: *There is a positive relationship between roles and responsibilities alignment and OU.*

H6c: *There is a positive relationship between trust and OU.*

H6d: *There is a positive relationship between respect and OU.*

The fifth set relationship this study was investigating involves DevOps culture and PBC-I. The proposed framework suggests that organisational culture has an influence on the personal characteristics of the DevOps practitioner. Hardgrave and Johnson (2003, pp. 332) have advanced that developers work in teams and that "team spirit" is instilled in their work culture thus leading to individuals tending to identify with the team. From this relationship, the following hypothesis was proposed:

H7: ***There is a positive relationship between culture and PBC-I.***

- *H7a:* *There is a positive relationship between open communication and PBC-I.*
- *H7b:* *There is a positive relationship between roles and responsibilities alignment and PBC-I.*
- *H7c:* *There is a positive relationship between trust and PBC-I.*
- *H7d:* *There is a positive relationship between respect and PBC-I.*

The last set of relationships that requires investigation involve OU as in the study by Hardgrave and Johnson (2003, p. 330) revealed that OU is influenced by both SN and PBC-I. For this reason, the following hypothesis is proposed:

H8: ***There is a positive relationship between SN and OU.***

H9: ***There is a positive relationship between PBC-I and OU.***

The approaches to a DevOps team that were adopted for this research study were presented in section 2.3.2. These are:

- A mixed responsibility approach in which both the development and operation responsibilities are assigned to all engineers.
- A mixed personnel approach consisting of developers and operators working together as a single DevOps team.

With these two approaches, it is important to further investigate how they influence the outcomes of the results. A new variable (a moderator) called 'job role' was used to measure the strength of the relationship between the dependent and independent variables. A moderator variable interacts with another independent or predictor variable to predict scores on and accounting for variance in a dependent or predicted variable (Salkind, 2007, p. 624). The job role has the following three levels:

Dev representing the developer roles;

Ops representing the operator roles; and

D&O which represents roles involving developer and operator at the same time.

The following effects were therefore hypothesised:

H10: The interaction between DevOps collaboration culture and SN is moderated by job role.

H10a: The interaction between DevOps collaboration culture and SN is moderated by *Dev* job role.

H10b: The interaction between DevOps collaboration culture and SN is moderated by *Ops* job role.

H10c: The interaction between DevOps collaboration culture and SN is moderated by *D&O* job role.

H11: The interaction between DevOps collaboration culture and OU is moderated by job role.

H11a: The interaction between DevOps collaboration culture and OU is moderated by *Dev* job role.

H11b: The interaction between DevOps collaboration culture and OU is moderated by *Ops* job role.

H11c: The interaction between DevOps collaboration culture and OU is moderated by *D&O* job role.

H12: The interaction between DevOps collaboration culture and PBC-I is moderated by job role.

H12a: The interaction between DevOps collaboration culture and PBC-I is moderated by *Dev* job role.

H12b: The interaction between DevOps collaboration culture and PBC-I is moderated by *Ops* job role.

H12c: The interaction between DevOps collaboration culture and PBC-I is moderated by *D&O* job role.

H13: The interaction between DevOps collaboration culture and INT is moderated by job role.

H13a: The interaction between DevOps collaboration culture and INT is moderated by *Dev* job role.

H13b: The interaction between DevOps collaboration culture and INT is moderated by *Ops* job role.

H13c: The interaction between DevOps collaboration culture and INT is moderated by *D&O* job role.

The variables proposed in this framework are complex in nature. In other words, there are different ways in which they can be evaluated. The several ways in which these variables could be measured and how they were evaluated in this research study is outlined in the section that follows.

3.6.3 Measurements of variables

The variables that were measured in this research study are: Open communication; Roles and responsibilities alignment; and Respect and Trust. The different ways in which these variables can be measured are discussed in this section. An approach that has been adopted in this research study for measuring these variables is also outlined. All variables were evaluated in a self-perspective manner, as perceived by the user.

3.6.3.1 *Open communication*

Open communication requires that every team member is kept informed about the software product throughout its development life cycle. Product development conflicts may arise as a result of poor communication—that is information that is received late, is unclear or is left out (Zhang et al., 2014, p. 17)—between Developers and Operators teams. Cheng et al. (2016, p. 273) defines openness as the degree to which the culture of a team is open to allow information to flow freely as needed. Team members (i.e. developers, quality assurance engineers, and system administrators) should discuss what they are working on during DevOps-style stand-up meetings (Walls, 2013, p. 15), to keep all the team members up-to-date. It is understandable that these meetings should be as frequent as possible so that every member is kept on the loop.

Another aspect of open communication is logistical—the availability of common physical space that allows for the chance of interactions (Walls, 2013, p. 16). Stand-up meetings are generally effective if held at a commonplace. This is corroborated by studies by Hummel et al. (2013, p. 345) which investigated the role of communication in agile development. These studies indicated that communication was effective in teams that are co-located, and allows collaboration by using whiteboards, status boards, and other informal communication media.

Two themes that stand out from the above discussion that are relevant to the DevOps team are:

- **Communication frequency**—team members need to be kept up to date about the progress and changes to the software product. If this information is communicated often enough, risks of communication conflicts that could arise can be avoided;
- **Availability of physical space to interact**—agile development promotes frequent face-to-face interactions of team members over documentation.

In this research study, communication frequency was adopted as an appropriate measure of open communication. This is because communication frequency is more likely to measure the openness of communication as required by DevOps and is expected to shed some light on how often information is exchanged from one person to the other. In the case of this research study, it is irrelevant how this information is exchanged. The communications may be through using formal or informal channels.

3.6.3.2 *Trust*

Trust is defined as the willingness of a group or individual to make themselves vulnerable to another group or individual, based on the confidence exhibited by the other party on the following characteristics—vulnerability, confidence, benevolence, reliability, competence, honesty and openness (Cheng et al., 2016, p. 271);

- **Willingness to risk vulnerability**—Trust is not taking risks per se, but rather a willingness to take a risk (Cheng et al., 2016, p. 271). Tschannen-Moran and Hoy (2000, p. 548) have pointed out the importance of coupling trust with interdependence by arguing that there is no need for trust if there is no interdependence.

- **Confidence**—In order for one person to trust another person to produce something that is beneficial to the trusting person, the trusting person must have confidence that the other person has the ability and intention to produce it.

- **Benevolence**— Benevolence is defined as the confidence that an individual's well-being or things the individual cares about will be protected and not harmed by the trusted party or group. Trust in this sense acts as an assurance that the trusted person will not exploit or take advantage of the vulnerability of the person who trusts.

- **Reliability**— The concept of reliability means that there is a sense of confidence that an individual's basic needs will be met in a positive way by the trusted person. It combines the sense of predictability—knowing what to expect from others—with benevolence bring a sense of confidence that the need will be met (Wilson et al., 2006, p. 19).

- **Competence**—this is the ability to perform as expected and according to the standards of the current assignment. To be able to trust, a person must feel that the individual or group being trusted possesses the capacity, skills, and resources to act in a reliable, benevolent manner.

- **Honesty**— this is related to the character, integrity, and authenticity of a person (Wilson et al., 2006, p. 19), and hence the fundamental facet of trust (Tschannen-Moran & Hoy, 2000, p. 558).

- **Openness**— this is the extent to which relevant information is not withheld (Tschannen-Moran & Hoy, 2000, p. 558). Openness may involve revealing personal information and thus making people vulnerable; however, confidence that all participants are facing similar risks reduces vulnerability.

Although all facets of trust mentioned above are important, Tschannen-Moran and Hoy (2000, p. 558) indicated that the weight given to the respective facets will depend on the nature of the interdependence and vulnerability in the relationship. This means one facet may be more important than the other or not relevant at all depending on the situation. Walls (2013, p. 6) identified the types of trusts needed by the DevOps team as being the following:

- Trust between the Operations and Development teams that ensures that the two teams are doing what they are supposed to do. This kind of trust is associated with confidence and competence trust. The one team needs to be confident that the other team has the ability and intention to do what needs to be done. The trusted team must demonstrate competency by showing that it has capacity, skills and resources to perform its assignments as required in a reliable and benevolent manner;

- Developers must trust the Quality Assurance (QA) team and believe that it is not just there to sabotage their success. This may be associated with reliability where one team trusts that its needs will be met in a positive way. Benevolence can also be observed in the sense that Development will have to have confidence that QA will protect their code; and
- A product manager needs to trust the Operations team to give objective feedback and matrix after the next deployment. Reliability is also the dominant trust fitting this scenario.

As previously stated, different kinds of trust variables can be associated with the above examples, depending on the facets that are being identified. This means that diverse people may perceive different trust variables that they consider to receive more weight than what is argued in this research study. Furthermore, other variables are embedded within one another; for example, reliability has confidence and benevolence characteristics. In summary, in this study, confidence was used as a measure of trust because confidence is considered to be the most meaningful in these circumstances.

3.6.3.3 *Respect*

Walls (2013, p. 6) points out the significance of team respect by highlighting that the contribution of every team member should be recognised, and no member should be afraid to speak because of fear of abuse. This characteristic relates to open communication. It is important that every member of the team feels that they are part of the team when discussions and decisions are made. If a person feels the sense of belonging to a group and the group values that person as their own, the person's willingness to contribute to the group would be affected accordingly. This is substantiated by Ellemers et al. (2013, pp. 21-37) in a study on how perceived respect affects positive team identity and willingness to invest in a team. Ellemers et al. (2013, p. 23) noted that being respected in a group does not define an individual's value to the group; instead, it is how an individual is valued as a member of a group. It was argued that it is the value of self for a group that determines the willingness to contribute to the group (Ellemers et al., 2013, p. 23). Respect in this research study was measured by how a person perceives respect from fellow team members based on that person's contribution to the team.

3.6.3.4 *Roles and responsibilities alignment*

Walls (2013, p. 16) stresses the significance of aligning role responsibilities of team members and keeping a schedule of this alignment that is up to date. Consequences of unclear roles are witnessed in fault recovery in siloed teams whereby time is wasted while trying to track responsible people. When people's roles and responsibilities are not clear, it becomes hard for the team to react promptly to arising situations. Role ambiguity represents a lack of explicit information regarding a particular role and may cause a negative effect on performance (Beauchamp et al., 2002, p. 229) among team members. Eys and Carron (2001, pp. 359-360) described four manifestations of ambiguity as being:

- **Scope of responsibility**—the role of team members' perception of a lack of clear information about the breadth of a member's duties or responsibilities. Team members should recognise their boundaries when it comes to their duties and the duties of others.

- **Behavioural responsibility**—the role of team members' perception of a lack of clear information about the behaviours required to fulfil the duties or responsibilities of a member. Put differently, members of a team should know what their role requirements are.

- **Evaluation of performance**—the role of team members' perception of a lack of clear information about how a team member's performance of role-related responsibilities is to be evaluated. This will encourage team members to self-evaluate their performance and where necessary do self-correction to fulfil their duties.

- **Consequences of not fulfilling responsibilities**—the role of team members' perception of a lack of clear information about the consequences of a failure to fulfil the responsibilities of team members. The team members should understand what or who will be affected by their absence of responsibility.

Since DevOps is targeted towards the removal of silos, it is a requirement that Developers and Operators work hand in hand from the beginning of the project to the delivery of the project. It is, therefore, important that roles are clearly defined to avoid the above mentioned ambiguities. In other words, the role and the responsibilities of the Operators at the beginning of the project and the role and responsibilities of the Developer during the deployment should be clear to every team member. If these roles are not clear, problems of blame-shifting may occur where members are not sure of who is to accept accountability at any stage of the project.

In addition, team members should understand how their work is going to be evaluated so that they do not have to account for people who failed to perform their duties. Walls (2013, p. 6) advices that the team should be incentivised based on their core goal of delivering a functional product and not reward Developers for producing lots of code, or punishing Operators for the code that does not work at production phase. Such an approach promotes the atmosphere of collaboration and teamwork.

In this research study, the alignment of roles and responsibilities was measured based on one of the four facets of ambiguity (i.e. the scope of responsibility). The reason for intentionally choosing the scope of responsibility is that it is in line with what Walls (2013, p. 6 & 16) was highlighted as being an important point in this category. Secondly, in the context of DevOps teams, performing duties and failure to perform them seem obvious once the scope of responsibility is understood.

Table 3.3 shows the framework variables that represents the DevOps culture adopted for this research study. Various perspectives that are used for measuring these variables exist, and the ones listed in Table 3.3 were used for this research study.

3.7 Chapter Summary

In this chapter, a proposed framework that was used for the investigation of the factors that influence Developers and Operators to accept and use the

Table 3.3 – Summary of variables measurements.

Variable	**Perspective**
Open communication	Frequency of use
Trust	Confidence
Respect	Contribution
Roles and responsibilities alignment	Scope of responsibility

DevOps approach to collaborative software development was presented. It was indicated that in order for an organisation to successfully adopt a new culture, OCM practices need to be followed. However, a discussion of these practices falls outside the scope of this research study. This chapter focused on identifying factors that would influence DevOps practitioners in willingly accepting a collaboration culture approach.

When identifying the contributing factors, it was deemed critical to explore theories that explained how human behavioural intentions are influenced. The two social science theories, which posit that social norm, social attitude and behavioural control influence the intended behaviour, are TRA and TPB. TRA and TPB are general in nature (they are applicable to most situations) and were adapted for technology studies by adding more constructs to predict the acceptance of technology by individuals. These technology acceptance models that were discussed in this chapter are: TAM and its variants, and UTAUT.

Another model that was looked at is the DOI model. Although DOI is not a technology acceptance model, an exploration of this model was deemed worthwhile because it explains how innovation is communicated and propagated over a period of time within a community. With all the relevant constructs checked, a prediction of whether the innovation would be spread successfully within the required period of time becomes a reality.

Once all the technology acceptance models explaining how individual users adopt and accept technology were discussed, the software development environment was explored. It was highlighted that several technology acceptance models have been tested in a different context, including software development environments. A model (ISDAM) constructed for this context was scrutinised. Not only was the model constructed from tried and tested constructs of earlier models, it was also purposefully designed for the software development environment. Using ISDAM as a logical foundation for this research study, new constructs were added to the model thus resulting in a new framework being built for the thesis.

In the new framework, new variables—open communication, roles and responsibilities alignment, respect and trust—were included. The framework indicated possible relationships between variables. How these variables truly influence the overall intention is an ideal subject for the next chapter.

Chapter 4

Research Methodology

4.1 Introduction

The previous chapters have provided detailed information showing why conducting this research study was significant and relevant. This chapter aims to outline the roadmap of how this study proceeded with the investigation of the research problem. In this chapter, the question of how the variables under investigation were addressed is explained in detail. The tools that were used in the investigation are also described and justified. This research methodology entails the approaches, procedures, sampling and data collection methods this study has employed.

This chapter is presented as follows: Section 4.2 provides an outline of which research process is using a "research onion" to illustrate the processes. During this illustration, the adopted research philosophy for this study is revealed. In addition, the research approach, strategy and design are also discussed. Section 4.3 describes the data collection instruments that were used and presents the settings under which this data collection was conducted. The analysis process of the collected data and the credibility of this study are discussed in Sections 4.4 and 4.5, respectively. The limitations of this study in terms of the methodology and the ethical consideration are presented in Sections 4.6 and 4.7, respectively.

4.2 Research Process

Saunders et al. (2009, p. 108) have developed what they called a "research onion", which was implemented in this study. This research onion constitutes distinct layers corresponding to different research processes. These layers resemble a literal onion, which can be peeled from the outer layer to gain access to inner layers. In this analogy, an effective progression through different research processes is achievable with such a design. Although the research onion was developed to give a clear meaning of research stages for effective formulation of research methodology, it can be adapted as required to meet the specific context of a researcher. Figure 4.1 depicts such adaptation to meet the context of this research study. According to the adapted research onion,

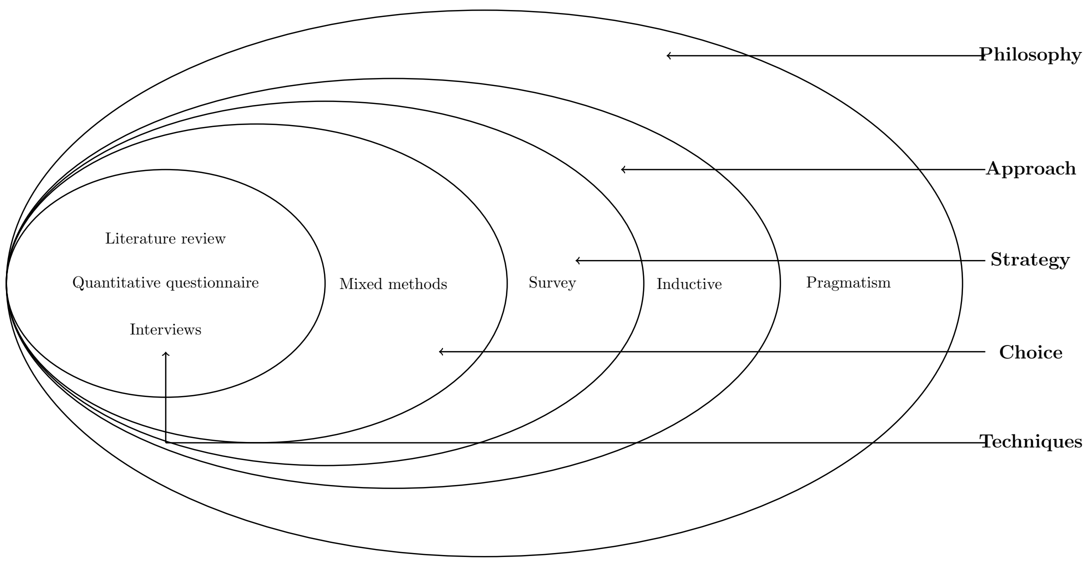


Figure 4.1 – Adapted Research Onion from Saunders et al. (2009, p. 108).

the outer layer is the research philosophy, followed by the research approach and the third layer is the research strategy. The last innermost layers show the methodological choice and data collection techniques. In the subsequent section, these distinct layers are briefly discussed.

4.2.1 Research philosophy

There exists a number of research paradigms and perspectives in the world of science. These include interpretivist vs positivist research, inductive vs deductive research, and qualitative vs quantitative studies. According to De Villiers (2012, pp. 239-240), a paradigm is a set of assumptions that provides a philosophical view, which leads to an organised study of a specific population. The paradigm serves the following purposes:

1. Establishment of standards for instruments, methodology and data collection and conducting of research;
2. Provision of procedures, philosophies and methods for similar contexts;
3. Guidance of researchers by specifying challenges within a discipline; and
4. Development of theories and models that permit researchers to solve the arising problems.

Creswell (2003, p. 5) states that there are three questions relating to research design that should be addressed. These questions are concerned with:

1. *The knowledge claims made by the researcher.* These knowledge claims are referred to as research paradigms—philosophical assumptions, epistemologies and ontologies—or research methodologies. A research paradigm is an important element in the research process. The basic beliefs that define a specific paradigm can be summarised by looking at the following three questions:

   (a) The ontological question asks what is the form and nature of reality. According to Creswell (2003, p. 6), ontology is what researchers claim about what knowledge is. It is the nature of social reality.

   (b) The epistemological question which asks what are the basic beliefs about knowledge. As far as Creswell (2003, p. 6) is concerned, epistemology is how this knowledge is acquired. It is the nature of knowing and the construction of knowledge.

   (c) The methodological question asks how a search is conducted on what an individual believes can be known. This process of studying this knowledge is methodology.

2. *The strategies that will inform the procedures to be followed.* Strategies of enquiry bring about the choice of a research design assumption about knowledge claims. They contribute to the overall research approach—the qualitative, quantitative and mixed methods approach. The research strategies undertaken by this study are justified in Section 4.2.3.

3. *Methods of data collection and analysis to be used.* This involves the selection of appropriate data collecting instruments. The choice of method to be used depends on whether the information to be collected is specified in advance or it emerges from participants (Creswell, 2003, p. 6).

Table 4.1 – Research paradigms (Saunders et al., 2009, p. 119).

	Positivist	**Realism**	**Interpretivist**	**Pragmatism**
Ontology: the researcher's view of the nature of reality or being	External, objective and independent of social actors	Is objective. Exists independently of human thoughts and beliefs or knowledge of their existence (realist), but is interpreted through social conditioning (critical realist)	Socially constructed, subjective, may change, multiple	External, multiple, view chosen to best enable answering of research question
Epistemology: the researcher's view regarding what constitutes acceptable knowledge	Only observable phenomena can provide credible data, facts. Focus on causality and law like generalisations, reducing phenomena to simplest elements	Observable phenomena provide credible data, facts. Insufficient data means inaccuracies in sensations (direct realism). Alternatively, phenomena create sensations which are open to misinterpretation (critical realism). Focus on explaining within a context or contexts	Subjective meanings and social phenomena. Focus upon the details of situation, a reality behind these details, subjective meanings motivating actions	Either or both observable phenomena and subjective meanings can provide acceptable knowledge dependent upon the research question. Focus on practical applied research, integrating different perspectives to help interpret the data
Axiology: the researcher's view of the role of values in research	Research is undertaken in a value-free way, the researcher is independent of the data and maintains an objective stance	Research is value laden; the researcher is biased by world views, cultural experiences and upbringing. These will impact on the research	Research is value bound, the researcher is part of what is being researched, cannot be separated and so will be subjective	Values play a large role in interpreting results, the researcher adopting both objective and subjective points of view
Data collection techniques most often used	Highly structured, large samples, measurement, quantitative, but can use qualitative	Methods chosen must fit the subject matter, quantitative or qualitative	Small samples, in-depth investigations, qualitative	Mixed or multiple method designs, quantitative and qualitative

The research philosophy presents a logical way in which data should be collected, analysed and used. In this way, the research philosophy helps in reaching an appropriate selection of a research methodology. Various research philosophies exist, and these include positivism, realism, interpretivism and pragmatism. Table 4.1 shows common paradigms which are most relevant for this research study. Differences between interpretivism and positivism, as suggested by Thornhill et al. (2009, p. 119), lie in the methods that are used for the research study in question. The positivist paradigm relies on surveys, experiments, and field studies. In contrast, the interpretivist paradigm allows the researcher to confide in ethnomethodological, phenomenographic, ethnographic and case research methods (Creswell, 2013, p.48). When it comes to selecting these methods, a number of factors come into play. These may include the training received by the researcher, social influence by immediate people a researcher is exposed to and methods that are likely to produce potential insight about the research being conducted. With all that said, Table 4.1 shows potential research methods that are relevant to each research paradigm. In the same vein, it is noteworthy that it is possible to use more than one method to collect and analyse data in a single study. This is referred to as triangulation. Methodological triangulation combines different research methods and theory of triangulation and makes explicit reference to more than one theoretical tradition to analyse data (Payne & Payne, 2004, p. 230). Within-methods triangulation occurs when variant forms of the same method are used, and between-methods triangulation occurs when different methods are used. This means that combining different methods such as quantitative and qualitative methods—mixed methods—is a form of between-methods triangulation. As philosophical underpinning of mixed methods, Creswell

(2003, p. 12) mentions pragmatism as a suitable research paradigm. In this instance, the research problem is more important than the research methods, where different approaches are used to derive knowledge about the research problem. In terms of DevOps research, 77% of the studies relied on pragmatism as their research philosophy(Guerrero et al., 2020, p. 55). The following interpretations hold for pragmatism (Creswell, 2003, p. 12):

1. It is not devoted to any one system of philosophy and reality.
2. A researcher has freedom of choice when it comes to methods, techniques, and procedures of research depending on contextual needs and purpose.
3. The world is not seen as an absolute unity. Different approaches to collecting and analysing data are adopted instead of committing to a single approach.
4. Truth is what works at that time. A researcher may employ both quantitative and qualitative data to better understand the research problem.
5. Looks at "what" and "how" of research based on the approach of the researcher. The purpose of mixing methods should be established first.
6. Research occurs in a different context and therefore mixing methods could be ideal in such situations where different theoretical lenses are required.
7. Pragmatists believe that asking questions about reality and laws of nature should be stopped and the subject should be changed.

In summary, pragmatism uncovers opportunities to use multiple methods, different world views, different assumptions, and different forms of collecting and analysing data (Creswell, 2003, p. 12). This study was aimed at identifying contributing factors and their relationships, for a successful DevOps collaboration framework. After the developed framework is built, it should be evaluated. Therefore, multiple methods were necessary to accomplish the objectives of this study. This study, therefore, followed a pragmatist philosophy as shown in the research onion illustrated in Figure 4.1.

4.2.2 Research approach

The second sphere of the research onion shown in Table 4.1 is the research approach. Two approaches to reasoning employ inductive and deductive methods (Gioia et al., 2013, p. 17). De Villiers (2012, p. 240) view inductive reasoning as a reasoning process whereby if the evidence supporting an argument is believed it supports but does not ensure the conclusion. Inductive reasoning is exploratory and open-ended, beginning with precise observations and resulting in wider generalised theories. This bottom-up approach is useful for detecting patterns leading to hypotheses that can be explored and ultimately providing convincing evidence to support a conclusion (Payne & Payne, 2004, p. 175). When there is no enough pre-existing knowledge in the area of the research, Creswell (2014, p. 34) recommends using the inductive approach.

On the other hand, deductive reasoning is narrower and is concerned with hypotheses confirmation (Payne & Payne, 2004, p. 170) using various techniques to test real-world theories. The deductive reasoning approach follows a top-down approach, starting with theories and resulting in hypotheses that are thereafter tested by Creswell (2013, p. 64). The generation of ideas is centred on individual experiences and theories, and the hypotheses that arise from a literature search of the specific study (De Villiers, 2012, p. 242).

As suggested by Creswell (2013, p. 65), both approaches can be combined to an extent that inductive exploratory and deductive confirmatory questions could be included in the same study. Creswell (2013, p. 65) has reasoned that qualitative questions remain mainly inductive questions, whilst quantitative questions are deductive hypotheses testing. This suggests that mixed qualitative and quantitative research methods include inductive and deductive elements Payne and Payne (2004, p. 175), where deduction shapes the argument and induction establishes agreement about one or more pieces of an argument (Payne & Payne, 2004, p. 176). The two approaches (inductive and deductive approaches) have a connection between them. Creswell (2013, p. 68) explains that inductive reasoning is used to show that a causal relationship exists and builds facts on which the deduction is formulated.

In summary, research may follow a deductive or inductive approach. A deductive approach begins with a given theory that is more general and moves towards more specific issues. Inductive approach, on the other hand, works the opposite way, starting with being more specific and generating a general theory. In the context of this research study, more general collaboration theories were studied and data was collected, which led to specific new knowledge contributions about collaboration in DevOps settings. This study was largely quantitative in nature, and deductive reasoning was predominantly used to validate identified success factors in the context of DevOps collaboration culture. Therefore, this study has adopted a deductive research paradigm approach.

4.2.3 Research strategy

Two common research strategies mentioned by Walliman (2006, p. 40) are exploratory and confirmatory approaches. Creswell (2013, p. 70) explains that in confirmatory studies, a researcher seeks to confirm pre-existing relationships. The exploratory approach is appropriate when a researcher is interested in clarifying the most common relationships and estimates any causal effects. A confirmatory research strategy is defined as a strategy that uses empirical analysis to confirm (or invalidate.) the proposed hypotheses (Walliman, 2006, p. 43). An exploratory research strategy, on the other hand, focuses on the closely related elements of evidence and theory (Walliman, 2006, p. 42). The confirmatory approach to a study is commonly supported by researchers with hypothetical and experimental backgrounds, while an exploratory approach of the study is inductive in nature and is commonly supported by those with an interpretivist alignment (Creswell, 2013, p. 72).

Confirmatory research is based on arithmetical inferences and the deductive approach of the descriptive statistics (De Villiers, 2012, p. 248). It starts by identifying hypotheses; thereafter these hypotheses are verified by answering particular questions (De Villiers, 2012, p. 248). According to Creswell (2013, p. 75), an exploratory research aims to generate and combine novel ideas, and it relies heavily on probability models that are developed directly from the data. In contrast, confirmatory research aims to evaluate hypotheses and confirm the validity of the assumptions in the research design. Flexible ways to examine data without any preconceptions are the subjects of the exploratory research approach. This approach relies heavily on graphical displays, allows data to suggest questions, focuses on indicators and approximates error magnitudes. The researcher needs to be open-minded in this regard. On the other hand, confirmatory research is based on hypothesis tests and formal confidence interval estimation. The hypotheses determine data collection methods and their emphasis is put on numerical computation ((Payne & Payne, 2004, p. 144). Also, the hypotheses are used to control variables and predict results (Payne & Payne, 2004, p. 145).

According to Creswell (2013, p. 77), both confirmatory and exploratory research can be either quantitative or qualitative. Walliman (2006, p. 44) explained how the phases in the process of research reflected confirmatory or exploratory strategies in both quantitative and qualitative research. These research process phases are as follows:

Phase 1: Research problem and question—while this is a confirmatory process within the quantitative approach, the qualitative approach is an exploratory process that focuses on descriptive statistics.

Phase 2: Data collection—quantitative confirmatory research employs instruments, observation, score-oriented closed-ended process and proposed hypotheses, while qualitative exploratory research can include interviews, observation, an open-ended process and video-oriented approaches.

Phase 3: Data analysis—quantitative confirmatory research relies on measures such as descriptive statistics and inferential statistics, while qualitative exploratory research adheres to procedures such as descriptive statistics (including classifying themes and seeking associations among themes (De Villiers, 2012, p. 250).

Phase 4: Data interpretation—quantitative confirmatory research focuses on

> the interpretation of the theory, while qualitative exploratory research relies on sense-making, asking questions and personal interpretation.

As indicated in the previous section, this study was largely quantitative in nature, and thus requires large amounts of data to be collected so that inferences can be made. This data needed to be collected in an economical way from a sizeable population. This study, therefore, used the confirmatory approach whereby surveys were used for the reasons explained above. Surveys are associated with the deductive approach (Al Zefeiti & Mohamad, 2015, p. 4), and are most frequently used for answering the 'who', 'what', 'where', 'how much' and 'how many' questions (Thornhill et al., 2009, p. 3). In addition, interviews were employed to try to explain the findings from the surveys.

4.2.4 Research design

Creswell (2003, p. 13) continues explaining the strategies of enquiry which should specifically provide a direction for proceeding in research design. It was explained that research can follow a quantitative, qualitative or mixed-method approach (Creswell, 2003, p. 13). Quantitative empirical research is usually associated with the use of administered surveys, laboratory based experiments, quantitative metrics, highly structured protocol simulations and hypothesis testing as its enquiry strategy (Olszewska et al., 2016, p. 260). The use of quantitative empirical studies is well developed in the natural sciences and quantitative researchers conduct the enquiry in an unbiased, objective manner, through the description of trends or an explanation of the relationship of the construct that is regarded as positivistic (De Villiers, 2012, p. 250).

Since quantitative research supports a positivist epistemological perspective, the researcher and research object examined are regarded as independent objects in a sense that the researcher is able to study the occurrences without influencing them or being influenced by the environment.

Qualitative data is usually collected in the form of words or images instead of numbers, as is the case for quantitative research. Payne and Payne (2004, p. 175) believe that qualitative methods were developed for the social sciences. The qualitative methods enabled social science researchers to study human behaviour and belief phenomena as they were believed to help researchers understand people and the social and cultural contexts within which they live. According to Knox (2004, 120), a qualitative strategy of enquiry permits a further definition of the study's nature and restrictions. This is supported by Henning et al. (2004, p. 34) when describing how a socio-technical perspective looks at people and technical features, how they are used, and how they interact. Since qualitative strategy supports an interpretivist epistemological perspective, the researcher and research object examined are interactively linked. This means that the researcher is able to study the occurrences while being influenced by the environment.

Mixed methods research, on the other hand, combines qualitative and quantitative research methods such as the use of qualitative and quantitative views, data collection, inference techniques and analyses for the broad purposes of the depth of understanding and corroboration of the study (Creswell, 2013, p. 79). Kaplan and Duchon (1988, pp. 574-575) are of the view that combining these methodologies can be done without violating fundamental paradigmatic assumptions.

Different viewpoints exist on connection level of the two strategies; at the data level described by Leedy and Ormrod (2005, p. 33) and at different research process phases level (Creswell, 2013, p. 79). Leedy and Ormrod (2005, p. 34) recommended a matrix method for mixing quantitative and qualitative research at the data collection level, where the grouping is centred on two types of decisions which are sequenced and prioritised. When the principal method is quantitative, but the use of a qualitative approach at the beginning is used to improve the effectiveness of the quantitative research methods that were used. The qualitative approach should be used to examine and develop the content of the quantitative questionnaire in order to confirm that the survey covers the important topics suitably. According to Creswell (2013, p. 80), mixing can happen at the following different phases within the research process:

Design stage: quantitative data can assist qualitative components, identifying members of a representative sample and spotting outlying observations (Johnson & Onwuegbuzie, 2004, p. 15). Equally, qualitative data can assist quantitative components with the concept as well as instrument development (Creswell & Clark, 2017, p. 188).

Data collection stage: whereas quantitative data can provide standard information and assist in avoiding bias, qualitative data can help facilitate the assessment of the generalizability of quantitative data and provide a new perspective on the findings.

Data analysis stage: in addition to grounding and modifying the theoretical perspective, qualitative data can fulfil an important role in interpreting, clarifying, describing and validating quantitative results.

Different types of mixed methods research can be represented on the qualitative-quantitative continuum as shown in Figure 4.2. Figure 4.2 illustrates the pure mixed, pure qualitative, and pure quantitative research methods. A mixed-methods researcher generally falls within the centre, representing the strongest or pure form. Mixed methods research that is principally qualitative (QUAL + quan) predominantly relies on a qualitative view of the research process but incorporates some quantitative methods to a lesser extent. Similarly, a principally quantitative (QUAN + qual) relies largely on a quantitative view of the research process, with less qualitative methods and data to add value to research studies.

A mixed-method approach was adopted for this research study. The mixed method is research where data is collected and analysed and conclusions drawn using both quantitative and qualitative approaches (Given, 2008, p. 526). Mixing these approaches can happen at different levels of research depending on the needs of the researcher. Creswell (2003, p. 16) mentions sequential—in which a researcher seeks to elaborate on or expand findings of one method with another in a sequential manner—as one of the three mix method strategies. The mixed method strategy is regarded as being explanatory sequential (Given, 2008, p. 526) when the researcher starts with quantitative data collection and analyses it; this means qualitative data collection and analysis is used to explain the results of the quantitative analysis. The opposite of such a type of mixed method strategy is called exploratory sequential.

An explanatory sequential mixed method design was employed in this research study. The quantitative data that was collected in the first phase was thereafter analysed using quantitative analysis techniques. The second phase involved the collection of qualitative data and subsequent data analysis it to help explain and hence suggest appropriate interventions to the findings of the first quantitative data collection phase.

4.3 Data Collection

The research instrument that was used to collect quantitative data for the first phase of this research study was the questionnaire involving which close-ended questions. The reason for choosing a questionnaire as the research instrument was that questionnaire are versatile in that they are cost effective, easy to manage as they cover small to vast populations, and can be used to gather large data (which is quantitative in this context) (Campbell et al., 2004, p. 146). Questionnaires are an efficient data collection technique, suitable for when knowing exactly what is needed and how the variables concerned are

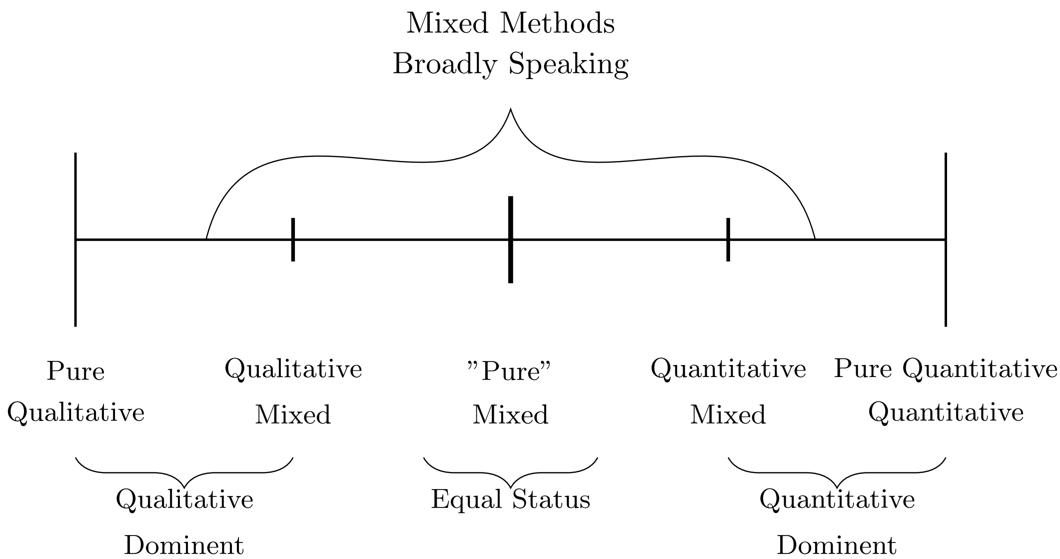


Figure 4.2 – Mixed methods types (Johnson, Onwuegbuzie & Turner, 2007, p. 124).

measured (Nishishiba et al., 2014, p. 100). Questionnaires include groups of questions that can be administered personally in a written format, distributed electronically, or emailed to the respondents. Although questionnaires are quick and easy to administer (Nishishiba et al., 2014, p. 100), careful attention is required to ensure the correct and accurate wording of the questions, the layout of the forms, and the ordering of the questions to ensure a valid outcome (Creswell, 2013, p. 83).

The second phase of the study was comprised of qualitative interviews. The purpose of this qualitative interviewing was to derive interpretations, and evaluate the findings of the first phase. A structured interview is a data collection technique which focuses on finding as much information to previously carefully questions. The communication tools of structured interviews can include telephone interviews, face-to-face interviews, or interviews conducted through a medium such as internet or cell phone (Nishishiba et al., 2014, pp. 100-102). Each means of approaching the interview has its strength and drawbacks in terms of time, clarity, cost, interviewer training and knowledge of computers.

In this research study, interviews were conducted with invited participants using a telephone or Skype, depending on the preferences of the participant.

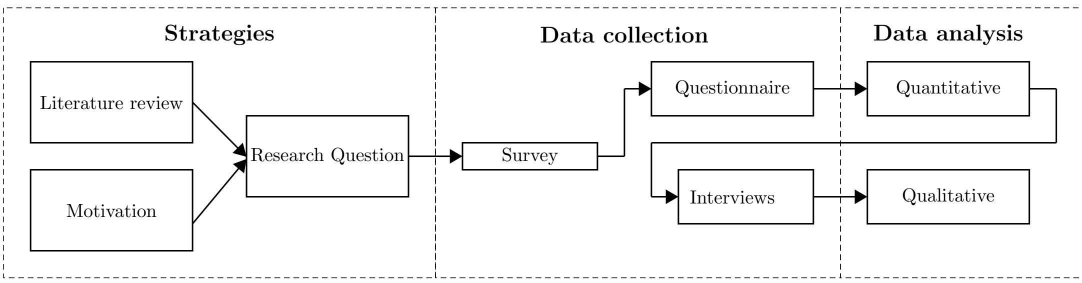


Figure 4.3 – Summary of the Research Design.

Reasons for utilizing these methods over the traditional face-to-face was as follows:

- The currently available technologies allow this form of interviewing to be conducted and thus eliminating the need for commuting between the locations of the participant and the interviewer.
- Time, being a limited resource in this research study, this method allowed for time to be managed effectively and efficiently by eliminating the time to travel to the location of the interviewee.
- Since no financial provisions in the form of a sponsor were made for this research study, the travelling costs were eliminated without necessarily compromising the quality of the research study.
- This study was conducted under the auspices of distance education university, which promotes and emphasises the use of technology to bridge the distance between the university and its students. Therefore, the researcher adopted this type of interviewing method in accordance with the university's stance on the use of technology in distance education.
- The nature of participant work. DevOps practitioners rely on the use of technology on their day to day work. Therefore using technology (like Skype) for interviewing purpose should not intimidate them.

McIntosh and Morse (2015, P. 7) indicated that telephone interviews have specific advantages cases where interviews need to be conducted with a geographically remote person. It has also been indicated that this form of interviewing is economical in terms of time and costs and thus make it much more efficient McIntosh and Morse (2015, P. 7).

Figure 4.3 shows a summary of the research design that this study has followed. Creswell (2003, pp. 21-22) suggests that the quantitative approach is appropriate when the research problem is geared towards identifying factors that influence an outcome or understanding the best predictors of outcomes. A qualitative approach is appropriate when there is a need to understand a concept or phenomenon. Mixed methods capture the best of both quantitative and qualitative approaches. Since this study involved the identification of success factors for collaboration in a DevOps environment, the mixed-methods approach was deemed appropriate and was therefore used for answering the research question of this research study.

Although this research is mostly quantitative, it was evident that it would benefit from a combination of approaches, as suggested by Creswell (2013, p. 80). To discover patterns and develop theories to gain a better understanding of the subject under investigation, this study employed qualitative methods (De Villiers, 2012, 251). Therefore, this research study was aimed at generating knowledge of human action in context through the use of qualitative data (Johnson & Onwuegbuzie, 2004, p. 18). The study predominantly used online questionnaires to quantitatively test the main success factors in DevOps collaboration culture acceptance. Conducting the research in various provinces in South Africa provided a more adequate understanding of the cultural and social context of the software development community. This was followed by interviews as an explanatory mechanism to the developed framework.

This section presents the data collection processes that were followed in this research study. In addition, it provides a brief discussion on the research settings, sampling and sampling size, and the participants of this research study. A discussion on how the pilot study was conducted, followed by the main study, is included. Section 4.3.6.1 below provides comprehensive information about the construction of this questionnaire. This is followed by the interviews discussion in Section 4.3.6.2.

4.3.1 Research settings

This study took place in South Africa. Creswell (2013, p. 83) views the target population as the whole group under study. In the context of this study, the target group were individuals who are practising DevOps philosophy in their different organisations. This study did not look at specific organisations or sectors *per se*. The participants included DevOps practitioners—Developers and Operators—with different responsibilities and roles.

4.3.2 Sampling

The target population for this study was all Developers and Operators who were directly involved with the actual development of the product within DevOps settings. To include all Developers and Operators in this population would be imposable as this number is too large to study at this level. Creswell (2013, p. 84) defines sampling as choosing participants from the study population. The purpose is to extract knowledge from the selected population representatives. Using this representative sample, the researcher is able to explain and describe the nature of the population.

Identifying these representative samples is a science on its own. There exist several techniques for selecting a representative sample from the entire population. Nishishiba et al. (2014, p. 78) categorise these techniques into probability and non-probability sampling. When a subject in the population has a chance of being selected as a candidate, it is referred to as probability sampling, which is also known as random sampling (Nishishiba et al., 2014, p. 79). The advantage of probability sampling is that it makes it possible to determine the extent to which the sample varies from the population (Payne & Payne, 2004, p. 210). In non-probability sampling, the representative candidate is selected based on personal or convenience judgement (Creswell, 2013, p. 84), sometimes referred to as deliberate sampling. Theoretical sampling deliberately selects participants according to the theoretical needs of the study (Lewis-Beck et al., 2004, p. 994).

In this research study, a form of non-probability sampling, which is also known as convenience or opportunity sampling, was used. This sampling method was used because participants of this research study were recruited from social media. Such an approach is regarded as opportunistic because only participants that are available on that platform could be reached. Budget constraints played a major role in the researcher opting for this technique.

This convenience sampling was followed with a purposive sampling criterion in which the participants were selected for interviews. Purposeful sampling is a non-probability sampling technique used in qualitative research to select participants that are knowledgeable about or experienced with a phenomenon of interest Creswell and Plano, 2011 as cited by Palinkas et al. (2015, p. 534). More experienced participants were selected to participate in the interviews.

4.3.3 Sample size

It is important to consider the sample size of this and any other research. As one has indicated, constrains such as budget and time, may limit the inclusion of the entire population in a study, and the question of the sample size needs to be borne in mind. Sample size depends on aspects mentioned by Nishishiba et al. (2014, p. 77) such as: nature and size of the population under study, resources, budget, time available, the required accuracy of the study, and the significance of the results. In addition, Creswell (2013, p. 86) advises that the sample size should be based on the type of research being carried out, although practical limitations may have an effect.

In this research, the sample of 540 respondents was regarded as being of a suitable size based on guidelines and standards set out by Payne and Payne (2004, pp. 200-203). According to Payne and Payne, the sample size should consider analysis methods that will be used, and the questionnaire size. Based on these and the level of confidence and the amount of error that can be tolerated, the sample size was estimated.

4.3.4 Participants

It is important that respondents of a study are able to provide the information required from them. To this end, respondents should possess specific skills such as cognitive skills (Rubin & Babbie, 2016, p. 55). This suggests that an understanding of written and spoken language and reacting accordingly is important.

In this research study, the research material was presented in English language, a language that is understandable to all the participants. In addition, participants were IT practitioners who were deemed to be capable of providing insight into the critical success factors of DevOps collaboration culture. It was believed that DevOps practitioners with at least one year experience in DevOps roles should be able to provide the required information for this research study. The participants were selected from various provinces in South Africa. Consequently, diverse races and cultures were included in the research study.

This study relied on primary data, that is, data collected and linked together for a particular research study to provide meaningful information (Creswell & Clark, 2017, p. 57). Such type of data can be collected through interviews, observations, and self-administered instruments (De Villiers, 2012, p. 240). In this research study, primary data were collected through the use of online questionnaires and interviews with the target population.

4.3.5 Pilot study

It is important to carry out a pilot study in order to remove or fix problems that could arise in the main study. Therefore, a pilot study can assist in resolving issues that could have cost more to resolve in the main study. In addition, a pilot study plays the role of checking the suitability of the research methods and research design, and thus enables the researcher to make changes, if required, to the questionnaire. A pilot study was carried out with 25 DevOps experts in the City of Tshwane area of South Africa. The purpose of the pilot study was to collect data to reduce risk and uncertainty. Results generated from the pilot study did not necessitate any changes to and a subsequent redesign of the questions.

The interview schedule was also piloted using two participants, and the necessary adjustments where made. This pilot was conducted to make sure of the following:

- All the questions that are necessary were included.
- The question elicited the type of expected responses.
- That there are no problems with the line of questioning (e.g. posing ambiguous questions that have different meaning to different people).

4.3.6 Main study

For the main research study, a total of 540 requests to complete the questionnaire were sent to potential respondents and only 312 were satisfactorily completed. The questionnaire consisted of two parts, A and B. In Part A, the respondents were asked to complete the demographic questions aimed at obtaining information about the moderating effects in the proposed framework. In Parts B, attitude questions based on DevOps collaboration were posed.

4.3.6.1 Quantitative questionnaire

The questionnaire consisted of a set of self-completion questions (See Appendix D for the whole list of questions.) which were administered and expected to be completed by the respondents online via email. Google forms

Table 4.2 – Summary of number assessment statements.

Methodology Adoption Construct	
Success factors	**Items**
Open Communication	6 items
Roles and responsibilities alignment	6 items
Respect	6 items
Trust	6 items
Subjective Norm	6 items
Organisational Usefulness	6 items
Perceived Behavioural Control-Internal	6 items
Intention to follow DevOps culture	8 items

were used to create and distribute the questionnaire to participants. The questionnaire was divided into eight sections with the first section focussing on bibliographic questions. Four variables that were in the first section of the questionnaire are gender, age, role, and experience. The remainder of the sections required the participants to capture using a five-point Likert scale their perceptions about DevOps collaboration factors.

The scale was a modification of a scale used in the study by Hardgrave and Johnson (2003). The participants indicated their level of agreement or disagreement with statements of the questionnaire. The survey instrument consisted of 54 statements.

These statements were designed to test DevOps collaboration culture using the following four constructs:

- Open Communication (OC);
- Roles Responsibility (RR);
- Respect (RE); and

- Trust (TR).

A detailed discussion on how these DevOps culture constructs were measured was presented in Section 3.6.3

The other five factors for DevOps collaboration culture acceptance were adopted from the model by Hardgrave et al. (2003), and the following constructs were used:

- Social Norm (SN);
- Organisational Usefulness (OU);
- Perceived Behavioural Control-Internal (PBC-I); and
- Intentions to use/follow DevOps culture (INT).

The number of items that were used to test each construct in the questionnaire are summarised in Table 4.2.

The low response rate in respect of using questionnaire as a data collection instrument remains a challenge. To maximise the response rate of the survey, the following precautionary measures were taken:

- Google forms that are available in most platforms were used to collect data.
- A reminder to complete the survey was issued two weeks after the first email was sent.
- Participants were made aware of the time required for the completion of the survey (15–20 minutes).

Table 4.3 – Construction of the interview questions.

Success factors	Scheduled question
Open Communication	How do you stay updated about DevOps projects you are working on with your team members?
Roles and responsibilities alignment	How important is the clear role responsibilities between DevOps developers and operators for the success of the project?
Respect	How your team members encourage you to contribute to the team?
Trust	How important is the trust between DevOps developers and operators for the success of the project?
Subjective Norm	How important is the influence of DevOps team members in the acceptance of DevOps?
Organisational Usefulness	How does the collaboration between Devs and Ops affect the success of the organisation?
Perceived Behavioural Control-Internal	How do one's skills and abilities affect the success of DevOps projects?

4.3.6.2 Qualitative interviews

The aim of this qualitative interview is to use DevOps experts to evaluate the developed framework. The study followed a hermeneutic interview approach, which is a form of phenomenological interviewing approach that seeks to examine the interpretative meaning aspects of the experiences of DevOps experts. Phenomenological interview approach generates detailed and in-depth descriptions of human experiences (Roulston, 2010, p. 16).

Semi-structured interviews were conducted in which participants were asked open questions and given the freedom to use their own words to formulate their answers. The construction of the questions for the interview is illustrated in Table 4.3.

4.4 Data analysis

This data analysis was conducted to identify the factors that are critical in the successful adoption of DevOps collaboration culture. The data analysis was performed to transform raw data that was collected into meaningful insights that would add value to the knowledge base. Conclusions were reached regarding the challenges and opportunities that emerged during the analysis process. Quantitative data analysis was conducted first and it was followed by qualitative data analysis. A brief discussion of the processes followed is presented in the following sub-sections.

4.4.1 Quantitative data analysis

The statistical package that was used to analyse the quantitative data was the IBM SPSS ver. 25 software. The raw data was read into SPSS and a database with variables names and type was created. To ensure correct and meaningful data, the raw data was subjected to verification and cleaning treatment prior to application of the statistical analysis method.

Once the data was cleaned and verified, the reliability and validity tests were performed on the instrument. Cronbach's Alpha was computed to test for reliability of the research instrument items. Cronbach's Alpha is used to assess internal constancy of the proposed decision variables (Huizingh, 2007, p. 243). Exploratory Factor Analysis was also computed as a measure of construct validity.

Further analysis was conducted using descriptive and inferential statistics. Descriptive statistics to calculate frequencies, means, and standard deviations were computed on items in the questionnaire. Inferential statistics were then conducted on the data to measure correlations between variables. In addition, a regression analysis was conducted to test the validity of hypotheses.

4.4.2 Qualitative data analysis

A deductive approach in which transcribed data was grouped using constructs from the previous hypothesis was followed during the analysis of the qualitative data. Content analysis was employed to analyse the data. Payne and Payne (2004, p. 51) described content analysis as a method that seeks to demonstrate the meaning of written or visual sources by systematically classifying their content into predefined categories and then quantifying and interpreting the sources. A description of how the content analysis was conducted is presented in Section 6.2.1.

4.5 Research credibility

Measures that were taken to show the credibility or the trustworthiness of this research are discussed in the subsections below. This credibility was measured in terms of validity and reliability. Since this was a mixed-method study, reliability and validity were measured differently for qualitative and quantitative studies. Therefore, these terms are discussed separately as follows: Section 4.5.1 is focussed on validity and reliability of the quantitative study and Sections 4.5.2 and 4.5.3 discuss the respective reliability and validity of the qualitative study.

4.5.1 Validity and Reliability of the quantitative study

Validity and reliability are two factors a researcher should be concerned with when conducting a research study (Payne & Payne, 2004, p. 196). Reliability is the extent to which research measurements can be repeated by numerous researchers, on different occasions and under different conditions (Drost et al., 2011, p. 106). Put differently, reliability is the consistency of measurement. Various methods for testing reliability in behavioural research exist. Drost et al. (2011, pp. 108-112) provides a brief discussion on the following reliability testing techniques: test-retest, alternative forms, split–halves, inter–rater reliability and internal consistency. Internal consistency uses Cronbach's Alpha to measure the consistency of questions in a questionnaire (Drost et al., 2011, p. 112). The internal consistency of the questionnaire that was used in the study was measured using Cronbach's Alpha.

Validity, on the other hand, is concerned with the meaningfulness of the research components. It ensures that the researcher measures what the researcher intends to measure. Drost et al. (2011, pp.114-121) summarises methods of measuring validity as: statistical conclusion validity, internal validity, construct validity, and external validity. Construct validity was the method used to test for validity in the quantitative study. To ensure that the constructs are valid, the following steps were followed in this research study:

Step 1: A conceptual framework (Figure 3.5) was constructed from the analysis of literature. The questionnaire design followed from the framework constructs.

Step 2: The study promoter evaluated the framework. In addition, the framework and its constructs were peer reviewed, presented and published in the conference proceedings.

Step 3: The construction of the data collection instrument was completed in collaboration with the statistician.

The questionnaire resulting from these steps was evaluated using face validity. Face validity as defined by Drost et al. (2011, p. 116), is "the subjective judgement on the operationalisation of a construct''.

The questionnaire used in this research study was, therefore, tested for two factors; validity was confirmed using EFA, whilst reliability was assessed using Cronbach's Alpha. Results emanating from this test are discussed in Chapter 5.

4.5.2 Reliability of the qualitative study

Reliability and validity are important in all research, including qualitative research. However, they address issues relating to the quality of data and appropriateness of the methods used. Reliability in qualitative studies attempts to ensure consistency, dependability and replicability of the qualitative research (Zohrabi, 2013, p. 259). In addition, Zohrabi (2013, p. 259) has added that obtaining similar results in qualitative studies is straightforward because it relies on numeric data. This is the opposite when dealing with qualitative data as the interpretations may differ from one researcher to another. As the result, Lincoln and Guba (1985, p. 288) advise that, instead of focusing on obtaining the similar results, the focus needs to be directed towards the dependability and consistency of the data. In this way, the reliability of the study was based on the credibility of the data collection process that leads to consistent and dependable results.

4.5.2.1 Dependability

Dependability is concerned with ensuring that research results are reproducible under similar conditions. Although it might seem impossible to replicate qualitative results, Lincoln and Guba (1985, pp. 316-317) suggest the use of the researcher's position and audit trail to ensure the dependability of the study. The researchers' position requires a provision of an explicit explanation of the various processes used during the enquiry. An audit trail, on the other hand, requires full details on how data is collected, how it is analysed, how the themes emerged and how to arrive at the given results.

To endure the reliability of the qualitative enquiry, this study followed the above suggestions. The researcher provided a comprehensive explanation of the processes followed to conduct the study. In addition, the journey on how the collection of data, the analysis of data, the formation of themes and interpretation of the results was explicitly stated.

4.5.3 Validity of the qualitative study

Validity in this context is based on the trustworthiness, utility and dependability that is placed on the study (Zohrabi, 2013, p. 258). In other words, validity is dependent on the researcher and the participants. Validity is concerned with whether the research is believable, true, and whether it measures what it is supposed to measure. Several validity factors, which were mentioned by Miles and Huberman (1994, pp. 278-279), are discussed in the following sub-sections.

4.5.3.1 Credibility

Credibility is concerned with the truth of the research findings, as explained by Lincoln and Guba (1985, p. 45). According to Miles and Huberman (1994, p. 278), to produce convincing findings, context-rich and meaningful descriptions should be provided. The five strategies for improving credibility are: prolonged engagement, persistent observation, triangulation, peer debriefing, negative case analysis, referential adequacy and member-checking Lincoln and Guba (1985, p. 45).

4.5.3.2 Confirmability

Confirmability is the degree to which the research findings can be confirmed or corroborated by others. To this end, confirmability ensures that the research findings are bias-free. Confirmability also refers to the extent to which a researcher is aware of, or accounts for, individual subjectivity or bias. To help minimise this bias, Miles and Huberman (1994, p. 67) recommended the preparation of a comprehensive precise description of data collection and analysis methods. On the other hand, Lincoln and Guba (1985, p. 48) have suggested the use of triangulation of methods and data sources, and practice reflexivity to carry out a conformability audit trail.

4.5.3.3 Transferability

Research findings are transferable if they could be applied to a similar situation. Transferability is analogous to external validity in that it demonstrates the extent to which findings can be applied to other contexts, or are generalised Miles and Huberman (1994, p. 70). To ensure transferability, Lincoln and Guba (1985, p. 51) have advised that researchers should give detailed research methodology that describes the data collection, analysis and sampling methods used.

As a result, this research study has adopted the following strategies which were found to be applicable to this study:

Triangulation: In this study, different methods (i.e. web surveys and interviews) were employed to collect data. Methodological triangulation was also used in the sense that the use of the mixed-method approach resulted in different data collection and data analysis techniques.

Peer debriefing: The study was conducted under the supervision of an experienced promoter who guided the research process. Expert blind peer reviews were conducted on published conference proceedings and presentations.

Vivid and broad descriptions: Information about the study is discussed in detail and the required information is disclosed to the readers.

Audit trail: All the processes of this study are documented, and the results used to generate reports are attached in the appendices.

With all that said, it can be concluded that measures were taken to increase the credibility of this study to an acceptable level. Although the credibility of this study was kept in mind, there are some limitations that need to be discussed in order for the reader to be clear on how they were dealt with.

4.6 Limitations

Qualitative data collection in the form of interviews can be a time-consuming process as it requires that an appointment with potential participants be made first. It is often not easy to acquire a time slot that suits both the researcher and the research participant, and this often leads to the number of interviewed participants to be low, as experienced in this research study. The second qualitative limitation of this research study is the location of the participants. Since this was a self-funded research study, the location of potential participants proved to be a constraint, and telephone and/or Skype interviews were the only option that was available to the researcher.

4.7 Ethical Issues

When conducting research, ethical and data protection issues should be taken into account. Ethical guidelines seek to work towards protecting the individuals, communities and environments involved in the studies against any form of harm, manipulation or malpractice.

This study was conducted under the auspices of the University of South Africa (UNISA), and its policies on ethics were therefore followed. Since this study involved human participants, UNISA's policy on the involvement of human participants was used as a guide. The policy is aimed at protecting human participants by using acceptable ethical standards in research involving human participants. An Ethical Clearance Certificate (see Appendix A) allowing the researcher to conduct this research study was obtained. The Certificate was issued by the UNISA's Research Ethics Committee.

The following ethical issues were considered and dealt with accordingly:

Informed consent This study relied on web questionnaires and telephone interviews. The participates were recruited on their individual basis and it was not necessary to request permission from their respective employers.

A cover letter (See Appendix B) to the anonymous web-based survey was included in the survey to allow participants to make a decision on whether they would like to continue with the survey or withdraw without facing negative consequences. By continuing with the survey, the participants were deemed to have given their consent.

With regard to telephone interviews, a consent form (See Appendix C) was read to the participant and the participants were asked if they would like to continue. If they responded with a 'yes', interviewing continued as planned otherwise the call was terminated after the explanation that this will not affect them negatively.

Harm and risk The nature of this study did not place participants nor the researchers in any position that could bring them harm or risks. The research was based on the opinions of participants relating to DevOps collaboration culture, and none of the participants was put in danger of any sort.

Honest and trust When conducting this research study, the researcher abided by the UNISA ethical guidelines to ensure the credibility of this study. Ethical guidelines for collecting and analysing research data were followed in an honest and trustworthy manner.

Privacy, confidentiality and anonymity In this study, the researcher ensured that the confidentiality and anonymity of all participants. To enforce this principle, no information that would identify the participant directly or indirectly was collected. The participants were made aware of their right to privacy, confidentiality and anonymity before their participation was recorded. It was also made clear that data collected from their participation will be used solely for the purpose of this study and their identity will not be revealed in publications resulting from this study.

Voluntary participation After assuring the participants on the precautions taken to protect their identity and their vulnerability, it was made clear to the participants that this research was only meant for the fulfilment of an academic obligation and that their participation was voluntary. The participants were informed of their right to withdraw their participation at any point of the research study.

Ethical clearance certificate

The appendix A shows the ethical clearance certificate that was issued for this study.

4.8 Chapter Summary

In this chapter, research processes were discussed using the research onion as a research process template. During this discussion, research concepts such as the research philosophy approach to research, the research strategies and designs were explained.

This study followed pragmatism as a research philosophy of choice since this research philosophy gives the researcher the freedom to combine different methodologies. This means that the research may use both qualitative and quantitative methodologies (mixed-methods study). Although this was a mixed-method study, it leaned mostly towards a qualitative study. The research study is regarded as a confirmatory study in that qualitative enquiry is conducted to explain and confirm the results of the quantitative enquiry. In addition, this study adopted a deductive approach in which more general theories were used to devise specific DevOps collaboration knowledge.

Data collection and analysis techniques used were explained in detail. The study relied on surveys as the data collection tool. During the surveys, a questionnaire and interviews were used to collect quantitative and qualitative data, respectively. Statistical techniques were used to analyse the quantitative data, whilst qualitative data was analysed using qualitative content analysis. Lastly, the processes used to ensure the credibility of this study and the ethical principles followed to protect this study and its participants were discussed.

The analysis and the findings of this study are outlined in the subsequent chapters. The results of the quantitative and qualitative studies are presented in Chapters 5 and 6, respectively.

Chapter 5

Quantitative Data Analysis and Interpretation of Findings

5.1 Introduction

This thesis followed a mixed method research design approach in which data was collected in two phases. Since this was an exploratory mixed methods study, quantitative data was collected and analysed first, which was thereafter followed by the collection and analyses of qualitative data. The first phase of data collection was conducted using a questionnaire as an instrument. This was followed by research interviews, which yielded qualitative data.

This chapter will discuss the statistical analysis of the results and the interpretations of the research findings from the quantitative data to answer the research questions. A Statistical Package for Social Sciences (SPSS) v25.0 was used to conduct the statistical analysis.

The chapter starts by discussing how raw data were treated cleaning during screening. This is followed by an analysis of the demographics of participants using descriptive statistics (see Section 5.3). To test for validity, Exploratory Factor Analysis (EFA) was employed. The Cronbach Alpha and a composite reliability test of the constructs were evaluated in Section 5.5. The constructs correlation was measured to determine if the relationships between constructs were significant. In Section 5.7, regression analysis was conducted to test the hypotheses. Lastly, a modified framework for acceptance of DevOps collaboration culture was presented. This framework showed the key elements of the DevOps collaboration culture that DevOps practitioners believe are important in such an environment.

Moderator variables were also investigated to determine their effect on the relationships associated with DevOps collaboration culture. For this investigation, the job role variable was analysed in detail, and the outcomes discussed accordingly.

5.2 Data screening

It is important to conduct data screening to identify missing data, the accuracy of data entry, normality and miscoded data. To avoid problems such as normality and linearity, missing values and outer influence are highlighted, as they could improve the R^2 values. Data screening was carried out to clean the data so that the statistical analysis procedures are precise and to ensure that estimates have a sound basis. During this screening process, the following cases were considered:

- Outliers: Frequency tables and Mahalanobis distance—a measure of the geometric distance between a given points on the graph—were computed and studied for univariate and multivariate outliers. Residual analysis was used to detect outliers with values above 3.5. No such outliers were found.

- Missing values: Frequency tables were used to detect missing values. In this case, the middle point was used to replace the missing value.
- Normality: Data was tested for a normal distribution before parametric tests were conducted. Data skewness—which measures the symmetry—and kurtosis—which measures whether data are peaked or flat—were within an acceptable value that is between -2 and $+2$.

During the filling of the questionnaire or the capturing of data on the system, the appropriateness of the data was taken into consideration. The data sets were examined for cases of univariate and multivariate outliers and where they were found biased results were removed. This was done to make meaningful decisions on the occurrence of such outliers. The normality of the data was also examined to understand what to do with cases of non-normality.

5.2.1 Sample data

A linear model must be carefully tested using diagnostic plots to confirm the validity of the assumptions of multiple regression and residual analysis (Huizingh, 2007, p. 300). The following assumptions are recommended by Huizingh (2007, p. 299) and Parker et al. (2015, p. 150) for a meaningful statistical analysis:

- analysing for linear functional form;
- having a proper representative sample;

- determination of fixed independent factors or variables and observations of the framework to confirm that there are no omitted factors;
- existence of equality of variance of the errors that provide homogeneity

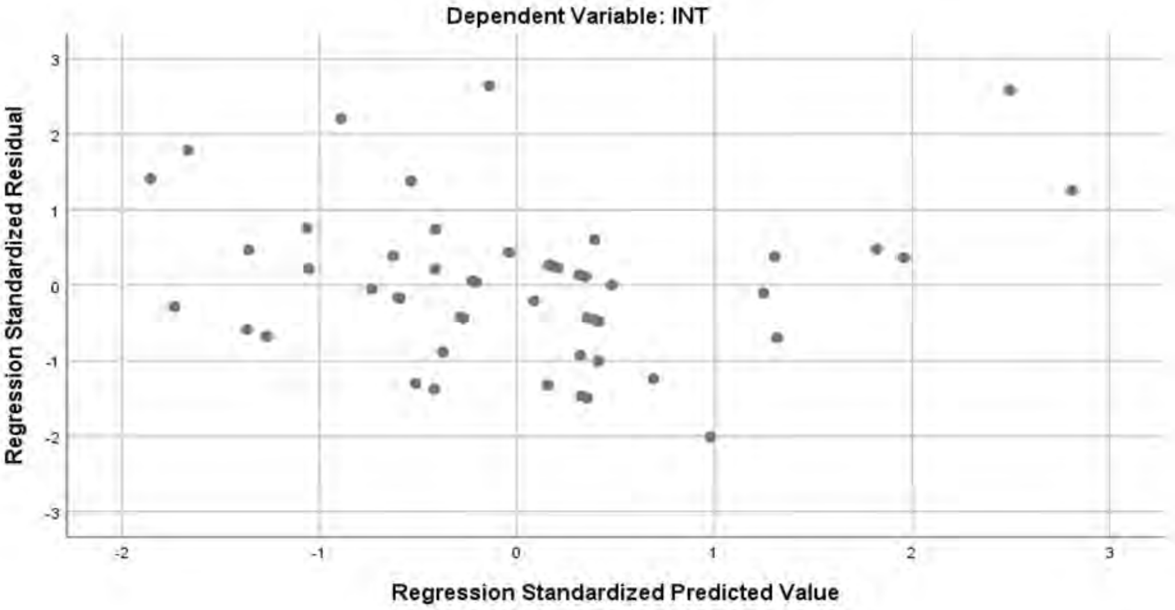


Figure 5.1 – Scatter plot diagnostic test.

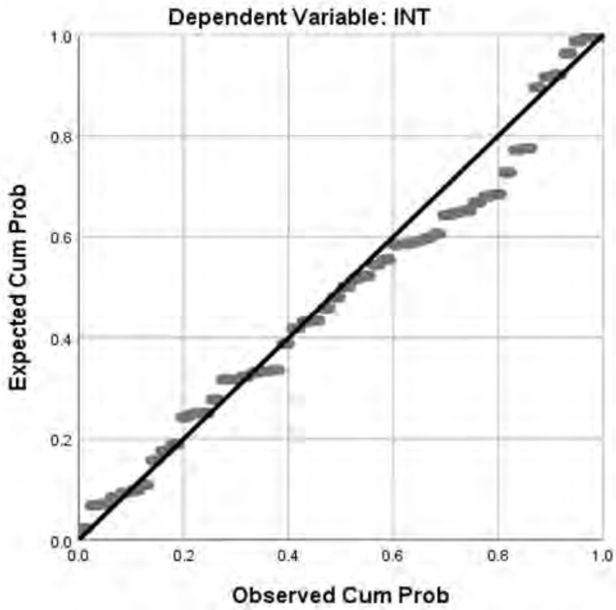


Figure 5.2 – A normal P-P Plot of regression standardised residual.

of the residual variance;

- normality of the residuals or errors, that is, checking whether data do not have multicollinearity and homoscedasticity;
- removing high correlation of the errors; and
- noticing and removing outliers.

Correlation matrix is associated with singularity and multicollinearity, which occur when decision variables are auto-correlated with values of 0.9 and above (Huizingh, 2007, p. 309). Multivariate and bivariate correlations were analysed and no bivariate correlations of 0.9 and above were found between the independent variables. Coefficients output and residual analysis were used to evaluate multivariate correlation. All tolerance readings were found to be above 0.3 and the variance inflation factors were below 3. Parameters were used to detect the outlier by comparison of values of the residuals in SPSS ($error = predicted - actual$) as being out of range of 3.5 and -3.5 of standardised residuals (Huizingh, 2007, p. 311).

To establish which errors were out of range, regression was run in SPSS v25.0 to determine the residual versus the fitted plot. Figure fig:Scatter-plot-diagnostic shows the resulting diagnostic test scatter plot. To detect the univariate and multivariate outliers, a Mahalanobis distance measure of the geometric distance between a given point on the graph and the centroid was used, using recommended assumptions made by Parker et al. (2015, p. 153). These outliers—multivariate and univariate—were examined using the residual analysis and no univariate and multivariate outlier cases with a residual out of the range 3.5 were found. It was therefore concluded that the multivariate outliers were random and that there was little danger in retaining these factors. The residuals indicate linearity, homoscedasticity and independency since the scores appear to have been organised between two parallel lines, as shown in Figure fig:A-normal-P-P. Multicollinearity is the high correlation among the explanatory decision variables that prevent their effects from being analysed, and was tested. As suggested by Parker et al. (2015, p. 153), the presence of multicollinearity makes it problematic to evaluate the influence of unidentified parameters giving significant errors for minor changes in data. Therefore, multicollinearity will lead to high significant standard errors and a high correlation coefficient that generates an R^2 value that is close to 1 or -1 (Huizingh, 2007, p. 301). This was evaluated by verifying whether the R^2value is near ± 1. The modal summary of SPSS produced an R^2 value of 0.529. When the R^2 value was computed for tolerance ($tolerance = 1 - R^2 = 0.471$), based on the criterion that the deletion would not improve the regression R^2, the resulting values showed non-existence of multicollinearity.

In addition, the descriptive analysis showed that the data kurtosis and skewness were within the acceptable value of ± 1, which meant that no data transformation was necessary. Furthermore, the expected normal probability plot was employed to assess multivariate normality. Based on the scores that are almost on the straight line (see Figure fig:A-normal-P-P), the normal P–P plot of the regression standardised residual was considered normal. The plotted scores appear to be close to the diagonal line, and scattering is, for this reason, regarded being normally distributed (Parker et al., 2015, p. 165). Thus, the null hypothesis, which declares that errors follow a normal distribution, is accepted. To this end, the assumptions for multiple regression and residual analysis are supported.

5.3 Respondents demographics

The survey was administered to 520 DevOps practitioners around South Africa. A total of 312 of questionnaires were returned by participants and were found to be suitable for analysis. Details about the characteristics of these respondents, in terms of their age, gender, job role and experience, are summarised in Table 5.1.

In the case of gender, most respondents were male with a representation of 82.7%; female respondents comprised 17.3% of the respondents. Respondents were also categorised according to the following age groups:

- Between 18 and 25;
- Between 26 and 36;
- Between 36 and 45; and
- 46 and above.

At 61.5%, the majority of the respondents were in the age group category 26 to 35. This was followed by the age categories of 36 to 45 (30.8%) and 18 to 25 (5.8%). The lowest number of respondents was recorded for the 46 and above age category.

The respondents were categorised further in terms of their job roles. That is, whether a respondent performs a developer, operator or both roles. The data gathered indicate that the majority of the participants (50.0%) were performing both developer and operator roles. This is not surprising as DevOps practitioners are generally encouraged to be able to carry out both roles. At 28.8% (operators) and 21.2% (developers), operators and developers were the second and third highest number of respondents, respectively.

The experience of the respondents was categorised as follows:

- Less than 1 year;
- Between 1 and 2 years;
- Between 3 and 5 years; and
- Above 5 years.

The majority of respondents (40.4%) had between three and five years of experience. Interestingly, 25.0% of the respondents have either one to two or over five years of work experience. Very few respondents (9.6%) had been working for less than one year.

5.4 Validity of the study

This section discusses the validity of the quantitative part of this study. Since this quantitative study relied solely on the use of a questionnaire, only the validity construct was tested using Exploratory Factor Analysis (EFA). The following criteria were used for deciding on the number of factors:

- A cumulative percentage explained by factors $> 60\%$,

Table 5.1 – Respondents characteristics (N=312).

Characteristic		Percentage	Frequency
Gender			
	Male	82.7	258
	Female	17.3	54
Age group			
	[18 and 25]	5.8	18
	[26 and 35]	61.5	192
	[36 and 45]	30.8	96
	[46 and above]	1.9	6
Job role			
	Developer	21.2	66
	Operator	28.8	90
	Both (Dev and Ops)	50.0	156
Experience			
	less than 1 yr	9.6	30
	1–2 yrs	25.0	78
	3–5 yrs	40.4	126
	above 5 yrs	25.0	78

- Eugen values > 1, and
- The significant decline in the Scree plot.

The Table 5.2 shows that a 82.8% cumulative variance, which is explained by eight (8) factors. These factors (the first eight on the table) have Eigenvalues larger than 1.

5.4.1 The Kaiser-Meyer-Olkin (KMO) value and Bartlett's test of sphericity

Table 5.3 shows the results of Bartlett's test of sphericity and the Kaiser-Meyer-Olkin (KMO) value. According to Pett et al. (2003, p. 231), the KMO

Table 5.2 – The cumulative variance explained for by the factors.

Component	Initial Eigenvalues			Extraction Sums of Squared Loadings			Rotation Sums of Squared Loadings		
	Total	% of Variance	Cumulative %	Total	% of Variance	Cumulative %	Total	% of Variance	Cumulative %
1	6.949	30.214	30.214	6.949	30.214	30.214	4.963	21.576	21.576
2	2.509	10.907	41.121	2.509	10.907	41.121	2.509	10.907	32.483
3	2.276	9.896	51.016	2.276	9.896	51.016	2.321	10.090	42.573
4	1.940	8.437	59.453	1.940	8.437	59.453	2.239	9.734	52.307
5	1.720	7.480	66.933	1.720	7.480	66.933	1.954	8.496	60.803
6	1.458	6.339	73.273	1.458	6.339	73.273	1.877	8.159	68.962
7	1.142	4.967	78.240	1.142	4.967	78.240	1.611	7.006	75.968
8	1.042	4.529	82.769	1.042	4.529	82.769	1.564	6.800	82.769
9	0.644	2.800	85.568						
10	0.577	2.508	88.076						
11	0.495	2.150	90.226						
12	0.409	1.779	92.005						
13	0.333	1.447	93.452						
14	0.300	1.306	94.758						
15	0.267	1.159	95.918						
16	0.243	1.055	96.972						
17	0.193	0.839	97.811						
18	0.142	0.617	98.429						
19	0.123	0.535	98.964						
20	0.111	0.480	99.444						
21	0.059	0.257	99.701						
22	0.044	0.189	99.891						
23	0.025	0.109	100.000						

Extraction Method: Principal Component Analysis.

Table 5.3 – Kaiser-Meyer-Olkin (KMO) and Bartlett's test.

KMO and Bartlett's Test		
Kaiser-Meyer-Olkin Measure of Sampling Adequacy.		0.572
Bartlett's Test of Sphericity	Approx. Chi-Square	1731.592
	df	78
	Sig.	0.000

Source: SPSS 25.0

value of 0.572 represents the degree of common variance among the eight variables, which is significant to conduct a factor analyses ($KMO > 0.5$). The p-value of Bartlett's test is below 0.01 ($p = 0.000$) and is significant at the 99% level of confidence. The p–value of Bartlett's test is, therefore, suitable for factor analysis thus suggesting that the correlation structure is significantly strong enough for factor analysis of items.

5.4.2 Communalities

The communalities of the items in the questionnaire are shown in Table 5.4. According to Pett et al. (2003, p. 148), communalities is the extent to which a single item correlates with the rest of the items in the construct. An item with the communalities value close to 1 signifies that the item correlates highly with the other items, whilst items with low communalities, i.e. those near 0.3, are eliminated as suggested by Pett et al. (2003, p. 148).

Using the Principal Component Analysis (PCA) as the extraction method, the communalities of all twenty three (23) items (see Table 5.4) are acceptable.

Table 5.4 – Communalities extraction.

	Communalities			Communalities	
Questions	**Initial**	**Extraction**	**Questions**	**Initial**	**Extraction**
Q5	1.000	0.747	Q37	1.000	0.858
Q6	1.000	0.798	Q41	1.000	0.721
Q11	1.000	0.857	Q42	1.000	0.797
Q12	1.000	0.853	Q43	1.000	0.861
Q16	1.000	0.865	Q44	1.000	0.824
Q21	1.000	0.897	Q46	1.000	0.830
Q22	1.000	0.892	Q47	1.000	0.722
Q23	1.000	0.869	Q48	1.000	0.808
Q24	1.000	0.738	Q50	1.000	0.937
Q29	1.000	0.775	Q53	1.000	0.946
Q30	1.000	0.848	Q54	1.000	0.716
Q36	1.000	0.877			

Extraction method: Principal Component Analysis (PCA)

The questionnaire was analysed using the EFA that is based on the PCA extraction method to inspect differences among the constructs. The reason for using PCA was to reduce the factors to a small set of composite variables. In addition, EFA was used to identify hidden dimensions or constructs, which may not be obvious from the direct analysis. Since the questionnaire conceptualisation was based on the literature review, four categories that were used to group the factors were also based on the literature. It was therefore important, as suggested by Pett et al. (2003, p. 92), to evaluate whether all these factors were loading in their categories and that their Eigenvalues were acceptable for inclusion in the final analysis.

Table 5.5 shows an extraction from the component matrix, with the results from the total variance explained. Each factor loadings on each rotated component that forms the categories of the classification is tabulated in Table 5.5. This resulted in eight factors that have Eigenvalues greater than 1.0.

Figure 5.3 shows the scree plot results after further analysis was conducted.

Table 5.5 – Results of Principal Component Analysis of success factors.

Category	Success factors	Rotated component matrix value	Total variance explained	Eigen values
SN	SN_1 SN_2	0.831 0.846	4.967	1.142
OC	OC_1 OC_2 OC_3	0.861 0.891 0.830	10.907	2.509
RR	RR_1 RR_2	0.898 0.901	7.480	1.720
OU	OU_1 OU_2	0.925 0.651	4.529	1.042
PBC-I	PBC_1 PBC_2	0.863 0.859	6.339	1.458
RE	RE_1 RE_2	0.818 0.843	9.896	2.276
TR	TR_1 TR_2	0.927 0.942	8.437	1.940
INT	INT_1 INT_2 INT_3 INT_4 INT_5 INT_6 INT_7 INT_8	0.783 0.691 0.870 0.785 0.805 0.683 0.842 0.779	30.214	6.949

Although a scree plot does not indicate clearly on which factors should be retained, a rule of thumb of disregarding all the factors possessing Eigenvalues less than 1 is generally adopted. That being said, a decision to extract the eight factors, as depicted in Figure 5.3, was reached.

5.4.3 Principal Factor Analysis (PFA)

The rotation method helps to identify the factors that load in each category (Jupp, 2006, p. 114).According to Jupp (2006, pp. 114-115) a factor is regarded as being loaded in a category if its loading value is greater than 0.3. The loading of an item indicates the extent to which that item contributes to the factor. The category extracted in Table 5.5 (TR) has the highest variance (30.214). This implies that this category has the most variance of the observed factors and therefore correlates well with many of the observed factors. In other words, in the case of this research study, TR plays a major and much more prominent role in DevOps collaboration culture than the rest of the other factors.

On the other hand, the next extracted category showing the highest variance of those factors that were not included in the TR category is OC. The implication of this is that this category correlates with many of those factors that were less correlated with the TR category. In the context of this research

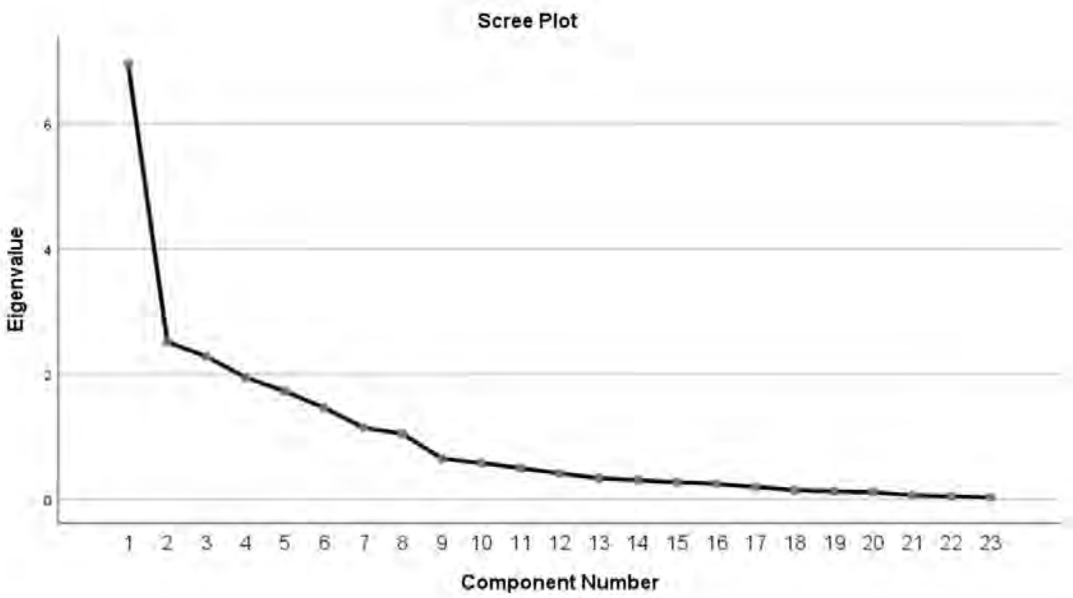


Figure 5.3 – Data sample scree plot.

study, this shows that the OC category factors have a strong influence on the DevOps collaboration culture, but many of these factors are independent of TR factors and their correlation may tend to be zero.

According to the PFA, there were no factors that indicated to have split loadings. These split loading factors loads in more than one category. The PCA of the twenty three (23) factor outputs and seven (7) iterations of extractions conformed to the eight (8) categories. Ultimately, factors were reduced from the initial twenty three (23) factors to eight (8) factors, resulting in a combined contribution of (82.8) of the total variance.

5.5 Reliability and correlation

To determine the degree of correlation between the decision variables and other measures that have been predicted in theory to correlate to them, this study employed a construct validity. Jupp (2006, pp.314-315) supports this in that it also determines whether these decision variables do not correlate with other variables that have not been theories not to correlate with them. Cronbach's Alpha was used to measure decision variables' reliability. Table 5.6 shows the summary reliability statistics for all the contributing factors in the research instrument.

5.5.1 Reliability of constructs

Decision variables were evaluated for their reliability before construct validity and correlation were tested. As suggested by Peng (2009, p. 428), Cronbach's

Table 5.6 – Reliability statistics.

Factor (Construct)	Cronbach's alpha
Construct 1: Trust (TR)	0.880
Construct 2: Open Communication	0.839
Construct 3: Respect (RE)	0.815
Construct 4: Intention (INT)	0.845
Construct 5: Role Responsibilities (RR)	0.728
Construct 6: Social Norm (SN)	0.794
Construct 7: Organisational Usefulness (OU)	0.770
Construct 8: Perceived Behavioural Control-Internal (PBC-I)	0.694

Alpha (α) was used to check for internal consistency of these decision variables. For the checking of the internal consistency of the decision variables, an item analysis was performed on the questionnaire items. This was done construct by construct to determine the Cronbach α values. These values were used as a measure of the reliability of the questionnaire. The three deciding criteria that were used to guide the inclusion or exclusion of the item in the questionnaire are as follows:

- Cronbach $\alpha > 0.8$, was considered a good reliability.
- Cronbach α of between 0.6 and 0.8, was considered as acceptable reliability.
- Cronbach $\alpha < 0.6$ was considered unacceptable reliability.

The construct by construct Cronbach's α values are discussed below. Tables 5.7 to 5.11 shows the construct Cronbach's α values of the items. It is noteworthy that only the acceptable items are shown in these tables.

Table 5.7 – Item statistics: Construct 1–TR.

Items	Mean	SD	Corrected item-Total correlation	Cronbach alpha (α) if Item deleted	Total Cronbach α
TR					**0.880**
Q49	72.06	55.231	0.604	0.824	
Q50	74.58	68.772	0.350	0.880	
Q51	72.12	55.472	0.592	0.825	
Q52	72.23	54.390	0.570	0.825	
Q53	74.62	68.263	0.332	0.877	
Q54	43.17	44.169	0.597	0.829	

5.5.1.1 Construct 1: Trust (TR)

For the TR construct, the overall Cronbach *a* value was computed to be 0.880 and was considered to be in the acceptable range. This Cronbach *a* value indicates a good reliability when the criterion indicated above is used. In cases where an individual item value is higher than this overall value, the item should be excluded from the list. All the items in Table 5.7 have Cronbach *a* values that are less than that of the threshold (0.880) and were therefore kept.

5.5.1.2 Construct 2: Open Communication (OC)

Construct 2 represents the OC factors. Table 5.8 shows the summarised Cronbach's α value as being 0.839, which is also acceptable since it indicates good reliability. Item Q13 displays a Cronbach's α value of 0.840, which is higher than the overall α. The implication of this is that should it be decided that this item is removed from this list, its removal would improve the overall Cronbach α value of that item. In this case, the Cronbach's α value

Table 5.8 – Item statistics: Construct 2–OC.

Items	Mean	SD	Corrected item-Total correlation	Cronbach alpha (α) if Item deleted	Total Cronbach α
OC					**0.839**
Q11	59.08	88.431	0.576	0.822	
Q12	58.88	88.308	0.593	0.821	
Q13	59.00	95.884	0.448	0.840	
Q14	59.25	92.484	0.472	0.829	
Q15	58.71	94.740	0.524	0.827	
Q16	58.79	96.476	0.445	0.831	

Table 5.9 – Item statistics: Construct 3–RE.

Items	Mean	SD	Corrected item-Total correlation	Cronbach alpha (α) if Item deleted	Total Cronbach α
RE					**0.815**
Q35	30.98	14.913	0.268	0.828	
Q36	31.06	13.366	0.453	0.807	
Q37	31.06	12.209	0.646	0.777	
Q38	30.98	12.289	0.718	0.766	
Q39	30.92	13.036	0.677	0.776	
Q40	31.19	13.005	0.401	0.822	

would move from 0.839 to 0.840. However, the 0.839 value is too close to the overall value (0.840) and its removal would therefore not be significant. That being said, the item was retained. All other items that significantly improved the total Cronbach α value were removed and are therefore not shown in this table.

Table 5.10 – Item statistics: Construct 5–INT.

Items	Mean	SD	Corrected item-Total correlation	Cronbach alpha (α) if Item deleted	Total Cronbach α
INT					**0.845**
Q41	71.92	57.023	0.622	0.826	
Q42	72.25	54.825	0.502	0.850	
Q43	72.23	53.117	0.815	0.814	
Q44	72.33	53.507	0.726	0.818	
Q45	72.23	54.853	0.707	0.846	
Q46	72.31	53.538	0.761	0.816	
Q47	72.42	54.264	0.647	0.821	
Q48	72.23	54.583	0.733	0.819	

5.5.1.3 Construct 3: Respect (RE)

Individual items Q35 and Q40 have Cronbach's α values of $0,828$ and 0.822, respectively, which are higher than that of the overall α value of 0.815 (See Table 5.9). This implies that should these items be removed, the overall Cronbach's α value would improve from 0.815 to 0.822 and 0.828, respectively. The two items were however retained as their removal was shown to be insignificant. The overall α value was also at an acceptable level. However, all other items that significantly improved the total Cronbach α value were removed and are therefore not shown in Table 5.9.

5.5.1.4 Construct 4: Intention (INT)

Intention construct contains twelve individual items. The overall Cronbach's α value was 0.845 which also represents good reliability. All of the individual items were observed to possess a good reliability (Cronbach's $\alpha > 0.8$). It

Table 5.11 – Item statistics: Construct 5–RR.

Items	Mean	SD	Corrected item-Total correlation	Cronbach alpha (α) if Item deleted	Total Cronbach α
RR					**0.728**
Q17	62.50	34.399	0.326	0.729	
Q18	63.10	33.618	0.410	0.709	
Q19	63.08	32.830	0.438	0.705	
Q21	63.04	33.182	0.477	0.704	
Q22	62.96	32.796	0.427	0.706	

was apparent from a comparison of the Cronbach's α value of the individual items with the overall Cronbach's α that items Q45 (Cronbach's α value of 0.853) and Q42 (Cronbach's α value of 0.850) were above the threshold. Both Cronbach's α values were however retained as their removal was deemed insignificant.

5.5.1.5 Construct 5: Role/Responsibilities (RR)

Another scale test for reliability was conducted on the construct RR. Three factors loaded in this construct with the overall Cronbach's $\alpha = 0.728$; As already indicated, this falls within the acceptable range. With five individual items, only one item (i.e. Q17 Cronbach's $\alpha = 0.729$) was above the threshold ($\alpha = 0.728$). This item was also retained for the simple reason that its removal was deemed insignificant. All other items that significantly improved the total Cronbach α value were removed and are therefore not shown in the table.

Table 5.12 – Item statistics: Construct 6–SN, 7–OU and 8–PBC-I.

Items	Mean	SD	Corrected item-Total correlation	Cronbach alpha (α) if Item deleted	Total Cronbach α
SN					**0.794**
Q5	11.79	6.804	0.572	0.759	
Q6	11.65	7.172	0.589	0.756	
Q7	11.98	5.614	0.674	0.707	
Q8	12.00	6.019	0.606	0.744	
Q10	11.58	9.505	0.706	0.743	
OU					**0.770**
Q23	13.25	3.777	0.522	0.740	
Q24	13.21	3.987	0.469	0.764	
Q25	13.54	2.796	0.656	0.669	
Q26	13.37	2.934	0.669	0.658	
Q27	10.94	5.148	0.384	0.683	
Q28	11.69	3.416	0.495	0.653	
PBC-I					**0.694**
Q29	10.94	5.148	0.384	0.683	
Q30	11.69	3.416	0.495	0.653	
Q31	11.02	5.305	0.436	0.663	
Q32	11.23	3.921	0.683	0.496	

5.5.1.6 Construct 6: Social Norm (SN)

As shown in Table 5.12, SN has the Cronbach α value of 0.814. As indicated in the table, all the six factors that were loaded into this construct were accepted

5.5.1.7 Construct 7: Organisational Usefulness (OU)

The overall Cronbach α value for this construct was 0.770 made up of six factors. These factors have a Cronbach α value that is less than the overall α and were therefore regarded as being acceptable.

5.5.1.8 Construct 8: Perceived Behavioural Control-Internal (PBC-I)

Lastly, the Cronbach α value of PBC-I was computed to be 0.694 as shown in Table 5.12. Four factors with Cronbach α values that are less that the threshold were accepted.

5.6 Correlation analyses

To test the strength of relationships existing between different constructs, a correlation matrix was calculated. Correlation is a measure of the linear relationship between two variables, with the correlation coefficient r as a measure of the association between two numerical variables, usually denoted as x and y (Jupp, 2006, p. 43). The correlation coefficient r indicates the strength and direction of a linear relationship and its value of r ranges from -1 to 1. In this research study, the following interpretations of r were used as a guide:

- $r = 1$ represents a perfect positive correlation;
- $r = 0$ represents no correlation and
- $r = -1$ represents a perfect negative correlation.

Table 5.13 – Pearson Correlation matrix. ($N = 312$).

		OC	RR	RE	TR	SN	OU	PBC	INT
OC	Pearson Correlation	1							
	Sig. (2-tailed)								
RR	Pearson Correlation	.114*	1						
	Sig. (2-tailed)	0.044							
RE	Pearson Correlation	.145*	.236**	1					
	Sig. (2-tailed)	0.010	0.000						
TR	Pearson Correlation	.337**	.150**	.530**	1				
	Sig. (2-tailed)	0.000	0.008	0.000					
SN	Pearson Correlation	.389**	.595**	.201**	0.047	1			
	Sig. (2-tailed)	0.000	0.000	0.000	0.405				
OU	Pearson Correlation	.664**	.245**	.720**	.620**	.254**	1		
	Sig. (2-tailed)	0.000	0.000	0.000	0.000	0.000			
PBC	Pearson Correlation	.252**	.552**	.480**	.781**	.143*	.526**	1	
	Sig. (2-tailed)	0.000	0.000	0.000	0.000	0.012	0.000		
INT	Pearson Correlation	.425**	-0.054	-0.108	.426**	-0.071	.140*	0.062	1
	Sig. (2-tailed)	0.000	0.340	0.056	0.000	0.211	0.013	0.278	

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

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

A positive correlation means that as the value of one variable increases, the value of the correlated variable also increases. A negative correlation on the other hand means that the correlated variables go in opposite directions; when one of the variables increases, the other one decreases and *vice versa*.

The size of r indicates the strength of the relationship. Thus, the larger the absolute value of r, the stronger the relationship. For the purpose of this study, the following guide was used:

- ± 1.00 represents a perfect correlation;
- ± 0.80 represents a strong correlation;

- ± 0.50 represents a moderate correlation;
- ± 0.20 represents a weak correlation; and
- ± 0 represents no correlation.

A Pearson correlation analysis was conducted to examine whether there is a relationship between the different variables. The variables examined were: OC, RR, RE, TR, SN, OU, PBC-I, and INT. The results of the Pearson correlation are tabulated in Table 5.13. Every variable correlates to itself at $r = 1$.

Only one relationship (i.e. the relationship between PBC-I and TR) revealed a significant strong positive whereby $r = 0.781$ and $p < 0.0001$ (shown as brown text in Table 5.13).

The second category of relationships those relationships that are significant, positive but moderate in strength. These relationships strengths are in the range $0.530 > r < 0.720$ and were significant at $p < 0.0001$. These relationships in their decreasing order are between: OC and OU (0.664), TR and OU (0.620), RR and SN (0.595), RR and PBC-I (0.552), RE and TR (0.480) and RE and PBC-I (0.480).

The third category is that of relationships that were weak in strength ($0.5 > |r| > 0.2$ as per the guideline) but are positive ($r > 0$) and significant ($p < 0.0001$). This category consists of ten (10) relationships at the range $[0.150, 0.426]$. These relationships, which are arranged in their decreasing order, are between: TR and INT (0.426), OC and INT (0.425), OC and SN (0.389), OC and TR (0.337), OU and SN (0.254), OC and PBC-I (0.252), RR and OU (0.245), RR and RE (0.236), RE and SN (0.201), and RR and TR (0.150).

The last category of relationships is negative correlations types of relationships whereby $r < 0$. The three relationships that exhibited this characteristic are between INT and RR (-0.054), INT and RE (-0.108), and INT and SN (-0.071). Although these relationships exist, their strength is too weak to be categorised as a weak relationship using the guideline outlined above. In addition, these relationships were not significant.

5.7 Regression analysis

Figure 5.4 shows the initial proposed framework which was reported by Masombuka and Mnkandla (2018). The collected data sample was fed into SPSS v25.0 against the framework. Regression analysis was carried out to test the framework.

Regression analysis is a collective name for methods that can be used for the modelling and analysis of numerical data consisting of values of a dependent variable (also called an outcome, measurement or criterion variable) and one or more independent variables (also known as explanatory variables or predictors). The dependent variable in the regression equation is modelled as a function of the independent variables, corresponding parameters and an error term. Regression is used for hypothesis testing and is also referred to as the modelling of causal relationships (Huizingh, 2007, pp. 298-299). The use of regression analyses relies heavily on the underlying assumptions being satisfied. Multiple hierarchical regression analysis was used to model the relationship between one continuous dependent variable and other continuous independent variables. The following assumptions of regression analysis were adopted for this research study:

- The error is assumed to be a random variable with a mean of zero.
- The independent variables are error-free, and the predictors are independent.
- The errors are uncorrelated, that is, the variance-covariance matrix of the errors is diagonal and each non-zero element is the variance of the error.
- The variance of the error is constant across observations (homoscedasti-

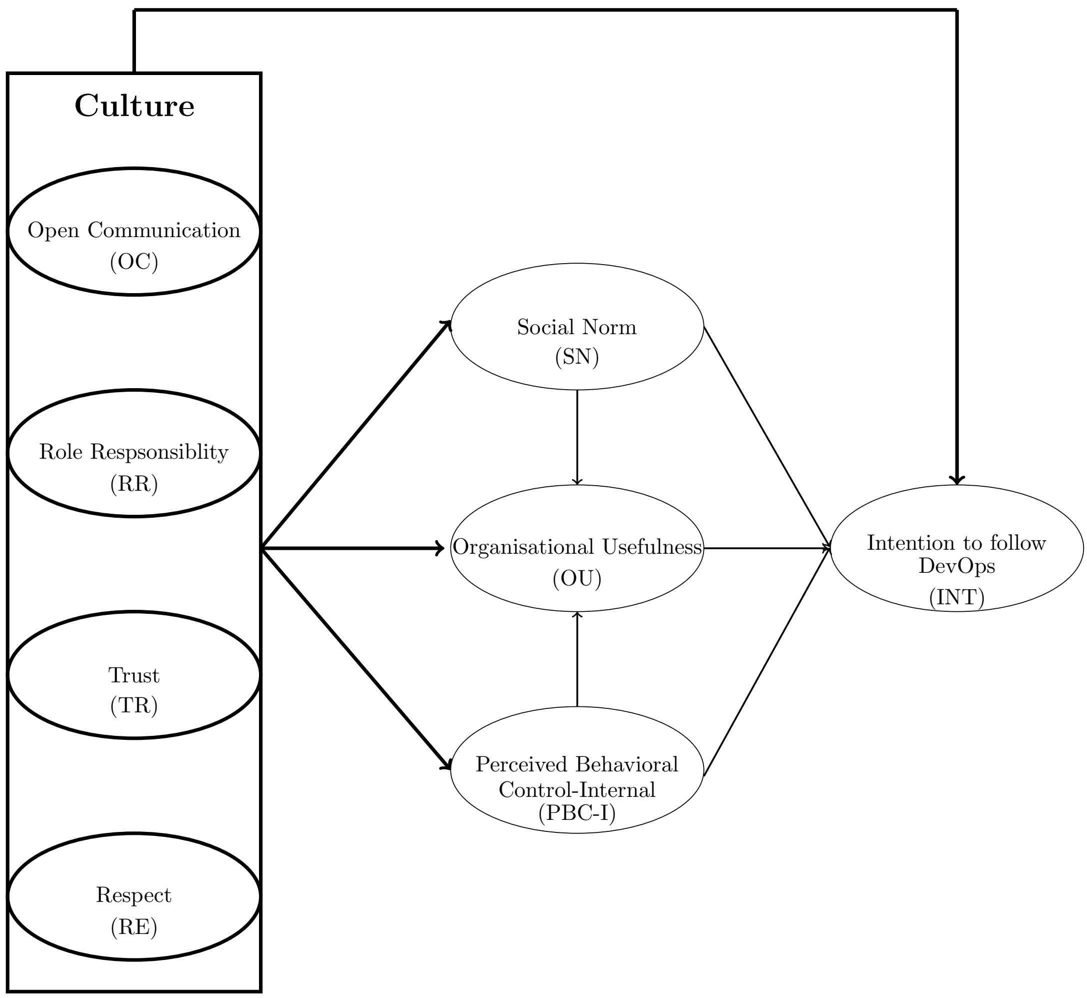


Figure 5.4 – The proposed framework (DCAM) (Masombuka & Mnkandla, 2018, p. 283).

city).

When testing theoretical models, it is important to examine beta scores in order to determine the importance of each variable relative to changes in the dependent variable diversity. "The regression coefficient B and the standardised coefficient beta (β) reflect the change in the dependent measure for each unit change in the independent variable. Comparison between regression coefficients allows for a relative assessment of each variable importance in the regression model" (Huizingh, 2007, p. 304). Both B and beta measure similar concepts, where B is the unstandardised coefficient, and beta is the value of the standardised regression coefficient calculated from standardised data. The standard error of the regression coefficient is an estimate of how much the regression coefficient will vary from samples of the same size taken from the same population. Therefore, instead of using B, it is more acceptable in statistical evaluations to look at the beta values for the estimate of the relative importance of each of the independent variables Huizingh (2007, p. 303).

Multicollinearity refers to the interrelations of predictor variables (Nishishiba et al., 2014, p. 271). High intercorrelation can lead to an increase in sensitivity to sampling and measurement errors (Reinard, 2006, p. 348). When multicollinearity increases, it complicates the interpretation due to the fact that it is more difficult to ascertain the effects of any single variable, because of the interrelationships that exist between variables (Reinard, 2006, p. 411). Multicollinearity is problematic if the variables under examination are not discriminant. In turn, this can cause inaccuracies in the multiple regression analysis. The Variance Inflation Factor (VIF) is the inverse of the tolerance value—instances of higher degrees of multicollinearity are reflected in lower tolerance values and consequently higher VIF values. The VIF translates the tolerance value, which directly expresses the degree of multicollinearity, into an impact on the estimation process: as the standard error increases, the confidence intervals around the estimated coefficients become larger, making it harder to demonstrate that the coefficient is significantly different from zero (Reinard, 2006, p. 375).

Tolerance scores range from 0 (perfect collinearity) to 1 (no collinearity) (Reinard, 2006, p. 375). Tolerance values in a range of 0 to 0.25 indicate a high degree of multicollinearity, and VIF levels equal to or greater than 4.0 indicate multicollinearity as well (Reinard, 2006, p. 374). The Durbin-Watson Test is a test for first-order serial correlation in the residuals of a time series regression. A value of 2.0 for the Durbin-Watson statistic indicates that there is no serial correlation (Vogt, 2011, p. 345). This is a significant phenomenon that indicates the degree to which the independent variables are sufficiently isolated from each other so that the regression values truly measure the contribution of each variable separately without possible cross-variable contamination.

5.7.1 Multiple hierarchical regression analysis to determine Intentions (INT)

Before regression analysis was carried out, the regression diagnostics were carried out in order to test the presumptions. Figure 5.5 presents the three output tests in the form of a histogram displaying the normality of the residuals, the Normal Probability plot (P-P plot) of regression and the scatter plot. The normality of the residuals of regression appears to be normally distributed. These residuals are the error terms or the differences between the observed value of the dependent variable and the predicted value. The P-P plot also confirms the normality of the data as shown on the plot—the little circles follow the normality line.

The residual scatter plot suggest the following about the appropriateness of the multiple linear regression:

- The residuals are scattered around the 0 line, suggesting that the assumption that the relationship is linear is reasonable.

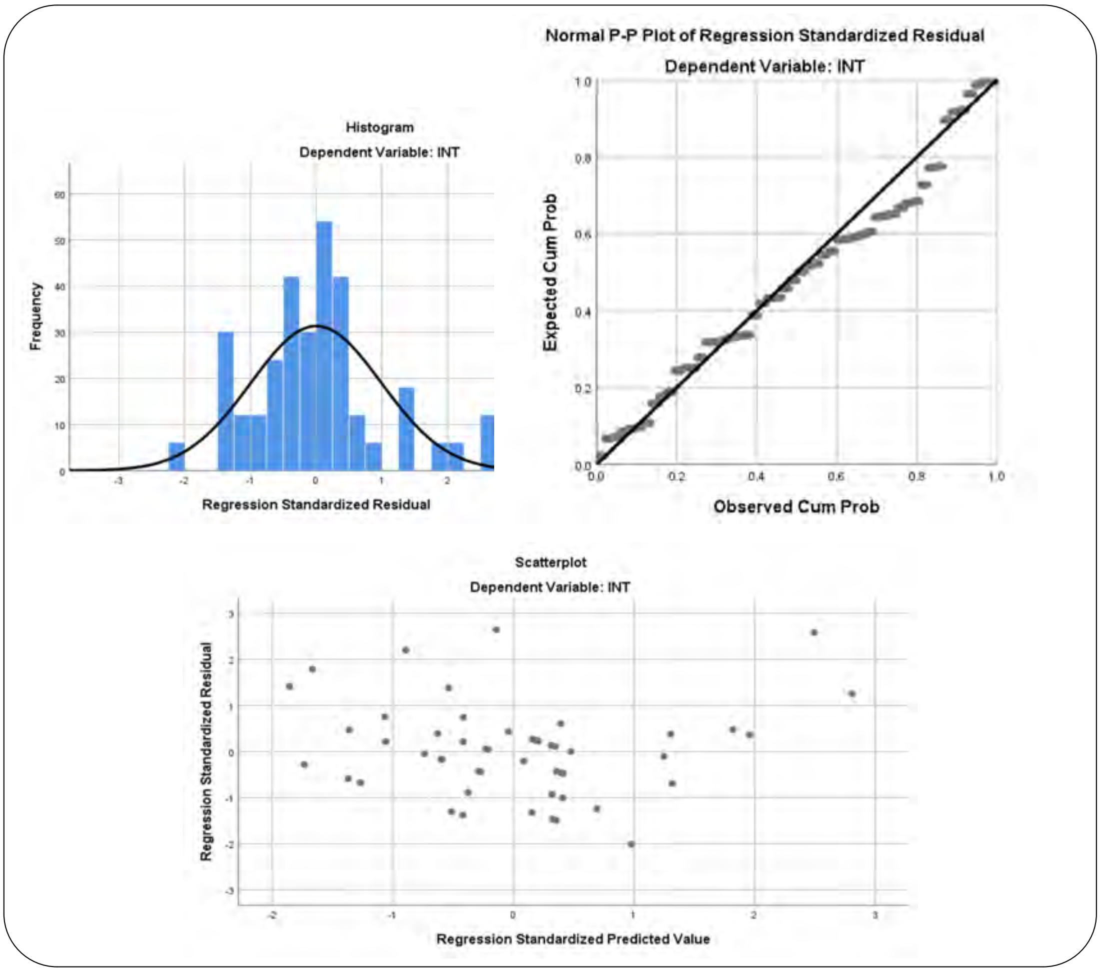


Figure 5.5 – Regression presumptions test (INT variable).

Table 5.14 – Regression analysis for intentions (INT) as a dependent variable.

(a) Model Summary(INT).

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate	Change Statistics					Durbin-Watson
					R Square Change	F Change	df1	df2	Sig. F Change	
1	0.727[a]	0.529	0.518	0.45098	0.529	48.793	7	304	0.000	2.256

(b) Coefficients[a] (INT).

	Model	Unstandardised Coefficients		Standardized Coefficients			95,0% Confidence Interval for B	
		B	Std. Error	Beta	t	Sig.	Lower Bound	Upper Bound
1	(Constant)	2.393	0.327		7.324	0.000	1.750	3.036
	SN	0.006	0.045	0.007	0.122	0.903	0.094	0.083
	OU	0.195	0.064	0.168	3.025	0.003	0.068	0.322
	PBC_I	0.099	0.038	0.146	2.616	0.009	0.024	0.173
	OC	0.164	0.022	0.305	7.529	0.000	0.121	0.207
	RR	0.036	0.037	0.039	0.964	0.336	-0.037	0.109
	RE	0.413	0.035	0.500	11.928	0.000	0.345	0.481
	TR	0.125	0.024	0.214	5.152	0.000	0.173	0.077

(c) ANOVA[a] (INT).

	Model	Sum of Squares	df	Mean Square	F	Sig.
1	Regression	69.465	7	9.924	48.793	.000b
	Residual	61.828	304	0.203		
	Total	131.294	311			

a. Predictors: (Constant), Culture(OC, RR, RE, TR), SN, PBC-I, OU

- The residuals form a roughly horizontal band around the 0 line. This explains the equal variance of error terms.
- No one residual stands out of the basic pattern suggesting that there are no outliers.

With all the assumptions satisfied, it was deemed appropriate to run regression analysis. The following regression model was tested:

$y_1 = (\beta_0 + \beta_{1xi1} + \beta_{2xi2} + \beta_{3xi3} + \beta_{4xi4}) + \epsilon_i$

where:

- y_i is the Intentions (INT) of a person i,
- x_{i1} is the Culture of a person i_i; where Culture is OC, RR, RE and TR,
- x_{i2} is the Social Norm (SN) of person i_i,
- x_{i3} is the Organisational Usefulness (OU) of a person i_i, and
- x_{i4} is the Perceived Behavioural Control-Internal (PBC-I) of a person i_i.
  and the independent error terms ϵ_i follows a normal distribution with mean 0 and equal variance σ^2. The variable ϵ is the value y is predicted to have when all the independent variables are equal to zero. β is the slope or the coefficient that describes the size of the effect the independent variable is having on the dependent variable y.

The output from the multiple regression analysis is shown in Tables 5.14a to 5.14c. The overall multiple regression model is significant at a 95% level of confidence with a p-value smaller than 0.05. The R^2 value is 0.518, which measures how much the above model fit the data analysed. Indicated in percentage figures, this means 51.8% of the variation in INT is reduced by taking into account Culture, SN, PBC-I and OU (see Table 5.14a). This is a statistically significant contribution as indicated by the Sig. F change value of (0.000), which is less than 0.05. The implication of this is that it is reasonable to reject the null hypothesis and accept the alternative hypothesis. The Durbin

Watson score of 2.256 regarded as the "ideal" Durbin Watson measure of independence (Vogt, 2011, p. 345) (see Table 5.14a), which indicates a high level of isolation among the independent variables of the model.

The ANOVA table indicates that the model as a whole (which includes both blocks of variables) is significant ($F(7; 304) = 48, 793, p < 0.05$) (see Table5.14c). This suggests that the model containing all the variables is more effective in predicting INT than not taking into account the four predictors (OC, RE, TR and OU).

The coefficients show to what extent variables contribute to the model or equation (see Table5.14b). In the significance column, the four variables that make a statistically significant contribution (less than 0.05) (see Table 5.14b) in order of importance are RE ($\beta = 0.500$), OC ($\beta = 0.302$), TR ($\beta = 0.214$) and OU ($\beta = 0.168$). This means that the slopes of these variables are statistically significant. From the above results, it is noticed that the DevOps culture elements constitutes the most influence. RE leads in terms of its influence signifying that it is the most important, followed by OC and then TR. OU has the least influence on the overall equation. The large β value afforded to respect indicates that it is the most critical element in terms of collaboration and teamwork. As reported by Ellemers et al. (2013), people are likely to contribute to a team when they feel that they are part of that team. This can be shown by how fellow teammates show respect to one another. The result is a team with a positive team identity. Teams exhibiting a positive identity may lead to increased productivity, as indicated by Carmeli et al. (2015).

When teammates are respected, they are free to communicate and contribute to the team. This is why communication remains a contributing factor in team collaboration. DevOps requires the collaboration between teammates to be open, meaning that there should be no boundaries that inhibit others from receiving or transferring information. Collaboration facilitates knowledge sharing, especially whereby the knowledge is transferred from the experienced members of the team to novices through open communication. The significance of communication styles to promote knowledge sharing was explored by de Vries et al. (2006). It is therefore crucial that communication is as open and free as possible in DevOps environments.

The last DevOps collaboration element that showed significance is trust. It is obvious that working together requires the element of trust amongst the engaging parties. In this context, the skills and the abilities of a particular team member should be trusted by the other team members so that all the team members are able to work freely knowing that the team member in question will play their part. Work conducted by Jong et al. (2016) has reported that trust promotes team productivity.

Besides the DevOps collaboration culture elements, the other construct that was significant in the determination of INT is OU. This means that intentions to use DevOps culture by the team increases when the team believes that by following this DevOps culture, the organisation would benefit. Indeed, DevOps encourages collaboration as the key element of its culture. Hence, studies by (Parker et al., 2015; Schuh et al., 2014) have shown that productivity can be boosted through collaboration. Hardgrave and Johnson (2003) has recognised OU as a significant determinant of INT when y the acceptance of information systems methodology by software developers was studied.

Not all the constructs were found to be significant. Small β values were not statistically significant were obtained for RR ($\beta = 0.039$), SN ($\beta = 0.007$) and

PBC-I ($\beta = 0.146$). Clearly defined roles and responsibilities should eliminate problems that may emerge from bringing the development and operations departments together. Unnecessary duplication of tasks and lack of accountability because of uncertainty about responsibilities and the scope of those responsibilities should be eliminated. Whereas Hoda et al. (2013) has shown that RR plays a significant role in improving performance, it was not demonstrated that RR significantly influence INT. Future studies should be conducted on this construct in this context.

SN was also found to be an insignificant determinant of INT. Although studies (Brown et al., 2010; Eckhardt et al., 2009*b*) have shown that positive social influence may lead to positive intention, this study did not find SN to be a statistically significant determinant of INT in this context. Similar results were reported by Hardgrave and Johnson (2003) upon an investigation of software developers intentions to follow information systems methodology.

PBC-I was, however, nearly significant as a p value of 0.009 with a $\beta = 0.146$. PBC-I are the skills one has to gain by accepting this culture. Collaboration facilitates team learning, as evidenced by a the study by Lindsjørn et al. (2016). The results of this study suggest that some DevOps practitioners agree with the fact that DevOps culture will also benefit them by building their skills—that may mean developers learning operators skills and vice versa. Future work is required to establish who is more eager to learn between the developers and operators. In corroboration of these results, Hardgrave and Johnson (2003) also did not find PBC-I to be significant.

5.7.1.1 Hypothesis 1 to 4

The first hypothesis this study intended to test is:

- **H1: There is a positive relationship between DevOps culture and intention to use DevOps.**

  In order to evaluate this hypothesis, the following sub-hypotheses were formulated:

  - H1a: There is a positive relationship between OC and INT to follow DevOps processes.
  - H1b: There is a positive relationship between RR alignment and INT to follow DevOps processes.
  - H1c: There is a positive relationship between TR and INT to follow DevOps processes.
  - H1d: There is a positive relationship between RE and INT to follow DevOps processes.

The hypothesis H1 regarding the influence of DevOps culture on INT was partially accepted. The hypotheses H1a, H1c, and H1d were accepted with only H1b rejected.

The following three hypotheses on the influence of SN, OU, and PBC-I on INT were also tested and were stated as follows:

- **H2: There is a positive relationship between SN and INT.**
- **H3: There is a positive relationship between OU and INT.**
- **H4: There is a positive relationship between PBC-I and INT.**

In this case, only H3 was accepted, whilst H2 and H4 were rejected as stated above. In summary, the determinants of INT, which were found to be statistically significant are H1a, H1c, H1d, and H3. That is, open communication,

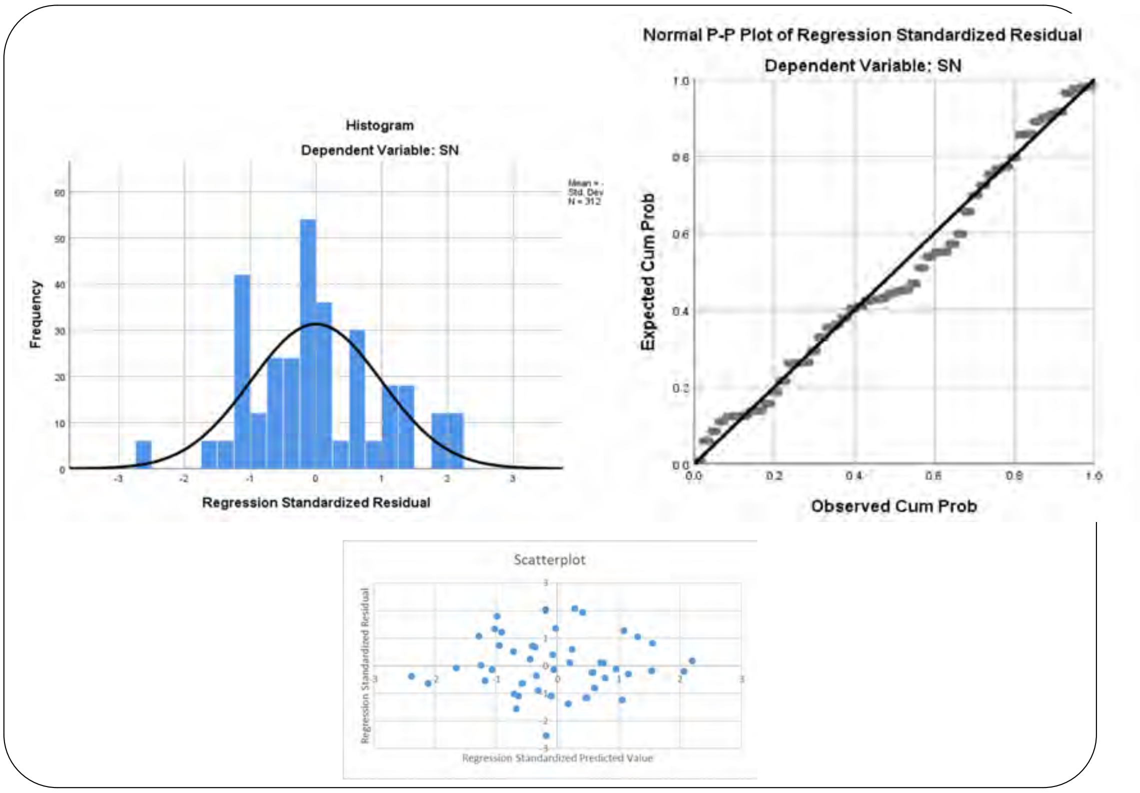


Figure 5.6 – Regression presumptions test (SN variable).

respect, trust and organisational usefulness are the determinants of intentions to follow the DevOps culture.

5.7.2 Multiple hierarchical regression analysis to determine Social Norm (SN)

As illustrated by the histogram in Figure 5.6, a normal distribution of residuals is demonstrated. The P-P plot also follows the normality line. In addition, the scatter plot shows that the residuals are scattered horizontally around zero and do not show any outliers. This suggests that it is appropriate to run a regression analysis.

Table 5.15 – Regression analysis for Social Norm (SN) as a dependent variable.

(a) Model Summary(SN).

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate	Change Statistics					Durbin-Watson
					R Square Change	F Change	df1	df2	Sig. F Change	
1	0.705[a]	0.497	0.490	0.32819	0.497	75.694	4	307	0.000	1.782

(b) Coefficients(SN).

		Unstandardised Coefficients		Standardized Coefficients			95,0% Confidence Interval for B	
Model		B	Std. Error	Beta	t	Sig.	Lower Bound	Upper Bound
1	(Constant)	1.850	0.198		9.352	0.000	1.461	2.239
	OC	0.219	0.024	0.387	8.953	0.000	0.171	0.267
	RR	0.510	0.039	0.553	13.230	0.000	0.434	0.586
	RE	0.083	0.028	0.142	2.911	0.004	0.027	0.139
	TR	0.231	0.048	0.241	4.805	0.000	0.325	0.136

(c) ANOVA(SN).

Model		Sum of Squares	df	Mean Square	F	Sig.
1	Regression	32.611	4	8.153	75.694	.000
	Residual	33.066	307	0.108		
	Total	65.676	311			

a. Predictors: (Constant), TR, RR, OC, RE

As mentioned in the preceding section (Section 5.7.1), the regression model can be metamerically presented as:

$$y_1 = (\beta_0 + \beta_{1xi1} + \beta_{2xi2} + \beta_{3xi3} + \beta_{4xi4}) + \epsilon_i$$

where:

- y_i is the Social Norm (SN) of person i;
- x_{i1} is the Open Communication (OC) of person i_i;

- x_{i2} is the Role/Responsibilities (RR) of person i_i;
- x_{i3} is the Respect (RE) of person i_i; and
- x_{i4} is the Trust (TR) of person i_i.The independent error term ϵ_i follows a normal distribution with mean 0 and equal variance σ^2. β is the slope.

The output from the multiple regression analysis is shown in Tables 5.15a to 5.15c. Once more, the overall multiple regression model is significant at a 95% level of confidence with a p-value that is smaller than 0.05. The R^2 value is 0.497 indicating that 49.7% of the variation in SN is reduced by taking into account OC, RR, RE and TR (see Table 5.15a). This is a statistically significant contribution as indicated by the Sig. F change value of 0.000. The Durbin Watson score of 1.782, which was rounded off to 2.000, which is still acceptable (see Table 5.14a). This indicates that there is no auto-correlation between residuals.

The ANOVA table (i.e. Table 5.15c) shows that this sub model is significant ($F(4;307) = 75.694, p < 0.05$).

The coefficients in Table 5.15b show how much the variables contribute to the model or equation. According to the significance column, all the variables make a statistically significant contribution (less than 0.05). In order of importance is RR ($\beta = 0.553$), OC ($\beta = 0.387$), RE ($\beta = 0.142$) and TR ($\beta = 0.241$). This means that the slopes of these variables are statistically significant. These results suggest that a team culture does influence the social norm of individuals in that team. Positive team culture should encourage positive attitudes in a team. This is supported by the study by Brown et al. (2010) which revealed that superior and peer influences have a significant effect on the social norm when studying the use of collaboration technology. Positive team culture can be built in an environment where the role and responsibilities of every team member are clear and unambiguous. This environment should encourage team respect, and trust whilst allowing communications to be open and fear-free. From the above results, one can conclude that DevOps culture plays a significant role in SN. Therefore, the model $y_1 = (\beta_0 + \beta_{1xi1} + \beta_{2xi2} + \beta_{3xi3} + \beta_{4xi4}) + \epsilon_i$ is acceptable.

5.7.2.1 Hypothesis 5

The original hypotheses formulation was as follows:

- **H5: There is a positive relationship between culture and individual subjective norm.**
  - H5a: There is a positive relationship between OC and SN.
  - H5b: There is a positive relationship between RR alignment and SN.
  - H5c: There is a positive relationship between TR and SN.
  - H5d: There is a positive relationship between RE and SN.

The results of the regression analysis support the above hypothesis. This means that there is a positive relationship between culture and social norm. Put differently, team culture is influenced by people around team members, and in turn, people around the team member are influenced by the culture they are submerged in.

5.7.3 Multiple hierarchical regression analysis to determine Organisational Usefulness (OU)

Figure 5.7 shows the test for regression assumptions. The histogram reveals a normal distribution of residuals. The P-P plot also follows the normality line. The scatter plot shows that the residuals are scattered horizontally around zero and do not show any outliers. This suggests that it is appropriate to run a regression analysis as its presumptions have been met.

The mathematical representation of the regression model is as follows:

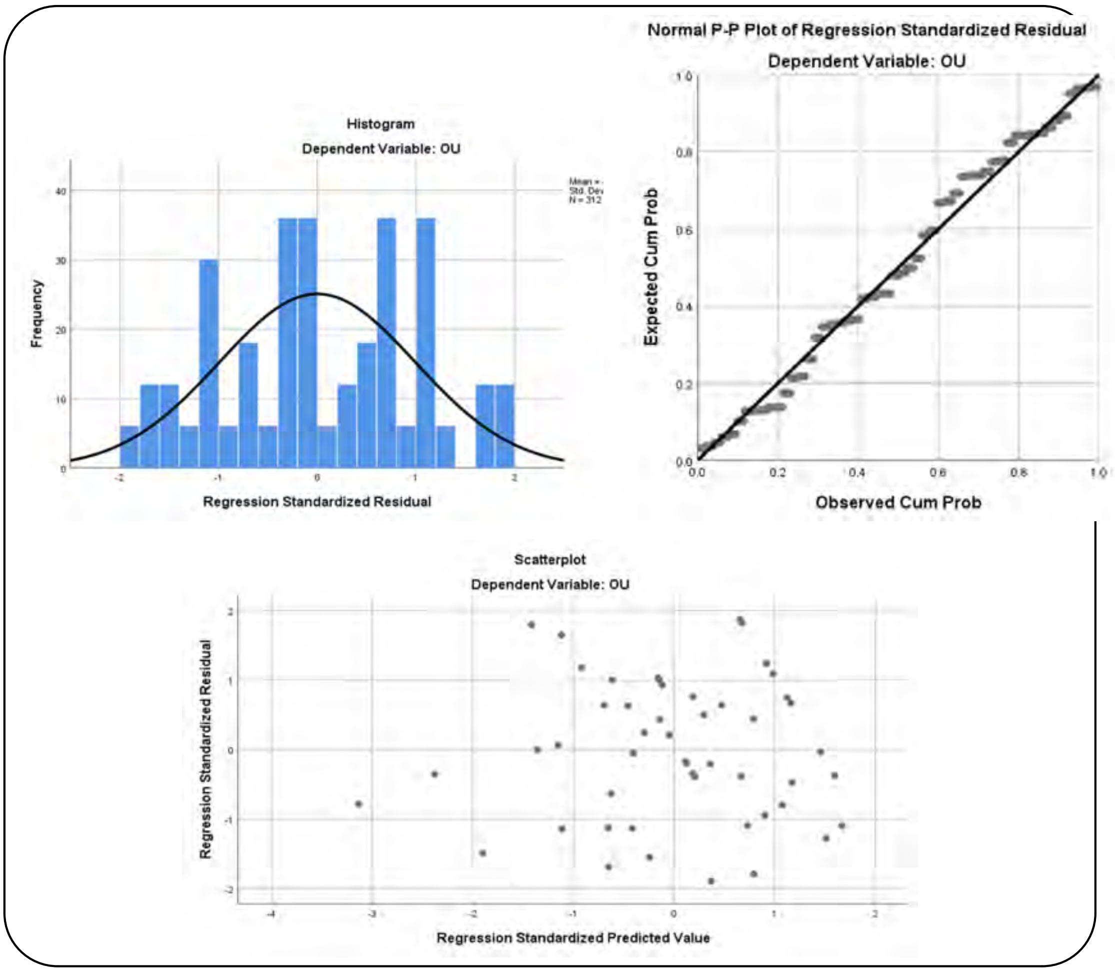


Figure 5.7 – Regression presumptions test (OU variable).

$$y_1 = (\beta_0 + \beta_{1xi1} + \beta_{2xi2} + \beta_{3xi3} + \beta_{4xi4}) + \epsilon_i$$

where:

- y_i is the Organisational Usefulness (OU) of person i;
- x_{i1} is the Open Communication (OC) of person i_i;
- x_{i2} is the Role/Responsibilities (RR) of person i_i;
- x_{i3} is the Respect (RE) of person i_i; and
- x_{i4} is the Trust (TR) of person i_i.
  The independent error terms ϵ_i follows a normal distribution with mean 0 and equal variance σ^2. β is the slope.

Tables 5.16a to 5.16c show the output from the multiple regression analysis of the above model. From Table 5.16a, it can be deduced that the overall multiple regression model is significant at a 95% level of confidence with a p-value that is smaller than 0.05. The R^2 value of 0.852 implies that 85.2%, of the variation in OU is reduced by taking into account OC, RR, RE and TR. This is a statistically significant contribution as indicated by the Sig. F change value of 0.000. The Durbin Watson score of 1.851, which was rounded off to 2.000, indicates that there is no auto-correlation between residuals.

The ANOVA table (see Table 5.16c) shows that the above model is significant ($F(4; 307) = 440.715, p < 0.05$).

The coefficients listed in Table 5.16b show how much the variables contribute to the model or equation. It can be seen that all the variables make a statistically significant contribution (less than 0.05), except for RR ($\beta = 0.031$) with

Table 5.16 – Regression analysis for Organisational Usefulness (OU) as a dependent variable.

(a) Model Summary(OU).

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate	Change Statistics					Durbin-Watson
					R Square Change	F Change	df1	df2	Sig. F Change	
1	0.923[a]	0.852	0.850	0.22497	0.852	440.715	4	307	0.000	1.851

(b) Coefficients[a] (OU).

		Unstandardised Coefficients		Standardized Coefficients			95,0% Confidence Interval for B	
Model		B	Std. Error	Beta	t	Sig.	Lower Bound	Upper Bound
1	(Constant)	0.228	0.136		1.685	0.093	-0.038	0.495
	OC	0.381	0.017	0.533	22.739	0.000	0.348	0.414
	RR	0.036	0.026	0.031	1.358	0.175	-0.016	0.088
	RE	0.416	0.020	0.563	21.303	0.000	0.377	0.454
	TR	0.165	0.033	0.137	5.007	0.000	0.100	0.229

(c) ANOVA(OU).

Model		Sum of Squares	df	Mean Square	F	Sig.
1	Regression	89.222	4	22.306	440.715	.000b
	Residual	15.538	307	0.051		
	Total	104.760	311			

a. Predictors: (Constant), TR, RR, OC, RE

$p = 0.175$.The order of importance of the significant variables is as follows: RE ($\beta = 0.563$), OC ($\beta = 0.533$) and TR ($\beta = 0.137$). These results suggest that respect is the most crucial element that will benefit the organisation when working in teams. Respect is accompanied by better communication between team members and trust is built and strengthened in the process. Teams operating in such an environment become stronger and positive, and this benefits the organisation since production is improved (Melo et al., 2013; Parker et al., 2015; Schuh et al., 2014). From the results described above, it can be concluded that all the elements of culture, except RR, play a significant role in OU.

5.7.3.1 Hypothesis 6

- **H6: There is a positive relationship between culture and OU.**
  - H6a: There is a positive relationship between OC and OU.
  - H6b: There is a positive relationship between RR alignment and OU.
  - H6c: There is a positive relationship between TR and OU.
  - H6d: There is a positive relationship between RE and OU.

The above hypothesis is partially accepted. This is because all the hypotheses were acceptable (except for RR (H6b)) since the results have shown that RR was not statistically significant.

5.7.4 Multiple hierarchical regression analysis to determine Perceived Behavioural Control-Internal (PBC-I)

Figure 5.8 shows the test for regression assumptions. The histogram shows normal distribution of residuals. The P-P plot also follows the normality line. The scatter plot shows that the residuals are scattered horizontally around zero and do not indicate any outliers. This suggests that it is appropriate to run regression analysis as its presumptions have been met.

The mathematical representation of the regression model is as follows:

$y_1 = (\beta_0 + \beta_{1xi1} + \beta_{2xi2} + \beta_{3xi3} + \beta_{4xi4}) + \epsilon_i$

where:

- y_i is the Perceived Behavioural Control -Internal (PBC-I) of person i;
- x_{i1} is the Open Communication (OC) of person i_i;
- x_{i2} is the Role/Responsibilities (RR) of person i_i;
- x_{i3} is the Respect (RE) of person i_i; and
- x_{i4} is the Trust (TR) of person i_i.

  The independent error terms ϵ_i follows a normal distribution with mean 0 and equal variance σ^2. β is the slope.

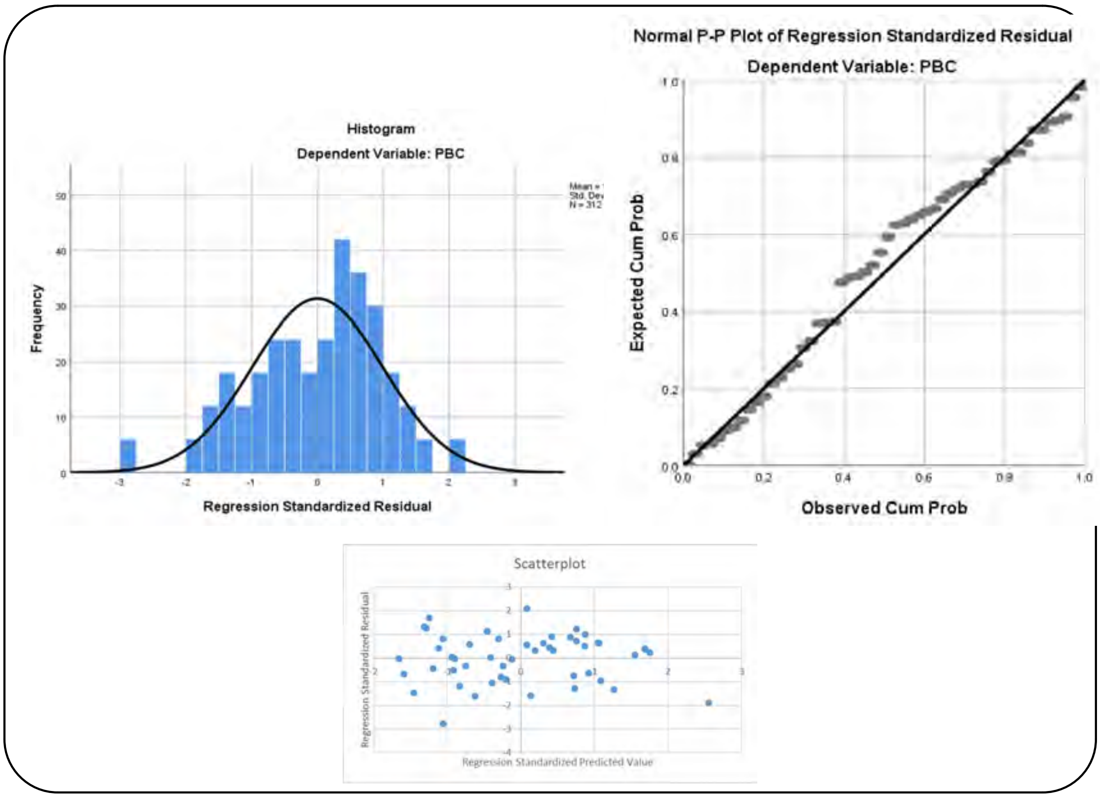


Figure 5.8 – Regression presumptions test (PBC-I variable).

Tables 5.17a to 5.17c show the output from the multiple regression analysis of the above model. According to Table 5.16a, the overall multiple regression model is significant at a 95% level of confidence with a p-value smaller than 0.05. The R^2 value is 0.806 implying that 80.6%, of the variation in PBC-I is reduced by taking into account OC, RR, RE and TR. This is a statistically significant contribution as indicated by the Sig. F change value of 0.000. The Durbin Watson score of 1.961, which was rounded off to 2.000, indicates no auto-correlation between residuals.

The ANOVA table (see Table 5.17c) shows that the above model is significant ($F(4;307) = 318.470$ with $p < 0.05$).

The coefficients in shows how much the variables contribute to the model or equation. According to Table 5.17b, only two of the four variables make a statistically significant contribution (less than 0.05). The two variables are (in order of importance): TR ($\beta = 0.734$), RR ($\beta = 0.449$). Trust amongst team members is important especially when members have something to gain or learn from other members. For the skills to be transferred from one team member to another, the member with the skills needs to be trusted. This notion is supported by a study by Carmeli et al. (2015), which reported just how important trust is in a team when sharing knowledge. When people feel respected and trusted, they are more likely to share their knowledge and skills to the benefit of others. Similarly, when role and responsibilities are clear, the party seeking the information will know where and how to search for the skills they need to complete their tasks. This will in turn improve productivity as suggested by Hoda et al. (2013). From the results mentioned above, it can be concluded that, save for OC and RE, all the elements of culture play a significant role in PBC-I.

Table 5.17 – Regression analysis for Perceived Behavioural Control-Internal (PBC-I) as a dependent variable.

(a) Model Summary(PBC-I).

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate	Change Statistics					Durbin-Watson
					R Square Change	F Change	df1	df2	Sig. F Change	
1	0.898[a]	0.806	0.803	0.27894	0.806	318.470	4	307	0.000	1.961

(b) Coefficients(PBC-I).

		Unstandardised Coefficients		Standardized Coefficients			95,0% Confidence Interval for B	
	Model	B	Std. Error	Beta	t	Sig.	Lower Bound	Upper Bound
1	(Constant)	-1.755	0.168		-10.436	0.000	-2.085	-1.424
	OC	-0.036	0.021	-0.046	-1.714	0.088	-0.077	0.005
	RR	0.567	0.033	0.449	17.295	0.000	0.502	0.631
	RE	-0.007	0.024	-0.008	-0.279	0.780	-0.054	0.041
	TR	0.959	0.041	0.734	23.516	0.000	0.879	1.039

(c) ANOVA(PBC-I).

Model		Sum of Squares	df	Mean Square	F	Sig.
1	Regression	99.116	4	24.779	318.470	.000b
	Residual	23.887	307	0.078		
	Total	123.002	311			

a. Predictors: (Constant), TR, RR, OC, RE

5.7.4.1 Hypothesis 7

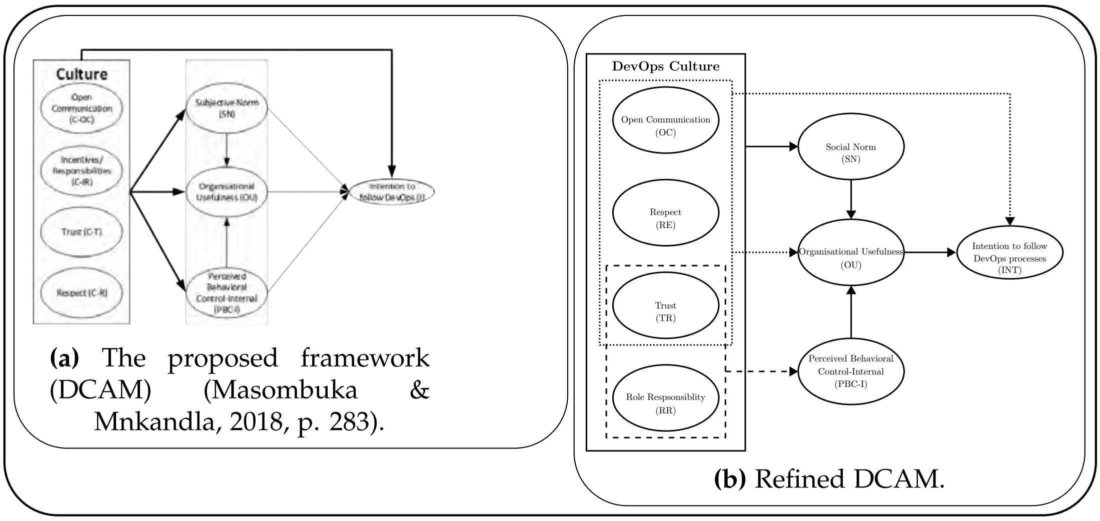
- **H4: There is a positive relationship between culture and PBC-I.**
  - H4a: There is a positive relationship between OC and PBC-I.
  - H4b: There is a positive relationship between RR alignment and PBC-I.
  - H4c: There is a positive relationship between TR and PBC-I.

– H4d: There is a positive relationship between RE and PBC-I.

The above hypothesis is also partially acceptable. That is, only two of the sub-hypotheses were accepted (H7b and H7c) and the other two were rejected (H7a and H7d).

5.8 Refined Framework

Figure 5.9 illustrates the transformation of our original proposed framework that was based on variables and relationships that were acceptable on the basis of the hypotheses presented above. The original proposed framework is shown in Figure 5.9a (the figure on the left), whist the transformed framework is depicted in Figure 5.9b (the figure on the right). From the two figures, one can see According to Figures 5.9a and Figure 5.9a, the variables that influence SN are OC, RE, TR and RR. These are all the variables that were initially proposed in the original framework. However, in the case of OU and PBC-I,



(a) The proposed framework (DCAM) (Masombuka & Mnkandla, 2018, p. 283).

(b) Refined DCAM.

Figure 5.9 – Proposed framework transformation.

RR was not significant when measured against OU, whilst OC and RE were not significant when measured against PBC-I. Thus, the refined framework shows the significant variables of OU as OC, RE, and TR and PBC-I variables as TR and RR. INT is influenced by variables OC, TR, RE from the DevOps culture and the variable OU.

Another set of variables that may affect the interaction of the mediating variable and the dependent variable is the moderator variable. A detailed discussion on the effects of these variables is presented in the next section.

5.9 The effect of the moderator variable (Job role)

The job role variable is divided into three levels, namely Developers (*Dev*), Operators (*Ops*) and both (*D&O*) roles. It is indicated in Table 5.1 that 21.2% of the total respondents came from the *Dev* group, 28.8% was from the *Ops* group, and the remainder (50%) was from the *D&O* group. The moderator variable was tested for its effect on SN, OU, PBC-I and INT on the DevOps culture variable using regression analysis. The results obtained after the run of the regression analysis and their interpretations, are presented in the subsequent sections.

5.9.1 Effects of Job Role on Social Norm (SN)

The hypothesis to be tested in this section is as follows:

> **H10: The interaction between DevOps collaboration culture and SN is moderated by job role.**

Table 5.18 – Job role as a moderator variable for SN.

(a) Model Summary.

Variable	R	R Square	Adjusted R Square	Std. Error of the Estimate
Dev	.672^a	0.452	0.416	0.73911
Ops	.368^a	0.135	0.092	0.58615
D&O	.592^a	0.351	0.335	0.63357

(b) ANOVA^{a,b}.

Model	Sum of Squares			df			Mean Square			F			Sig.		
	Dev	*Ops*	*D&O*	*Dev*	*Ops*	*D&O*	*Dev*	*Ops*	*D&O*	*Dev*	*Ops*	*D&O*	*Dev*	*Ops*	*D&O*
Regression	27.495	4.251	34.089	4	4	4	6.874	1.063	8.522	12.583	3.093	21.231	.000^a	.020^a	.000^a
Residual	33.323	27.142	63.022	61	79	157	0.546	0.344	0.401						
Total	60.818	31.393	97.111	65	83	161									

(c) Coefficients^{a,b}.

Model	Unstandardised Coefficients						Standardized Coefficients								
	B			Std. Error			Beta			t			Sig.		
	Dev	*Ops*	*D&O*	*Dev*	*Ops*	*D&O*	*Dev*	*Ops*	*D&O*	*Dev*	*Ops*	*D&O*	*Dev*	*Ops*	*D&O*
(Constant)	3.620	2.737	2.069	1.060	0.716	0.471				3.414	3.824	4.393	0.001	0.000	0.000
OC	0.595	0.063	-0.148	0.088	0.067	0.045	0.671	0.111	-0.248	6.750	0.935	-3.287	0.000	0.353	0.001
RR	0.326	0.136	0.613	0.153	0.134	0.069	0.262	0.125	0.580	2.135	1.010	8.889	0.037	0.316	0.000
RE	0.148	0.199	0.032	0.365	0.084	0.081	0.074	0.301	0.031	0.406	2.362	0.400	0.686	0.021	0.689
TR	-1.071	-0.093	0.016	0.454	0.182	0.121	-0.379	-0.059	0.011	-2.359	-0.509	0.134	0.022	0.612	0.894

The hypothesis is divided into the following sub-hypotheses:

H10a: The interaction between DevOps collaboration culture and SN is moderated by *Dev* job role.

H10b: The interaction between DevOps collaboration culture and SN is moderated by *Ops* job role.

H10c: The interaction between DevOps collaboration culture and SN is moderated by *D&O* job role.

Table 5.18 shows the results from the run of regression analysis to determine the effect of the job role as the moderator variable on the dependent variable SN. The summaries of the model for *Dev*, *Ops* and *D&O* are tabulated in Table 5.18a. According to Table 5.18a., the R^2 values—which measure how well the regression model fits the data—for these levels are 0.452, 0.135, 0.351 for *Dev*, *Ops* and *D&O*, respectively. The implication of this is that 45.2%, 13.5% and 35.1% of the variation in SN is explained by taking into account *Dev*, *Ops* and *D&O*, respectively. From these numbers, it is clear that developers have the most influence on the relationship between DevOps culture and SN (see Section 5.7.2). Operators had the least influence.

Table 5.18b shows that the results obtained from the analysis of variance for the job variable on the three levels in relation to the SN variable. It is clear from Table 5.18b that all job role levels are statistically significant (p-value is less than 0.05). This implies that the null hypothesis can be rejected. The F-values for the three levels are 12.583, 3.093 and 21.231, respectively.

The coefficients table (Table 5.18c) gives an indication of how much contribution do these levels make to the regression model in relation to predictors of SN. The SN predictors are OC, RR, RE and TR. The following associations are significant:

- OC: *Dev* ($p = 0.00$, $\beta = 0.671$) and *D&O* ($p = 0.001$, $\beta = 0.045$);

- RR: *Dev* ($p = 0.037$, $\beta = 0.262$) and *D&O* ($p = 0.000$, $\beta = 0.580$);
- RE: *D&O* ($p = 0.021$, $\beta = 2.362$), and $Ops(p = 0.021$, $\beta = 2.362)$; and
- TR: *Dev* ($p = 0.022$, $\beta = -2.359$).

As shown in Table 5.18c, *Ops* is only significant in relation to RE; however, this is not the case for *Dev*. Nevertheless, the summaries of the model summaries indicate that all levels are significant in terms of the regression models. It can be concluded that the regression model presented is supportable and therefore the main hypothesis is acceptable. The three sub hypotheses are also acceptable. The next section looks at how this moderator variable affects the OU.

5.9.2 Effect of Job Role on Organisational Usability (OU)

The hypothesis to be tested in this section is as follows:

> **H11: The interaction between DevOps collaboration culture and OU is moderated by job role.**

The hypothesis is divided into the following sub-hypotheses:

> H11a: The interaction between DevOps collaboration culture and OU is moderated by *Dev* job role.
>
> H11b: The interaction between DevOps collaboration culture and OU is moderated by *Ops* job role.
>
> H11c: The interaction between DevOps collaboration culture and OU is moderated by *D&O* job role.

Table 5.19 – Job role as a moderator variable for OU.

(a) Model Summary

Variable	R	R Square	Adjusted R Square	Std. Error of the Estimate
Dev	.501[a]	0.251	0.202	0.35206
Ops	.548[a]	0.301	0.265	0.48112
D&O	.270[a]	0.073	0.049	0.55410

(b) ANOVA[a,b].

Model	Sum of Squares			df			Mean Square			F			Sig.		
	Dev	*Ops*	*D&O*	*Dev*	*Ops*	*D&O*	*Dev*	*Ops*	*D&O*	*Dev*	*Ops*	*D&O*	*Dev*	*Ops*	*D&O*
Regression	2.530	7.856	3.796	4	4	4	0.633	1.964	0.949	5.104	8.485	3.091	.001a	.000a	.018a
Residual	7.561	18.286	48.204	61	79	157	0.124	0.231	0.307						
Total	10.091	26.143	52.000	65	83	161									

(c) Coefficients[a,b].

	Unstandardised Coefficients						Standardized Coefficients								
Model	B			Std. Error			Beta			t			Sig.		
	Dev	*Ops*	*D&O*	*Dev*	*Ops*	*D&O*	*Dev*	*Ops*	*D&O*	*Dev*	*Ops*	*D&O*	*Dev*	*Ops*	*D&O*
(Constant)	3.620	2.737	2.069	1.060	0.716	0.471				3.414	3.824	4.393	0.001	0.000	0.000
OC	0.595	0.063	-0.148	0.088	0.067	0.045	0.671	0.111	-0.248	6.750	0.935	-3.287	0.000	0.353	0.001
RR	0.326	0.136	0.613	0.153	0.134	0.069	0.262	0.125	0.580	2.135	1.010	8.889	0.037	0.316	0.000
RE	0.148	0.199	0.032	0.365	0.084	0.081	0.074	0.301	0.031	0.406	2.362	0.400	0.686	0.021	0.689
TR	-1.071	-0.093	0.016	0.454	0.182	0.121	-0.379	-0.059	0.011	-2.359	-0.509	0.134	0.022	0.612	0.894

a. Predictors: (Constant), TR, OC, RR, RE

b. Dependent Variable: OU

As in the previous section, Table 5.19 shows the results of the regression analysis for determining the effect of the moderator variable on the dependent variable, in this case OU. The model summaries are tabulated in Table 5.19a. The R^2 values were computed to be 0.251,0.301,0.073 for the *Dev*, *Ops*, and

D&O job roles, respectively. This indicates that 25.1%, 30.1% and $7,3\%$ of the variation in OU is explained by taking into account *Dev*, *Ops* and *D&O*, respectively. The R^2 value for the *D&O* job level is very low indicating that this job role level makes very little contribution to the model.

Operators have a greater influence on the outcomes of the relationship between DevOps culture and OU (as discussed in Section 5.7.3 above.). These results are the opposite of what was found in the relationship between DevOps culture and SN, in which developers took the lead. In addition to that, *D&O* job level has the least influence in this case, although the majority of the respondents came from *D&O* job level group, This means that they don't believe that DevOps culture is useful to their organisations, than operators and developers.

Although the *D&O* level makes very little contribution to the regression model , this level contribution–and the two *others*-are still statistically significant shown in Table 5.19b. In addition, the F-values and the significance of these levels are depicted in 5.19b as follows: $Dev(F = 5.104, p = 0.001)$, $Ops(F = 8.485, p = 0.00)$, and $D\&O(F = 3.091, p = 0.018)$. Such values of p allow for the acceptance of the three sub-hypotheses and thus confirming the primary hypotheses and the rejection of the null hypothesis.

The coefficient values listed in Table 5.19c indicate how much contribution the job role levels make to the entire regression model. The following associations are significant:

- OC: *Dev* ($p = 0.000$, $\beta = 0.671$) and *D&O* ($p = 0.001$, $\beta = -0.248$);
- RR: *Dev* ($p = 0.037$, $\beta = 0.262$) and *D&O* ($p = 0.000$, $\beta = 0.580$);

- RE: *Ops* ($p = 0.021$, $\beta = 0.301$); and
- TR: *Dev* ($p = 0.022$, $\beta = -0.379$).

In summary, it can be concluded that the regression model presented above is supportable. The three sub-hypotheses are also acceptable and the research hypothesis is therefore accepted. The next section looks at how this moderator variable affects the OU.

5.9.3 Job role effect on Perceived Behavioural Control-Internal (PBC-I)

The hypothesis to be tested in this section is as follows:

> **H12: The interaction between DevOps collaboration culture and PBC-I is moderated by job role.**

The hypothesis is divided further into the following sub-hypotheses:

> H12a: The interaction between DevOps collaboration culture and PBC-I is moderated by *Dev* job role.
>
> H12b: The interaction between DevOps collaboration culture and PBC-I is moderated by *Ops* job role.
>
> H12c: The interaction between DevOps collaboration culture and PBC-I is moderated by *D&O* job role.

Table 5.20 – Job role as a moderator variable for PBC-I.

(a) Model Summary.

Variable	R	R Square	Adjusted R Square	Std. Error of the Estimate
Dev	.927[a]	0.859	0.850	0.41032
Ops	.279[a]	0.078	0.031	0.88988
D&O	.325[a]	0.106	0.083	0.81912

a. Predictors: (Constant), TR, OC, RR, RE

b. Dependent Variable: PBC-I

(b) ANOVA[a,b].

Model	Sum of Squares			df			Mean Square			F			Sig.		
	Dev	*Ops*	*D&O*	*Dev*	*Ops*	*D&O*	*Dev*	*Ops*	*D&O*	*Dev*	*Ops*	*D&O*	*Dev*	*Ops*	*D&O*
Regression	62.548	5.263	12.436	4	4	4	15.637	1.316	3.109	92.879	1.662	4.634	.000c	.167c	.001c
Residual	10.270	62.559	105.341	61	79	157	0.168	0.792	0.671						
Total	72.818	67.821	117.778	65	83	161									

(c) Coefficients[a,b].

Model	Unstandardised Coefficients						Standardized Coefficients								
	B			Std. Error			Beta			t			Sig.		
	Dev	*Ops*	*D&O*	*Dev*	*Ops*	*D&O*	*Dev*	*Ops*	*D&O*	*Dev*	*Ops*	*D&O*	*Dev*	*Ops*	*D&O*
(Constant)	-2.519	1.825	2.755	0.589	1.087	0.609				-4.280	1.680	4.524	0.000	0.097	0.000
OC	-0.124	-0.198	0.131	0.049	0.102	0.058	-0.128	-0.238	0.199	-2.535	-1.941	2.248	0.014	0.056	0.026
RR	0.932	0.137	0.251	0.085	0.204	0.089	0.684	0.086	0.216	11.002	0.671	2.813	0.000	0.504	0.006
RE	-2.759	-0.165	0.211	0.203	0.128	0.105	-1.264	-0.170	0.182	-13.622	-1.293	2.012	0.000	0.200	0.046
TR	4.132	0.596	-0.375	0.252	0.277	0.157	1.336	0.259	-0.238	16.394	2.153	-2.391	0.000	0.034	0.018

Regression analysis was carried out using SPSS ver 25.0 to determine the effect of the moderator variable on the dependent variable PBC-I. These computations are tabulated in Table 5.20. The model summaries are displayed in Table 5.20a. *Dev* has the highest R^2 value. Not only is this value the highest, it is a strong value at 0.859. Eighty five point nine percent (85.9%) of the effects of the job role variable is contributed by *Dev*. On the other hand, *Ops* and *D&O* have the respectiveR^2 values of 0.078 and 0.106, which and contribute 7.8% and 10.6%, respectively.

Most developers believe that following a DevOps culture may benefit them in terms of knowledge and skills they are likely to acquire. Operators do not show interest in these skills transfer. These results support the results from Section 5.9.1 which means that from the social interactions knowledge and skills can be transferred.

The regression model was significant for two levels, *Dev* and *D&O* with $p = 0.000$ and $p = 0.001$, respectively (see Table 5.20b). *Ops* was however not significant ($p = 0.167$). The F-values were found to be as follows: $Dev(F = 92.879)$, $D\&O(F = 4.634)$ and $Ops(F = 1.662)$. Only two of the sub-hypotheses are accepted, which means the main hypothesis is partially accepted.

The coefficients table (Table 5.20c) illustrates how much contribution the job role levels are making towards the regression model. The following associations are significant:

- OC: *Dev* ($p = 0.000$ $\beta = -4.280$) and *D&O* ($p = 0.026$, $\beta = 2.248$);
- RR: *Dev*($p = 0.000$ $\beta = 11.002$) and *D&O* ($p = 0.006$,$\beta = 2.813$);
- RE: *Dev*($p = 0.000$, $\beta = -13.622$), $D\&O(p = 0.046$, $\beta = 2.012)$; and
- TR: *Dev*($p = 0.000$, $\beta = 16.394$), $Ops(p = 0.034, \beta = 2.153)$ and $D\&O(p = 0.018,\beta = -2.391)$.

In summary, the main hypothesis was partially accepted as one sub-hypothesis was rejected (H7b). The final moderator effect that is between DevOps collaboration culture and the intention variable is discussed in the next section.

5.9.4 Effects of Job Role on Intention (INT)

The hypothesis to be tested in this section is as follows:

> **H13: The interaction between DevOps collaboration culture and INT is moderated by job role.**

The hypothesis is divided further into the following sub-hypotheses:

> H13a: The interaction between DevOps collaboration culture and INT is moderated by *Dev* job role.
>
> H13b: The interaction between DevOps collaboration culture and INT is moderated by *Ops* job role.
>
> H13c: The interaction between DevOps collaboration culture and INT is moderated by *D&O* job role.

Results of the regression analysis of the last moderator for determining the effect of job role on INT variable are tabulated in Table 5.21. As in the preceding sections, the model summaries are tabulated in Table 5.21a. The three variable levels showed moderate R^2 values ranging from 0.408 for *Ops* level to the highest R^2 value of 0.598 for *Dev* and 0.472 for *D&O*. Developers are more likely to accept the DevOps culture, followed by *D&O* and then operators. This is not surprising because developers have turned to favour the

Table 5.21 – Job role as a moderator variable for INT.

(a) Model Summary.

Variable	R	R Square	Adjusted R Square	Std. Error of the Estimate
Dev	.767[a]	0.588	0.561	0.42895
Ops	.638[a]	0.408	0.378	0.70124
D&O	.774[a]	0.598	0.588	0.47208

a. Predictors: (Constant), TR, OC, RR, RE

b. Dependent Variable: INT

(b) ANOVA[a,b].

Model	Sum of Squares			df			Mean Square			F			Sig.		
	Dev	*Ops*	*D&O*	*Dev*	*Ops*	*D&O*	*Dev*	*Ops*	*D&O*	*Dev*	*Ops*	*D&O*	*Dev*	*Ops*	*D&O*
Regression	16.049	26.724	52.122	4	4	4	4.012	6.681	13.031	21.806	13.586	58.469	.000c	.000c	.000c
Residual	11.224	38.848	34.989	61	79	157	0.184	0.492	0.223						
Total	27.273	65.571	87.111	65	83	161									

(c) Coefficients[a,b].

Model	Unstandardised Coefficients						Standardized Coefficients								
	B			Std. Error			Beta			t			Sig.		
	Dev	*Ops*	*D&O*	*Dev*	*Ops*	*D&O*	*Dev*	*Ops*	*D&O*	*Dev*	*Ops*	*D&O*	*Dev*	*Ops*	*D&O*
(Constant)	0.612	-1.756	0.612	0.351	0.856	0.351				1.743	-2.050	1.743	0.083	0.044	0.083
OC	0.010	0.069	0.010	0.034	0.081	0.034	0.017	0.084	0.017	0.285	0.855	0.285	0.776	0.395	0.776
RR	-0.031	0.589	-0.031	0.051	0.161	0.051	-0.031	0.375	-0.031	-0.610	3.663	-0.610	0.542	0.000	0.542
RE	0.042	0.138	0.042	0.060	0.101	0.060	0.043	0.145	0.043	0.704	1.370	0.704	0.483	0.174	0.483
TR	1.003	0.747	1.003	0.090	0.218	0.090	0.741	0.330	0.741	11.102	3.423	11.102	0.000	0.001	0.000

SN and PBC-I interactions, which gives opportunities to learn and transfer knowledge and skills. Following DevOps culture will promote collaboration which is supported by the developers' group.

All the contributions to the regression model were statistically significant with

$p = 0.000$ for all the levels. The F-values were found to be 21.806, 13.586, and 58.469 for -0.702 *Dev*, *Ops*, and *D&O*, respectively (See Table 5.21b). An analysis of these values of p has led to the conclusion that the three sub-hypotheses are acceptable and thus confirm the main hypotheses.

The beta values listed in Table 5.21c have led to the following conclusions regarding the significant associations:

- RR: *Ops* ($p = 0.00$, $\beta = 0.375$); and
- TR: *Dev* ($p = 0.00$, $\beta = 0.741$), $Ops(p = 0.001, \beta = 0.330)$, and $D\&O(p = 0.000, \beta = 0.741)$.

Note that only the significant associations are reported. Interestingly, OC was not significant for all the levels and only TR was significant at all levels, of which *Dev* and *D&O* were significant on this predictor (TR) only. *Ops*, on the other hand, was also significant on the RR predictor. In summary, it can be concluded that the regression model presented above is supportable. The three sub-hypotheses are also acceptable and the research hypothesis is therefore accepted. However, it should be borne in mind that not all the elements of DevOps collaboration culture contributed to the model.

The results presented in this section showed that developers are more likely to accept DevOps culture that operators. The reason may be because they are interested in improving their developer skills and that is facilitated by collaboration environment that DevOps will bring. Uden and Dix (2004, 102) acknowledges that software developers are expected by employers to engage in a life long learning during their careers. To the knowledge of the researcher, there is no study that compares these groups that exist to help explain these results.

5.10 Chapter summary

In this chapter, the demographics of the respondents who participated in this study were discussed. These demographics included the age groups, gender, job roles and the experience of the participants. Raw quantitative data collected using a questionnaire was processed using various statistical techniques with the view to eliminate any discrepancies in the collected data.

The research instrument was tested for reliability using the Cronbach's Alpha technique. Construct validity was tested based on cumulative percentage, Eugen values and significance of the scree plot. Exploratory factor analysis, using principal factor analysis as the rotation method, was used to validate the collected data before it was subjected to further analysis.

The collected data was further analysed using correlation analysis to test the strength relationships existing between different constructs. A Pearson correlation analysis was conducted in this case. Regression analysis was used for modelling and hypotheses testing. The original proposed framework was refined, and a new framework was presented based on hierarchical regression analysis.

Furthermore, the framework was tested for the influence of moderator variables, particularly the job role variable. A more detailed framework emerged and was presented.

In the next chapter, an interpretation of the qualitative data derived from the interviews is presented. The results obtained in this chapter were investigated further so that more clarity can be obtained for a better understanding of these results.

Chapter 6

Qualitative Data Analysis and Interpretation of Findings

6.1 Introduction

Details about quantitative data analysis and the results thereof were presented in the previous chapter. The quantitative data was analysed using the statistical software package SPSS ver. 25. Different techniques were used to transform the raw data into useful information that can add value to the knowledge base in the DevOps field.

In this chapter, qualitative data was used to give more meaning to the information obtained in the previous chapter. It should be borne in mind that this research was a sequential mixed method study in which quantitative results are further explained using qualitative methods to gain a better understanding of the conclusions made. Furthermore, the main themes that emerge from the transcriptions of in-depth interviews were identified to produce meaning to the findings of the quantitative study reported in Chapter 5. Using qualitative content analysis of the responses from the respondents was carried and the corresponding findings are reported in this chapter.

6.2 Qualitative Data Analysis

There are several techniques that can be employed analysing qualitative data. These techniques include the following (Byrne, 2002, n.p.):

Narrative analysis: this method takes narratives—ways in which people express and communicate their experiences—as the object of the research. It looks at the content of the narrative (a story), usually in the form of a textual transcript and how this narrative is organised.

Discourse analysis: the method is about how people use language in communication. Similar to narrative analysis, discourse analysis also focuses on the content and the form. The core focus of this analysis is language use.

Conversation analysis: whilst discourse analysis looks at any form of discourse, the interest of conversational analysis lies in processes of interpersonal communication between individuals. This form of analysis does not include written text, although it focuses on verbal and non-verbal interactions. It looks at ways that people use to produce and interpret social interactions.

Visual analysis: this analysis deals with images as the object of the research. Its concerns represents the content of the images being studied, the arrangements of elements in them, the processes used to produce the image and the social context around their production.

Content analysis: this analysis method focuses on any form of communication including images and text. The method essentially looks at the frequencies of certain content and uses coding to generate measurements from the qualitative materials.

Thematic analysis: during the analysis of the qualitative materials, themes are identified and content is categorised. During thematic analysis, preliminary scanning of the materials is completed and followed by the development of thematic categories. The elements are coded into these themes.

Of all the quantitative data analysis methods explained above, thematic and content analysis remain the most relevant methods for this study. Both thematic and content analysis are focussed on communication in which interviews are the source of data for the research study. Vaismoradi et al. (2013, p. 399) are of the view that thematic and content analysis are suitable for a researcher wanting to use a low level of interpretation.

6.2.1 Content analysis

Payne and Payne (2004, p. 51) described context analysis as a method that seeks to demonstrate the meaning of written or visual sources by systematically classifying their content into predefined categories and then quantifying and interpreting the sources. Downe-Wamboldt, 1992 as cited by Hsieh and Shannon (2005, p. 1278) describes the goal of qualitative content analysis as being to provide knowledge and understanding of what is being studied. This goal is in line with that of this qualitative enquiry, which is to explain the findings on the quantitative study as discussed in the previous chapter.

Hsieh and Shannon (2005, p. 1278) mentions the following three approaches to qualitative content analysis:

Conventional analysis is used by the researcher to describe a phenomenon. It is usually used when there is limited information about the object being studied because, instead of relying on pre-existing themes, the researcher is free to explore new themes emerging from the data.

Directed analysis is used by the researcher when there is some pre-existing theory or research about a phenomenon. In such cases, the aim would be to further the understanding of that phenomenon. This is the case for this research study because some initial knowledge was obtained from the quantitative study, and further understanding is needed to support or extend prior findings. Hsieh and Shannon (2005, p. 1281) has explained that the goal of direct analysis is to validate or extend a framework or theory.

Summative content analysis is used by the researcher who wants to understand the contextual use of words or content. The aim is not to infer meaning but to explore usage. Since this study aims to infer meaning and understand the quantitative findings, summative content analysis is not an ideal choice for this research study.

Qualitative content analysis employing a direct approach is the qualitative data analysis technique that is appropriate for the analysis of qualitative interviews for this research study. Themes that were identified and subsequently tested during the quantitative enquiry were validated using this method. The findings of the explanation study are presented below.

6.3 Analysis of Interviews

Six telephone tape-recorded interviews were converted into text format (transcripts). This type of format conversion was carried out to allow for context analysis procedures. Purposive sampling was used; in this case, the participants were representatives of the DevOps managers. The reason for recruiting DevOps managers was to elicit information from the experts whose aim is to get these two departments to work together without taking a side. The researcher took a decision not to interview any member of any of the two teams to avoid the members of any these teams defending their native silo culture by blaming the other party. Therefore, information was elicited from DevOps managers because the collaboration of the two parties is something that is envisioned by DevOps managers. In other words, DevOps managers were viewed to be well positioned to provide a much more sensible explanation of the quantitative results presented in the previous chapter.

Qualitative content analysis involves a set of systematic and transparent procedures for processing data, and enables the valid and reliable inferences to be produced (Zhang & Wildemuth, 2009, p. 3). Although the steps for conducting a qualitative content analysis are dependent on the research goals, making it more flexible, the following steps have been suggested for making this type of analysis much more flexible (Zhang & Wildemuth, 2009, p. 3):

Step 1: Preparing the data.

The data format needs to be transformed into text format. This step is necessary because various forms of data can be analysed using qualitative content analysis. In the context of this research study, all interviews recordings were transcribed accordingly. Transcription is a time-consuming process. In accordance with recommendations of Zhang and Wildemuth (2009, p. 3), only information that is relevant to this research study was transcribed. Mayring (2014, pp. 45-47) give a discussion on several available transcribing techniques. A selective protocol is explained as a technique whereby only specific topics of the interview are transcribed (Mayring, 2014, p. 45); this technique that was used in this research study.

Step 2: Defining the unit of analysis.

This involved finding unit text within the transcribed data that were used during the classification. Mayring (2014, p. 51) has listed the following units, which are important in qualitative content analysis:

- coding unit determines the smallest component of a material that can be assessed and the minimum portion of text to fall within a category;
- context unit determines the maximum text component to fall within a category; and
- recording unit determines which portions are confronted with one system of categories (Mayring, 2014, p. 51). In the deductive category assignments, a recoding unit could be the people interviewed (Mayring, 2014, p. 52).

Zhang and Wildemuth (2009, p. 321) are of the view that qualitative content analysis usually uses individual themes instead of words, sentences and paragraphs. This theme can be as small as a single word and as large as the entire document being analysed. Following this idea, themes were used as a unit of analysis in this research study. All the responses to the same questions were read and important themes were highlighted. This was done in an incremental fashion until all question were covered.

Step 3: Developing categories and a coding scheme.

According to Zhang and Wildemuth (2009, p. 3) these categories and coding skills can be generated from the transcribed text and related studies from past and existing theories. The preliminary framework of this study was used to generate the following coding list:

- Open Communication (OC)
- Role/Responsibilities (RR)
- Respect (RE)
- Trust (TR)
- Social Norm (SN)
- Organisational usefulness (OU)
- Perceived behavioural control (PBC-I)

These categories were entered into ATLAS.ti software, which was used to assist with the content analysis.

Step 4–6: Testing the coding scheme on sample text, code all the text and assessing the coding consistency.

This is done to validate the coding scheme early in the process. With all the themes highlighted in the transcripts, one transcript was selected for testing purposes. A transcription from one participant was selected and coded into the scheme on ATLAS.ti. After the coding was done, it was realised that the scheme was adequate for the rest of the respondents to be included in the scheme. The remaining responses were therefore entered into the software for analysis. The consistency of the coding scheme was rechecked after the coding of all the data.

The following coding scheme was produced:

- Open Communication (OC)
  - Knowledge sharing
  - Learning
  - Feedback/Status updates
- Roles/Responsibilities (RR)
  - Understand the contribution of the team member
  - Accountability
  - Clarity
  - alignment
  - Expectations
- Respect (RE)
  - Sense of belonging
  - Valued contributions

- Productivity/creativity
- Reduce conflict

- Trust (TR)
  - Expectations/People's skills and abilities
  - Decision making
  - Productivity
- Social Norm (SN)
  - Compliance
- Organisational usefulness (OU)
  - Boost productivity
  - Innovation
  - Flexibility
  - Service quality
- Perceived Behavioural Control-Internal (PBC-I)
  - Diversity
  - Enhance certain skills/communication/thinking/conflict resolution

Step 7–8: Drawing conclusions from the coded data and reporting the findings

When the researcher was satisfied that the coding was done correctly, attempts were made to make sense out of the themes identified. The requisite findings are discussed in the section that follows.

6.3.1 Open communication

The results of the regression analysis presented in Section 5.7.1 indicated that OC is the one DevOps culture factor that is important in a DevOps environment. To gain an understanding of what makes OC, one of the key elements of DevOps culture, the following question was posed to the DevOps managers during the interviews:

How important is the communication between DevOps developers and operators for the success of the project?

The results obtained shows that communication between the two groups is important in that it gives a supporting component of the DevOps culture, which requires collaboration. The interviewed participants indicated that, when open communication is supported, it benefits the teams by allowing them to share ideas and knowledge. Knowledge transfer can easily occur when people with the knowledge are willing to transfer their knowledge, when given the platform to do so. The novice can also acquire the knowledge from the experienced members, assuming that they are free and are encouraged to request this knowledge when required. This proposition is supported by the study conducted by de Vries et al. (2006), which suggests that the communication styles influence the willingness and eagerness to share knowledge between team members.

The second aspect of OC is that it allows team members to stay updated about the project. This is important because such an approach enables effective feedback about the project status to be given to the DevOps team thus allowing the team to timely resolve any issues that may delay the project.

Such type of delays can be mitigated when every team member is encouraged to speak out on any issues without the fear of being intimidated so that they can be supported as and when required.

Allowing every team member an opportunity to communicate freely with the rest of the team gives the team an opportunity to appreciate each other better. Taking turns to speak during meetings gives a platform for team members to know the roles and the responsibilities of everyone involved. Ultimately, such an approach will make each team member to understand the roles and responsibilities of other members. When these roles are understood, the value of every role—and the person performing that role—will receive more respect from other team members and thus instil trust amongst the team members.

6.3.2 Roles/Responsibilities

The second question that was posed to the interviewees was:

How important is the clear role responsibilities between DevOps developers and operators for the success of the project?

The main theme that emerged from the responses is an understanding of every team member's contribution to the project. It is important to appreciate and understand the value of every team member's contribution. When these are clearly defined, it makes it easier for the team to work together knowing precisely who does what and thus eliminate the unnecessary duplication of tasks which may result in conflict arising from accountability functions. Any issues arising during the project have a higher chance of being resolved as quickly as possible when the responsible member is instantly alerted without any undue delay. Of course, the success rate is much higher when the role and responsibilities are clearly defined.

Although the role and responsibility alignment forms part of the DevOps collaboration culture, the qualitative study was found not to significantly influence INT. RR may however still play a significant role in improving performance as suggested by Hoda et al. (2013).

6.3.3 Respect

The third question that was posed to respondents was:

How important is the respect between DevOps developers and operators for the success of the project?

This question was intended to establish the role played by respect in the relationship between the team members. During this investigation, it was revealed that when team members feel respected, in other words, when their presence in the team is valued, they are more likely to feel welcomed and needed and thus contribute positively to the team. Such sense of belonging allows the team members to be much more creative without the fear of being sidelined by the team. This phenomenon is in line with the DevOps collaboration culture that encourages the participation of all team members. A study by Ellemers et al. (2013) has revealed that the action readiness of a team is determined by two aspects, namely: the willingness of the team member to invest or contribute to the team and the positive team identity. Whereas the willingness to contribute is influenced by how a member is valued in the teams, team identity is affected by how a member values the team.

When all the members participate enthusiastically g, the productivity of the team is boosted. An increase in productivity can be regarded as another supporting factor for team respect. It is evident from a study conducted by Carmeli et al. (2015) that respect in a team is an engine for new ideas and it promotes productivity.

With that said, it is clear that respect can be a team breaker if not given consideration by team leaders. This is because respect encourages willingness, which in the case of this research study could be regarded as intentions. To build respect in teams, it is important that every member is allowed to communicate freely with other team members. This type of communication is also bound to instil trust amongst team members.

6.3.4 Trust

The relative question was:

How important is the trust between DevOps developers and operators for the success of the project?

For the project to be successful, team members need to trust each other to allow them to deliver on their promises. The rest of the team will have to rely on the skills and abilities of the team members. Therefore, it is important for team members to be reliable. Being reliable is an important element of collaboration because certain aspects of collaborative work is delegated to a particular group. This makes the team to rely on each other for the success of whatever collaborative work that is being undertaken. Productivity is therefore enhanced because support is given to the members who need it, which substantially engenders this trust. A study by Jong et al. (2016), which investigated the effect of trust on team performance, has revealed that trust has a positive influence on team performance thus confirming the link between productivity and trust.

Decisions can be taken reliably without delaying the project schedule when there is trust amongst the members and decision makers. In a good collaborating team, it should be made clear who should take decisions and how others will be affected because their roles and skills are known and trusted.

Communication, respect and trust to go together when working in teams. When a member receives respect from fellow team members, communication becomes the norm and this facilitates working together and building trust amongst team players. When there is mutual trust within a team, respect becomes effortless and team communication improves. Lastly, when a member is free to communicate openly in a team, trust is built as team members begin to appreciate and respect each other better. In the context of collaboration, knowing the responsibilities of every team player completes the circle collaboration, which is envisioned by the DevOps movement.

6.4 Social Norm

To get clarity on the social norm, the following question was posed to the interviewees:

How important is the influence of DevOps team members in the acceptance of DevOps?

In their response, the participant indicated that the power of the group should not be underestimated. These groups can use their power to influence the behaviours of team members by providing information on how a team member is expected to behave within that particular group. On a positive note, the influence of the group can exert pressure which may encourage other members to comply with the goals and the vision of the team. Those with power and influence obtained through their personal characteristics and positions they occupy in the group, may influence the behaviour of other team members, and ultimately the entire team.

It is therefore apparent that positive influences may lead to positive intentions on the entire team. However, the power to influence may depend on the influencer. Eckhardt et al. (2009*b*) has established that there is a difference in the power of the influence from different groups of influencers (Superiors, Peers and IT department) when adopting IT by the HR department. Superiors were found to possess more power followed by peers, whilst the IT department showed the weakest power of influence of the three groups. This notion was corroborated by a similar study by Brown et al. (2010) which investigated the use of collaboration technology supported; superiors and peers were found to possess a significant power to influence others.

Although social norm appears to be crucial as people influence one another, the quantitative findings of this study were not significant. SN was found not to have a significant influence on INT. This may be explained by a weak contribution of the IT department on HR department, which was observed in the study of Eckhardt et al. (2009*b*). A study by Hardgrave and Johnson (2003, p. 330) also found that SN did not statistically influence INT when studying the acceptance of information system development processes by software developers.

6.5 Organisational Usefulness (OU)

To investigate the results obtained for OU during the quantitative study, the following question was answered:

How does the collaboration between Devs and Ops affect the success of the organisation?

The interviews revealed that the collaboration between the two teams is notable to the organisation for several reasons. The first reason relates to an increase in productivity resulting from the workload being appropriately shared amongst team members. This distribution of the workload can be managed according to the skills and strengths of team members. Several studies (Melo et al., 2013;Schuh et al., 2014;Parker et al., 2015) have shown how productivity can be boosted using teamwork.

The second reason is that collaboration promotes innovation, which in turn benefits the organisation. A creative environment is fostered when team membranes bounce ideas off each other to devise a solution. During such brainstorming sessions, unique and creative ideas emerge and are then adopted for the benefit of the company. The third reason identified is the flexibility brought by the collaboration environment, which allows skills transfer between members. Disruptions are weakened as a result of a team member pulling off the project. This, in turn, boosts the morale of the team members and lowers the employee turnover rate, and ultimately benefiting the organisation. The last reason is the service and/or product quality. Since everyone is involved from the beginning of the development of the product to the end of the cycle, the evaluation of that product is not left to the end of the product development cycle. The product evaluation is continued from start to finish. This also promotes a sense of ownership of and accountability to the final product on the part of the team members.

The quantitative study revealed that OU was statistically significant in the determination on INT. This corroborated the results of Hardgrave and Johnson (2003).

6.6 Perceived Behavioural Control-Internal (PBC-I)

The last question was:

How do one's skills and abilities affect the success of DevOps projects?

Working in collaboration allows exposure to diverse points of view, which benefit the team and individuals working on that project. The sharing of viewpoints, skills, and knowledge stands to build more effective teams with capable team players. Therefore, collaboration enhances learning from the team, which can help junior personnel to rapidly acquire the necessary technical skills in a short space of time. When surveyed by Lindsjørn et al. (2016), Agile software development teams were of the view that collaboration facilitated their learning in an optimistic way. Other soft skills that are enhanced by collaboration teamwork are communication, critical thinking, and conflict resolution skills. Communication is central to the DevOps collaboration culture, therefore, intensifying this skill—by encouraging team collaboration—will strengthen the DevOps culture. Mastering the art of collaboration will also improve the critical thinking and conflict resolution skills of the team members.

With all that said, it can be concluded that (PBC-I) is a factor that is necessary for the acceptance of DevOps collaboration culture. Simply put, an understanding of the benefits that will be accrued from the adoption of the DevOps collaboration culture will in all likelihood lead to acceptance of the change of traditional organisational culture to the DevOps culture. The acceptance of DevOps culture will not only benefit the adopting organisation, but also the individuals who practice it.

Similar to SN, however, the empirical results of the quantitative study as reported in Chapter 5 did not produce enough compelling evidence to statistically conclude that PBC-I significantly influences INT. These results are in agreement with those published by Hardgrave and Johnson (2003), which involved an investigation of information system development methodology acceptance.

6.7 Chapter Summary

The aim of this chapter was to provide an explanation of the results obtained for the quantitative data analysis presented in the previous chapter. Qualitative data gathered through the use of telephonic interviews were analysed following qualitative data analysis techniques. Various qualitative techniques were discussed in this chapter and the techniques that were deemed suitable for this study were identified.

The technique that was used to analyse the qualitative data was qualitative content analysis. The procedures for conducting content analysis were chronicled and adopted for the analysis of the data. The analysis revealed that open communication is important in DevOps collaboration culture because it fosters the culture of knowledge sharing, which is important in DevOps because the developers need to share knowledge with the operators and *vice versa*. Communication also facilitates learning occurring amongst the members. Feedback and project status update should also be communicated to the entire team on a regular basis as they will mitigate project delays as it allows problems to be communicated early and dealt with accordingly.

Clarity of roles and responsibilities contribute in assisting the team to appreciate the contributions that every team member is making to the project. This also assists team members to know what the team expects from them. The workload can be distributed accordingly, and it makes accountability for tasks to be aligned with the rightful owner.

It was also highlighted that people are more willing to contribute to teamwork when they feel that they are part of that team. Therefore, respect plays a role in making people feel welcomed in the team. In other words, productivity is increased when people are shown respect by team members because then they are willing to contribute positively to the team and avoid conflicting situations.

Also, rusting fellow team members with their skills and abilities could potentially boost productivity. Productivity stands to increase when team members deliver on what is expected of them and this makes decision making easier because team members trust that decisions are taken by the knowledgeable people who have the success of the project in their hearts.

Social norm has also shown that people have the power to influence each other. On a positive note, this influence could lead to compliance with respect to the team's obligation to the organisation. To the organisation, the usefulness of DevOps is important because it may lead to an increase in productivity and product quality. DevOps encourages innovation whilst maintaining the flexibility of the team by allowing knowledge sharing and skills transfer in a collaboration environment. Not only does this sharing and transfer of knowledge and skills benefit the organisation, team members are also supported in terms of upgrading their technical and soft skills. Thus, the perceived behavioural control-internal is supported by the DevOps collaboration culture.

Chapter 7

Discussion of Results

7.1 Introduction

In this chapter a discussion and interpretations of the research findings of results that were presented in Chapter 5 and 6 are presented. The hypotheses findings are discussed as relevant to the framework of this study. In addition, the research questions that this study has attempted to answer are assessed on the basis of the achievement of the research objectives.

The main research question for this study was geared towards identifying factors that needed to be included in a framework to support a successful DevOps collaboration environment. This study was motivated by DevOps being a new software development strategy that organisations are adopting due to its wide ranging benefits. A review of the literature indicated an absence of a framework that organisations can follow to guide the successful adoption and of DevOps and its concomitant potential benefits.

The study was conducted in South Africa. In a quest to address the aim of this research project, research sub-questions were posed and their answers were presented in Section 7.3. The remaining layout of this chapter is as follows: Section 7.2 provides a detailed discussion on the research findings in line with the hypotheses evaluated. The hypotheses in relation to moderator factors are presented in Section 7.2.2. A constructive discussions on the empirical findings in relation to the research questions are discussed in Section 7.3. The final research framework, which a deliverable of this thesis, is presented in Section 7.3.

7.2 Discussion of the results and findings

The hypotheses required for answering the research questions were explained in Section 3.6.2 and tested in Chapter 5. The summarised outcomes of the tests are tabulated in Table 7.1. The discussion and interpretation of these research findings are presented in the following section.

Table 7.1 – Summarised research findings from hypothesis.

Proposed Hypotheses		**Results**
H1:	**There is a positive relationship between DevOps culture and intention to use DevOps.**	**Partially Supported**
H1a:	There is a positive relationship between open communication and intention to follow DevOps processes.	Supported
H1b:	There is a positive relationship between role and responsibility alignment and intention to follow DevOps processes.	Not Supported
H1c:	There is a positive relationship between trust and intention to follow DevOps processes.	Supported
H1d:	There is a positive relationship between respect and intention to follow DevOps processes.	Supported

Table 7.1 – Summarised research findings from hypothesis.

Proposed Hypotheses		Results
H2:	There is a positive relationship between SN and INT.	Not supported
H3:	There is a positive relationship between OU and INT.	Supported
H4:	There is a positive relationship between PBC-I and INT.	Not supported
H5:	**There is a positive relationship between culture and individual subjective norm (SN).**	**Supported**
H5a:	There is a positive relationship between open communication and SN.	Supported
H5b:	There is a positive relationship between role and responsibility alignment and SN.	Supported
H5c:	There is a positive relationship between trust and SN.	Supported
H5d:	There is a positive relationship between respect and SN.	Supported
H6	**There is a positive relationship between culture and OU.**	**Partially Supported**
H6a:	There is a positive relationship between open communication and OU.	Supported
H6b:	There is a positive relationship between role and responsibility alignment and OU.	Not Supported
H6c:	There is a positive relationship between trust and OU.	Supported
H6d:	There is a positive relationship between respect and OU	Supported
H7:	**There is a positive relationship between culture and PBC-I.**	**Partially Supported**

Table 7.1 – Summarised research findings from hypothesis.

Proposed Hypotheses		Results
H7a:	There is a positive relationship between open communication and PBC-I.	Not Supported
H7b:	There is a positive relationship between role and responsibility alignment and PBC-I.	Supported
H7c:	There is a positive relationship between trust and PBC-I.	Supported
H7d:	There is a positive relationship between respect and PBC	Not Supported
H8:	There is a positive relationship between PBC-I and OU	Supported
H9:	There is a positive relationship between SN and OU.	Supported

7.2.1 Discussion and interpretation of the hypotheses

Hypotheses were formulated based on the following DevOps collaboration culture elements:

- Openness Communication (OC);
- Roles /Responsibilities (RR);
- Respects (RE); and
- Trust (TR).

It was hypothesised that these elements would influence the behavioural intention (INT). In addition to these elements, the following elements were also hypothesised to be the determinants of behavioural intention (INT):

- Subjective Norm (SN);
- Organisational Usefulness (OU); and
- Perceived Behavioural Control-Internal (PBC-I).

In the sub-sections that follow, factors that did not show any influence on INT are discussed. This is followed immediately by a discussion on those factors that were found not to influence INT.

7.2.1.1 Factors that fail to influence Behavioural Intention

SN was found to have no insignificant influence on behavioural intention (H5). This was in contradiction with the expectation that social influence has a positive effect on INT. This means that superiors and peers may have an influence on other behavioural intentions or actions, but certainly not on the acceptance of DevOps culture. These findings are in line with those reported by Hardgrave and Johnson (2003) on acceptance of information system development. On the contrary, a study conducted by Guardado (2012) on the acceptance and adoption of processes by software practitioners revealed that social influence was a significant determinant for process adoption and acceptance.

Collaborative environments encourage social interactions that are trustful and respectful. When a team member is groomed in such environments, it is likely that that member's behaviour will be influenced by the superiors and peers, and hence, that member's intentions are more likely to be team intentions. Perhaps the timing of this study falls within a period where acceptance is at early stages in terms of the Diffusion of Innovation theory (Rogers, 2003, p. 35), where there are not enough adopters to motivate and influence late adopters.

Similar to PBC-I, when looking at intention influencers, it was established in this study that PBC-I has no influence on INT (H6). Similar findings on acceptance of information system development methodology were reported by Hardgrave and Johnson (2003).

The role and responsibilities did not confirm any significant acceptance of DevOps culture (H1b). This means that the role a person occupies does not have a significant influence on whether or not, to accept the DevOps culture. Although it is suspected that certain roles would encourage specific behaviour, the results of this study did not reveal that at this stage.

7.2.1.2 Factors influencing behavioural Intention (INT)

The findings of this study indicate that, save for RR, all the elements of DevOps collaboration culture have a positive effect on the behavioural intention to accept of DevOps culture (H1 partially supported in Table 7.1). These findings complement that of Ben Othman et al. (2016) on their study of the organisational culture effects on the acceptance of Agile methodology, which revealed that group culture played a role in the acceptance of Agile methodologies. This group culture is human relations centred and promotes flexibility, trust and encourages participation Ben Othman et al. (2016, p. 18), which is equivalent to DevOps collaboration culture. On another study in a similar context, Strode et al. (2009) found that an organisational culture which values, amongst other things, feedback and learning, trustful social interactions and collaboration, leads to the acceptance of Agile methodology.

Furthermore, the study by Strode et al. (2009) revealed that OU also has an influence on intentions (H6). People are more likely to carry on and perform the activities that will benefit their organisation. In other words, those activities are necessary for the survival of the organisation. The results of the study by Hardgrave and Johnson (2003) concur with those of the current as they show that OU is a significant determinant of intentions on their study on the acceptance of information system development processes. Following an investigation of user acceptance of IS project management methodologies, Mohan and Ahlemann (2011) reported that OU, which is referred to in the study as a task-related utilitarian value, provides instrumental value to the user (e.g. increasing performance, efficiency and productivity) (Mohan & Ahlemann, 2011, p. 915).

It is evident that DevOps collaboration culture determinants (i.e. open communication, respect and trust) are important the determinants of the intention to follow DevOps. It is therefore crucial for the DevOps adopters to invest in these elements for the successful acceptance of DevOps and thus observe the benefits of DevOps in their organisations. In addition to the culture, social influence, organisational usefulness, and perceived behavioural control-internal are also important for the acceptance of DevOps. As already mentioned, superiors and peers are great influencers of behavioural intentions, and organisations should educate their staff members on the importance of adopting DevOps in the organisation (OU) and how the use of DevOps will benefit them (PBC-I).

7.2.1.3 Factors influencing Social Norm (SN)

It has been iterated in this study how collaborative environments foster communications, create a supportive environment, and encourage knowledge sharing and learning, to name a few. Trust and respect feature prominently in collaborative environments, and this suggests that social influence plays a significant role in such environments. Results generated from this study also suggest that the DevOps collaboration culture (OC, RR, RE and TR) has an effect on the social norm; this notion is supported by hypothesis H2. These findings corroborate the findings of Eckhardt et al. (2009*b*) and Brown et al. (2010) as mentioned in Section 6.4.

Since SN is the determinant of OU, it is important that DevOps managers nourish this collaboration culture amongst team members. This can be achieved by offering training to team members Hardgrave and Johnson (2003, p. 334) , especially on DevOps collaboration elements. Other than encouraging open communication between members without them feeling intimidated, team members should show respect towards each other and strife not to be judgemental (Walls, 2013, p. 15). The team members should support one another in order to create a stimulating working environment.

7.2.1.4 Factors influencing Perceived Behavioural Control-Internal (PBC-I)

The hypothesis H7 was partially supported by RR (H7b) and TR (H7c) whilst OC (H7a) and RE (H7d) were rejected. The results indicate that roles and responsibilities influence PBC-I. It is noteworthy that PBC-I is the resources that a person possesses internally that will allow that person to successfully carry out the duties of that person. In other words, these are skills a member is equipped with to perform the role and responsibilities of that particular member. It is therefore understandable that a person must possess specific skills for that person to take up a specific role. With that said, it is not surprising that RR influences PBC-I. Holding specific roles in a team may boost the morale and motivation of the person occupying that role to acquire more skills for better recognition in terms of being promoted. Similarly, TR, which was measured in terms of skills possessed and ability to perform duties, influences the control a person has.

Communication and respect were found not to have a significant influence on PBC-I. A possible explanation for this is that both elements do not require skills for controlling a certain behaviour. In other words, these elements are unconditional. To illustrate this phenomenon by way of an example, a person can be trusted on a certain skill that that person possesses. However, a collaboration culture requires that team member be respected, irrespective of their roles or the skills they possess. The same notion applies to communication.

PBC-I as a determinant of OU should also be taken into consideration. Since PBC-I is influenced by RR and TR, managers should take measures to clearly define role and responsibilities whilst instilling a culture of trust amongst team members (Hardgrave & Johnson, 2003, p. 334). Clear roles and responsibilities and promoting team trust will lead to high PBC-I (Walls, 2013, p. 15), which in turn improve the acceptance of the DevOps culture.

7.2.1.5 Factors influencing Organisational Usefulness (OU)

Save for RR (H6b), DevOps collaboration culture elements have a positive effect on OU and they were found to be significant (H6a, H6c, and H6d). These results suggest that OU is highly dependent on OC, TR and RE. It is therefore very important for a DevOps organisation to allow members of the DevOps team to communicate openly, trust each other and show respect to fellow team members (de Vries et al., 2006;Carmeli et al., 2015; Jong et al., 2016). By creating such an enabling environment, the DevOps organisation will be able to promote the sharing of knowledge and the transfer of skills amongst team members. Ultimately, such an approach will benefit the DevOps organisation thus resulting in increased productivity without compromising product quality, stimulating innovation and maintaining team flexibility whilst keeping staff turnover at minimal rates.

The findings of this study also showed that OU is dependent on SN and PBC-I (H8 and H9). DevOps influencers, managers and peers, should be willing to support the new culture (DevOps) in an organisation. Thus, the effects of SN on OU will be noticed. These results are in agreement with those reported by Hardgrave and Johnson (2003). Similarly, for the effect of PBC-I on OU, training, mentoring and support of academic programmes should be implemented to improve PBC-I and therefore, its effect (Hardgrave & Johnson, 2003, p. 332).

7.2.2 Discussion of the hypotheses in relation to moderating factors

Hypotheses H10 to H13 (Tabulated in Table 7.2) illustrates the effect of the moderator on the hypothesised relationships. The job role is the moderator variable that these hypotheses relate to. Three levels of this moderator variable were examined, namely: *Dev* representing the developers, *Ops* repres-

Table 7.2 – Moderator hypothesises summaries.

No.	Hypothesis	Acceptance
H10	**The interaction between DevOps collaboration culture and SN is moderated by job role.**	Accepted
H10a:	The interaction between DevOps collaboration culture and SN is moderated by *Dev* job role.	Accepted
H10b:	The interaction between DevOps collaboration culture and SN is moderated by *Ops* job role.	Accepted
H10c:	The interaction between DevOps collaboration culture and SN is moderated by *D&O* job role.	Accepted
H11:	**The interaction between DevOps collaboration culture and OU is moderated by job role.**	Accepted
H11a:	The interaction between DevOps collaboration culture and OU is moderated by *Dev* job role.	Accepted
H11b:	The interaction between DevOps collaboration culture and OU is moderated by *Ops* job role.	Accepted
H11c:	The interaction between DevOps collaboration culture and OU is moderated by *D&O* job role.	Accepted
H12:	**The interaction between DevOps collaboration culture and PBC-I is moderated by job role.**	Partially supported
H12a:	The interaction between DevOps collaboration culture and PBC-I is moderated by *Dev* job role.	Accepted
H12b:	The interaction between DevOps collaboration culture and PBC-I is moderated by *Ops* job role.	Rejected
H12c:	The interaction between DevOps collaboration culture and PBC-I is moderated by *D&O* job role.	Accepted
H13:	**The interaction between DevOps collaboration culture and INT is moderated by job role.**	Accepted
H13a:	The interaction between DevOps collaboration culture and INT is moderated by *Dev* job role.	Accepted
H13b:	The interaction between DevOps collaboration culture and INT is moderated by *Ops* job role.	Accepted
H13c:	The interaction between DevOps collaboration culture and INT is moderated by *D&O* job role.	Accepted

enting the operators, and *D&O* representing job role that combines both the *Dev* and *Ops* roles.

H10 stated that the interaction between DevOps collaboration culture and

SN is moderated by job role. The test results for this hypothesis and its sub-hypotheses were all supported. This implies that whilst a relationship exists between DevOps collaboration culture and SN, this association is influenced by the job role. In other words, this association is dependent on whether the participant (the person that is being measured) is a developer, operator, or performs both roles. This is similar to the interaction between DevOps collaboration culture and OU, and INT (H11 and H13). In other words, the influence of DevOps culture on OU or INT is affected by whether the participant performs a *Dev*, *Ops*, or *D&O* role. Hence, H11, H13 and their sub-hypotheses were all supported.

H12 was partially supported. It stated that the interaction between DevOps collaboration culture and PBC-I is moderated by job role. During the sub-hypotheses testing, H12b was found to be insignificant and was soundly rejected. The assessment of *Ops* as a moderating job role did not produce enough evidence to suggest that *Ops* participants can affect the relationship between DevOps collaboration culture and PBC-I.

7.2.3 Conclusions Related to the Hypotheses

Results of this study confirm some of the previous findings by several researchers which suggest that DevOps collaboration culture (open communication, respect and trust), and organisational usability, are key success factors and have an influence on intention to follow DevOps processes. These factors should be included in the final research framework of this study. Except for RR, the DevOps collaboration culture elements in this study support the assertion of building a DevOps culture by Walls (2013). Furthermore, this research study highlighted that job role factors influence other variables in the research framework, indicating that the *Dev* role is more likely to have a positive orientation towards the key success factors influencing DevOps acceptance.

7.3 Findings and discussions relating to the research questions

The main research question that was asked in this study was: *What are the factors that must be included in a coherent framework to support a successful collaborative culture in DevOps environment?*

The results generated from this study have revealed the following supporting factors:

- DevOps collaboration factors in which OC, RE and TR were found to be significantly influencing the intentions for the acceptance of DevOps culture. Respect (H1d; $\beta = 0.500$; $p < 0.000$) was found to have the most influence, followed by open communication (H1a; $\beta = 0.305$; $p < 0.000$) and then the trust (H1c; $\beta = 0.214$; $p < 0.000$).
- Organisational usefulness factors (H1; $\beta = 0.168$; $p < 0.005$) are also the relevant success factors that influence a successful DevOps culture. Therefore, OU should also constitute an essential part of the framework.

In addition to the aforementioned factors, in terms of the influence on OU, the following factors were found to be relevant and possess the influence:

- DevOps collaboration factors, in which OC, RE and TR were found to have a significant influence. As in the influencers of INT, respect (H6d; $\beta = 0.563$; $p < 0.001$) appeared to have the most influence, followed by open communication (H6a; $\beta = 0.533$; $p < 0.001$) and trust (H6c; $\beta = 0.137$; $p < 0.001$).

- Social norm factors (H9; $\beta = -0.333$; $p < 0.001$) are the success factors that indirectly influence INT by affecting OU, which in turn directly influences INT.

- Perceived behavioural control-Internal factors (H8; $\beta = 0.205$; $p < 0.001$) are similar to social norm factors mentioned above in that they have an indirect influence INT via OU.

In terms of SN, the following factors were found to have a significant influence.

- All the DevOps collaboration culture factors, that is OC (H5a; $\beta = 0.387$; $p < 0.001$), RR (H5b; $\beta = 0.553$; $p < 0.001$), RE (H5c; $\beta = 0.142$; $p < 0.005$), and TR (H5d; $\beta = 0.241$; $p < 0.001$) have an influence on SN.

Lastly, in terms of PBC-I, the following factors were found to have a significant influence.

- DevOps collaboration culture factors in which RR (H7b; $\beta = 0.449$; $p < 0.001$), and TR (H7c; $\beta = 0.734$; $p < 0.001$) remained the only influential factors.

This study has answered the main research question that was posed at the beginning of the thesis. The factors indicated above are the key factors that should be included in the framework. Research sub-questions were used to assist in the answering of the main question. These sub-questions were:

SQ_1: *What are the characteristics of DevOps collaborative culture?* This question was answered in Chapter 2 of this thesis.

SQ_2: *What factors encourage a collaborative culture in a DevOps environment?* This question was answered in Chapter 3, 5 and 6 of this thesis.

SQ_3: *How do the factors interact with each other?* This question was answered in Chapter 5 and 6 of the thesis.

SQ_4: *How can a collaborative culture be implemented in DevOps environment?* This question was answered in Chapter 6 and 7 of this thesis.

SQ_5: *Does the proposed framework supports the collaborative culture in DevOps environment?* This question was answered in Chapter 6 and 7 of this thesis.

According to the results generated from this research study, the most influential factors with regards to DevOps culture relate to the influence of DevOps collaboration culture. This indicates that DevOps adopters, who are typically managers, should ensure that human factors are taken into consideration for the success of the implementation of DevOps within their organisations. The managers should provide the necessary support and provide training to workers (Hardgrave & Johnson, 2003, p. 332), especially training that is related to collaboration practices. Employees should be encouraged to collaborate with one another by promoting activities that will positively enhance this cultural practice. Open communication, respect and trust should be built and maintained throughout the organisation as recommended by Walls (2013, p.15).

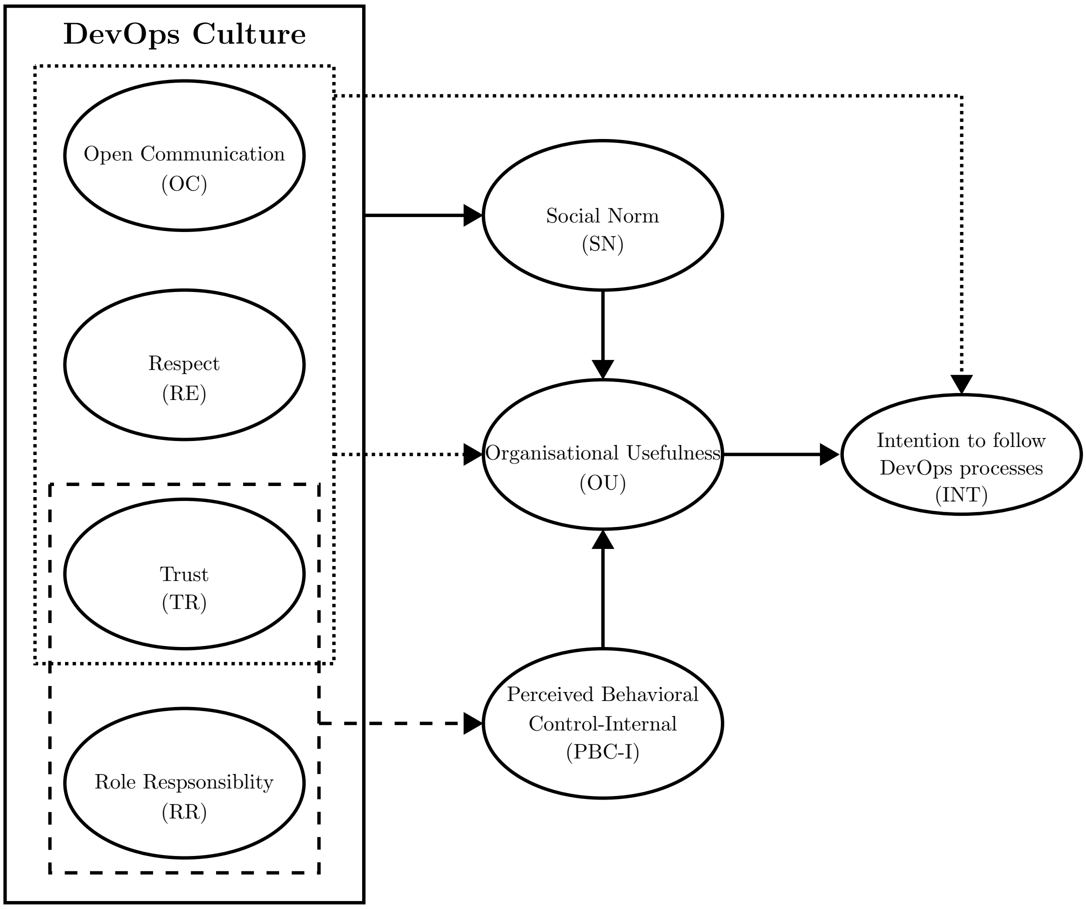


Figure 7.1 – The final framework (DCAM).

7.4 The objective of the study

As stated in Section 1.4, the primary objective of this research study was *to propose a framework that will allow the acceptance of the DevOps collaboration culture within DevOps environments*. This objective was accomplished through the following sub-objectives:

- *To investigate DevOps collaboration requirements.* This objective was achieved in Chapter 2 of this thesis.

- *To determine the necessary factors that promote a successful collaborative culture.* This objective was achieved in Chapter 3 of this thesis.

- *To evaluate the key success factors for their interaction with one another.* This objective was achieved in Chapter 5 of this thesis. This objective was achieved through the test of hypotheses shown in Table 7.1 and 7.2 of this chapter. Table 7.2 shows the hypotheses that tested the moderator effects of job role. The supported and rejected hypotheses are summarised on these tables.

- *To determine how the interaction of the contributing factors contributes to successful collaboration in a DevOps environment.* Chapter 5 and 6 of this thesis achieved this objecti*ve.*

- *To propose a comprehensive framework for a collaborative DevOps acceptance.* This objective was achieved in Chapter 2 through to Chapter 7 of this thesis.

The final product of this thesis is depicted in the form of a framework model in Figure 7.1. The framework can be used as the guideline for software development managers who are planning to transit to the DevOps culture. It can also be used by organisations that have already adopted DevOps and need to improve its implementation.

7.5 Chapter Summary

This chapter provided an interpretation and explanation of the results that were presented in Section 1.4. The implications of these results were also discussed. Specifically, the findings as they relate to the research questions, the hypotheses and the research objectives of this study were reviewed. The chapter has indicated how the research study has arrived at its conclusion.

The following chapter, which is the concluding chapter discussed how this thesis met its objectives. The valuable contributions of this study will make to the various areas are detailed in the next chapter.

Chapter 8

Conclusion and Recommendations

8.1 Introduction

This chapter provides an evaluation of the research study by re-stating the research rationale to the findings of the study. The processes that were followed to reach the outcomes are described and the contributions this study makes to the software engineering body of knowledge and to DevOps philosophy in particularly are articulated. The knowledge generated and recommendations made from in this research study could potentially influence policies of existing and future adopters of DevOps. In addition, research opportunities arising from the limitations on this study are identified and presented future enquiry. The research significance, suitability, methodology, contribution and attainment of the research questions are also explained.

8.2 Summary of the thesis

The purpose of this research was to develop a coherent framework to support a successful collaborative culture in a DevOps environment. The proposed study was found to be significant and relevant in that it would produce guidelines to organisations that are in the process of or have implemented DevOps.

After highlighting in Chapter 1 the importance and relevance of this research study in the field of software engineering and DevOps philosophy, research questions that needed to be answered in order to pave the way for the envisaged framework were identified. A review of these questions is presented in Section 8.3.

The theoretical framework that was adopted in this study was presented in Chapter 3. The models that were used in previous studies to assess how human behavioural intention to use or accept new technologies were explored. These models included TRA, TPB, TAM, UTAUT, DOI, and ISDAM. Furthermore, it was shown that ISDAM is the most appropriate model to build on as a base framework. The reason behind this assertion is that ISDAM was built from a combination of some of these models which were tested in different contexts. Secondly, ISDAM was built and tested in the information system development environment, which is more relevant to this research study. The framework (DCAM) was proposed (Masombuka & Mnkandla, 2018) and was evaluated in subsequent chapters.

The methodology that was used to test and validate the proposed framework was presented in Chapter 4. A mixed-method approach was adopted in which quantitative and qualitative data was collected and analysed was followed as the research design. Web-based questionnaires were used to collect quantitative data. Qualitative data were, on the other hand, collected using structured interviews conducted over the telephone and Skype. Using mixed methods strengthens the reliability and validity of this research study.

The collected data collected using the questionnaire and interviews were analysed using quantitative analysis techniques and qualitative content analysis, respectively. This mixed methods approach was conducted in a sequential manner starting with an initial quantitative investigation being conducted and followed immediately thereafter by the qualitative study. The results of the quantitative and qualitative studies were presented in Chapter 5 and 6, respectively. The results of this research study have revealed that open communication, respect and trust are statistically significant. In addition, organisational usefulness was also found to be a significant contributing factor to the acceptance of the DevOps culture.

8.3 How the research questions were answered

The main research question was formulated as follows:

What are the success factors that must be included in a coherent framework to support a successful collaborative culture in a DevOps environment?

In order to investigate this question, the following five sub-questions were derived:

SQ_1: *What are the characteristics of DevOps collaborative culture?*

SQ_2: *What factors encourage collaborative culture in a DevOps environment?*

SQ_3: *How do the contributing factors interact with each other?*

SQ_4: *How can a collaborative culture be implemented in a DevOps environment?*

SQ_5: *Does the proposed framework supports the collaborative culture in a DevOps environment?*

SQ_1 was answered through a compilation of a literature review in Chapter 2. The reviewed literature revealed four elements that constitute the DevOps collaboration culture, namely open communication, roles and responsibility alignment, respect and trust. Section 2.3.5 gave a detailed explanation of these constituents of DevOps collaboration culture.

SQ_2, SQ_3 and SQ_4 were dealt with in Chapter 5. Statistical techniques were used to determine the identified factors and the relationships between them, which responded to SQ_2, SQ_3 and SQ_4. Details of the techniques that were used to address these research question are provided in Chapter 5.

The last research question was answered in Chapter 6. Following the development of the framework, qualitative content analysis was employed to confirm and explain the framework in Chapter 6.

All the research questions of this research study (i.e. RQ_1 to RQ_5) were answered and the methodology used to arrive at those answers was documented in the relevant sections. It was quite evident that answering all the sub-questions played the dual role of simultaneously answering the main question. Therefore, it can be concluded that this research study has answered all the relevant questions and the main question that drove this study.

8.4 Evaluation of the methodological contribution

During the evaluation of the methodological contribution, the research was tested for the following:

- chosen research methods;
- research theme;
- topic suitability;
- unit of analysis; and
- data collected for analysis

A comprehensive discussion on the above criterion is presented in the section that follows.

8.4.1 Relevance of the used approach

The primary objective of this study was to develop a framework for a successful DevOps collaboration culture. Since DevOps is still fairly new, not much academic work has been conducted in this area. Existing academic literature was used to identify DevOps collaboration elements from which a framework was developed.

Special focus was placed on DevOps collaboration culture in South Africa and DevOps practitioners in this country were, to this end, recruited to participate in this research study. Questionnaire and interview based surveys were used for data collection purposes. Exploratory factor analysis was used to reduce the factors to a tolerable size. Regression analysis was used to explain the influence of the variables concerned. The methods employed were deemed appropriate for this research study and they made a meaningful contribution to knowledge relating to software engineering and DevOps philosophy.

The theoretical framework that was adopted in this study was ISDM. The adoption of the ISDM was effected after assessing a number of several such as TRA, TPB, TAM and UTAUT. ISDM was modelled for software development environments in which the software developers themselves are the actual users of the technology, that is, the information system development processes in the case of this research study. In addition, ISDM is based on a combination of these well-evaluated theories.

Moderating factors in which the job role was evaluated were introduced in this framework. The final framework and the effects of the moderator variable were assessed using regression analysis. Therefore, this empirical analysis of the data provided a methodological contribution to the field.

8.4.2 Research theme of the study

As already mentioned, the objective of this study was to investigate the factors that influence the successful adoption of the DevOps collaboration culture. At the time of conducting this study, this researcher was not aware of any other study of collaboration in this context. With DevOps placing a huge emphasis on collaboration culture, a need exists for a better understanding of the factors that influence a successful DevOps collaboration culture in organisations for efficient adoption and acceptance of this culture. Any changes in culture come with resistance. Such resistance may lead to organisations incurring wasteful expenditure and losing fruitful time due to employees clinging onto their old habitual culture. In light of that, a framework for successful adoption of DevOps collaboration culture is relevant to software engineering and DevOps philosophy.

8.4.3 Fitness of the topic in the knowledge building process

This framework built in this study is informed by the views of DevOps practitioners on the DevOps collaboration culture in South Africa. A web-based questionnaire and telephonic/Skype interviews were used to gather responses from DevOps practitioners on their perceptions on DevOps.

The study was developed around a research framework based on ISDM theory which was, constructed from various existing theories, including the TPB, TRA, and UTAUT. The resulting framework was then used to determine the critical success factors contributing to the successful acceptance of DevOps collaboration culture through an empirical study. Therefore, an understanding of these success factors and recommendation of better strategies to address the impact of these factors on DevOps environments is a major step in the knowledge building of software engineering systems.

8.4.4 Relevance of the unit of analysis

As much as a DevOps collaboration culture is being adopted in many organisations, the success or failure of the adoption of DevOps relies on the team members who need to transit to the new culture. In the context of this study, the decision to analyse the perceptions of DevOps team members who are directly affected by this cultural change was deemed appropriate.

8.4.5 Collected data in relation to needed research findings

In this research study, both secondary and primary data were used. The literature review, which was used to gather the secondary data has proven to be significant because the data were collected from accredited sources such as journals, books and conference proceedings. The articles that were accessed from these sources had previously been peer reviewed by different software engineering, professionals, scholars and experts and were therefore regarded as being trustworthy.

The primary data were collected through interviews and questionnaires conducted with DevOps practitioners. The web based questionnaire was sent to DevOps professionals through social media, and interviews were conducted over the telephone and Skype. Therefore, the data analysed in this research study was found to be appropriate and suitable to generate quality results and findings.

8.5 Importance of the Research

Software engineering research or any other research for that matter, should not only be important to professionals in that field but to other readers who may find the subject interesting, in terms of both style and content. The importance of the actual content of the research is normally evaluated based on the degree to which it engages the reader's curiosity and its potential to create awareness and encourage discussions on that research matter.

8.5.1 Applicability of the Topic and Research Output

The topic of this research study is of great importance to DevOps management and professionals because there are no research studies, at least to the knowledge of this study researchers, that have modelled factors for DevOps collaboration culture. Therefore, this research study has provided an important contribution through the development of a DevOps collaboration culture framework.

8.5.2 Contribution to Discussions on Current matters

Organisations are increasingly adopting and implementing DevOps practices in their business environment. The need for an organisational culture change to a DevOps culture where collaboration is the norm has led to the development of a research framework that is proposed in this research study. Since the adoption of DevOps philosophy is relatively new, especially in South Africa, this research study is therefore timeous and relevant.

8.6 Research Contribution of the Study

In this research study, the aim was to develop and validate a framework that can be used as a guideline for DevOps adopters. This framework was therefore developed as the result of undertaking this research study. Organisations adopting or planning to adopt DevOps can use it as a guideline for successful adoption.

8.6.1 Contributions to DevOps Software Engineering Field

The framework developed in this study was informed by the literature review, tested theoretical frameworks, as well as surveys in the form of a questionnaire and interviews. The conceptual effect of this study is that it showed that ISDM was a suitable underpinning reference model to use when exploring the acceptance of information system development approaches, including DevOps. The ISDM was developed from a combination of relevant theories which were tested in various contexts, and it was evaluated in an information system development environment. The produced framework, DCAM, developed in this study is evident that it is suitable for use in DevOps environments in developing countries context as it has been based on data from South Africa.

To reiterate, this study employed a combination of literature review and surveys (questionnaire and interviews) to identify and analyse the factors for successful DevOps collaboration culture acceptance. This mixed-methods approach that was adopted for this study led to an improvement in the validity and reliability of this study. With that said, the use of qualitative and quantitative approaches to this study signifies a methodological contribution in the software engineering field. To the best knowledge of the researcher, no study involving DevOps that is comparable to this research study has previously been conducted in this context.

8.6.2 Contributions to DevOps Managers

This study has identified and assessed the important factors that need to be given attention by DevOps managers in order to adopt DevOps culture advantageously. These factors include: Open Communication, Role and Responsibility Alignment, Respect, Trust, Social Norm, Perceived Behavioural Control-Internal and Organisational Usefulness. It was confirmed in this study that these factors are the key contributors to behavioural intention to accept DevOps collaboration culture; however, some of these factors do not directly influence Intentions.

The managers must instil the culture of open communication, respect and trust between team members in order to minimise the risk of resistance to the new culture. In addition, managers should clearly define the roles and responsibilities of all team members to reduce the uncertainties that may arise and often come with blame-shifting. The empirical results of the study provide new insights into the success factors influencing DevOps practitioners with regards to the software development in DevOps settings, which is still in its infancy. A well-documented policy framework and an implementation guide should be drawn up by these organisations to minimise adoption failure by paying attention to the factors identified.

8.6.3 Methodology contribution

The use of the DCAM as an extension of the ISDAM and its validation as the theoretical framework provides a new mechanism for the identification and confirmation of the success factors of DevOps collaboration culture. In addition, the extraction of these factors using literature review and qualitative content analysis of the interviews provide methodological support to this study. This methodological approach forms a base for empirical confirmatory analysis that could be used by organisations implementing the DevOps culture. The methodology used during this research study is a key contributor to the body of knowledge of DevOps success factors in the field of software engineering.

8.6.4 Theoretical contribution

In this study, the ISDAM was extended by including the DevOps collaboration culture elements as new decision variables, thereby formulating a new theoretical framework. The proposed framework (DCAM), which is formulated from numerous previous theories and models, investigated the factors affecting the acceptance of DevOps collaboration culture. Therefore, such an investigation contributes to the literature on software development generally, and DevOps environments in particular.

8.6.5 Practical contribution

Organisations incur costs when introducing new technologies; thus, it is important that such technologies are fully utilised to see a return on investment. The adoption and implementation of DevOps culture is not different. However, when decision makers are informed of these factors affecting the acceptance of DevOps culture and the impacts of these factors in real practice, the risks of rejection by the intended practitioners can be minimised. The theoretical framework developed by this study acts as a standard and guideline for appropriate implementation of DevOps collaboration culture within the organisations. Therefore, DevOps managers will be able to rely on informed decisions to prepare and predict any future adoption of DevOps and thus, mitigate possible risks associated with such an adoption.

8.7 Research Impact on Policy and Organisations

Up until now, factors influencing DevOps culture have remained a fairly new area in software engineering and has therefore not been extensively investigated. The reason that is attributed to this is that DevOps on its own is still new and many organisations are still using trial and error methods for its implementation. As a result, there are currently no documented or tried and tested standards on the implementation of DevOps. Various organisations use different approaches in the implementation of DevOps thus making it difficult for an organisation to adopt implementation strategies of other rival organisation. Simply put, DevOps is not a one size fits all solution for every organisation.

Nevertheless, this study serves as a guideline that organisations could use to improve their policies for the process of DevOps adoption. Findings emanating from this study show that any future successful DevOps collaboration culture places more emphasis on bridging the role played by the individuals within software development organisations. Therefore, the research framework in this study provides insight into understanding the roles carried out by key actors in the DevOps culture.

8.8 Limitations of the Research Study

The researcher of this study acknowledge the following as the limitations of this study:

1. The research framework does not factor in DevOps managers and their personalities and emotions.

2. The research was carried out in a developing country context using South Africa as a framework for a developing country. Therefore, its representation may not be a true reflection of all developing countries. For this reason, the results of this study may not be generalisable to all developing countries.
3. The research framework relies on the assumption that humans are rational beings that make systematic judgments.
4. Unconscious motives are unaccounted for by the research framework.
5. There might be differences between the approaches used among the study population of the provinces of South Africa.
6. The data used in this research study was collected at one point in time (cross-sectional survey only). The sample was studied only once.
7. Further data are required in future to compare the factors from different organisations located within the borders of South Africa.
8. The study demonstrated a need to carry out research using longitudinal studies. Prospective studies could focus on investigating how DevOps management and the perceptions of experts on these factors vary over time. This may assist in explaining the success trends of DevOps culture.
9. Due to resource constraints, the research was restricted to South Africa.
10. No follow up was done with participants of this research study due to anonymity of the participants.

8.9 Recommendations

In terms of the findings emanating from this research study, the following recommendations are made:

1. DevOps management and practitioners need to use DevOps culture in software development projects based on DevOps collaboration culture policy guidelines. They must focus on the most important factors, such as open communication, which allows a DevOps team member an equal opportunity to be heard, irrespective of the role or rank of the team member in the organisation. The communication policies should be tailored in a way that supports the disadvantaged and encourages the voiceless to be heard.

2. Roles should be clearly defined in terms of their scope and responsibilities that come with a role. This should be documented and made accessible to everyone entitled to this information. Team building exercises should be pursued with the intention of encouraging team participation, which promotes team trust and respect of fellow team members. Instead of effecting individual incentives, team incentives should be encouraged to promote the culture of collaboration.

3. Employees should be encouraged to respect and trust each other for the success of the DevOps projects. Team-building exercises should be encouraged.

4. This study provides researchers with important knowledge regarding the practice of DevOps practitioners. This knowledge can be used for

further research and for encouraging DevOps managers and practitioners to employ DevOps culture more efficiently and effectively in a sustainable way.

5. Corporate sponsors should be encouraged to reward, with appropriate incentives, team members who motivate their team members to implement DevOps culture for further improvement of the culture of their DevOps teams.

8.10 Future Research of the Study

In view of the findings of this research study, the following recommendations are made for future research:

1. A comparative study of different types of organisations. As it was mentioned that DevOps does not have a one size fits all implementation approach, it would be interesting to study the difference DevOps requirements of the different organisations.

2. Future studies could assess DevOps collaboration culture by applying longitudinal surveys rather than a slice-time method which was used in this research study. Employing data collected over a longer period of time will help researchers to forecast possible trends in DevOps organisations.

3. Post-intention approach research in which respondents are asked about what they intend to do and to articulate a very specific plan about how they could go about attaining their goal should be considered. In this

way, an individual is forced to be realistic in their planning by considering their environments. Such an approach could assist in providing better predictions of the practitioner's behaviour when implementing DevOps culture.

4. Future investigations must examine the risks and challenges faced by practitioners, which result from the collaboration culture.

5. The effects of variables such as educational level, on these relationships should be investigated.

8.11 Conclusion

This research study was conducted to identify the factors that influence the acceptance of DevOps culture. This investigation was conducted using secondary data in the form of literature review and surveys as primary data. The literature review was used to identify these factors, and their influence was assessed using surveys. From the review of the literature, factors that made up DevOps collaboration culture were identified as open communication, role and responsibility alignment, respect and trust. The influence of these factors on the acceptance of DevOps culture was investigated further using a mixed methods research design.

Quantitative data was collected using web-based questionnaires and qualitative data were acquired through interviews. Since this research study was essentially an explanatory mixed method study, the quantitative data was collected and analysed prior to the collection and analysis of qualitative data.

The analysis of the quantitative data revealed that open communication, respect and trust were all significant. In addition, organisational usefulness was also found to be significantly influencing intentions to accept the DevOps culture. Qualitative methods were employed to confirm the quantitative results.

These findings are important to organisations that have just adopted or are considering future adoption of DevOps. The finding gives a guideline on how to prepare employees for this transition to the new culture. By incorporating the recommendations of this study into their transition policies, DevOps management can minimise the risks of rejection of this culture by the affected employees. Past research has shown that people factors are regarded as common major challenges faced by DevOps adoption. Therefore, by investing in human resources, DevOps managers are likely to achieve successful implementation of DevOps.

In conclusion, all the hypotheses, objectives and research questions were answered and the findings of this study may assist to promote the successful adoption of DevOps and thus enable organisations to realise the true benefits of the culture of DevOps collaboration.

References

Acuña, S. T., Gómez, M. and Juristo, N. (2009), 'How do personality, team processes and task characteristics relate to job satisfaction and software quality?', *Information and Software Technology* **51**(3), 627–639. 36

Adler, P. S. (2003), 'Practice and process: the socialization of software development', **2003**(1), C1–C6. 23

Agarwal, R. and Prasad, J. (2000), 'A field study of the adoption of software process innovations by information systems professionals', *IEEE Transactions on Engineering Management* **47**(3), 295–308. 68

Ahmad, Z., Hussain, M., Rehman, A., Qamar, U. and Afzal, M. (2015), Impact minimization of requirements change in software project through requirements classification, *in* 'Proceedings of the 9th International Conference on Ubiquitous Information Management and Communication', pp. 1–5. 1

Ajzen, I. (1991), 'The theory of planned behavior', *Organizational Behavior and Human Decision Processes* **50**(2), 179–211. 56

Akbar, M. A., Mahmood, S., Shafiq, M., Alsanad, A., Alsanad, A. A.-A. and Gumaei, A. (2020), 'Identification and prioritization of DevOps success factors using fuzzy-AHP approach', *Soft Computing* . 9

Al-Gahtani, S. S., Hubona, G. S. and Wang, J. (2007), 'Information technology (IT) in Saudi Arabia: Culture and the acceptance and use of IT', *Information and Management* **44**(8), 681–691. 64

Al-Mamary, Y. H., Al-nashmi, M., Hassan, Y. A. G. and Shamsuddin, A. (2016), 'A Critical Review of Models and Theories in Field of Individual Acceptance of Technology', *International Journal of Hybrid Information Technology* **9**(6), 143–158. 65, 66

Al Zefeiti, S. M. B. and Mohamad, N. A. (2015), 'Methodological considerations in studying transformational leadership and its outcomes', *International Journal of Engineering Business Management* **7**(1), 1–11. 106

Alrawashdeh, T. A., Elbes, M. W., Almomani, A., ElQirem, F. and Tamimi, A. (2019), 'User acceptance model of open source software: an integrated model of oss characteristics and utaut', *Journal of Ambient Intelligence and Humanized Computing* pp. 1–13. 63

Attaran, M. (2004), 'Exploring the relationship between information technology and business process reengineering', *Information & Management* **41**(5), 585–596. 6

Bagozzi, R. P. (1992), 'The self-regulation of attitudes, intentions, and behavior', *Social Psychology Quarterly* **55**(2), 178–204. 54

Barnhart, D., Barrett, T., Sachs, J. and Will, P. (2009), Development and operation of a micro-satellite dynamic test facility for distributed flight operations, *in* 'AIAA SPACE 2009 Conference & Exposition', p. 6443. 4, 25

Beauchamp, M. R., Bray, S. R., Eys, M. A. and Carron, A. V. (2002), 'Role ambiguity, role efficacy, and role performance: Multidimensional and mediational relationships within interdependent sport teams', *Group Dynamics* **6**(3), 229–242. 90

Ben Othman, H., Zouaoui, M. and Hamdoun, M. (2016), 'Organizational culture and the acceptance of agile methodology', *Proceedings - 2016 International Conference on Digital Economy: Emerging Technologies and Business Innovation, ICDEc 2016* pp. 16–23. 77, 218

Bento, F., Tagliabue, M. and Lorenzo, F. (2020), 'Organizational silos: A scoping review informed by a behavioral perspective on systems and networks', *Societies* **10**(3), 56. 25

Bjarnason, E., Wnuk, K. and Regnell, B. (2011), Requirements are slipping through the gaps–A case study on causes & effects of communication gaps in large-scale software development, *in* '2011 IEEE 19th international requirements engineering conference', pp. 37–46. 37

Blom, T and Viljoen, R. (2016), Human Reactions to Change, *in* 'Proceedings of the 2016 IAABR/Academic Oasis–Palm Beach International Conference on Business, Economics, Finance and Accounting'. 53

Brown, D. R. and Harvey, D. (2011), *An experiential approach to organization development*, Prentice Hall Upper Saddle River, NJ, Englewood Cliffs, N.J. 53

Brown, S. A., Dennis, A. R. and Venkatesh, V. (2010), 'Predicting collaboration technology use: Integrating technology adoption and collaboration research', *Journal of Management Information Systems* **27**(2), 9–54. 166, 170, 207, 220

Byrne, D. (2002), Data analysis and interpretation, *in* 'Research Project Planner', SAGE Publications. 194

CA Technology (2013), Techinsights report : What smart businesses know about DevOps, techreport, TechInsights Report. 4

Campbell, A., McNamara, O. and Gilroy, P. (2004), *Practitioner Research and Professional Development in Education*, SAGE Publications. 110

Carmeli, A., Dutton, J. E. and Hardin, A. E. (2015), 'Respect as an engine for new ideas: Linking respectful engagement, relational information processing and creativity among employees and teams', *Human Relations* **68**(6), 1021–1047. 164, 177, 205, 222

Chan, K. Y. and Thong, J. Y. L. (2007), 'An Integrated Framework of Individual Acceptance of Agile Methodologies', *PACIS 2007 Proceedings* . 77

Chau, P. Y. (1996), 'An empirical investigation on factors affecting the acceptance of CASE by systems developers', *Information and Management* **30**(6), 269–280. 68

Cheng, X., Yin, G., Azadegan, A. and Kolfschoten, G. (2016), 'Trust Evolvement in Hybrid Team Collaboration: A Longitudinal Case Study', *Group Decision and Negotiation* **25**(2), 267–288. 85, 87

Creswell, J. (2013), Qualitative, quantitative, and mixed methods approaches, *in* 'Research design', SAGE Publications. 100, 103, 104, 105, 107, 108, 111, 113, 114, 115, 116

Creswell, J. (2014), *A concise introduction to mixed methods research*, SAGE Publications. 102

Creswell, J. W. (2003), *Research design: qualitative, quantitative, and mixed methods approaches*, 3rd edn, SAGE Publications. 14, 97, 98, 100, 101, 102, 106, 109, 113

Creswell, J. W. and Clark, V. L. P. (2017), *Designing and conducting mixed methods research*, SAGE Publications. 108, 117

Dasgupta, S. and Gupta, B. (2010), 'Organizational culture and technology use in a developing country: An empirical study.', p. 565. 6, 18, 64

Davis, F. D. and Venkatesh, V. (1996), 'A critical assessment of potential measurement biases in the technology acceptance model: Three experiments', *International Journal of Human-Computer Studies* **45**(1), 19–45. 58

De Villiers, M. R. (2012), Models for interpretive information systems research, part 2: Design research, development research, design-science research, and design-based research - a meta-study and examples, *in* 'Research Methodologies, Innovations and Philosophies in Software Systems Engineering and Information Systems', IGI Global, pp. 238–255. 97, 102, 103, 104, 105, 106, 113, 117

de Vries, R. E., van den Hooff, B. and de Ridder, J. A. (2006), 'Explaining knowledge sharing', *Communication Research* **33**(2), 115–135. 165, 202, 222

Debbiche, F., Wrang, M. and Sinkala, K. (2019), 'Accelerating software delivery in the context of requirements analysis and breakdown for devops: A multiple-case study'. 22

Debois, P. (2012), 'Devops areas - codifying devops practices'. [accessed: 25 April 2015].
URL: *http://www.jedi.be/blog/2012/05/12/codifying-devops-area-practices/* 4, 31

DeGrandis, D. (2011), 'Devops: So you say you want a revolution?', *Cutter IT Journal* **24**(8), 34. 37

Diaz, J., López-Fernández, D., Perez, J. and González-Prieto, Á. (2020), 'Why are many business instilling a devops culture into their organization?', *arXiv preprint arXiv:2005.10388* . 33

Dijkstra, O. (2013), Extending the Agile Development Discipline to Deployment - The Need For a Holistic Approach, Master's thesis, Utrecht University. 8, 9, 24, 25, 35

Dipboye, R. L. (2018), *The emerald review of industrial and organizational psychology*, Emerald Publishing Limited. 45

Dreesen, T., Linden, R., Meures, C., Schmidt, N. and Rosenkranz, C. (2016), Beyond the Border: A Comparative Literature Review on Communication Practices for Agile Global Outsourced Software Development Projects, *in* '2016 49th Hawaii International Conference on System Sciences (HICSS)', pp. 4932–4941. 37

Drost, E. A. et al. (2011), 'Validity and reliability in social science research', *Education Research and perspectives* **38**(1), 105. 124, 125

Durodolu, O. O. (2016), 'Technology acceptance model as a predictor of using information system'to acquire information literacy skills', *Library Philosophy and Practice* . 54, 57, 58

Dyck, A., Penners, R. and Lichter, H. (2015), 'Towards Definitions for Release Engineering and DevOps', *2015 IEEE/ACM 3rd International Workshop on Release Engineering* pp. 3–3.
URL: *http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=7169442* 27

Eckhardt, A., Laumer, S. and Weitzel, T. (2009*a*), 'Who influences whom?

analyzing workplace referents' social influence on it adoption and non-adoption', *Journal of information technology* **24**(1), 11–24. 55, 63

Eckhardt, A., Laumer, S. and Weitzel, T. (2009*b*), 'Who influences whom? analyzing workplace referents' social influence on it adoption and non-adoption', *Journal of Information Technology* **24**(1), 11–24. 166, 207, 220

Edwards, D. (2010), 'What is DevOps', *Retrieved* **3**, 2014. 5, 8

Elbayadi, M. E. (2014), Relational leadership, DevOps, and the post-pc era: toward a practical theory for 21st century technology leaders, PhD thesis, Antioch University. 4, 9, 25

Ellemers, N., Sleebos, E., Stam, D. and de Gilder, D. (2013), 'Feeling Included and Valued: How Perceived Respect Affects Positive Team Identity and Willingness to Invest in the Team', *British Journal of Management* **24**(1), 21–37. 90, 164, 204

Erich, F., Amrit, C. and Daneva, M. (2014), 'DevOps Literature Review', *University of Twente, Tech. Rep* p. 27. 5, 8, 9, 25, 33, 34, 52

Eys, M. A. and Carron, A. V. (2001), 'Role ambiguity, task cohesion, and task self-efficacy', *Small Group Research* **32**(3), 356–373. 90

Farroha, B. and Farroha, D. (2014), A framework for managing mission needs, compliance, and trust in the DevOps environment, *in* '2014 IEEE Military Communications Conference', IEEE. 28, 29, 36

Ferreira, S., Collofello, J., Shunk, D., Mackulak, G. and Wolfe, P. (2003), Utilization of process modeling and simulation in understanding the effects of requirements volatility in software development, *in* 'International Workshop on Software Process Simulation and Modeling, Portland, Oregon'. 22

Fiampolis, P. and Groll, J. (2016), 'Bringing together dev and ops', *Training Journal* pp. 12–15. 38

Fishbein, M. and Ajzen, I. (1975), 'Belief, Attitude, Intention and Behaviour: An Introduction to Theory and Research', *Reading MA AddisonWesley* **6**(2), 480. 55, 57

Fisher, S. L. and Howell, A. W. (2004), 'Beyond user acceptance: An examination of employee reactions to information technology systems', *Human Resource Management* **43**(2-3), 243–258. 7

Foltz, P. (2018), 'Automating the assessment of team collaboration through communication analysis', *Design recommendations for intelligent tutoring systems* **6**, 197–185. 5

Fowler, J. (2009), 11 Ethical Issues in Survey Research, *in* 'Survey Research Methods (4th ed.)', 0 edn, SAGE Publications, pp. 162–170. 16

Frank Lafasto, C. L. (2001), *When Teams Work Best: 6,000 Team Members and Leaders Tell What It Takes to Succeed*, SAGE Publications. 5, 41, 45, 46

Ghazizadeh, M., Lee, J. D. and Boyle, L. N. (2012), 'Extending the Technology Acceptance Model to assess automation', *Cognition, Technology & Work* **14**(1), 39–49. 58

Gioia, D. A., Corley, K. G. and Hamilton, A. L. (2013), 'Seeking Qualitative Rigor in Inductive Research: Notes on the Gioia Methodology', *Organizational Research Methods* . 102

Given, L. M. (2008), *The Sage encyclopedia of qualitative research methods*, SAGE Publications. 109

Google.com (2019), 'DevOps google trends'. Accessed [November, 2019].
URL: *https://trends.google.com/trends/explore?q=DevOps&geo=US* 24

Gottesheim, W., Strasse, F. and Gottesheim, W. (2015), 'Challenges, benefits and best practices of performance focused DevOps', *LT 2015 - Proceedings of the 4th ACM/SPEC International Workshop on Large-Scale Testing, in Conjunction with ICPE 2015* p. 3. 35

Gratton, L. and Erickson, T. J. (2007), 'Eight ways to build collaborative teams', *Harvard business review* **85**(11), 100. 6, 47

Guardado, D. (2012), Process Acceptance and Adoption by IT Software Project Practitioners, PhD thesis, Capella University. 2, 6, 7, 22, 34, 52, 53, 64, 70, 75, 217

Guerrero, J., Zúniga, K., Certuche, C. and Pardo, C. (2020), 'A systematic mapping study about devops', *Journal de Ciencia e Ingeniería* **12**(1), 48–62. 101

Hardgrave, B. C., Davis, F. D. and Riemenschneider, C. K. (2003), 'Investigating determinants of software developers' intentions to follow methodologies', *Journal of Management Information Systems* **20**(1), 123–151. 56, 68, 69, 70, 120

Hardgrave, B. C. and Johnson, R. A. (2003), 'Toward an information systems development acceptance model: the case of object-oriented systems development', *Engineering Management, IEEE Transactions on* **50**(3), 322–336. 52, 71, 72, 73, 75, 80, 81, 82, 119, 165, 166, 207, 209, 210, 217, 218, 219, 220, 221, 222, 228

Henning, E., Van Rensburg, W. and Smit, B. (2004), *Finding your way in qualitative research*, Van Schaik Pretoria. 107

Hoda, R., Noble, J. and Marshall, S. (2013), 'Self-organizing roles on agile software development teams', *IEEE Transactions on Software Engineering* **39**(3), 422–444. 166, 177, 204

Hsieh, H.-F. and Shannon, S. E. (2005), 'Three approaches to qualitative content analysis', *Qualitative Health Research* **15**(9), 1277–1288. 195, 196

Huisman, M. (2004), 'Factors that affect the use and acceptance of systems development methodologies by system developers', *ACIS 2004 Proceedings* p. 57. 69

Huisman, M. and Iivari, J. (2002), 'The Individual Deployment of Systems Development Methodologies', *Caise 2002* pp. 134–150. 69, 70, 75

Huizingh, E. (2007), *Applied Statistics with SPSS*, SAGE Publications. 122, 135, 137, 138, 157, 159

Humble, J. and Molesky, J. (2011), 'Why enterprises must adopt devops to enable continuous delivery', *Cutter IT Journal* **24**(8), 6. 30, 31, 38

Hummel, M., Rosenkranz, C. and Holten, R. (2013), 'The role of communication in agile systems development: An analysis of the state of the art', *Business and Information Systems Engineering* **5**(5), 343–355. 86

Hüttermann, M. (2012*a*), Beginning DevOps for Developers, *in* 'DevOps for Developers', Springer, pp. 3–13. 3, 18, 26, 28, 32

Hüttermann, M. (2012*b*), *DevOps for developers*, Springer-Verlag Berlin and Heidelberg. 4, 5, 8, 25, 52

Iivari J., I. N. (2011), 'The relationship between organizational culture and the deployment of agile methods', *Information and Software Technology* **53**(5), 509–520. 68

Im, I., Hong, S. and Kang, M. S. (2011), 'An international comparison of technology adoption: Testing the UTAUT model', *Information and Management* **48**(1), 1–8. 64

Javed, T., Maqsood, M. e. and Durrani, Q. S. (2004), 'A study to investigate the impact of requirements instability on software defects', *ACM SIGSOFT Software Engineering Notes* **29**(3), 1. 1, 22

Johnson, R. B. and Onwuegbuzie, A. J. (2004), 'Mixed methods research: A research paradigm whose time has come', *Educational researcher* **33**(7), 14–26. 108, 113

Jones, C. M. and McCarthy, R. (2009), 'Utilizing the technology acceptance model to assess employee adoption of information systems security measures', *Issues in Information Systems* **11**(1), 182. 7

Jones, C. M., McCarthy, R. V., Halawi, L. and Mujtaba, B. (2010), 'Utilizing the technology acceptance model to assess the employee adoption of information systems security measures', *Issues in Information Systems* **11**(1), 9. 53

Jong, B. A. D., Dirks, K. T. and Gillespie, N. (2016), 'Trust and team performance: A meta-analysis of main effects, moderators, and covariates.', *Journal of Applied Psychology* **101**(8), 1134–1150. 165, 206, 222

Jupp, V. (2006), *The SAGE Dictionary of Social Research Methods*, SAGE Publications, London. accessed[03 October 2017].

URL: *https://methods.sagepub.com/reference/the-sage-dictionary-of-social-research-methods* 146, 147, 154

Kamuto, M. B. and Langerman, J. J. (2017), Factors inhibiting the adoption of devops in large organisations: South african context, *in* '2017 2nd IEEE International Conference on Recent Trends in Electronics, Information & Communication Technology (RTEICT)', IEEE, pp. 48–51. 9

Kaplan, B. and Duchon, D. (1988), 'Combining qualitative and quantitative methods in information systems research: a case study', *MIS quarterly* pp. 571–586. 107

Karahanna, E., Agarwal, R. and Angst, C. (2006), 'Reconceptualizing Compatibility Beliefs in Technology Acceptance Research', *MIS Quarterly* **30**(4), 781–804. 58

Kashada, A., Ehtiwsh, E. and Nakkas, H. (2020), 'The role of technology acceptance model (tam) towards information systems implementation success: A meta-analysis', *The International Journal of Engineering and Science (IJES)* **9**(01), 30–36. 53

Khan, A. A. and Shameem, M. (2020), 'Multicriteria decision-making taxonomy for devops challenging factors using analytical hierarchy process', *Journal of Software: Evolution and Process* p. e2263. 5, 9, 34, 52

Khatib, T. M. (1997), 'Organizational culture, subcultures, and organizational commitment.', *Dissertation Abstracts International Section A: Humanities and Social Sciences* **57**(11-A), 4952. 53

Knabe, A. P. (2012), *Applying Ajzen's theory of planned behavior to a study of online course adoption in public relations education*, Marquette University. 56

Knox, K. (2004), 'A researcher's dilemma-philosphical and methodological pluralism', *The Electronic Journal of Business Research Methods* **2**(2), 119–128. 107

Komma, R. C. (2010), *Quality cost oriented software testing*, California State University, Long Beach. 22

Kozar, K. A. (1989), 'Adopting Systems Development Methods: An Exploratory Study', *Journal of Management Information Systems* **5**(4), 73–86. 68

Lai, P. C. (2017), 'The Literature Review Of Technology Adoption Models And Theories For The Novelty Technology', *Journal of Information Systems and Technology Management* **14**(1), 21–38. 54, 60

Laura-georgeta, T. (2008), 'Change Management – Resistance To the Change', *Economics Science Series* **17**(4), 622–625. 53

Lê-Quôc, A. (2011), 'Metrics-driven devops', *Cutter IT Journal* **24**(12), 24. 34

Leedy, P. D. and Ormrod, J. E. (2005), *Practical research*, Pearson Custom. 108

Leite, L., Rocha, C., Kon, F., Milojicic, D. and Meirelles, P. (2019), 'A survey of devops concepts and challenges', *ACM Computing Surveys (CSUR)* **52**(6), 1–35. 9

Leonard-Barton, D. (1987), 'Implementing structured software methodologies: a case of innovation in process technology.', *Interfaces* **17**(3), 6–17. 68

Lewis-Beck, M., Bryman, A. and Futing Liao, T. (2004), *The SAGE Encyclopedia of Social Science Research Methods*, SAGE Publications, Thousand Oaks, California. 115

Lincoln, Y. S. and Guba, E. G. (1985), *Naturalist inquiry*, SAGE Publications. 125, 126, 127, 128

Lindsjørn, Y., Sjøberg, D. I., Dingsøyr, T., Bergersen, G. R. and Dybå, T. (2016), 'Teamwork quality and project success in software development: A survey of agile development teams', *Journal of Systems and Software* **122**, 274–286. 166, 209

Liu, Y., Li, C. and Liu, W. (2014), 'Integrated Solution for Timely Delivery of Customer Change Requests: A Case Study of Using DevOps Approach', *International Journal Science and Technology* **7**(2), 41–50. 24, 25, 28

Luz, W., Pinto, G. and Bonifácio, R. (2019), 'Adopting DevOps in the real world: A theory, a model, and a case study', *Journal of Systems and Software* **157**. 26, 35

Markus, M. L. (2004), 'Technochange management: using IT to drive organizational change', *Journal of Information technology* **19**(1), 4–20. 6

Maroukian, K. and Gulliver, S. (2020*a*), 'Exploring the link between leadership and devops practice and principle adoption', *Advanced Computing: An International Journal* **11**(4). 46

Maroukian, K. and Gulliver, S. R. (2020*b*), 'Leading devops practice and principle adoption', *arXiv preprint arXiv:2008.10515* . 26

Masombuka, T. and Mnkandla, E. (2018), A DevOps collaboration culture acceptance model, *in* 'Proceedings of the Annual Conference of the South African Institute of Computer Scientists and Information Technologists on SAICSIT', ACM Press. 76, 157, 158, 179, 232

Mathieson, K. (1991), 'Predicting user intentions: Comparing the technology acceptance model with the theory of planned behavior', *Information Systems Research* **2**(3), 173–191. 75

Mattessich, P. W. and Monsey, B. R. (1992), *Collaboration: what makes it work. A review of research literature on factors influencing successful collaboration.*, ERIC. 5, 41

Matturro, G., Raschetti, F. and Fontán, C. (2015), Soft skills in software development teams: a survey of the points of view of team leaders and team members, *in* 'Cooperative and Human Aspects of Software Engineering (CHASE), 2015 IEEE/ACM 8th International Workshop on', pp. 101–104. 36

Mayring, P. (2014), 'Qualitative content analysis: Theoretical foundation, basic procedures and software solution', *Qualitative Social Research* . 198

McIntosh, M. J. and Morse, J. M. (2015), 'Situating and constructing diversity in semi-structured interviews', *Global Qualitative Nursing Research* **2**, 233339361559767. 112

McLagan, P. A. (2002), 'Success with change.', *T+ D* **56**(12), 44–53. 6

McLeod, L. and MacDonell, S. G. (2011), 'Factors that affect software systems development project outcomes', *ACM Computing Surveys* **43**(4), 1–56. 43, 44

Melo, C. D. O., Cruzes, D. S., Kon, F. and Conradi, R. (2013), 'Interpretative case studies on agile team productivity and management', *Information and Software Technology* **55**(2), 412–427. 174, 208

Miles, M. B. and Huberman, A. M. (1994), *Qualitative data analysis: An expanded sourcebook*, SAGE Publications, Thousand Oaks. 126, 127, 128

Mohan, K. and Ahlemann, F. (2011), A theory of user acceptance of is project management methodologies: Understanding the influence of psychological determinism and experience, *in* '10th International Conference on Wirtschaftsinformatik, A. Bernstein and G. Schwabe', pp. 912–922. 219

Neely, S. and Stolt, S. (2013), Continuous Delivery? Easy! Just Change Everything (well, maybe it is not that easy), *in* 'Agile Conference (AGILE), 2013', pp. 121–128. 26

Nishishiba, M., Jones, M. and Kraner, M. (2014), *Research Methods and Statistics for Public and Nonprofit Administrators: A Practical Guide*, SAGE Publications, London. 111, 114, 115, 116, 159

Nugent, S. L. (2012), 'Building and leading a culture of collaboration: An analysis of the influence of a company's social networking tools on employee collaboration.', *Dissertation Abstracts International Section A: Humanities and Social Sciences* **73**(4-A), 1482. 5, 6, 18, 41, 42, 53

Nybom, K., Smeds, J. and Porres, I. (2016), On the impact of mixing responsibilities between devs and ops, *in* 'Agile Processes, in Software Engineering, and Extreme Programming', Springer International Publishing, pp. 131–143. 27, 28, 38

Oliveira, T. and Martins, M. (2011), 'Literature review of Information Technology Adoption Models at Firm Level', *Electronic Journal of Information ...* **14**(1), 110–121. 66

Olszewska, M., Heidenberg, J., Weijola, M., Mikkonen, K. and Porres, I. (2016), 'Quantitatively measuring a large-scale agile transformation', *Journal of Systems and Software* **117**, 258–273. 106

OxfordDisctionaries.com (2016), 'Oxford Dictionaries'. 26

Page, S. E. (2008), *The difference: How the power of diversity creates better groups, firms, schools, and societies*, Princeton University Press. 45

Palinkas, L. A., Horwitz, S. M., Green, C. A., Wisdom, J. P., Duan, N. and Hoagwood, K. (2015), 'Purposeful sampling for qualitative data collection and analysis in mixed method implementation research', *Administration and policy in mental health and mental health services research* **42**(5), 533–544. 115

Pant, R. (2009), 'Organizing a Digital Technology Department of Medium Size in a Media Company.'. 5, 8, 25, 26

Parker, D. W., Holesgrove, M. and Pathak, R. (2015), 'Improving productivity with self-organised teams and agile leadership', *International Journal of Productivity and Performance Management* **64**(1), 112–128. 135, 137, 138, 165, 174, 208

Payne, G. and Payne, J. (2004), *Key Concepts in Social Research*, SAGE Publications. 15, 100, 102, 103, 105, 107, 115, 116, 123, 124, 195

Peng, C. (2009), *Data Analysis Using SAS®*, SAGE Publications. 147

Penners, R. and Dyck, A. (2015), 'Release Engineering vs. DevOps-An Approach to Define Both Terms', *Full-scale Software Engineering* p. 49. 26, 27

Pett, M., Lackey, N. and Sullivan, J. (2003), *Making Sense of Factor Analysis*, SAGE Publications. 141, 143, 144

Ragunath, P., Velmourougan, S., Davachelvan, P., Kayalvizhi, S. and Ravimohan, R. (2010), 'Evolving A New Model (SDLC Model-2010) For Software

Development Life Cycle (SDLC)', *IJCSNS International Journal of Computer Science and Network Security* **10**(1), 112–119. 23

Rajapakse, J. (2011), 'Extending the unified theory of acceptance and use of technology (UTAUT) model', *The 4th International Conference on Interaction Sciences* pp. 47–52. 64

Reinard, J. (2006), *Communication Research Statistics*, SAGE Publications. 159, 160

Richards, S. B., Frank, C. L., Sableski, M.-K. and Arnold, J. M. (2016), Other models of collaboration ma ry-k at e sableski, *in* 'Collaboration Among Professionals, Students, Families, and Communities', Routledge, pp. 170–187. 5

Riemenschneider, C. K., Hardgrave, B. C. and Davis, F. D. (2002), 'Explaining software developer acceptance of methodologies: A comparison of five theoretical models', *IEEE Transactions on Software Engineering* **28**(12), 1135–1145. 68, 69, 70

Rogers, E. M. (2003), *Diffusion of Innovations*, 5th edition edn, Simon and Schuster Inc. 65, 69, 218

Rosen, E. (2007), *The culture of collaboration: Maximizing time, talent and tools to create value in the global economy*, Red Ape Pub. 5, 6, 34, 41, 42, 43, 46, 53

Roulston, K. (2010), *Reflective Interviewing: A Guide to Theory and Practice*, SAGE Publications, London. 121

Rubin, A. and Babbie, E. R. (2016), *Empowerment series: Research methods for social work*, Cengage Learning. 116

Rütz, M. (2019), 'Devops: A systematic literature review', *no. August* . 23, 25

Salkind, N. (2007), *Encyclopedia of Measurement and Statistics*, SAGE Publications. 83

Saugatuck Technology (2014), Why DevOps Matters : Practical Insights on Managing Complex & Continuous Change, Technical report, Saugatuck Technology In. October. 6, 52

Saunders, M., Lewis, P. and Thornhill, A. (2009), *Research methods for business students*, Pearson education. 96

Schach, S. R. (2002), *Object-oriented and classical software engineering*, McGraw-Hill New York. 3, 22

Schein, E. H. (2009), *The corporate culture survival guide*, Vol. 158, John Wiley & Sons. 43, 53

Schuh, G., Potente, T., Wesch-Potente, C., Weber, A. R. and Prote, J.-P. (2014), 'Collaboration mechanisms to increase productivity in the context of industrie 4.0', *Procedia CIRP* **19**, 51–56. 165, 174, 208

Senapathi, M., Buchan, J. and Osman, H. (2018), 'DevOps capabilities, practices, and challenges: Insights from a case study', **137700**. 9

Senarath, A., Grobler, M. and Arachchilage, N. A. G. (2019), 'Will they use it or not? investigating software developers' intention to follow privacy engineering methodologies', *ACM Transactions on Privacy and Security (TOPS)* **22**(4), 1–30. 58

Shamow, E. (2011), 'Devops at advance internet: How we got in the door', *Cutter IT Journal* **24**(8), 13. 52

Strode, D. E., Huff, S. L. and Tretiakov, A. (2009), The impact of organizational culture on agile method use, *in* 'Proceedings of the 42nd Annual Hawaii International Conference on System Sciences, HICSS'. 218, 219

Sudhakar, M. (2005), Managing the impact of requirements volatility, mathesis, Umea University, Department of Computing Science. 22

Swartout, P. (2014), *Continuous Delivery and DevOps–A Quickstart Guide*, Packt Publishing Ltd. 26

Taherdoost, H. (2018), 'A review of technology acceptance and adoption models and theories', **22**, 960–967. 54

Taylor, S. and Todd, P. A. (1995), 'Understanding information technology usage: A test of competing models', *Information Systems Research* **6**(2), 144–176. 58

Techbeacon (2015), '7 DevOps roles you need to succeed'. Accessed [2018-05-10].
URL: *https://techbeacon.com/aboutus* 29

Tessem, B. and Iden, J. (2008), Cooperation between developers and operations in software engineering projects, *in* 'Proceedings of the 2008 international workshop on Cooperative and human aspects of software engineering', pp. 105–108. 5, 23, 33

Thornhill, A., Saunders, M. and Lewis, P. (2009), *Research methods for business students*, fitfth edition edn, Prentice Hall, New York. 100, 106

Trompenaars, F. and Woolliams, P. (2004), *Managing change across corporate cultures*, Capstone Chichester. 6, 53

Tschannen-Moran, M. and Hoy, W. K. (2000), 'A Multidisciplinary Analysis of the Nature, Meaning, and Measurement of Trust', *Review of Educational Research* **70**(4), 547–593. 87, 88

Uden, L. and Dix, A. (2004), 'Lifelong learning for software engineers', *International journal of continuing engineering education and life-long learning* **14**(1), 101–110. 191

University of Melbourne (2003), 'Software development life cycle'. Accessed [2016-01-14].
URL: *https://www.unimelb.edu.au/accessibility/users/development* 22

Vaismoradi, M., Turunen, H. and Bondas, T. (2013), 'Content analysis and thematic analysis: Implications for conducting a qualitative descriptive study', *Nursing & Health Sciences* **15**(3), 398–405. 195

Van Gelder, S. (2011), The effectiveness of knowledge sharing and collaboration in creating high performance work teams, PhD thesis, Pepperdine University. 37

Vangen, S. and Huxham, C. (2008), 'Nurturing collaborative relations: Building trust in interorganizational collaboration', *The Journal of Applied Behavioral Science* **39**(3), 490–504. 37

Venkatesh, V. (2000), 'Determinants of perceived ease of use: Integrating perceived behavioral control, computer anxiety and enjoyment into the technology acceptance model', *Information systems research* **11**(4), 342–365. 58, 59, 60

Venkatesh, V. and Bala, H. (2008), 'Technology acceptance model 3 and a research agenda on interventions', *Decision Sciences* **39**(2), 273–315. 60, 61

Venkatesh, V., Thong, J. and Xin, X. (2016), 'Unified Theory of Acceptance and Use of Technology: A Synthesis and the Road Ahead', *Journal of the Association for Information Systems* **17**(5), 328–376. 63

Vogt, W. (2011), *SAGE Quantitative Research Methods*, SAGE Publications. 160, 164

Waddell, D. and Sohal, A. S. (1998), 'Resistance: a constructive tool for change management', *Management Decision* **36**(8), 543–548. 53

Walliman, N. (2006), *Social Research Methods*, SAGE Publications. 104, 105

Walls, M. (2013), *Building a DevOps culture*, " O'Reilly Media, Inc.". 4, 6, 27, 35, 36, 38, 77, 85, 86, 88, 89, 90, 92, 220, 221, 224, 228

Ward, R. (2013), 'The application of technology acceptance and diffusion of innovation models in healthcare informatics', **2**(4), 222–228. 66

Wettinger, J., Breitenbücher, U. and Leymann, F. (2014), DevOpSlang–bridging the gap between development and operations, *in* 'Service-Oriented and Cloud Computing', Springer, pp. 108–122. 9, 25

Willis, J. (2010), 'What devops means to me', Web. acceessed[14 Februdary 2016].
URL: *https://blog. chef. io/2010/07/16/what-devops-means-to-me/.* 4, 5, 33, 52

Wilson, J. M., Straus, S. G. and McEvily, B. (2006), 'All in due time: The development of trust in computer-mediated and face-to-face teams', *Organizational behavior and human decision processes* **99**(1), 16–33. 88

Winer, M. and Ray, K. (1994), *Collaboration Handbook: Creating, Sustaining, and Enjoying the Journey.*, ERIC. 5, 46, 47, 48

Yuen, Y. Y., Yeow, P. H. P., Lim, N. and Saylani, N. (2010), 'Internet banking adoption: Comparing developed and developing countries', *Journal of Computer Information Systems* **51**(1), 52–61. 64

Zhang, X., Stafford, T. F., Dhaliwal, J. S., Gillenson, M. L. and Moeller, G. (2014), 'Information & Management Sources of conflict between developers and testers in software development', *Information & Management* **51**(1), 13–26. 85

Zhang, Y. and Wildemuth, B. M. (2009), 'Qualitative analysis of content', *Applications of social research methods to questions in information and library science* **308**, 319. 197, 198, 199

Zohrabi, M. (2013), 'Mixed method research: Instruments, validity, reliability and reporting findings.', *Theory & practice in language studies* **3**(2). 125, 126

Appendix A

Ethical clearance

UNISA university of south africa

UNISA COLLEGE OF SCIENCE, ENGINEERING AND TECHNOLOGY'S (CSET) RESEARCH AND ETHICS COMMITTEE

12 September 2018

Ref #: 044/KTM/2018/CSET_SOC
Name: Mr Koos Themba Masombuka
Student #: 32637004
Staff #:

RECEIVED
2018 -09- 13
OFFICE OF THE EXECUTIVE DEAN
College of Science, Engineering and Technology

Dear Mr Koos Themba Masombuka

Decision: Ethics Approval for 5 years (Humans involved)

Researchers: Mr Koos Themba Masombuka, P.O. Box 2139, Empumalanga, 0458, 32637004@mylife.unisa.ac.za, +27 81 475 6679, +27 11 670 9123

Project Leader(s): Prof E Mnkandla, mnkane@unisa.ac.za, +27 11 670 9059

Working title of Research:
A model for successful collaboration in Development and Operations (DevOps) Environments

Qualification: PhD in Computer Science

Thank you for the application for research ethics clearance by the Unisa College of Science, Engineering and Technology's (CSET) Research and Ethics Committee for the above mentioned research. Ethics approval is granted for a period of five years, from 12 September 2018 to 12 September 2023.

1. The researcher will ensure that the research project adheres to the values and principles expressed in the UNISA Policy on Research Ethics.
2. Any adverse circumstance arising in the undertaking of the research project that is relevant to the ethicality of the study, as well as changes in the methodology, should be communicated in writing to the Unisa College of Science, Engineering and Technology's (CSET) Research and Ethics Committee. An amended application could be requested if there are substantial changes from the existing proposal, especially if those changes affect any of the study-related risks for the research participants.

Open Rubric

University of South Africa
Preller Street, Muckleneuk Ridge, City of Tshwane
PO Box 392 UNISA 0003 South Africa
Telephone: +27 12 429 3111 Facsimile: +27 12 429 4150
www.unisa.ac.za

3. The researcher(s) will conduct the study according to the methods and procedures set out in the approved application.
4. Any changes that can affect the study-related risks for the research participants, particularly in terms of assurances made with regards to the protection of participants' privacy and the confidentiality of the data, should be reported to the Committee in writing, accompanied by a progress report.
5. The researcher will ensure that the research project adheres to any applicable national legislation, professional codes of conduct, institutional guidelines and scientific standards relevant to the specific field of study. Adherence to the following South African legislation is important, if applicable: Protection of Personal Information Act, no 4 of 2013; Children's act no 38 of 2005 and the National Health Act, no 61 of 2003.
6. Only de-identified research data may be used for secondary research purposes in future on condition that the research objectives are similar to those of the original research. Secondary use of identifiable human research data requires additional ethics clearance.
7. No field work activities may continue after the expiry date (12 September 2023). Submission of a completed research ethics progress report will constitute an application for renewal of Ethics Research Committee approval.
8. Field work activities may only commence from the date on this ethics certificate.

Note:

The reference number 044/KTM/2018/CSET_SOC *should be clearly indicated on all forms of communication with the intended research participants, as well as with the Unisa College of Science, Engineering and Technology's (CSET) Research and Ethics Committee.*

Yours sincerely

Dr. B Chimbo

Chair: Ethics Sub-Committee SoC, College of Science, Engineering and Technology (CSET)

Prof I. Osunmakinde

Director: School of Computing, CSET

Prof B. Mamba

Executive Dean: CSET

University of South Africa
Preller Street, Muckleneuk Ridge, City of Tshwane
PO Box 392 UNISA 0003 South Africa
Telephone: +27 12 429 3111 Facsimile: +27 12 429 4150
www.unisa.ac.za

Appendix B

Cover Letter To An Online Anonymous Web-based Survey

UNISA university of south africa

Ethical clearance #: 044/KTM/2018/CSET_SOC
Research permission #:

COVER LETTER TO AN ONLINE ANONYMOUS WEB-BASED SURVEY

Dear Prospective participant,

You are invited to participate in a survey conducted by Mr KT Masombuka under the supervision of Prof. E Mnkandla a professor in the Department of information systems towards a PhD at the University of South Africa.

The survey you have received has been designed to study the acceptance of DevOps collaboration culture. You were selected to participate in this survey because you are a DevOps Developer or Operator. By completing this survey, you agree that the information you provide may be used for research purposes, including dissemination through peer-reviewed publications and conference proceedings.

It is anticipated that the information we gain /from this survey will help us to build a model for the acceptance of DevOps collaboration culture. You are, however, under no obligation to complete the survey and you can withdraw from the study prior to submitting the survey. The survey is developed to be anonymous, meaning that we will have no way of connecting the information that you provide to you personally. Consequently, you will not be able to withdraw from the study once you have clicked the send button based on the anonymous nature of the survey. If you choose to participate in this survey it will take up no more than 15 minutes of your time. You will not benefit from your participation as an individual, however, it is envisioned that the findings of this study will help organisations to successfully adopt DevOps practices. We do not foresee that you will experience any negative consequences by completing the survey. The researcher(s) undertake to keep any information provided herein confidential, not to let it out of our possession and to report on the findings from the perspective of the participating group and not from the perspective of an individual.

University of South Africa
Preller Street, Muckleneuk Ridge, City of Tshwane
PO Box 392 UNISA 0003 South Africa
Telephone: +27 12 429 3111 Facsimile: +27 12 429 4150
www.unisa.ac.za

The records will be kept for five years for audit purposes where after it will be permanently destroyed. Electronic versions will be permanently deleted from the hard drive of the computer. You will not be reimbursed or receive any incentives for your participation in the survey.

The research was reviewed and approved by the School of Computing Ethics Review Committee. The primary researcher, Mr KT Masombuka, can be contacted during office hours at 011 670 9123. The study leader, Prof. E Mnkandla, can be contacted during office hours at 011 670 9059. Should you have any questions regarding the ethical aspects of the study, you can contact the chairperson of the School of Computing Ethics Research Committee, Dr B Chimbo at 011 670 9105. Alternatively, you can report any serious unethical behaviour at the University's Toll Free Hotline 0800 86 96 93.

You are making a decision whether or not to participate by continuing to the next page. You are free to withdraw from the study at any time prior to clicking the send button.

Consent Form

UNISA university of south africa

CONSENT TO PARTICIPATE IN THIS STUDY

I, ___________________ (participant name), confirm that the person asking my consent to take part in this research has told me about the nature, procedure, potential benefits and anticipated inconvenience of participation.

I have read (or had explained to me) and understood the study as explained in the information sheet.

I have had sufficient opportunity to ask questions and am prepared to participate in the study.

I understand that my participation is voluntary and that I am free to withdraw at any time without penalty (if applicable).

I am aware that the findings of this study will be processed into a research report, journal publications and/or conference proceedings, but that my participation will be kept confidential unless otherwise specified.

I agree to the recording of the <insert specific data collection method>.

I have received a signed copy of the informed consent agreement.

Participant Name & Surname…………………………………………… (please print)

Participant Signature……………………………………………… .Date…………………

Researcher's Name & Surname……………………………………… (please print)

Researcher's signature…………………………………………….. Date…………………

University of South Africa
Preller Street, Muckleneuk Ridge, City of Tshwane
PO Box 392 UNISA 0003 South Africa
Telephone: +27 12 429 3111 Facsimile: +27 12 429 4150
www.unisa.ac.za

Appendix D

Questionnaire

DevOps Collaboration Culture Questionnaire

Section 1: Bibliographic information

1. **Age:** 18-25☐ 26-35 ☐ 36-45☐ 46- above☐
2. **Gender: Male☐ Female ☐**
3. **Current Role**: Developer☐ Operator☐ Both☐
4. **Experience in the above role:**

Less than 2 years☐ 2-5 years☐ 6-10 years ☐ above 10 years☐

Section 2: Social Norm

5. Most people who are important to my career would approve my use of DevOps collaboration.
6. Most people who are important to my career would tend to encourage me to use DevOps collaboration.
7. Most people who are important to my career would admire my use of DevOps collaboration.
8. Most people who are important to my career would tend to encourage me to accept DevOps collaboration.
9. Most people who are important to my career would appreciate my use of DevOps collaboration.
10. Most people who are important to my career would tend to encourage me to embrace DevOps collaboration culture.

Section 3: Open Communication

11. I approve frequent meetings with my team members.
12. Meeting with my team members regularly is easy for me.
13. Meeting with my team members regularly is problem free.
14. Regular meetings are good for the project.
15. I would attend all scheduled meetings with team members.
16. I would promote regular meetings with team members.

Section 4: Roles and Responsibility alignment

17. My roles and responsibilities to the team are clear.
18. My roles and responsibilities to the team are easy for me.
19. Understanding my roles and responsibilities is beneficial to the team.
20. Understanding the scope of my roles and responsibilities is my priority.
21. I understand my role and responsibilities to the team.
22. I understand the scope of my role and responsibilities to the team.

Section 4: Organisational Usability

23. My use of DevOps collaboration would make my organisation more successful.
24. My use of DevOps collaboration would be beneficial for my organisation.
25. The benefits to the organisation of using DevOps collaboration would substantially outweigh the cost.
26. My use of DevOps collaboration would make my organisation competitive
27. DevOps collaboration is useful to my organisation.
28. My using of DevOps collaboration will be helpful to my organisation

Section 5: Perceived Behavioural Control-Internal

29. Based on my personal background, using DevOps collaboration practices would be easy for me.
30. Based solely on my knowledge, skills and abilities, using DevOps collaboration practices would be problem free.
31. Based on my personal background, using DevOps collaboration practices would be straightforward.
32. Based solely on my knowledge, skills and abilities, using DevOps collaboration practices would be straightforward.
33. My background allows me to use DevOps collaboration comfortably.
34. My knowledge, skills and abilities are favourable for DevOps collaboration.

Section 4: Respect

35. I feel free to contribute my opinions to the team.
36. My team members value my contributions to the team.
37. My team members respects ideas I bring into the team.
38. My team members makes me feel part of the team.
39. I enjoy the attention my team member give me.
40. The respect I receive from my team members in high.

Section 5: Intentions to follow DevOps

41. It is my personal goal to follow DevOps collaboration culture.
42. My personal level of commitment to following DevOps collaboration culture is high.
43. My personal intention to use DevOps collaboration culture is high.
44. Following DevOps collaboration culture make me feel positive.
45. I feel comfortable using DevOps collaboration.
46. I like DevOps collaboration practices
47. My commitment to DevOps collaboration practices is strong.
48. I need to use DevOps collaboration culture

Section 6: Trust

49. I have confidence in my team members.
50. I trust the abilities and skills of my team members.
51. My team members are reliable.
52. I trust the intentions of my team members on this team.
53. The intentions of my team members on this team are good.
54. My level of trust to this team is high.

Language Editor Certificate

THEEDITINGDOCTOR(PTY)LTD

COMPANY REGISTRATION NO.: 2017/053873/07; ADDRESS: 48 Acacia Glen, Trefnant Road, Ormonde, 2091, Republic of South Africa;
MOBILE: + 27 (0) 61 544 2641; WHATSAPP: + 27 (0) 61 540 6472; E-MAIL: info@TheEditingDoctor.com

EDITORIAL CERTIFICATE

This certificate serves to confirm that the manuscript listed below was edited by one or more professional scientific editors with a PhD. The University or the student's supervisor(s) can contact us for a copy of the edited document that was submitted to the author(s).

Manuscript Title:

A model for successful collaboration in DevOps environments

Authors:

Themba Masombuka

Date Issued:

08 January 2020

Certificate Number:

21-122019 UNISA

TheEditingDoctor(Pty)Ltd works with researchers from academic and research organizations in the fields of science, engineering and technology to get their work ready for examination and/or publication in internationally acclaimed English journals. Our areas of specialization include Editing (i.e. editing poor syntax, poor general structure, repetition, redundancy, contradiction or inconsistency) and Rewriting (i.e. rewriting of phrases, sentences, paragraphs or sections to achieve better flow, more coherence clearer logic, and more appropriate vocabulary).

Turnitin Report