A Leadership Model for DevOps Adoption within Software Intensive Organisations

HENLEY BUSINESS SCHOOL
THE UNIVERSITY OF READING

A thesis submitted to the University of Reading in fulfilment of the requirements for the degree of Doctor of Philosophy in Informatics and System Science

Krikor Maroukian

2022

Declaration

I confirm that this is my own work and the use of all material from other sources has been properly and fully acknowledged.

Krikor Maroukian

Acknowledgements

The realisation of a grandiose endeavour, such as the multi-year committed research time and dedication to produce a doctoral thesis, would not have been possible without the psychological and spiritual encouragement of my family – Lorin (my wife) and my youngster duality Lia and Arek. I immensely appreciate all the kind words of encouragement by my brother, Khazak, Julia and sounds of baby Alex. The continuation of another generation producing a doctoral thesis after my father (Prof. Emeritus Hampik Maroukian) defended his, back in 1983 at National and Kapodistrian University of Athens, Greece following the award of an MSc at Louisiana State University in 1975, is a celebratory moment for the family-tree. I also know how proud this achievement makes my mother (Shogher) and mother-in-law (Talin).

I am thankful to my supervisor Assoc. Prof. Stephen R. Gulliver for his continued support, impeccable judgment, wise words of advice, and encouragement during this journey. I commend his availability to always find time to schedule meetings with me, his resiliency to contribute with insightful ideas as well as to provide unbiased feedback from an entirely objective perspective. I will reminisce those in-depth philosophical discussions and meetings, which evolved around several research areas, and which led to a different thinking process in what proved to be later the fulfilment of a beautifully orchestrated series of academic publications – to him I recite "*be ahead of the times through endless creativity, inquisitiveness and pursuit of improvement.*" – from The Toyoda Precepts (one of Five Main Principles of Toyoda). Without you the research wouldn't have been realised.

I also extend my thanks to the extended BISA team which I've grown to appreciate through the path of metamorphosis to become a researcher; Prof. Yinshan Tang, Prof. Kecheng Liu and Prof. Keiichi Nakata, for their continued support especially during the annual PhD progress presentations and transfer viva. Special thanks to the Admin team of Informatics Research Center (IRC) including IRC PhD colleagues who invited me to be involved in their research and also shared research insights and practices with me - especially Dr. Buddhi Pathak and Dr. Samnan Ali. Special thanks to Dr. Harris Apostolopoulos who has always been an ingenious pillar of support, extending a helpful arm when challenging times have required it. A warm note of appreciation to Konstantinos Tsoukalas for his strong willingness to keeping me on track with my aspirational planning on individual health and wellbeing.

I also extend my thanks to the extended BISA team which I've grown to appreciate through the path of metamorphosis to become a researcher; Prof. Yinshan Tang, Prof. Kecheng Liu and Prof. Keiichi Nakata, for their continued support especially during the annual PhD progress presentations and transfer viva. Special thanks to the Admin team of Informatics Research Center (IRC) including IRC PhD

colleagues who invited me to be involved in their research and also shared research insights and practices with me - especially Dr. Buddhi Pathak and Dr. Samnan Ali. Special thanks to Dr. Harris Apostolopoulos who has always been an ingenious pillar of support, extending a helpful arm when challenging times have required it. A warm note of appreciation to Konstantinos Tsoukalas for his strong willingness to keeping me on track with my aspirational planning on individual health and wellbeing. My utmost appreciation and respect goes out to all the Microsoft Worldwide Connected Communities who circulated and actively participated in the research-based data collection process. Lastly, I'd like to extend a thankful note to - DevOps Dojo Community of Practice members - who widely disseminated and shared research results and outcomes as part of the DevOps Dojo Lean Product.

A sincere and wholehearted 'Thank You' to all of you.

Related Publications

Publications related to the research described herein this thesis, have been published in journals, conference proceedings or are currently under revision of respectable journals. These are described in tabulated in table 0.1.

Table 0.1 – Published work relating to the content of the Thesis.

#	Conference/Journal	Paper Title	Link	Result
CON.01	Proceedings of the 9th International Conference on Information Technology Convergence and Services (ITCSE 2020). AIRCC, Computer Science and Information Technology, 41-56. *(ISBN : 978-1-925953-19-0)*	Leading DevOps Practice and Principle Adoption	http://airccse.org/csit/V10N05.html	Presented on Sunday, 31st of May 2020
CON.02	Proceedings of the 9th International Conference on Computing and Pattern Recognition (ICCPR 2020)	Towards Practice and Principle Adoption Through Continuous DevOps Leadership	http://www.jsoftware.us/vol16/429-TE01.pdf	Presented on Sunday, 1st of November 2020
CON.03	Proceedings of the 15th International Conference on Software Engineering Advances (ICSEA 2020)	Defining Leadership and its Challenges while Transitioning to DevOps	https://www.iaria.org/conferences2020/ICSEA20.html	Presented on Thursday, 22nd of October, 2020
CON.04	Proceedings of the European Symposium on Software Engineering (ESSE 2020) *(ISBN: 978-1-4503-7762-1)*	The link between transformational and servant leadership in DevOps-oriented environments	https://dl.acm.org/doi/10.1145/3393822.3432340	Presented on Thursday, 5th of November, 2020 and Published in Dec 2020
CON.05	Proceedings of the European Symposium on Software Engineering (ESSE 2021) ACM Conference Proceedings *(ISBN: 978-1-4503-8506-0)*	Synthesis of a Leadership Model for DevOps Adoption	https://dl.acm.org/doi/10.1145/3501774.3501783	Presented on Sunday, 21st of November, 2021 and Published in Feb 2022
JRN.01	Advanced Computing: An International journal (ACIJ)	Exploring the Link Between Leadership and DevOps Practice and Principle Adoption	http://airccse.org/journal/acij/vol11.html	Published in June 2020
JRN.02	Journal of Software. JSW, ISSN 1796-217X, Vol. 16, No. 1, Jan. 2021.	Practice and principle adoption through DevOps leadership	http://www.jsoftware.us/index.php?m=content&c=index&a=show&catid=224&id=3018	Published in Jan 2021

Abstract

The research, undertaken in organisational environments within IT-oriented culture and highly structured processes, outlines challenges and benefits associated to the adoption of Agile, Lean, and DevOps practices and principles. Realizing the adoption of DevOps practices and principles is no longer restricted to technology-specific skills. Studies indicate that successful DevOps adoption is part of continuous organisational transformation at various levels, and that includes a shift in cultural and behavioural patterns, process-driven perspectives, and toolchain usage readiness. There are also DevOps models to suggest an adoption roadmap for organisations to follow through the transitional path from existing highly structured processes to agile and lean approaches. However, there is a considerable lack of validated adoption models which are inclusive of leadership styles, traits, characteristics and the connection to adoption success or failure. This thesis details the explanation of product development approaches and using a mixed methods approach aims to provide proof and evidence to support the answers towards three research questions.

The approach collected data through thirty interviews with industry practitioners, who were from ten countries working in nine different industry sectors. Almost two-thirds of interviewees had practiced DevOps. A set of agile, lean and DevOps practices and principles, which organisations choose to include in their DevOps adoption journeys were identified. The most frequently adopted structured service management practices, contributing to DevOps practice adoption success, indicate that those with software development and operation roles in DevOps-oriented organisations benefit from existence of highly structured service management approaches such as ITIL®. Furthermore, coded themes were generated based on the thirty interviews to expand understanding of relevant factors and produce the structure of an online survey.

The analysis and evaluation of the online survey (n=250) confirmed some of the initial findings of the interviews and expanded viewpoints on other perceived outcomes. Out of the total 250 participants, 81% had 10+ years of professional experience and two-thirds were practicing DevOps. 73% of participants were from Europe and 76% had held previous leadership positions. The aim of the survey was to unveil leadership-specific observations on characteristics and factors that would indicate certain reasoning behind challenges faced by organisations while transitioning to DevOps.

The research questions which evolved around (R1) an understanding of how productivity can be improved for software product development teams, indicated that there is a specific set of service management, project management and product management practices and principles to take into account. Furthermore, evidence produced from the qualitative and quantitative studies confirmed (R2) that DevOps-oriented organisations have mainly preferred to extend the structured approaches

previously adopted such as ITIL®v3. The online survey produced significant evidence (R3.a) of industry practitioners' desire to have a leadership role for the purposes of DevOps practice and principle adoption. (R3.b) The emergent leadership style pertinent to the transition of IT-focused organisations to DevOps condensed to the linkage of transformational and servant leadership.

The observations from the confirmatory study of the online survey (n=250) contributed to the design and development of a conceptual model which emphasizes. The conceptual model was validated using PLS-SEM to improve understanding of significance and predictive power of construct validity and corresponding manifest variables. The final, model evaluation research stage of three focus group interviews (n=19), indicated industry practitioner consensus on the validated model in a range of 70% - 79%.

The thesis outcomes formulate a leadership model towards the fulfilment of DevOps adoption within organisations. The thesis outcomes also aim to support the transitional efforts to DevOps, and commitment of software-intensive i.e. enterprise with an IT organisation. In this way, it become possible to enhance the competence level of an organisation's adoption capability, guide DevOps adoption leadership through its upskilling journey, and achieve the cultural shift of mindset to enable continuous and habitual change.

Contents

Declaration ii
Acknowledgements iii
Related Publications v
Abstract vi
List of Figures xii
List of Tables xiv
List of Abbreviations xvii
Chapter One 18
Introduction 18
1.1. Overview 18
1.2. Research Background and Motivation 18
1.3. Research Problem 20
1.4. Outline of Research Questions, Aim and Objectives 21
1.5. Mapping of Research Questions, Aim and Objectives 23
1.6. Expected Research Contribution 23
1.6.1. Theoretical Contribution 25
1.6.2. Methodological Contribution 25
1.6.3. Practical Contribution 25
1.7. Thesis Structure 26
Chapter Two 28
Literature Review 28
2.1. Chapter Overview 28
2.2. Structured Approaches to Software Product Development 28
2.2.1. IT Service Management 28
2.2.2. IT Project Management 29
2.2.3. Adoption Challenges 31
2.3. Agile Product Development 32
2.3.1. Agile Software Development 32
2.3.2. A Retrospective View of Agile 34
2.3.3. Adoption Benefits 36
2.3.4. Adoption Challenges 37
2.4. The Lean Movement 38
2.4.1. The Lean Mindset 38
2.4.2. History of Lean Management 39
2.4.3. Adoption Benefits 42
2.4.4. Adoption Challenges 42

2.5. DevOps and its Adoption 43
2.5.1. Defining DevOps 43
2.5.2. Measuring DevOps 49
2.5.3. Adoption Benefits 52
2.5.4. Adoption Challenges 53
2.6. Leadership Styles and Traits in Software-intensive Organisations 56
2.6.1. Definitions of Leadership 56
2.6.2. Transactional Leadership 57
2.6.3. Transformational Leadership 58
2.6.4. Servant Leadership 60
2.6.4 DevOps Adoption Leadership 60
2.7. Synopsis of Considerations on Research Aims and Questions 61
2.8. Chapter Summary 61
Chapter Three 63
Research Design and Methodology 63
3.1. Chapter Overview 63
3.2. Research Philosophy and Paradigms 63
3.2.1. Positivist Philosophy 64
3.2.2. Interpretivist Philosophy 65
3.2.3. Pragmatist Philosophy 66
3.2.4. Selected Research Philosophy 66
3.3. Research Design and Methods 67
3.3.1. Literature Review Search Design 67
3.3.2. Phase One (Exploratory study) – Qualitative Method (Interviews) 69
3.3.3. Phase Two (Confirmatory study) – Quantitative Method (Survey) 72
3.3.4. Phase Three (Model Evaluation) – Qualitative Method (Focus Groups) 76
3.4. Research Stages Flow 80
3.5. Ethical Considerations 80
3.6. Validation 81
3.6.1 Potential Risk to Internal Validity 81
3.6.2 Potential Risk to External validity 81
3.7. Chapter Summary 82
Chapter Four 83
Exploratory Study Analysis and Outcomes 83
4.1. Chapter Overview 83
4.2. Introduction 83
4.3. Interview Process and Structure 83
4.4. Analysis and Evaluation 86

4.4.1. Interviewee background segmentation 86
4.4.2. Exploring and depicting the current research state 88
4.4.3 Thematic Analysis 98
4.4.4. Analysis of Interview Discussions 101
4.6 Chapter Summary 106
Chapter Five 107
Quantitative Research Analysis and Outcomes 107
5.1 Chapter Overview 107
5.2 Introduction 107
5.3 Survey Data Collection Process 107
5.4. Analysis and Evaluation 109
5.4.1. Survey Background Segmentation 109
5.4.2. Confirming Exploratory Study Findings 110
5.4.3 Design of a Conceptual Model of DevOps Adoption Leadership 115
5.4.4 Conceptual Model Hypotheses 119
5.5. Multivariate Data Analysis 120
5.6. Designing the PLS-SEM Model 131
5.6.1 Partial Least Squares – Structural Equation Modelling (PLS-SEM) 131
5.6.2 Develop Equation Path Model 133
5.6.3. Assess Measurement Model 135
5.6.4. Assess Structural Model for Significance 138
5.7 Chapter Summary 143
Chapter Six 144
Model Evaluation 144
6.1. Chapter Overview 144
6.2. Introduction 144
6.3. Analytic Frameworks in Focus Groups 144
6.4. Conducting Focus Groups and Model Evaluation 147
6.4.1. Planning and Conducting Focus Groups 147
6.4.2. Model Evaluation 149
6.5. Focus Group Data Collection Process 156
6.6 Chapter Summary 157
Chapter Seven 158
Conclusions and Future Work 158
7.1. Research Contribution 158
7.1.1 Theoretical Contribution 158
7.1.2. Methodological Contribution 158
7.1.3. Practical Contribution 159

7.2. Research Aim and Contributions 159
7.3. Discussion on Research Aim and Contributions 161
7.4 Future Work 165
References 167
Appendix A – Interview Questions 178
Appendix C – Consent to Interview Participation 190
Appendix D – Survey Questions 191
Appendix E – Terms of Survey Participation 196
Appendix F –Consent to Survey Participation 198
Appendix G – Focus Group Registration Questions 199
Appendix H – In-session Focus Group Questions 200
Appendix I –Consent to Focus Group Participation 202

List of Figures

Figure 1-1	Research Thesis Structure.	**24**
Figure 2-1	Process-based service management models.	**29**
Figure 2-2	Scrum methodology.	**35**
Figure 2-3	State of DevOps Report Participants in DevOps Teams.	**46**
Figure 3-1	The research onion.	**64**
Figure 3-2	Qualitative research – forms of interview.	**69**
Figure 3-3	Model Validation Research Stage.	**78**
Figure 3-4	Flow of research stages.	**80**
Figure 4-1	Job role of interview participants (interviewee count: 30).	**87**
Figure 4-2	Industry of interview participants (interviewee count: 30).	**88**
Figure 4-3	Experience level relevant to interview context in customer	**89**
Figure 4-4	Competence of interviewee according to practices (interviewee count: 30).	**90**
Figure 4-5	Most beneficial practices in order of preference (interviewee count: 30).	**91**
Figure 4-6	Competence of interviewee according to principles (interviewee count: 30).	**92**
Figure 4-7	Most beneficial principles in order of preference (interviewee count: 30).	**93**
Figure 4-8	Responses on whether agility and leanness can become an extension or replacement to highly structured approaches to software development (interviewee count: 30).	**94**
Figure 4-9	Areas to address in a DevOps transformation initiative (interviewee count : 30).	**95**
Figure 4-10	DevOps definition according to interviewees (interviewee count: 30).	**96**
Figure 4-11	Skills and capabilities required for DevOps adoption leadership (interviewee count: 30).	**97**
Figure 4-12	Organisational teams that should be part of a DevOps adoption process (interviewee count: 30).	**98**
Figure 4-13	Comparison of online interview transcription results between Temi and NVivo.	**99**
Figure 4-14	Coded themes generated in NVivo12.	**100**
Figure 5-1	DevOps adoption stage of survey participants (n=250).	**111**
Figure 5-2	DevOps adoption metrics indicated by survey participants.	**113**
Figure 5-3	Conceptual Model for DevOps Adoption Leadership.	**116**
Figure 5-4	Holistic view of continuous practices in business, development, operations, and innovation.	**117**
Figure 5-5	Classification Chart of Multivariate Techniques.	**120**

Figure 5-6	Redundancy analysis of ‘Metrics’ Manifest Variable.	**133**
Figure 5-7	Redundancy analysis of ‘Practices’ Manifest Variable.	**133**
Figure 5-8	Redundancy analysis of ‘Principles’ Manifest Variable.	**134**
Figure 5-9	Redundancy analysis of ‘Skills’ Manifest Variable.	**134**
Figure 5-10	Formative Measurement Model.	**136**
Figure 5-11	Formative Structural Model.	**139**

List of Tables

Table 0-1	Published work relating to the content of the Thesis.	**iv**
Table 1-1	Mapping of research questions to aims and objectives.	**23**
Table 2-1	Chaos Report Project Failure Figures.	**30**
Table 2-2	Lean principles relevant to software development.	**39**
Table 2-3	Types of waste in Lean IT.	**41**
Table 2-4	Diversity of DevOps definitions.	**44**
Table 2-5	Agile, Lean and DevOps mapping.	**49**
Table 2-6	Leadership definitions.	**56**
Table 2-7	Characteristics of Transformational Leaders.	**59**
Table 3-1	Comparison of three research philosophies.	**65**
Table 3-2	Qualitative research - forms of interview	**72**
Table 4-1	Interview participant profile.	**85**
Table 4-2	Research to interview questions mapping.	**86**
Table 4-3	ITSM process significance to value delivery of software development (interviewee count: 30).	**102**
Table 5-1	Research Questions to Survey Questions Mapping.	**109**
Table 5-2	Decision-making role in DevOps adoption process.	**112**
Table 5-3	Characteristics of Leadership Styles.	**115**
Table 5-4	Survey questions structure by type mapped to statistical analysis technique.	**121**
Table 5-5	Spearman Rho and Pearson correlation for the model construct 'DevOps Adoption Leader'.	**122**
Table 5-6	Spearman Rho and Pearson correlation for the model construct 'Practices and Principles'.	**122**
Table 5-7	Spearman Rho and Pearson correlation for the model construct 'Practices and Principles'.	**123**
Table 5-8	Spearman Rho and Pearson correlation for the model construct 'Practices and Principles'.	**123**
Table 5-9	Spearman Rho and Pearson correlation for the model construct 'Practices and Principles'	**123**
Table 5-10	Spearman Rho and Pearson correlation for the model construct 'DevOps Adoption Leader Metrics'.	**124**
Table 5-11	Spearman Rho and Pearson correlation for the model construct 'DevOps Adoption Leader Metrics'.	**124**

Table 5-12	Spearman Rho and Pearson correlation for the model construct 'DevOps Adoption Leader Metrics'.	**124**
Table 5-13	Spearman Rho and Pearson correlation for the model construct 'DevOps Adoption Leader Metrics'.	**125**
Table 5-14	Spearman Rho and Pearson correlation for the model construct 'DevOps Adoption Leader Metrics'.	**125**
Table 5-15	Spearman Rho and Pearson correlation for the model construct 'DevOps Adoption Leader Metrics'.	**125**
Table 5-16	Spearman Rho and Pearson correlation for the model construct 'DevOps Adoption Leader Metrics'.	**126**
Table 5-17	Spearman Rho and Pearson correlation for the model construct 'DevOps Adoption Leader Metrics'.	**126**
Table 5-18	Spearman Rho and Pearson correlation for the model construct 'DevOps Adoption Leader Metrics'.	**126**
Table 5-19	Spearman Rho and Pearson correlation for the model construct 'DevOps Adoption Leader Metrics'.	**127**
Table 5-20	Spearman Rho and Pearson correlation for the model construct 'DevOps Adoption Leader Skillset'.	**127**
Table 5-21	Spearman Rho and Pearson correlation for the model construct 'DevOps Adoption Leader Skillset'.	**127**
Table 5-22	Spearman Rho and Pearson correlation for the model construct 'DevOps Adoption Leader Skillset'.	**128**
Table 5-23	Spearman Rho and Pearson correlation for the model construct 'DevOps Adoption Leader Skillset'.	**128**
Table 5-24	Spearman Rho and Pearson correlation for the model construct 'DevOps Adoption Leader Skillset'.	**128**
Table 5-25	Spearman Rho and Pearson correlation for the model construct 'DevOps Adoption Leader Skillset'.	**129**
Table 5-26 a-c	Spearman Rank Hypothesis Testing.	**128-130**
Table 5-27	Model Constructs and Indicator List and Descriptions.	**137**
Table 5-28	Outer VIF values of model.	**138**
Table 5-29	Formative Constructs PLS-SEM Testing Results.	**140**
Table 5-30	Assessment of Formative Structural Model Relevance of Path Relationships (Total Effects).	**140**
Table 5-31	Formative Path Relationship R^2.	**141**
Table 5-32	Formative Constructs f^2.	**141**

Table 5-33	Blindfolding and Predictive Relevance Q^2 (Cross-validated Redundancy).	**142**
Table 5-34	Effect Size q^2 (MV Prediction Summary).	**142**
Table 6-1	Model Validation for all Research Lifecycle Phases.	**146**
Table 6-2	Focus Groups Participant List.	**148**
Table 6-3	Focus Groups Participant Agreement Levels for Conceptual Model.	**155**
Table 6-4	Research questions to focus group survey questions mapping.	**156**
Table 7-1	Research questions to findings mapping.	**160**
Table 7-2	Characteristics of Leadership Styles.	**164**

List of Abbreviations

BCS	British Computer Society
CSF	Critical Success Factor
ITIL®	IT Infrastructure Library
ITSM	IT Service Management
itSMF	IT Service Management Forum
GDPR	General Data Privacy Regulation
KPI	Key Performance Indicator
LM	Lean Management
KP	Lean Production
SLO	Service Level Objectives
TPS	Toyota Production Systems
TBP	Toyota Business Practices
PMI	Project Management Institute
PRINCE®	Projects IN Controlled Environments
PMBOK®	Project Management Body of Knowledge
TDD	Test Driven Development
UC	Underpinning Contract

Chapter One
Introduction

1.1. Overview

In a world where every enterprise is striving to deliver its promised value to its customers in a rapid, reliable, and repeatable fashion, and in the face of plethora of global socioeconomic challenges and multi-waves of pandemic and crisis lockdowns, it feels like it's a non-pragmatic goal to achieve. Reengineering and optimising the way value is delivered, by placing the workforce at the epicentre of informed decision making, i.e. to lead the adoption of agility and leanness in the everyday workspace, can become a precursor when facing uncertain change. In that context, software product development, especially in knowledge-intensive industry domains, is at a precipice. Software has become pervasive in day-to-day human activities, and the world economy is now dependent on software use. This in turn has increased the importance of having software-intensive products and services, which are consumed in a useful, secure, and reliable manner, consistently throughout operational use. Recently, the ability of an organisation to continuously and rapidly deliver new suitable and innovative software features, has become an important factor of competition in the software industry market. Given enough time, any software-intensive organisation can develop great software products and services. Due to this, speed to market is one of the most important success factors in the software product development industry.

This introductory chapter establishes key aspects of research background and motivation in section 1.2, which led the researcher of the thesis to embrace a logical approach to the research problem in section 1.3. Additionally, the research questions, aims and objectives are outlined in section 1.4 and mapped in section 1.5 for increased clarity and conciseness of what topics are to be investigated regarding agile, lean and DevOps adoption practices and principles. The expected research contribution in Section 1.6 is thoroughly explained in the dimensions of theoretical contribution in subsection 1.6.1, methodological contribution in subsection 1.6.2, and practical contribution in subsection 1.6.3.

1.2. Research Background and Motivation

The Fourth Industrial Revolution, characterised by the growing utilisation of disruptive digital technologies, is transforming the world of work. Research by Harvard Business Review (HBR 2017) suggests that

globally about half of the jobs performed today by humans will be disrupted in some way by automation, and a report from the World Economic Forum (WEF 2020) stated that 42% of the core job skills required today are set to substantially change by 2025. In addition, leading cultural change will be key to digital business transformation (Larjovuori et al., 2018). Within this dynamically changing business world, use of software management is playing a much larger and more strategic role in shaping how companies compete, with large 'traditional' organisations finding themselves limited in their ability to respond to market and customer needs.

Software has become pervasive in day-to-day human activities, and the world economy is now dependent on software use. This in turn has increased the importance of having software-intensive products and services that are useful, secure, and reliable at all times during operational use. The 1990s saw the birth of pre-agile approaches, such as the Rapid Unified Process (Kruchten, 1999) and XP (Beck, 2000; Fowler, 1999), which eventually led to Agile Software Development, characterised mainly by lightweight, flexible, adaptive processes linked to rapidly changing corporate business environments aiming to eliminate waste (Poppendieck and Poppendieck, 2003). The traditional 'waterfall' approach to release and deployment management requires a release cycle of 6-18 months, which shifts focus to reactive natured maintenance activities and support of bug fixes. This practically means that operations teams in Information Technology (IT) organisations are not focused on project or product-based tasks which are executing IT strategy and ultimately lead to the fulfilment of corporate strategic aspirations. There is, however, a lack of development of new feature development, i.e., change of features or functions that would fundamentally change the program architecture (Alahyari et al., 2019).

A retrospective view of the last twenty years of software product development practices and principles shows that a decline of Extreme Programming (XP) publications has been succeeded by the gradual increase (since about 2009) in the popularity of agile and lean practices; such as SCRUM (Sutherland et al., 2017) and Kanban (Anderson, 2010). Moreover, two other areas that seem to be gaining popularity are technical debt and code smells, which address software product development and code maintenance sub-optimisation - in terms of agile team velocity to deliver sprint artefacts for the minimum-viable product. Furthermore, certain agile practices, e.g., pair-programming since 2003, user stories since 2003, test-driven development since 2007, and code refactoring since 2009, are relatively stable. In addition, DevOps, Continuous Integration, Continuous Deployment, Continuous Delivery are characterised as 'hot research topics', with considerable increases in popularity since 2014 (Rodriguez, 2019).

Leading DevOps practice and principle adoption has become a fundamental element to the success of DevOps teams (Poppendieck and Poppendieck, 2003) (Maroukian and Gulliver, 2020). A recent systematic

literature review indicates that a lack of DevOps approach use, a lack of effective management, and a lack of trust and confidence, are considered in the top 10 most significant issues at hand with DevOps adoption transformational initiatives within organisations (Khan et al., 2022). A high-performing organisation is characterised by adoption of DevOps practices by multiple teams and departments, high responsiveness to mean-time-to-recover from product system failure, i.e., end-user experience degradation, mean-time-to-market, change failure rate, and embedding security deep into the source code (Geurts, 2016). However, there is still limited research outlining the leadership style, traits, competencies, and skillset accompanied with high-performing DevOps-oriented organisations. Speed in the development and delivery of new software features provides the opportunity to respond quickly to customer needs, business opportunities, and get quick feedback about the new software features (Schlossnagle, 2017).

1.3. Research Problem

Organisations are increasingly involved with ways to unlock the full potential of their workforce. With regard to this, IT organisations have increased the importance of having software-intensive products and services that are secure, reliable and adding value to the business at all times during operational use. Software intensive organisation is defined as the system where software contributes essential influences to the design, construction, deployment, and evolution of the system as a whole (IEEE 2000). IT organisations are under pressure to both support existing products and develop new versions of these products with additional features and fewer defects. Therefore, the IT organisation is becoming increasingly accountable of delays experienced in delivering value to customers. Furthermore, there seems to be a significant need for business demand to be translated to frequent releases, powered by automated build, testing and deployment processes where automation reduces required effort to setup new product releases. In addition, business demand should be translated to more daily code commits-to-deployments with improved quality assurance, enhanced collaboration and communication means, improved visibility of implemented features to the customer, including testing with customers.

Structured IT service management frameworks such as ITIL®4 (AXELOS, 2019), project management frameworks such as PRINCE2® (AXELOS, 2017), and PMBOK® (PMI, 2017), have been introducing heavily structured approaches to the management of new products and services, within the IT organisation of various industries. The numerous decision points, process roles, compliance controls and documentation requirements that these structured approaches have created, have led to considerable accumulation of ‘technical debt’ e.g. most activities are labelled as ‘urgent’ and require expedited treatment and maintenance overheads, such as keep-the-lights-on activities, within the IT organisation. Consequently, development, quality assurance, information security and operations teams have had to face, on a daily basis, increasing

operational overheads and collaboration barriers. This situation has inhibited the IT organisation's productivity and indicated weaknesses in the current IT organisation's model to accomplish projects 'on time-on budget-on scope' (OTOBOS) and deliver product value to rapidly shaping market demand and end-user communities.

Agile software development (Beck 2000) (Poppendieck and Poppendieck, 2003), lean product development (Shingo et al., 1988), and DevOps (Kim, 2018; Forsgren et al., 2018) practices and principles all aim to identify the value adding activities in the change, release and deployment processes of an organisation, eliminate waste introduced by structured service management and project management approaches and aim for drastic decrease of time to recover from service outage experienced by customers and end-user populations. Therefore, there is clearly a major industrial need to examine whether structured management approaches require an extension to agility and leanness or whether these 'traditional' service management practices should be deemed obsolete in today's software-intensive and knowledge-intensive modern organisation and entirely replaced. The transformation from a framework or process-led organisational environment to the adoption of groups of best practices, entails a significant shift in mindset, skillset and ultimately toolset usage.

1.4. Outline of Research Questions, Aim and Objectives

Following the outline of the problem in Section 1.3, the developed research questions can be synthesised in the context of software product development and its overlap with IT service management its practices and principles. Firstly, a research question should investigate the set of practices and principles from Agile, lean and DevOps communities can help highly structured, 'traditional' organisations achieve increased productivity levels such as decreased time-to-market products and new features. Secondly, another research question is to identify whether Agile, Lean and DevOps practices form a continuation of structured service management approaches or a replacement. Thirdly, one final research question should focus on uncovering as much information as possible on the type of leadership style(s) which is required to succeed when adopting DevOps practices and principles.

The three research questions have a main research aim to investigate the impact of the adoption of Agile, Lean and DevOps practices and principles in business environments and explore how it is possible to lead adoption efforts through a specific model design for product development teams.

The aim can be mapped to objectives with a purpose to achieve an accurate mapping of the three research questions to research aims and objectives. Initially, a comprehensive list of benefits and challenges can be described for the transition journey from a structured approach to an agile and lean approach in software-

intensive organisations for software product development teams. Additionally, there is an identification process for the specific practices and principles, skills, and metrics, which together contribute to the success of DevOps adoption and time-to-market product value. Determining the required type of DevOps leader role skills and metrics in order to champion an organisation's efforts to adopt DevOps practices and principles is essential.

The holistic scoping and modeling of all aforementioned areas of DevOps adoption transition journeys will be meticulously considered as part of a sequence of three research questions which are thoroughly presented in tabulated format in Section 1.5.

1.5. Mapping of Research Questions, Aim and Objectives

Following the description of the research questions, aim and objectives the pertinent mapping is shown in Table 1-1.

Table 1-1 Mapping research questions to aim and objectives.

Research Aim	Research Questions	Research Method	Research Objectives
Investigation of the impact of the adoption of Agile, Lean and DevOps practices and principles in business environments and explore how it is possible to lead adoption efforts through a specific model design for product development teams	RQ1) Which agile, lean and DevOps practices and principles can improve productivity for software product development teams in software-intensive organisations that have adopted a structured service management approach?	RM1) Design and conduct interviews with thirty (30) practitioners from consultancy firms and customer organisations from ten (10) different countries between period Sep 2018 – Jan 2019.	RO1) The benefits and challenges that can be uncovered during a transition process from a structured to an agile and lean approach in software-intensive organisations for software product development teams.
	RQ2) Are DevOps-oriented environments and their teams an extension or a replacement of process-driven structured service management approaches?	RM2) Same as above and additionally, design and conduct an online survey with a maximum of 250 responses from practitioners of varying geographical regions between period Jul 2019 – Dec 2019.	RO2) Identify the specific practices of structured service management that can contribute to the success of DevOps adoption and time-to-market in the software development product lifecycle.
	RQ3) a) Can Leadership affect DevOps adoption within a software-intensive organisation? b) What is the leadership style that can be attributed to the DevOps adoption leader role?	RM3) Same as previous two aims and additionally develop a DevOps adoption leadership approach accompanied by a set of practical guidelines based on Qualitative and Quantitative Research evaluation and analysis.	RO3) Determine, through a validated model, the practices, principles, skillset, and metrics required of DevOps leaders to champion an organisation's efforts to adopt DevOps practices.

1.6. Expected Research Contribution

The thesis consists of eight chapters, presented in **Figure 1-1**, which depicts the flow and linkage in between the chapters.

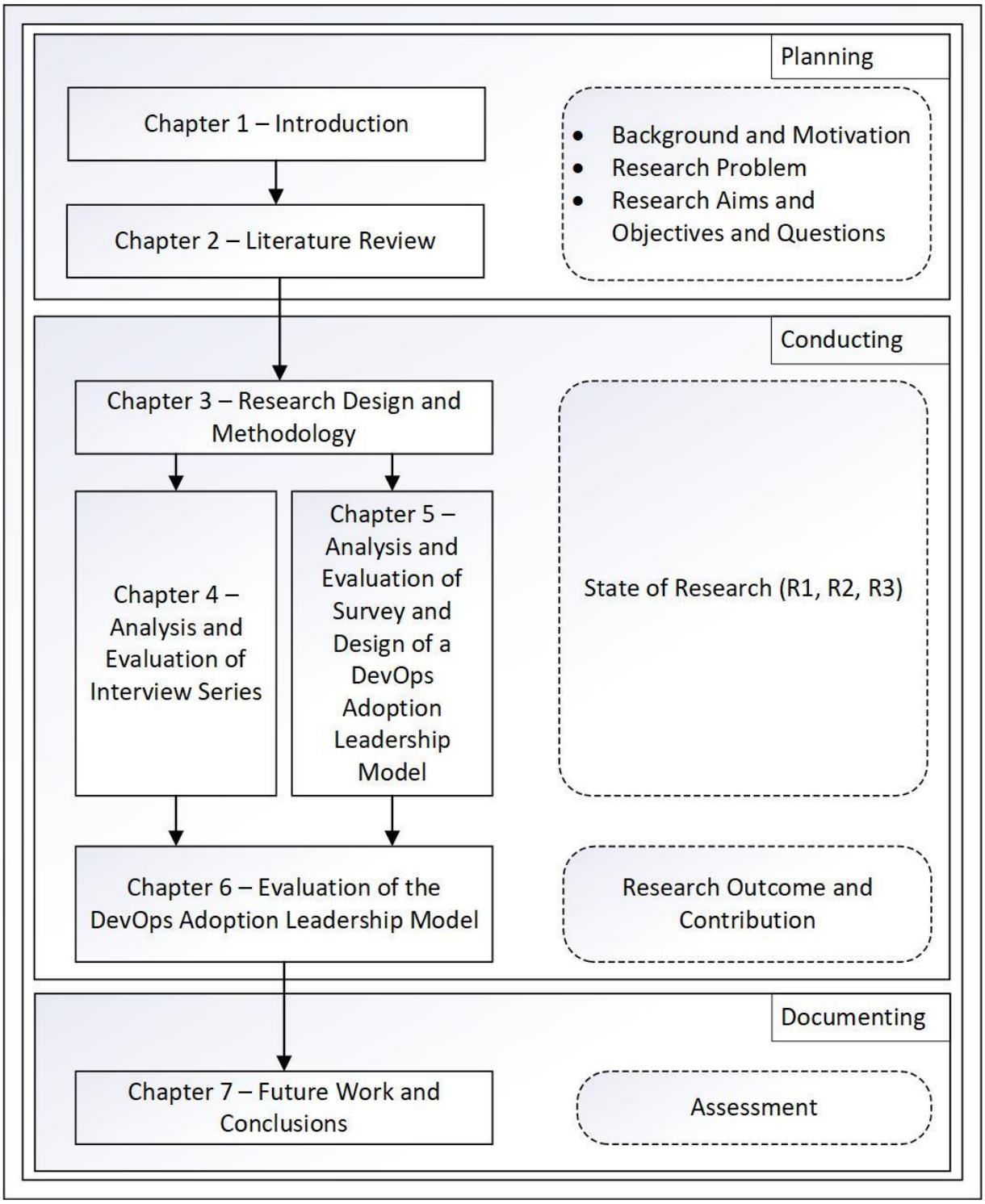


Figure 1-1 Research Thesis Structure.

1.6.1. Theoretical Contribution

The research reviews and analyses theories, models of legacy service management, agile software development, lean IT and DevOps practices and principles. The research reviews lead to a thorough investigation of the relationship between software product development team culture, behaviours and habits, and the potential to build a novel adoption capability and lead the adoption of new practice and principles. Consequently, key theoretical research contributions would include the (1) design and development of a novel approach to DevOps adoption leadership and (2) proper guidance to actually utilise such an approach. In the global, vast corporate landscape, which is increasingly deciding to embrace the transition towards a DevOps oriented environment, there is a very limited set of models to show how to lead DevOps adoption efforts by organisational teams which are committed to develop a capability that selectively applies a leadership style. This thesis investigates the impact of the adoption of Agile, Lean and DevOps practices and principles in business environments and explores how it is possible to lead adoption efforts through a specific model design for product development teams.

1.6.2. Methodological Contribution

The thesis aims to develop an approach to design, build, and model the capability for a software product development team to adopt DevOps practices and principles and lead its adoption. In addition, the thesis aims to enrich academic-oriented research which currently lacks both exploratory and confirmatory studies on the topic of DevOps adoption leadership. The validated model could be used systematically to help industry practitioners and organisations define the DevOps leadership role in the context of software product development lifecycle. This approach will help software developers, software testers, information security officers and operational roles to focus on developing the growth mindset to be inclusive of new behaviours and habits that trains the workforce to transform a challenging right-brain task into a routine left-brain task - thus achieving the process of unlearning and relearning a software product development skillset. This consistent effort of skillset mutation – reskilling, leads to the shift of mindset from legacy approaches, in service management, to the enablement of agility and leanness in everyday software product development activities.

1.6.3. Practical Contribution

Currently, there is very limited peer-reviewed and/or published validated guidance to support software product development teams to move towards DevOps practice and principle adoption by taking care of the required shift of mindset and skillset at technology-agnostic level. Moreover there isn’t any indication of the leadership style and traits that can be expected to be part of the DevOps adoption leader’s character. This thesis aims to support the DevOps adoption transitional efforts, and commitment of software-intensive

organisations - to enhance the competence level of an organisation's adoption capability, guide DevOps adoption leadership through its upskilling journey and achieve the cultural shift of mindset to enable habitual change.

1.7. Thesis Structure

The thesis consists of seven chapters, as presented in Figure 1-1, which depicts the flow and linkage between the chapters.

Chapter One introduces the research topic; the key drivers and motivation of the research undertaken, and identifies the research problem. Chapter one also specifies the research questions coupled with aims and objectives that are considered.

Chapter Two develops the motivation and background established in the introductory section, keeping in reference the research questions. Chapter 2 takes note of the frameworks, methodologies, and process-driven approaches; i.e. presenting software-intensive organisations since the end of 1980s in the context of service management. In addition, particular mention is made to agile software product development approaches and lean management practices and principles. The most up-to-date advances in the domain of software product development is covered as an emerging set of practices and principles widely known as DevOps. Relevant leadership styles and traits are also examined in the context of adoption of new practices and principles for software product development teams.

Chapter Three explains how this research will be conducted, and illustrates the epistemological and methodological assumptions that govern the research design. The application, examination and justification of a mixed methods approach is described, which is also mapped to a specific set of qualitative and quantitative data. A discussion on benefits and limitations of each research method is also included. Moreover, internal, and external validity is also discussed.

Chapter Four aims to evaluate the current state of research in structured service management frameworks and methodologies – including agile, and lean practices - through a series of thirty interviews carried out with practitioners from software-intensive organisations. A transcript analysis of the interviews transposed to thematic analysis is also presented produced from NVivo.

Chapter Five extends the investigation of current state of research in DevOps by establishing the connection of qualitative research described in Chapter Four, to the structure of a published online survey which is attempted by 250 highly experienced industry practitioners. The results and outcomes of partial

least squares (PLS) PLS in structural equation modelling (SEM) analysis are outlined, leading to the design and development of a conceptual model for DevOps Adoption Leadership.

Chapter Six examines key concepts derived from chapters four and the model from chapter five to in a series of three focus groups which evaluates and validated the constructs and manifest variables that constitute the DevOps Adoption Leadership model. The high degree of agreement among focus groups participants is discussed, and serves as the baseline of a set of guidelines for software-intensive organisations willing to embrace DevOps practices and principles with specific measurements in place.

Chapter Seven discusses provides a conclusion to the thesis including related future work aspirations. This chapter outlines the research that has been carried out, linking content with the three research questions - initially posed in chapter one and justified in chapter two – and defined project aims and objectives. Chapter seven summarises the theoretical, methodological, and practical research contributions, and finished with critical discussion concerning different aspects and perspectives that can potentially give birth to further academic research.

Chapter Two

Literature Review

2.1. Chapter Overview

This chapter provides a detailed review of structured service management approaches, agile software development, lean IT, DevOps and relevant leadership styles, traits, and characteristics. Initially an overview of pre-agile and process-driven highly structured service management approaches in software product development is provided including a retrospective view of their birth and growing importance in the worldwide IT service management community and intersections with project management. Moreover, agile methodologies and models are discussed, which is followed by a discussion of lean management, lean production, and lean product; development coupled with mention of historical aspects of its birth in the car manufacturing industry and the transposition to non-manufacturing industries including the software industry. Additionally, DevOps models and relative practices and principles are clearly explored and reviewed. The last section of the chapter entails leadership styles, traits and characteristics are explained and illustrated with a series of reviewed common design theories. Finally, since the research literature forms an essential part of this research, some aspects are considered in greater detail in research chapters linking to specific research objectives presented in section 1.5.

2.2. Structured Approaches to Software Product Development

2.2.1. IT Service Management

IT Service Management (ITSM) is the discipline that aims to design, deliver, operate, and improve the way IT is used within and organisation. The primary aim of ITSM good practices is to ensure the right processes, people, and technology are in place, i.e. so that an organisation can meet its business goals. Globally known ITSM frameworks include:

- IT Infrastructure Library® (Axelos, 2019)
- VeriSM™ (Agutter, 2017)
- Microsoft Operation Framework™ (Microsoft 2008)
- Microsoft DevOps Dojo (Microsoft 2021)
- Integrated Service Management (ISM) Method (Hoving and Bon, 2012)
- Universal Service Management Body of Knowledge™ (Clayton 2008)
- ISO/IEC 20000:2018 International Standard (ISO/IEC 2018)

Figure 2-1 shows a timeline of the development of process-based service management models which clearly indicate the multiple iterations of evolution that preceded the most recent ITIL4® publication in 2019. Service management is "the management approach adopted by an organisation to deliver value to consumers though quality products and services." (Agutter, 2017). The most widely known ITSM practice-driven framework is the IT Infrastructure Library (ITIL®), currently in its fourth edition. ITIL® provides a framework of best practice for IT service management. ITIL was originally developed in the 1980s by the Cabinet Office, formerly UK Office of Government and Commerce (OGC), however has become recognised and used world-wide; focusing on the delivery and support of IT services appropriate to the business needs of the organisation. ITIL® includes guidelines for the entire lifecycle of an IT service, from concept to retirement or replacement. ITIL® coins the term *Service* as "a means of delivering value to customers by facilitating outcomes customers want to achieve without the ownership of specific costs and risks" (Axelos, 2019).

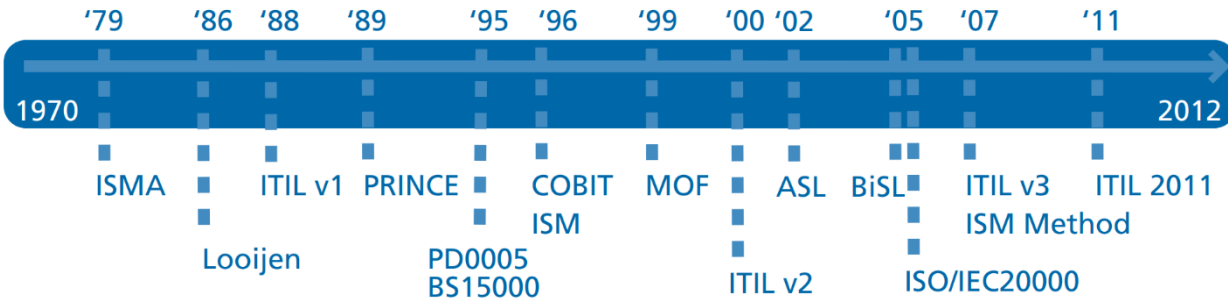


Figure 2-1 Process-based service management models.

Management should not be taught as a set of new production techniques and principles but rather to achieve systematic continuous evolution and improvement across the organisation by developing repeatedly and consistently applied behavioural routines (Rother, 2009).

2.2.2. IT Project Management

There are certain predominant global project management frameworks such as PMBOK® and PRINCE2® which have a significant impact and contribution to global teams performing according to a set of project goals with specific deliverables, whether that refers to IT, finance, banking, manufacturing, e-commerce, transportation, construction, healthcare sectors and other sectors. ISO21500 is an international standard to provide generic guidance, explain core principles and what constitutes good practice in project management. It can be used by any type of organisation, including public, private or community organisations, and for any type of project, irrespective of complexity, size, and duration. The PMP Handbook states that PMBOK® can be used with PRINCE2®. Both PRINCE2® and

PMBOK™®acknowledge each other's existence in their educational content, and attempt to position themselves as complementary products i.e. PRINCE2® as a 'methodology' and PMBOK® as a 'standard'. More specifically, **PR**ojects **IN C**ontrolled **E**nvironments (PRINCE2®) is a process-driven project management methodology, which was first published in 1990 with a focus on IT projects. PRINCE2® was revised for general project management and released in 1996, and significantly updated in 2009, i.e. to apply a defined sets of principles, themes and processes.

Globally established project management methodologies, such as PRINCE2®, provide information on what needs to be in place for an organisation, or a project team, at setup; i.e., to facilitate the project to its successful accomplishment in terms of scope, cost, and time. PRINCE2® also describes how to conduct effective project management following a specific set of rules.

According to the Anderson Economic Group (AEG) study commissioned by the Project Management Institute (PMI), by 2027 the project management-oriented labour force in seven project-oriented sectors is expected to grow by 33 percent; i.e., nearly 22 million new jobs. By 2027, employers will need nearly 88 million individuals in project management-oriented roles. China and India will represent more than 75 percent of the total project management-oriented employment sector. Talent shortages in the profession can potentially create risks of nearly US$208 billion in GDP over the 10-year period (2017-2027) in the 11 countries examined. Moreover, the Chaos report, published on an annual basis, provides a clear insight into global project failure rates with successful projects marginally increasing Year-on-Year (YoY), see Table 2-1.

Table 2-1 Chaos Report Project (Standish Group et al. 2020).

	Successful	Challenged	Failed
2015-2020	35%	47%	19%
2010-2014	38%	43%	19%
2005-2009	33%	46%	21%
2000-2004	30%	51%	19%
1995-1999	27%	43%	30%

Software project management involves the fulfilment of business requirements with humans interacting to build a set of specifications. Human interaction in turn can lead to frequently changing and ambiguously recorded information where different parties can provide different meaning to similar terminology. Such reasons significantly increase the possibility of failure to deliver enterprise systems, with failure rates reaching 70% of large-scale projects (McManus and Wood-Harper, 2008).

An extension to the previously stated failure factors of IS projects is identified by Pankratz and Basten, with fifty-four (54) failure factors in IS projects and grouped these factors in ten (10) categories applying data-driven qualitative content analysis (Pankratz et al., 2013). These were also grouped in eight (8) dimensions. Moreover, McManus and Wood-Harper (2008) showed that poorly defined requirements analysis is the main reason behind IT project failure, and that requirements related IT project failure broadly fall into two distinct areas:

- Requirements' specifications are not accurately mapped to the business issue that needs to be addressed;
- The software design lacks precise requirements specifications.

2.2.3. Adoption Challenges

Structured IT service management (ITSM) frameworks, such as the IT Infrastructure Library - ITIL® (Axelos, 2019), and project management frameworks, such as PRINCE2® (Axelos, 2017) and PMBOK® (Project Management Institute, 2017) have been introducing numerous decision-making roles and gates in IT organisations; and have as such allowed more delays in the product development lifecycle. In addition, accountability in structured approaches supports increased culpability in process ownership, which although leads to accountability reduces flexibility, since all changes require the approval of multiple stakeholders. This approach leads to accumulating delays and thus unwanted lead time which the customer experiences in terms of poor response times it takes to either fulfil requests, resolve incidents, or deploy changes. Furthermore, structured approaches to change, release, and deployment management of new products and services within the IT industry, has led to the innate proclivity to be blameful within post implementation reviews, or within post-project delivery lessons-learned meetings.

Agile, lean and DevOps principles and practices aim to identify value and non-value adding activities within ITSM processes. Specifically, the identification regards the end-to-end ownership of associated roles, processes, and technology (Willis 2010) to the software product development lifecycle (Bass et al., 2015; Dyck et al., 2015; Kersten, 2018). IT organisations willing to adopt agility discover that the more structure they introduce in their defined processes that leads to restricted agility (Horlach et al., 2017). This can be interpreted as a sign for a necessary extension, or shift, from structured service management practices towards agility and leanness. The transition, i.e. from a framework or process-led organisational environment to the adoption of groups of best practices, entails a significant shift in individual and organisational mindset. There needs to be a clear organisation-specific roadmap on the types of practices and principles that need to be adopted, including i) team structures that need to be applied, and ii) leadership styles that can help guide others towards agility and leanness adoption.

2.3. Agile Product Development

2.3.1. Agile Software Development

During the 1990s, individuals with a desire to think and act outside the structured approaches imposed in project and product management began forming the agile community. Agile is a term used to describe approaches to software development emphasising incremental delivery, team collaboration, continual planning, and continual learning. The term "Agile" was coined in 2001 in the Agile Manifesto (Beck et al., 2001). The manifesto set out to establish principles to improve the existing software development approaches. Agility aimed at solving a lot of the issues that were created in information intensive organisations by structured approaches. Essentially, the Agile Manifesto declares four value statements representing agile practices as follows:

- **Individuals and interactions over processes and tools –** valuing people more highly than processes or tools is easy to understand because the people respond to business needs and drive the development process. If the process or tools drive development, then the team is less responsive to change, and less likely to meet customer needs. Communication is an example of the difference between valuing individuals versus process. In the case of individuals, communication is fluid and happens when a need arises. In the case of process, communication is scheduled and requires specific content.
- **Working software over comprehensive documentation** – in waterfall environments, software developers spend a considerable time-consuming period creating detailed documentation such as technical specifications. That was before developers commenced work on coding. And while documentation isn't a bad thing, there comes a point when you should focus on providing your customers with working software. Working software over comprehensive documentation – historically, in waterfall-based environments, extensive amount of time was spent on product documentation prior to commencing work on coding. Some examples include, technical specifications, technical requirements, linkage to project charter's business case; including interface design documents, test plans, documentation plans, and required approvals. The list was extensive and was a cause for the long delays in development. Agile does not eliminate documentation, but streamlines it in a form that gives the developer what is needed to do the work without getting bogged down in minutiae. Agile documents requirements as user stories, which are sufficient for a software developer to begin the task of building a new function. The Agile Manifesto values documentation, but it values working software more.
- **Customer collaboration over contract negotiation** - customer collaboration over contract negotiation – the accurate alignment between what a contract stated, what the product does, and what

the customer expects is essential. With development models such as Waterfall, customers negotiate the requirements for the product, often in great detail, prior to any work starting. This means that the customer is involved in the process of development before the software development team begins coding, and after it has been delivered, but not during the process. The Agile Manifesto focuses on continuous development coupled with a feedback loop with customers. This makes it far easier for development to meet the needs of the customer. Agile methods can include the customer at intervals for periodic demos, but a project could just as easily have an end-user as a daily part of the team and attending all meetings, ensuring the product meets the business needs of the customer.

- **Responding to change over following a plan** - responding to change over following a plan - software development teams should have the ability to reach consensus and decisions to change direction whenever they need to, with a flexible roadmap that reflects that decision-making process.

The Twelve Principles are the guiding principles for the methodologies of the Agile Movement and they are prescriptive enough to indicate that decentralized decision making from high authority to agile teams, customer-centricity, human-to-human collaborative interactions, minimum viable products made up of smaller batch sized work items is crucial to the success of the product development lifecycle . They describe a culture in which change is welcome, and the customer is the focus of the work. They also demonstrate the movement's intent as described by Alistair Cockburn, one of the signatories to the Agile Manifesto, which is to bring development into alignment with business needs. The twelve principles of agile development include (Beck et al., 2001):

- **Customer satisfaction through early and continuous software delivery** – Customers are happier when they receive working software at regular intervals, rather than waiting extended periods of time between releases.
- **Accommodate changing requirements throughout the development process** – The ability to avoid delays when a requirement or feature request changes.
- **Frequent delivery of working software** – Scrum accommodates this principle since the team operates in software sprints or iterations that ensure regular delivery of working software.
- **Collaboration between the business stakeholders and developers throughout the project** – Better decisions are made when the business and technical team are aligned.
- **Support, trust, and motivate the people involved** – Motivated teams are more likely to deliver their best work than unhappy teams.
- **Enable face-to-face interactions** – Communication is more successful when development teams are co-located.
- **Working software is the primary measure of progress** – Delivering functional software to the customer is the ultimate factor that measures progress.

- **Agile processes to support a consistent development pace** – Teams establish a repeatable and maintainable speed at which they can deliver working software, and they repeat it with each release.
- **Attention to technical detail and design enhances agility** – The right skills and good design ensures the team can maintain the pace, constantly improve the product, and sustain change.
- **Simplicity** – Develop just enough to get the job done for right now.
- **Self-organising teams encourage great architectures, requirements, and designs** – Skilled and motivated team members who have decision-making power, take ownership, communicate regularly with other team members, and share ideas that deliver quality products.
- **Regular reflections on how to become more effective** – Self-improvement, process improvement, advancing skills, and techniques help team members work more efficiently.

The intention of Agile is to align development with business needs, and the success of Agile is apparent. Agile projects are customer focused and encourage customer guidance and participation. As a result, Agile has grown to be an overarching view of software development throughout the software industry and an industry all by itself.

Agile Software Development (ASD) emerged in 2001 as an evolutionary practice to existing structured approaches. The new practice advocated for iterative short-cycled development increments and continuous integration as opposed to structured engineering stage-gate models (Poppendieck and Poppendieck, 2003; Rodriguez et al. 2019). Organisations that are in the business of developing and producing services, or products to their customers, have been adopting traditional software development lifecycle (SDLC) practices specific to their organisational culture. In doing so, the shortcomings of these structured approaches have exposed slow reactive responsiveness to a constant shift of demand to which organisations are subject.

2.3.2. A Retrospective View of Agile

In 1986, Hirotaka Takeuchi and Ikujiro Nonaka published “The New Production Development Game”; and coined the term ‘Scrum’ as part of agile product development (Takeuchi et al., 1986). From a series of interviews with organisation members, Takeuchi and Nonaka (1986) learned that leading companies indicate six characteristics when managing their product development process, i.e. built-in instability, self-organising project teams, overlapping development phases, multi-learning, subtle control and organisational transfer of learning. The term “Scrum” is short for “scrummage” and originates from the sport of rugby where “scrum” is a method of restarting play in rugby that involves players packing closely together with their heads down and attempting to gain possession of the ball. Moreover, Scrum is “a

framework within which people can address complex adaptive problems, while productively and creatively delivering products of the highest possible value." (Sutherland and Schwaber, 2017). Scrum is commonly used as an agile product development approach in software-intensive organisations. In the mid 90's, Agile Software Development (ASD) techniques emerged under the influence of lean manufacturing practices of the Japanese car manufacturing industry. ASD has been characterised as the collection of a set of agile practices, agile methods, project size, project performance and project constraints (Joseph and Santana, 2016). In this context, Scrum a lean product development approach (Carvalho, 2011) gained popularity among the software development community. Moreover, Scrum was applicable to other industries such as aviation, retail, transport, etc. Scrum. developed by Jeff Sutherland, Ken Schwaber, and Mike Beedle, is based on the six characteristics (Schwaber, 1995) of results flexibility where late changes to requirements can be negotiated and accepted, deadline flexibility to allow time adjustments for attained milestones, small-sized teams typically of 5 ± 2 individuals, frequent peer reviews of produced work by team members, cooperation by focusing on improved communication skills and shift of mindset and lastly, object orientation.

Scrum does not require or provide any specific technique for the development phase. Scrum only establishes a group of rules and management practices that should be adopted to shift from waterfall software project management to agile software development. The Scrum management practices are product backlog, daily scrum, sprint, sprint planning meeting, sprint backlog and sprint review meeting. Figure 2-2 depicts the various Scrum-based processes, including the aforementioned management practices.

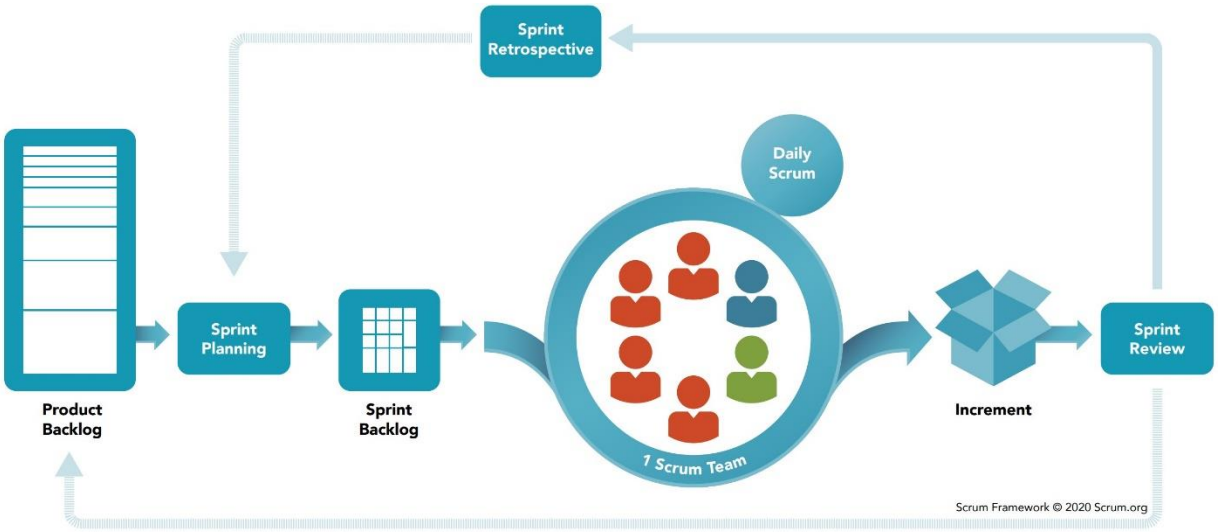


Figure 2-2 Scrum methodology (Scrum.org, 2022).

Recent developments in ASD have led to rapid and continuous software engineering practices, which refer to the organisation developing its own capability to address development, release, and continuous learning cycles from concurrently applied software development cycles (Rodriguez et al., 2019).

It is essential to understand what Agile is not. Microsoft's DevOps Resource Center (Techtarget, 2018) defines that:

- Agile practices and method do not aim for cowboy coding. Agile should not be confused with a discovery process of resolving issues based on what the software development team faces at any given moment in time during a sprint. That's not the type of autonomy a team should enjoy. While Agile values autonomy for individuals and teams; Agile emphasises aligned autonomy, ensuring the delivery of increased value through increased autonomy. Agile also requires both a Definition of Done (DoD) and value delivered to customers in every sprint. DoD resembles the pre-agreed software development team-wide criteria, i.e. to develop a common understanding of when work is accomplished and can be delivered as an outcome of a sprint.
- Agile practices should not be adopted without rigor and planning. On the contrary, Agile methodologies and practices typically emphasize discipline in planning. The key is continual planning throughout the project, not just planning up front. Continual planning ensures the team can learn from the work they are executing, thus maximising planning ROI (Return On Investment).
- Agile is not an excuse for the lack of a roadmap. This point has been detrimental to the agile community overall. Organisations and teams following an Agile approach absolutely know where they are heading and the results they want to achieve. Recognising change as a part of the process (an agile approach) is different from pivoting in a new direction every week, sprint, or month.
- Agile is not development without specifications. It's necessary in any project to keep the software development team aligned concerning "why" and "how" work will be accomplished and delivered. An agile approach to specifications, includes, ensuring those specifications are "right-sized" and reflect appropriately how the team will sequence and deliver work.

2.3.3. Adoption Benefits

There are several benefits associated with ASD approaches. The five most popular regard increase in customer satisfaction coupled by decrease in number of complaints, improvement in communication and increase in cooperation among team members, improvement in product quality, increase in team productivity and increase in project return on investment (Poppendieck, 2003; Carvalho and Mello, 2011). Moreover, in terms of agile practices, methods, and principles adoption, there is a set of Test-Driven Development (TDD) benefits; that complement the aforementioned ASD approaches result in decreased

code cyclomatic complexity, improved quality (external[1] and internal) of the same products and code refactoring, which also leads to higher quality outcomes. Moreover, there is evidence that code developed by Test-last programmers, when compared to test-first programmers, has more defects, and that test-first programmers are more likely to write software in more and smaller units that are less complex and more 'testable'. TDD also contributes towards decreased defect density (Sfetsos and Stamelos, 2010), i.e. a lower bug cap per product release or ideally close-to-zero-bug. Additionally, the bug cap ensures that technical debt caused by software release defects is rarely carried forward to future sprints, and the team can continuously learn and improve from diagnosed mistakes. Bug cap represents bugs in the bug backlog. It does not include bugs found and fixed within the sprint, in which a feature is developed (Microsoft Docs, 2022). These benefits can lead to a reduction in the overall development effort, improved developers' productivity, and decreased avoidable fault cost (Sfetsos and Stamelos, 2010).

On the other hand, when it comes to ASD adoption, industry professionals indicate that improving communication through agile events (formerly known as ceremonies circa, 2017) - like daily stand-ups, sprint planning meeting, and sprint retrospectives - are preferred to technical elements such as pair-programming and TDD (Julian et al., 2019). The development of the body of knowledge of ASD on its specific practices, principles and methods indicated that TDD was one of the most reviewed areas (Rodriguez et al., 2019).

2.3.4. Adoption Challenges

The key challenges that agile project management and product development research has considered is revolved mainly around the methods of migration from Waterfall to Agile environments (Spundak, 2014; Almeida, 2017). The dilemma, which agile practice adoption carries, is whether a big-bang or gradual adoption approach should be selected (DevOps Podcast, 2015). In terms of the gradual adoption, introducing agile practices into existing "traditional" organisations can be hard. Agile practice encourages agility at the agile rituals level e.g. sprints, stand-ups, and retrospectives - accompanied by task tracking and visualisation i.e. at an operational level (Julian et al., 2019). This can lead to agile practice adoptions that do not enjoy the entire set of benefits that agility can bring to organisational capabilities. For instance, references to "Water-scrum-fall" or "Scrumbut"; a term that denotes the application of selective Scrum roles, events, artefacts, and rules (Julian et al., 2019). In fact, the risk in the case of tailoring adoption approaches can lead to practice adoption failure.

[1] External Quality in a control setting was measured by the number of passed acceptance tests or the total number of defects or number of defects/KLOC (defect density). Internal quality was measured by different code metrics such as code size, cyclomatic complexity, coupling and cohesion.

The agile practice adoption journey eventually led to teams embracing Scrum or Kanban. This journey remains part of a continuous improvement cycle providing feedback to teams assessing how they work with newly adopted agile practices and how they can continuously improve (Fitzgerald et al., 2014). In fact, receptiveness to change increases as teams become more experienced with their practices (Julian et al., 2019). In contrast, the big-bang approach, which practically means the adoption of agile practices from Day 1, could work better with experienced teams, but can result to an initial productivity drop (Julian et al., 2019). Notably, the main challenges to the adoption of agile practices are focused on three areas namely; resistance, outside forces and regulatory compliance and requirements associated to the type of work (Julian et al., 2019).

Moreover, the adoption achievement process requires to take into account several variables, such as the goal of the team, the situation, project scope, project budget, etc., which can constrain the selection of a practice to be adopted (Kiv et al., 2019). More importantly, the changes that need to take place can occur in five dimensions divided into technology-oriented and team-oriented practices. On the one hand, software development and engineering practices are aimed to improving results and outcomes obtained from time and effort committed to accomplished work with technology tool usage. On the other hand, team, management approach, reflective and cultural practices aim to improve team performance, shift attitude and habits and align committed time and effort towards specific goals.

2.4. The Lean Movement

2.4.1. The Lean Mindset

The term “Lean” was coined in 1988 by John Krafcik (Krafcik, 1988) and popularised in 1990 by James P. Womack (Womack and Jones, 1990), with the aim to remove the following ‘waste’ areas (Poppendieck and Poppendieck, 2003) in knowledge-intensive and software-intensive organisations: partially completed work; unnecessary product features; relearning/reskilling the workforce; poor knowledge handoff; task switching; delays; product defect; and (a later addendum) underutilised talent of the workforce.

Lean IT aims to transpose the same approaches to waste to software development, i.e., to eliminate or reduce their impact on product development ‘lead times’ to market delivery. In comparison to ASD, it is notable that Lean Software Development (LSD) was an incremental improvement on top of ASD (Rodríguez et al., 2014). LSD relevant principles are denoted in Table 2-2.

Table 2-2 Lean principles relevant to software development.

Lean Software Development Principles (Poppendieck and Poppendieck, 2003)[b]	The Principles of Product Development Flow (Reinertsen, 2009)	The Kanban Principles (Anderson, 2010)
Eliminate waste Build quality in Create knowledge Defer commitment Deliver fast Respect people Optimise the whole	Use an economic view Manage queues Exploit variability Reduce batch size Apply WIP (work in progress) constraints Control flow under certainty Use fast feedback Decentralise control	Visualise the workflow Limit WIP Manage flow Make process policies explicit Improve collaboratively (using models and the scientific method)

2.4.2. History of Lean Management

The roots of Lean Enterprise stretch as far back as 1908, i.e. to a time when Henry Ford's Ford Motor Company was designing and producing Ford Model T automotive cars. The grandiose Model T mass production plan was successful because it provided inexpensive transportation, which symbolised both innovation and modernisation for the rising middle classes in the US. The set of practices and principles employed by Henry Ford's automotive production factories developed to what is known Ford Production System (FPS). Moreover, FPS became the baseline synthesis of lean manufacturing (Levinson, 2002). Henry Ford extended organisational considerations to human psychology, and aimed at the development of the inclusive work environment, i.e. where each and every factory employee partnered with the organisation to achieve its goals.

Lean Manufacturing in the 1920s was very deeply represented by Ford Production System (FPS). During World War II, while America was being drawn into the war, the Roosevelt administration asked Ford Motor Company to manufacture components for the B-24 Liberator bomber. Henry Ford contributed significantly in the architectural and development plans for the construction of an assembly and manufacturing plant in Willow Run. where one B-24 bomber was being produced, at its production peak, every hour.

Following World War II, FPS was transformed by Toyota into two pillars known as i) Just In Time (JIT) and ii) Jidoka aka autonomation - making kanban boards, kaizen (continuous improvement), and poka-yoke

(error-proofing) a key part of the Toyota Production System (TPS) (Shingo and Dillon, 1988; Ohno, 1988). The exploration and understanding of FPS by the Japanese after WWII was over, was succeeded in the 1960s by Americans like Edward Deming (Liker, 2004), visiting Toyota manufacturing sites to better comprehend TPS or Toyota Business Practices (TBP) - the collection of TPS practices as they became to be called.

Instead of the Ford push system, Toyota created a unique pull system, which then became the backbone of Toyota Production System (TPS) and lean manufacturing, to avoid overproduction and meet the diversified customer demands. TPS targeted at removing any kind of waste and inconsistency in the production system. TPS is not only elimination of non-value adding activities from the process but also improving the quality of the product with help of Jidoka (Jasti and Kodali, 2016).

Lean manufacturing was based on Lean Production (LP) (Krafcik, 1988). LP addresses elimination of waste and makes the process more streamlined and flow more efficiently (Liker, 1996). Lean has expanded ever since vertically - into accounting, marketing, HR, IT, design and R&D, and logistics - and horizontally - into domains including service, health, government, and banking. Lean manufacturing aimed at bringing together people, material, and mechanical resources at the right time to accomplish the job, while reducing cost and bring continuous improvement. It strived to eliminate seven kinds of waste; i.e. Over production; Waiting time; Transportation; Processing; Inventory; Motion; and Product defects.

Lean Manufacturing types of waste can be transposed to Lean IT types of waste as shown in Table 2-3 (Poppendieck and Poppendieck, 2003). Moreover, the "waste element' described in Table 2-3 is highlighting and categorising the non-value adding tasks to which an organisation is committing time and effort through its workforce. Notably, lean management is a consistent and constant attempt at identifying improvement areas and including them in the next improvement action cycle, known as 'kaizen bursts'. The small step improvements lead to significant gains in productivity when their summation over a period of time is holistically calculated. For instance, in managing change requests, there can be a centralised function of approvers who qualify a change based on its type and decide on a sequence of actions to approve a change execution. Identifying delays in such a process is considered a commonality among industry practitioners. Therefore, attempts to gradually provide the right autonomy level to the change approving function to disassociate human effort from 'work' that produces delays is considered an orthological step towards 'waste elimination' in lean management.

Table 2-3 Types of waste in Lean IT.

Waste element	Examples	Business outcome
Defects	Unauthorised system and application changes.	Poor customer service, increased costs.
	Substandard project execution.	
Overproduction (overprovisioning)	Unnecessary delivery of low-value applications and services.	Business and IT misalignment, Increased costs, and overheads: energy, data centre space, maintenance.
Waiting	Slow application response times.	Lost revenue, poor customer service, reduced productivity.
	Manual service escalation procedures.	
Non-Value Added Processing	Reporting technology metrics to business managers.	Miscommunication.
Transportation	On-site visits to resolve hardware and software issues.	Higher capital and operational expenses.
	Physical software, security, and compliance audits.	
Inventory (excess)	Server sprawl, underutilised hardware.	Increased costs: data centre, energy; lost productivity.
	Multiple repositories to handle risks and control.	
	Benched application development teams.	
Motion (excess)	Fire-fighting of repeat problems within the IT infrastructure and applications.	Lost productivity.
Employee knowledge (unused)	Failing to capture ideas / innovation.	Talent leakage, low job satisfaction, increased support, and maintenance costs.
	Knowledge and experience retention issues.	
	Employees spend time on repetitive or mundane tasks.	

The adoption of LP in an end-to-end production process, i.e. from the suppliers to the delivery point to the customer, has given rise to the concept of the Lean Enterprise (LE) (Womack and Jones, 1994). LE does not restrict to organisational departments, but instead extends in an enterprise-wide fashion. There is an

inherently dynamic nature to leanness and its most current form is expressed in LE (Papadopoulou and Ozbayrak, 2005). Evidently, what 'lean' means in all its forms and especially LE, carries a certain degree of ambiguity with an antiquated vision of leanness (Papadopoulou and Özbayrak, 2005). Developing a Lean Enterprise is all about eliminating friction, and waste, in the value stream (Martin and Osterling, 2014), and reducing the time taken to deliver a product or service to market consumers (Mutjaba et al., 2010).

2.4.3. Adoption Benefits

Benefits of lean management have been transferred to non-manufacturing industries as well such as healthcare (Zarbo and D'Angelo, 2007). Moreover, lean practices and principles can be applied to any type of industry on a global level (Womack et al., 1990; 1996). Apart from the countless manufacturing companies, lean process management has been adopted in many different industries, as diverse as insurance, IT, healthcare, etc. However, non-manufacturing industries have not incorporated lean process as extensively as in manufacturing organisations (Puvanasvaran, 2010). Additionally, adoption of lean practices in the manufacturing industry can lead to higher level of cognitive readiness for change, resourcefulness and confidence, positive affect at work and overall job satisfaction including higher satisfaction with organisation and management, opportunities for development, job content, working facilities, supervisor, and income (Lipińska-Grobelny and Papieska, 2012).

Evidently, the manufacturing, automobiles and auto components industries have been experiencing the largest adoption rate of such practices and therefore research coverage is widespread compared to other non-manufacturing industries. However, there seems to be some potential in non-manufacturing industries such as services and information communication technology, electronics, and software development (Jasti and Kodali, 2016).

2.4.4. Adoption Challenges

There have been lean practices which have been adopted mainly by large-scale organisations compared to small-medium sized organisations (SMEs) based on the organisation size definition of the European Commission (EC, 2003). Frameworks and models such as that of Åhlström (1998), Womack and Jones (2003), and Karim and Arif-Uz-Zaman (2013) but most of them are derived from the experiences of large corporations. Yusof and Aspinwall (2000) approved that smaller companies cannot just imitate, in total, the approaches adopted by their large counterparts. There is therefore a real need to develop a new framework for Small-medium enterprises (SME) (Belhadi et al., 2016).

2.5. DevOps and its Adoption

Applications built over the years carry historical design assumptions, such as that a few hours of downtime for maintenance upgrades every month is acceptable. DevOps offers an unprecedented opportunity for organisations to transform their Software Development lifecycle to increase efficiency and meet end-users' changing expectations. DevOps attempts to redefine the foundations of software development and management recasting the approach concerning development of every element (Ravichandran et al., 2017); even in cloud services provisioning (Rajkumar et al., 2016). The reformation that DevOps brings, with its set of developed practices and principles, also extends to the customer experience.

2.5.1. Defining DevOps

The term 'DevOps' was first coined in 2009 and there are , different terms and theoretical accounts that have been used to study the DevOps definition and its constituents (Bass et al., 2015; Smeds et al., 2015; Dyck et al., 2015; de França et al., 2016; Lwakatare[a] et al., 2016; Jabbari et al., 2016), see Table 2-4. Although, the definition of DevOps is unclear it is also continuously evolving (Alahyari et al., 2019). Literature highlights the definition diversity that exists and is strongly associated to DevOps with acronyms such as 'DevSecOps' (Myrbakken and Colomo-Palacios, 2017), 'DevSecOps' (Development-Security-Operations), 'BizDevOps' (Business-Development-Operations) (Drews et al., 2017) and 'DevNetOps' (Dyck et al., 2015).

There is, however, a research and industrial need to develop a better understanding of what is included in the scope of DevOps (Dingsøyr and Lassenius, 2016). DevOps has been described as being: a new role within a software organisation (Kerzazi and Adamns, 2016); a movement (Lwakatare et al., 2016) for change in software industry (De França et al., 2016); a set of software development practices (Bass et al., 2015); a le-agile approach (Xiaofeng et al., 2012) – i.e., the combination of the lean and agile paradigms; and High Velocity IT, which ITIL4® defines as involving techniques for valuable investments, fast development, resilient operations, co-created value and assured conformance (AXELOS 2019). Several definitions of DevOps have been published, see Table 2-4. The majority of the descriptions specify DevOps as a term that is used to emphasise the collaboration between software development and operations.

Table 2-4 Diversity of DevOps definitions.

Source	Definition of DevOps	Focus of Definition
Bass et al., 2015	DevOps is a set of practices intended to reduce the time between committing a change to a system and the change being placed into normal production while ensuring high quality	Goal-oriented (fast delivery of quality software)
Dyck et al., 2015	DevOps is 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 the delivery of changes	Means-oriented (empathy, cross-functional collaboration); and goal-oriented (operate resilient systems, accelerate change delivery)
Smeds et al., 2015	A set of engineering process capabilities supported by certain cultural and technological enablers	Means-oriented (engineering capabilities)
De Franca et al., 2016	DevOps is a neologism, representing a movement of ICT professionals addressing a different attitude regarding software delivery through the collaboration between software systems development and operations functions, based on a set of principles and practices, such as culture, automation, measurement and sharing'	Means-oriented (attitude, cross-functional collaboration)
Jabbari et al., 2016	DevOps is a development methodology aimed at bridging the gap between Development and Operations, emphasising communication and collaboration, continuous integration, quality assurance and delivery with automated deployment, utilising a set of development practices	Means-oriented (cross-functional collaboration, automated deployment)
Microsoft, 2022	DevOps is the union of people, process, and technology to continually provide value to customers.	Means oriented (continuous value delivery)

However, there is published research that downplays the fact of not having consensus over a DevOps definition. and that there is no need for "extending" acronyms like DevSecOps, SecDevOps, or DevNetOps (Dyck et al., 2015) (Microsoft, 2021). There are many similar cases where some people thought a definition is not needed, e.g. the agile movement. Cultural enablers, used to promote the adoption of DevOps practices, are required, such as focus on decision making, customer focus, engineering practices, learning and development, leadership, team recognition, innovation, guilds, and performance feedback (Jones et al., 2016; Kamuto and Langerman, 2017; Bezemer et al., 2019). Moreover, to achieve performance gains, while adopting DevOps, the following are shown to be essential (Ravichandran et al., 2017):

- **Tightened feedback loops between Development and Operation teams -** The traditional waterfall approach focused on one-way communication from software and product development teams to quality assurance (testing) and operations teams with little feedback on how these teams and end-users perceive the experiences delivered. DevOps encourages a bi-directional collaboration where continuous feedback mechanisms allow developers to promptly identify, analyse, and act on outcomes following their code-deployment delivery;
- **Established practices of automated performance monitoring** – The protruding DevOps practices in automation do not only regard automated standardised test cases which measure performance in live production environments but also include continuous monitoring and self-healing resilience capabilities, should an incident become the cause of service outage to the customer and its community of end-user consuming that service;
- **Measurement of key performance metrics in Continuous Integration, Test and Ops teams** – One of the key facets in DevOps is the type of metrics that teams adopt to achieve their product's deployable increments, its testing, and monitoring;
- **Shared tools and performance metrics across teams -** In DevOps teams task-switching, interactions with multiple tools and understanding cross-functional metrics is considered unnecessary non-value adding activities from the customer's perspective. Therefore, every attempt and effort to minimise or even eliminate as much of that types of unwanted waste is considered beneficial to the behaviours that teams should be using, including shared or unified tool usage (as reskilling on multiple tools is considered unwanted waste) and enterprise-wide metrics (i.e. performing against a common purpose).

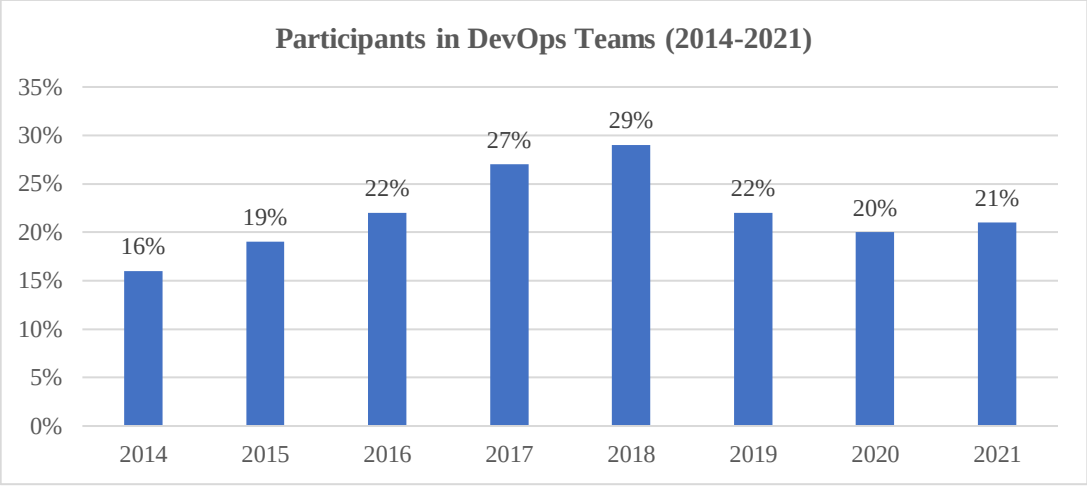


Figure 2-3 State of DevOps Report Participants in DevOps Teams.

According to the State of DevOps Report (Puppet, 2021), up until 2018 there was an increasing inclusion of respondents identifying themselves as working in DevOps teams, which deflated slightly in 2019 and for the past three years it has stabilised at 2016 levels, see Figure 2-3. There is also evidence in the same report to suggest that the drop in individuals identifying themselves as not part of a DevOps team is that there is a growing feeling that these individuals considered themselves part of an emerging theme of teams - stream-aligned or engineering team. It is worth noting that the State of DevOps Report poses evidence that industry practitioners have experienced rise in salaries when their job title includes "DevOps". On the contrary, job titles such as "System Administrator" have declined 38% since 2014 when compared to 2021. Similar trends exist for job titles including "Operations" and "Infrastructure Engineer" terms, see Figure 2-3 (Puppet, 2021).

DevOps Practices, Principles and Maturity Frameworks

The adoption of DevOps practices and principles requires several factors to be taken into account since DevOps is more than just a mindset but rather patterns of DevOps practices (Lwakatare et al., 2016). A recent systematic literature review (Sánchez-Gordón and Colomo-Palacios, 2018) found that DevOps Cultures all possess seven popular characteristics, i.e. 1) Communication, 2) Collaboration, 3) Feedback (Continuous and immediate), 4) Responsibility (personal/mutual), 5) Improvement cycle, 6) Sharing Knowledge, and 7) Transparency. Interestingly, they all evolve around the notion of 'Empathy', and the dissolvement of boundaries between self and other.

In agile software development there is a distinction between practices and influence (Kropp et al., 2018), which can be extended by a lean principles background that form a prerequisite for successful DevOps adoption (Lwakatare et al., 2016). Furthermore, research shows that advisory skills, testing skills, analysis skills, functional skills, social skills, decision making skills, and full stack development skills, all tend to result in successful DevOps cross-functional teams (Wiedemann and Wiesche, 2018). This can be further complemented by a set of practices - common amongst development and operations teams - and a set of principles (e.g. social aspects, automation, quality assurance, leanness, sharing measurement) (De França et al., 2016). This extension from agile and lean to DevOps, is closely linked to the CAMS (Culture-Automation-Measurement-Sharing) model originally coined by John Willis and Damon Edwards (Willis, 2010) and later refined by Jez Humble within CALMS (Culture-Automation-Lean-Measurement-Sharing).

CALMS refers to the shift of mindset, skillset, and ultimately toolset, which is needed to be acquired in order to effectively adopt a new set of practices and guiding principles. CALMS has become the most popular model among DevOps adoption (Humble et al., 2011). This orientation requires gradual and minor changes in an organisation's daily operations. For companies to move from structured to agile structures in

software development, there needs to be first an adoption stage of agile practices and a shift to smaller cross-functional teams, and later, when a certain level of maturity is attained, DevOps practices can be adopted, such as automated system integration and continuous integration (Rodríguez et al., 2017). When continuous integration is in place, customers express an interest in receiving enhancements and bug fixes more frequently. Therefore, adoption of continuous delivery practices is required. The final step occurs when the organisation not only releases software continuously, but also develops mechanisms to conduct rapid experimentation to drive innovation. There is a research-based agile, lean IT and DevOps roadmap that has been produced in the context of continuous software engineering which identifies 1) Business strategy and planning (which includes continuous planning – continuous budgeting), 2) development (which includes continuous integration – continuous delivery – continuous deployment – continuous verification – continuous testing – continuous compliance – continuous security – continuous evolution), 3) operations (which includes continuous use – continuous trust – continuous monitoring), and 4) improvement and innovation (which includes continuous improvement – continuous innovation – continuous experimentation) (Fitzgerald et al., 2017).

Adidas, for example, has developed its own DevOps Maturity Framework (Adidas, 2020), which extends the CALMS model as follows:

- **Development**
  - Crawl – there is no version control of source code artifacts.
  - Walk – source code or other assets under version control.
  - Run – source code and all product artifacts are versioned and stored in artifact repository.
- **Product and Processes**
  - Crawl - there is no customer feedback gathered.
  - Walk - customer gathered on an ad-hoc basis.
  - Run - customer feedback systematically gathered after all releases.
- **Management and Monitoring**
  - Crawl - no monitoring in place.
  - Walk - application or infrastructure performance monitored without alerting mechanism setup.
  - Run - application and infrastructure performance monitored with enabled alerting mechanism.
- **Culture**
  - Crawl - time, effort and resources are not committed to experimentation.
  - Walk - irregular scheduled events are blocked for team experimentations such as hackathons.
  - Run - regular scheduled events blocked for team experimentation, such as monthly hackathons.
- **Architecture**

- Crawl - Monolithic application with a high level of interdependencies.
- Walk - Rearchitect from a monolithic solution to a microservice-based architecture.
- Run - System is independent of direct dependencies. Any existing dependencies are tied to open standards and not tied to technologies and frameworks (e.g. Java RPC).

CALMS shares similarities with another model that involves a specific set of categories namely: agility, automation, collaborative culture; also called DevOps Culture (Sánchez-Gordón and Colomo-Palacios, 2018), continuous measurement, quality assurance, resilience, and sharing and transparency (Luz et al. 2019). This can be further extended to include collaboration in terms of empathy (Lwakatare et al., 2016), respect, trust, responsibility and incentive alignment and open communication (Masombuka and Mnkandla, 2018). There are recurring studies to suggest that the lack of a 'collaborative culture' is detrimental to the success of DevOps teams and DevOps practice and principle adoption in an organisation (Rajkumar et al., 2016; Wiedemann and Wiesche, 2018; Luz et al., 2019).

The DevOps Institute's Collective Body of Knowledge (CBOK) focuses on three pillars, i.e. DevOps, Lean, and Leadership (DevOps Institute, 2022). In addition, successful adoption of DevOps requires agile software development (Lwakatare, 2016). For practitioners in the industry, there is a decline of interest in XP, and a steady increase in SCRUM over time. Between 2006 and 2015, there was an increase in interest concerning continuous integration, however, there was a sharp increase in DevOps adoption in the same decade (Dingsøyr and Lassenius, 2016) with a noticeable increase in academic publications on DevOps Culture (Sánchez-Gordón and Colomo-Palacios, 2018). This sharp increase has most likely been triggered by DevOps leaders who have acquired the transformational acumen required to contribute to the design, influence, and motivate cultural transformation, which is proven to be a critical success factor in DevOps adoption; making DevOps a multidisciplinary topic that requires application of a mix of skills, practices, and principles (Jones et al., 2016). Gaining an improved understanding of how the three disciplines of agile, lean and DevOps interact Table 2-5. outlines the mapping between the three communities.

Table 2-5 Agile, Lean and DevOps Mapping (adapted from Puppet, 2021).

Practice Category	**Practice**
Extreme Programming (XP)	Sustainable pace, Signup, iterations, velocity, frequent releases, user stories, continuous integration, collective ownership, simple design, refactoring, Test-Driven Development (TDD)
Teams	Project charter, sustainable pace, Scrum of Scrums, Niko-niko, pair programming, team room, heartbeat retrospective, facilitation, team
Lean	Lead time, Kanban board, Definition of Done (DoD)
Scrum	Iterative development, timebox, iterations, daily meeting, three questions, burndown chart, task board, Definition of Ready, point estimates, relative estimation, planning poker, backlog, backlog grooming
Product Management	Personas, story mapping, story splitting, user stories, 3 Cs, INVEST, Incremental development
DevOps	Continuous Deployment, Continuous Integration, Automated build, version control
Design	Ubiquitous language, simple design, refactoring, rules of simplicity, quick design session, CRC cards
Testing	Role-feature, Given-When-Then, Behaviour-Driven Development (BDD), Acceptance Test Driven Development (ATDD), Acceptance tests, Mock objects, TDD, Unit tests, Exploratory testing, usability testing
Fundamentals	Team, iterative development, incremental development, version control

2.5.2. Measuring DevOps

Metrics in traditional highly structured corporate environments produce development cycles that focus a lot on defect density of the software product. Defect density, however, is not the most effective way to measure quality in the context of software product development (Herring, 2018; Kersten, 2018). The effect that traditional approaches have had to software development is that ‘surrogation’ can lead to enterprise strategy being replaced with metrics (HBR 2019), with employees consciously aiming to contribute to local optima rather than global corporate optima to increase flow in the value stream (Goldratt, 1994).

When senior management first establish a key performance metric of ‘least defects in deployable code’ into a production environment, and announce the downsizing of the quality assurance team (Herring, 2018), software development teams commonly express significant differences in behavioural patterns of developers and testers. Software development should be attempting to get closer to the metrics most

frequently utilised to evaluate the speed with which releases can move to production environments before performance inefficiencies starts to appear (Herring, 2018). Additionally, software development pipeline health is essential to maintaining high quality software. Measurement approaches in DevOps teams include, but are not limited to, source code version control, optimum branching strategy, static analysis, >80% code coverage, vulnerability scan, open-source scan, artifact version control, auto provisioning, immutable servers, integration testing, performance testing, build deploy testing automated for every commit, automated rollback, automated change order, zero downtime release, feature toggle (Nygard et al., 2019).

In addition to the aforementioned points, there is increased research interest in understanding how DevOps teams measure cognitive load using relative domain complexity without measuring lines of code produced, number of modules, classes, or methods (Kersten, 2018). This can be further complemented by flow metrics i.e., flow distribution, flow velocity, flow time, flow load, flow efficiency (Gruver, 2016), which represent the proportion of each flow work item being active in each sprint. In particular, flow velocity measures features, defects, risks and technical debt in the product development flow whereas flow time resembles lead time and process time as defined in value stream maps (Martin and Osterling, 2014). Moreover, flow load represents active or waiting work in the value stream, and flow efficiency is the result of measuring flow load, i.e., duration of work inactivity in the value stream (Kersten, 2018).

Furthermore, flow can be considered for optimisation purposes at the requirements planning level, new and unique work, and any kind of repetitive routine type of work. Moreover, optimising it requires fast feedback and a focus on end-to-end cycle time for an all-round customer feedback. Workflow can be further categorised according to the Deployment Pipeline stages (Gruver, 2016) as: Requirement/Planning; Environments; Testing.

Requirement/Planning

Several organisations never setup a test environment for upcoming new product features, maintenance work, defect fixes, etc., let alone setting up a test environment which in many cases lacks consistency with production environments. The process of creating test environments is one of the main candidates for automation purposes (Gruver 2016). Questions that Gruver (2016) raised include:

- What percentage of the organisations capacity is spent on documenting requirements and planning?
- What is the amount of requirements inventory waiting for development, roughly, in terms of days of supply?
- What percentages of the requirements are reworked after originally defined?
- What percentages of the delivered features are being used by the customers and are achieving the expected business results?

Environments

An area where a lot of inefficiencies are usually observed is the quality assurance of a product. Questions that Gruver (2016) raised include:

- Time from environment request to delivery;
- How frequently the new environments are required;
- The percent of time environments need fixing before acceptance;
- The percent of defects associated with code vs environment vs deployment cs database cs other at each stage in the DP.

Testing

Questions that Gruver (2016) raised include:

- The time it takes to run the full set of testing;
- The repeatability of the testing (false failures);
- The percent of defects found with unit tests, automated system tests, and manual tests;
- The time it takes the release branch to meet production quality;
- Approval times;
- Batch sizes or release frequency at each stage.

Another dimension to DevOps can be Microsoft's perception on the triage of people, process and technology while providing a strong focus on the following five DevOps habits (Microsoft, 2021):

1. Customer obsession (practicing zero-distance by establishing continuous feedback channels by the end-user community towards the software product development team);
2. You build it, you love it (extreme ownership and accountability of code committed and deployed in production by developers) ;
3. Align outcomes, not outputs (the adoption of Objective Key Results instead of the traditional key performance indicators);
4. Get clean, stay clean (continuously reduce manual maintenance cost in terms of committed time and effort by the software product development teams);
5. Shift quality left and right (move from manual to automate software product testing).

In regard to the five habits, 1) flow of customer value entails automated testing, Continuous Integration (CI), Continuous Deployment (CD) and release engineering and management. 2) scaling that in terms of agile to self-managing teams and feature crews regards team autonomy and enterprise alignment. 3) within Microsoft feature crews, another habit is to refine and reprioritise backlog items through usage monitoring,

telemetry, Testing in Production (TIP) and stakeholder feedback. In fact, evidence collected from production environments include all aforementioned steps for backlog refinement plus the use of feature flags and continuous experimentation, regarded as one of DevOps key practices. 4), managing technical debt concerns peer code reviews, automated testing, continuous measurement, and agile documentation. Application performance management and Infrastructure-as-Code (IaC), coupled with configuration management and automated recovery, plays a big role in achieving a production first mindset. Finally, 5) IaC, automated scaling, sandboxing for development and test environments as well as the usage of microservices and containers make Infrastructure a flexible resource to work with while adopting DevOps practices and principles.

The aforementioned literature on DevOps metrics at the team structure-process-toolset level should be taken into account in a cross-functional manner and be communicated transparently to both leadership and engineering teams to establish progress and quality in a consistent format (Herring et al., 2015).

2.5.3. Adoption Benefits

DevOps adoption benefits include: More implemented features; Frequent releases; Powered by automated build, testing and deployment processes; Automation reduces required effort to setup releases; More daily commits of code; Improved quality assurance; Enhanced collaboration and communication; Improved visibility of implemented features to the customer; Testing with real customers; Enables continuous experimentation; Improved well-being of DevOps teams (Riungu-Kalliosaari et al., 2016).

One of the most important practices that automation offers in DevOps-practice-driven environments is continuous deployment (CD). Organisations seeking to adopt CD practices need to understand the underlying principles it brings to people's mindset, processes, and tooling approaches. Agile, lean, open source and internet speed development practices have resulted to creating enablers for continuous deployment such as parallel deployment, high-capability tools for deployment and testing automation and proactive customers and product managers (Rodriguez et al., 2019). Moreover, shorter-time-to-market, rapid feedback, customer satisfaction, increased efficiency, improves quality focus and more effective progress monitoring and quality can become forefront benefits for continuous deployment (Rodriguez et al., 2019). Additionally, cultural enablers that can promote adoption of DevOps practices are required such as a clear decision making and customer focus, appropriate use of engineering practices, learning and development, leadership, team recognition, innovation, guilds, and performance feedback (Kamuto and Langerman, 2017). Moreover, to achieve performance gains while adopting DevOps practices are (Gottesheim, 2015):

- Tighten the Feedback Loops between Dev and Ops;

- Establish a practice of automated performance monitoring;
- Measure key performance metrics in CI, Test and Ops;
- Share tools and performance metrics across teams.

Gottesheim (2015) discusses the need for a shared understanding to performance metrics across all teams, i.e. a blameless culture that reduces i) incident resolution times since it minimises the effects of blameful-culture, and ii) the monitoring of performance metrics, analysis and action; the next steps taken. Service Level Management (AXELOS, 2019) plays an important role to defining Service Level Objectives (SLO) and Service Level Agreements (SLA) between the service supplier (e.g. IT Organisation) and the customer organisations. Almost every low-level, ad-hoc monitor and every high-level executive KPI (Key Performance Indicator) can be articulated in terms of "service level" (Schlossnagle, 2017). Therefore, it seems that Service Level-driven metrics for DevOps give birth to the need of reviewing underpinning contracts (UC) of suppliers of the service supplier. For instance, in the old traditional approach a release schedule was based on a waterfall model, the shift of mindset that DevOps brings in a transformation towards agile teams, e.g. lean thinking and continuous release cycles, should be incorporated in the way this new structure is supported by third party organisations.

2.5.4. Adoption Challenges

In general, organisations and industry IT practitioners place DevOps in high regard, but DevOps practices and principles adoption is associated with challenges. These challenges can arise mainly from a combination of necessity in maintaining a legacy system, lack of senior management buy-in, managerial structure, and resistance (Jones et al., 2016). DevOps adoption challenges include, but are not limited to, the insufficient communication, deep-seated company culture, industry constraints and feasibility, heterogeneous environments (Poppendieck et al., 2003). Moreover, a Delphi study of forty-two (42) Norwegian experts indicated a comprehensive list of problems influencing poor cooperation between software development and operations (Iden et al., 2011), however, the most serious problems in poor software development – operations cooperation – included the following aspects:

- **Operations not being involved in the requirements specifications** – the early involvement of operations team members in technical requirements specifications firstly, requires the timely dissemination of requirements documentation to the operations team(s) from the product manager / owner, architect, developer, etc. The next step is to welcome any thoughts, ideas and opinion that could avert potential future operational 'technical debt' when the specific product is released and deployed in production environments delivering on the promise of new or improved customer experiences.

- **Poor communication and information flow** – mainly characterised by the absence or lack thereof cross-functional collaboration mechanisms within the corporate environment which leads teams to perform as 'silo' entities instead of collectively aiming to fulfil the enterprise-wide purpose that delivers on the business strategy of the organisation as a single whole entity. That is exactly how customers should, transparently, perceive interactions with those cross-functional teams.
- **Unsatisfactory test environment** – can denote the poor state or complete absence of User Acceptance Testing, staging or preproduction environments which do not meet realistic criteria to run functional (each function of the software application operates in conformance with the requirement specification) and non-functional test cases (e.g., for stress, load, performance testing). These test cases can be manual, semi-automated, or fully automated. In a DevOps-oriented team ideally, these are automatically provisioned through scripts that create test environments and following successful completion of testing period, decommission the test environment, similar to an on-demand testing approach.
- **Lack of knowledge transfer** – empowering and encouraging the design and development of a centralised knowledge repository (per product), with a network of knowledge article contributors per area or product expertise can eliminate a lot of the required manual knowledge transfer of 1:1 and 1:many sessions by shifting the knowledge acquisition experience to self-servicing and on-demand knowledge dissemination models.
- **Products being put into production before they are complete** – the release of new or improved products can be regarded as production-deployable artefacts that should be consumed by the end-user or customer community. The testability of products bearing new or improved functionality has to be performed at the preproduction stage with the involvement of actual and carefully selected end-users who champion the testing process of their own community's future experiences.
- **Operational routines not being established prior to deployment** – the automation of certain process steps that define how new changes will be applied in production environments following deployment can lead to standardisation of manual tasks and thus elimination of committed time and effort which in a setting of multiple deployments (per day) can lead to significant 'technical debt' spurring out of the development team(s), and leading to operations team(s) becoming a bottleneck of the deployment process.

The hierarchical approach of organisational structures that welcome static team structures can also become a bottleneck to information flow. A shift of mindset is required. For instance, to overcome DevOps adoption challenges, Nike.com designed a shift of mindset for its workforce. The Nike.com case study indicated that developers of software product development teams were supporting the DevOps adoption movement when suddenly they were faced with the fact of supporting incidents caused by their code on live servers

(Techtarget, 2018). In fact, that created a lot of tension and resistance to the DevOps practices adoption roadmap and it was eventually driven only through the right senior management support.

Moreover, obstacles to flow can also be characterised as anything that acts as an impediment to cognitive load of a DevOps team topology (Skelton and Pais, 2019). Cognitive load refers to the amount of working memory being used at any one moment within a team structure. Flow challenges can be due to disengaged teams, software too big for team structure, confusing organisational design options, team getting pulled into too many directions, painful reorganisation every few years, flow is blocked by certain factors and too many reactive-natured surprises for the team to handle (Skelton and Pais, 2019).

For modern software companies, speed facilitates fast and repeatable software development and delivery processes (Feitelson, 2013). Complexity of performance engineering approaches is a barrier for wide-spread adoption by practitioners. Accordingly, performance engineering approaches must be lightweight and must smoothly integrate with existing tools in the DevOps pipeline (Bezemer et al., 2019). This is evident by the emergence and the growing interest of a continuous deployment paradigm in the software industry. Continuous deployment entails the capability of an organisation to deliver new software features at multiple times and in the shortest time possible. DevOps is an approach that has been reported to enable the continuous deployment paradigm as it embodies a set of useful principles crucial to the development and deployment of software (Humble and Molesky, 2011). Practices that have posed as barriers to continuous deployment include time pressure, increased technical debt, customer unwillingness to update and conflicting goals between rapid released and achieving high reliability and test coverage. In addition, the adoption challenges that have also been identified in large scale organisations are cultural barriers, risk of disintermediation of roles, lack of DevOps education and awareness, resistance to change, silo mentality and lack of strategic direction from senior management (Kamuto and Langerman, 2017). Additionally, other points which pose as barriers are existence of blame-culture, communication difficulties, and delays in producing software releases (see Bass et al., 2015; Smeds et al., 2015; Dyck et al., 2015; de França et al., 2016; Lwakatare[a] et al., 2016; Jabbari et al., 2016).

The State of DevOps Report (Puppet, 2021), produces three segments of organisations in the DevOps adoption journey, low, middle and high performers. The challenges faced by low performing organisations regard organisational resistance to change, followed by legacy architecture, shortage of skills, limited or lack of automation, and unclear goals or objectives. Moreover, middle performers cite shortage of skills, legacy architecture, organisational resistance to change, and limited or lack of automation as the primary blockers to better DevOps practices. Lastly, high performers culture does not pose as a barrier any longer and therefore the two major blockers to DevOps adoption are legacy architecture and a shortage of skills.

2.6. Leadership Styles and Traits in Software-intensive Organisations

2.6.1. Definitions of Leadership

Leadership is mostly defined as the process of influencing a group toward the achievement of goals and directing the organisation to make it more cohesive and coherent (Bass, 1997). A leader carries out such a process by applying his/her leadership qualities, such as values, beliefs, character, knowledge, skills, ethics, experience, and culture. Leaders inspire people, move them to action, and change the world. Leadership is a social process that is highly complex.

Table 2-6 Attributed definitions of Leadership.

Researchers	Definition of Leadership
Blackmar (1911)	The centralisation of effort in one person.
Bernard (1927)	Focuses the attention of group members into the desired direction.
Copeland (1942)	The art of influencing.
Knickerbocker (1948)	Consists of a relationship between an individual and a group.
Stogdill (1950)	The process of influencing the activities of an organised group in its effort toward goal setting and goal achievement.
Bennis (1959)	Induces a subordinate to behave in a desired manner.
Bass (1961)	An individual's effort to change the behaviour of others.
Tannenbaum (1961)	Interpersonal influence toward the attainment of a specific goal or goals.
Katz and Kahn (1966)	An influential increment over and above compliance with the routine directives of the organisation.
Burns (1978)	Transforms followers, creates visions of the goals that may be attained and articulates for the followers, ways to attain those goals. Leadership individuals mobilise resources to arouse, engagement and satisfy the motives of followers.
Pondy (1989)	A form of social influence.
Schein (1992)	The ability to start evolutionary change processes that are more adaptive.
Bass (1994)	An interaction and leaders are agents of change whose acts affect other people more than people's acts affect them.
Drucker (1998)	Needs of a leader. The only definition of a leader is someone who has followers.
Vroom and Jago (2007)	Refers to a potential or capacity to influence others.
Jung (2013)	The alignment of subordinates' activities and their motivational activation for goal attainment.

Leadership includes influence processes involving determination of the group's objectives, motivating task behaviour in pursuit of these objectives, and influencing group maintenance and culture (Yukl, 1989). Burns (1978) defined leadership as "inducing followers to act for certain goals that represent the values and the motivations, the wants and needs, the aspirations and expectations of both leaders and followers." Table 2-6 summarises leadership definition provided by researchers, which cover more than one century of academic work on the subject.

There are various leadership styles that should be taken into account when considering DevOps – especially if a highly structured organisation is attempting to adopt agile, lean and DevOps practices and principles. A non-exhaustive list of those leadership styles is provided:

- Servant Leadership (Greenleaf, 2002) – a term first coined by Greenleaf in 1970 proposes that service to followers is the primary responsibility of leaders and the essence of ethical leadership on which servant leadership forms a strong basis. Servant leadership in corporate environments is about helping others to accomplish shared objectives by facilitating individual development, empowerment, and collective work that is consistent with the health and long-term welfare of followers (Yukl, 2020). Servant leadership was developed as a theory of ethical leadership which is comprised of values such as integrity, altruism, humility, empathy and healing, personal growth, fairness and justice, empowerment;
- Authentic Leadership (Avolio et al., 2005) – is another theory derived from ethical leadership and its most widely accepted definition is " a pattern of leader behaviour that draws upon and promotes both positive psychological capacities and a positive ethical climate, to foster greater self-awareness, an internalised moral perspective, balanced processing of information and relational transparency on the part of leaders working with followers, fostering positive self-development (Yukl, 2020);
- Transformational Leadership (Sahu et al., 2018) – first coined by James McGregor Burns in 1978, describes how effective leader inspire and transform followers by appealing to their ideas and emotions (Yukl, 2020).

2.6.2. Transactional Leadership

Transactional leadership generally uses organisational bureaucracy, policy, power, and authority to maintain control; this style of leadership is occasionally referred to as authoritative (Bennet, 2009). Transactional leaders emphasise work standards, assignments, and task-oriented goals. In addition, transactional leaders tend to focus on task completion and employee compliance, and these leaders rely quite heavily on organisational rewards and punishments to influence employee performance (Hinkin, 1998). They explain what is required of them and what compensation they will receive if they fulfil these

requirements (Bass, 1990). Transactional leadership focuses on ways to maintain the status quo and manage the day-to-day operations of a business. Transactional leadership seeks to guide and motivate their subordinates to achieve established goals by clarifying role and task requirements, and assisting employees to identify what must be done to achieve desired results (Robbins and Sanghi, 2005). Moreover, transactional leadership gives the opportunity to subordinates to fulfill their own self-interest, minimise workplace anxiety, and concentrate on clear organisational objectives such as increased quality, customer service, reduced costs, and increased production (Sadeghi and Pihie, 2012).

Transactional leadership does not focus on identifying the organisation's goals and how employees can work toward and increase their productivity in alignment with these goals, thus increasing organisational profitability (Avolio et al., 1991). Followers are motivated and corrected by the leaders' transactional actions. Egri and Herman (2000) defined the main concern of transactional leaders as being that of the accomplishment of the subordinates' task performance in terms of meeting organisational goals and objectives. Leaders gain the commitment of employees by giving them contingent rewards. Accordingly, Kuhnert and Lewis (1987) suggested that effective transactional leaders must regularly fulfil the expectations of their followers.

The basis of transactional leadership is a transaction or exchange process between leaders and followers. The transactional leader recognises followers' needs and desires and then clarifies how those needs and desires will be satisfied in exchange for meeting specified objectives or performing certain duties (Dinibutun, 2020). Thus, followers receive rewards for job performance, while leaders benefit from the completion of tasks (Vecchio, 2002). Because of these transactional relationships, some of the theories explained in the previous section can be considered as transactional theories, such as path-goal theory and initiating structure. Transactional leadership involves a commitment to 'follow the rules'; therefore, transactional leaders maintain stability within the organisation rather than promoting change (Vecchio, 2002).

There is a temporary process for transactional leadership. Once a transaction is complete, the relationship between the leader and subordinates can end or be redefined for the next transaction (Lussier Achua, 2001). Therefore, the nature of this kind of leadership style is based on a short-term relationship between leader and follower.

2.6.3. Transformational Leadership

Transformational leadership comprises four dimensions: idealised influence, inspirational motivation, intellectual stimulation, and individualised consideration (Bass, 1998). Such leaders promote and motivate

their followers by projecting and communicating attractive visions, common goals, and shared values (Bass and Riggio, 2005). Idealised influence is the leader's ability to build loyalty and devotion among the team members, assisting them to identify with the leader. Inspirational motivation relates to the ability of the leader to provide a vision to its followers and motivate them to work in that direction. Intellectual stimulation activates the followers to be risk-taking and innovative at work. Individualised consideration, is related to the behaviour of the leader to pay attention to the individual needs of the followers.

The State of DevOps Report, published by DORA in 2017, discovered a correlation between transformational leadership and organisational performance (Puppet, 2021). Although transformation leadership still comprises of the dyadic relationship of leader-follower there are certain transformational leader characteristics, see Table 2-7., which indicate that there is a higher performance achievement when compared to transactional leadership. A strong association between transformational leadership behaviour and desirable outcomes can lead to that improved performance (Sahu et al., 2018).

Table 2-7 Characteristics of Transformational Leaders.

Researchers	Characteristics of Transformational Leaders
Burns (1978)	Provide change and movement in an organisation
Pondy (1989)	Broaden and elevate the interests of their constituencies
Kuhnert and Lewis (1987)	Are capable of having profound and extraordinary effects on people by causing a shift in the beliefs, needs and values of followers; thus, followers have the potential to become leaders themselves.
Den Hartog et al. (1997)	Transform the organisation by defining the need for change, creating new visions, mobilising commitment to these visions and by providing awareness of the organisational vision and goals.
Eisenbach et al. (1999)	Are referred to as change agents.
Egri and Herman (2000)	Tend to direct specific activities as much as to alter moods, to evoke symbolic images and expectations and to inspire desired objectives.
Antonakis et al. (2003)	Are proactive, raise follower awareness for collective interests and motivation followers to achieve out of range goals.
Avolio and Bass (2004)	Inspire followers to go beyond their own self-interests for the good of the organisation with their vision. They heighten the awareness of followers with vision they create and the strategies for reaching them. They develop higher level needs for followers such as achievement, autonomy, and affiliation, which can be both work and not work related.

2.6.4. Servant Leadership

In 1970, Robert Greenleaf published an essay entitled "The Servant As Leader" that introduced the term "servant leadership. "Servant leadership is a holistic leadership approach that engages followers in multiple dimensions such as relational, ethical, emotional, spiritual in order to empower them to grow into what they are capable of becoming" (Eva et al., 2019). Servant leadership was developed as a theory of ethical leadership which is comprised of values such as integrity, altruism, humility, empathy, healing, personal growth, fairness and justice, empowerment, etc. (Yukl and Gardner, 2020). In that context, a characteristics and trait groups, which represent servant leadership, includes: Empathy; Active Listening; Healing; Awareness; Persuasion; Conceptualisation; Foresight; Stewardship; Commitment to the growth of people; Building community.

The servant leader commits time and effort to understand each follower's background, core values, beliefs, and behavioural patterns not only in the professional but also in the personal domain (Eva et al., 2019). Additionally, charismatic, and transformational leaders attempt to communicate their leadership qualifications through 1) appealing to follower values, 2) communicating in symbolic ways that are clear and vivid, and 3) displaying emotional conviction and passion for the mission (Yukl and Gardner, 2020). Lastly, there is research to suggest that leadership development programs should consider the benefits of servant leadership due to its gender-neutral style and synergistic ability to develop leaders as skilled mentors (Sims et al., 2020).

2.6.4 DevOps Adoption Leadership

Leading DevOps practice and principle adoption has become a fundamental element to the success of DevOps teams (Bass et al., 2015; Maroukian and Gulliver, 2020). A high-performing organisation is characterised by adoption of DevOps practices by multiple teams and departments, high responsiveness to mean-time-to-recover from product system failure, i.e., end-user experience degradation, mean-time-to-market, change failure rate, and embedding security deep into the source code (Geurts, 2016). However, there is still limited research outlining the leadership style, traits, competencies, and skillset accompanied with high-performing DevOps-oriented organisations. There is limited research outlining the leadership style, traits, competencies, and skillset accompanied with organisation who decided to formulate a path towards transitioning to a DevOps-oriented enterprise with a specific adoption roadmap which also included a training plan and communication plan per affected stakeholder role.

The State of DevOps Report (Puppet, 2021) stated of a correlation between transformational leadership and organisational performance which indicated that the more an organisation becomes inclusive of the

transformational leadership style the higher performance is noted in terms of rapid deployments in productions environments and higher resiliency to failure is achieved. Transformational leadership comprises of four dimensions: idealised influence, inspirational motivation, intellectual stimulation, and individualised consideration and the leader aims to inspire and transform followers by appealing to their ideas and emotions (Yukl and Gardner, 2020). In addition, the State of DevOps Report conveys that DevOps leaders with a servant leadership mentality inspired better team performance (Puppet, 2021). In essence, the leader should be serving rather than being served, and, therefore, creates an environment of trust, collaboration, and reciprocal service, which ultimately leads to higher performance (Greenleaf, 2002). Servant leadership was developed as a theory of ethical leadership, which is comprised of values, such as integrity, altruism, humility, empathy and healing, personal growth, fairness, justice, and empowerment (Yukl and Gardner, 2020).

2.7. Synopsis of Considerations on Research Aims and Questions

The research aim, outlined in Section 1.5, states that this research aims to "*understand the effect that Agile, Lean and DevOps practice and principle adoption have on structured service management processes and how it is possible to lead adoption efforts through a specific model design for product development teams in a software-intensive organisation.*" Chapter Two has provided a detailed account of agile software development, lean IT and DevOps practices and principles, benefits, and challenges adoption. The detailed accounts of research-based evidence of agile software development, lean IT and DevOps practices and principles indicate that there is evidence to support productivity improvements for software product development teams in software-intensive organisations that have adopted a structured service management or 'waterfall' product development approach (RQ1). However, it is also unclear whether a process-driven structured service management approach can deliver benefits to a DevOps-oriented environment and its teams (RQ2). There is also very limited evidence to indicate a relationship between how Leadership affects DevOps adoption within a software-intensive organisation, and which leadership style is more attributable to a potential DevOps adoption leader role (RQ3). Therefore, to establish the link between structured service management approaches with agile, lean, and DevOps approaches an exploratory study should be designed which carefully considers aspects of that relationship. In addition, the exploratory study should be inclusive of leadership aspects present in DevOps adoption which is, currently, lacking research community publications.

2.8. Chapter Summary

This chapter has developed the motivation and background introduced in Chapter One, keeping in reference the research questions. The chapter takes note of the frameworks, methodologies, and process-driven

approaches, present in agile, lean and DevOps global communities and which focus on software-intensive organisations in the context of service management and project management. In addition, particular mention is made to agile software product development approaches and lean management practices and principles. The most up-to-date advances in the domain of software product development is covered as an emerging set of practices and principles widely known as DevOps. Relevant leadership styles and traits are also examined in the context of adoption of new practices and principles for software product development teams. Evidently, there are academic studies to support 'DevOps adoption' frameworks, methodologies, models, Dojos. However, that plethora converts to shortage, when considering the research topic of 'DevOps adoption leadership'. The chapter concludes with an summary of content and definition and justification of the research questions, as stated out in Chapter One.

Chapter Three
Research Design and Methodology

3.1. Chapter Overview

This chapter develops and outlines the research design by identifying the appropriate research philosophy, methods, and techniques. The chapter starts by highlighting a number of research philosophies. This chapter also presents the benefits and limitations of each research philosophy followed by an illustration and justification of the selection of the research philosophy. A clear description of the research methods and techniques, which are directly applied in this research, are also provided. Lastly, an account of the thesis ethical considerations and research threat validity is provided.

3.2. Research Philosophy and Paradigms

Research philosophy refers to a system of beliefs and assumptions about the development of knowledge. Moreover, before pursuing any kind of research a researcher must think about and reflect on their own set of assumptions and beliefs in relation to the world and the research phenomena. Research paradigms reflect the different way in which researchers view the world and ultimately lead to selection of the research methods used to perform the research. A range of research assumption exist, these include:

- *Ontological* assumptions, which regards the way in which the researcher studies and see the research artefacts themselves such as organisations, practitioners, etc.;
- *Epistemological* assumptions, which explain and understanding how we know what we are aware of (Crotty, 1998) and acquire the truth and acceptable knowledge (Straub, 2009);
- *Axiology* assumptions, which relates to the perception of the researcher's values and ethics including the research participants;
- *Methodological* assumptions, which regards the logical step-by-step approach taken to select research methods taking into account the end-to-end process (Crotty, 1998).

Research paradigms include positivism, realism and interpretivism (Bryman and Bell, 2018) – see figure 2.1. Additionally, pragmatism can be considered (Saunders et al., 2018). The three research paradigms that are taken into account as part of the thesis regard positivism, interpretivism, and pragmatism since, in the field of *Information Systems*, these three can be used to guide the development of specific research (Niehaves, 2005).

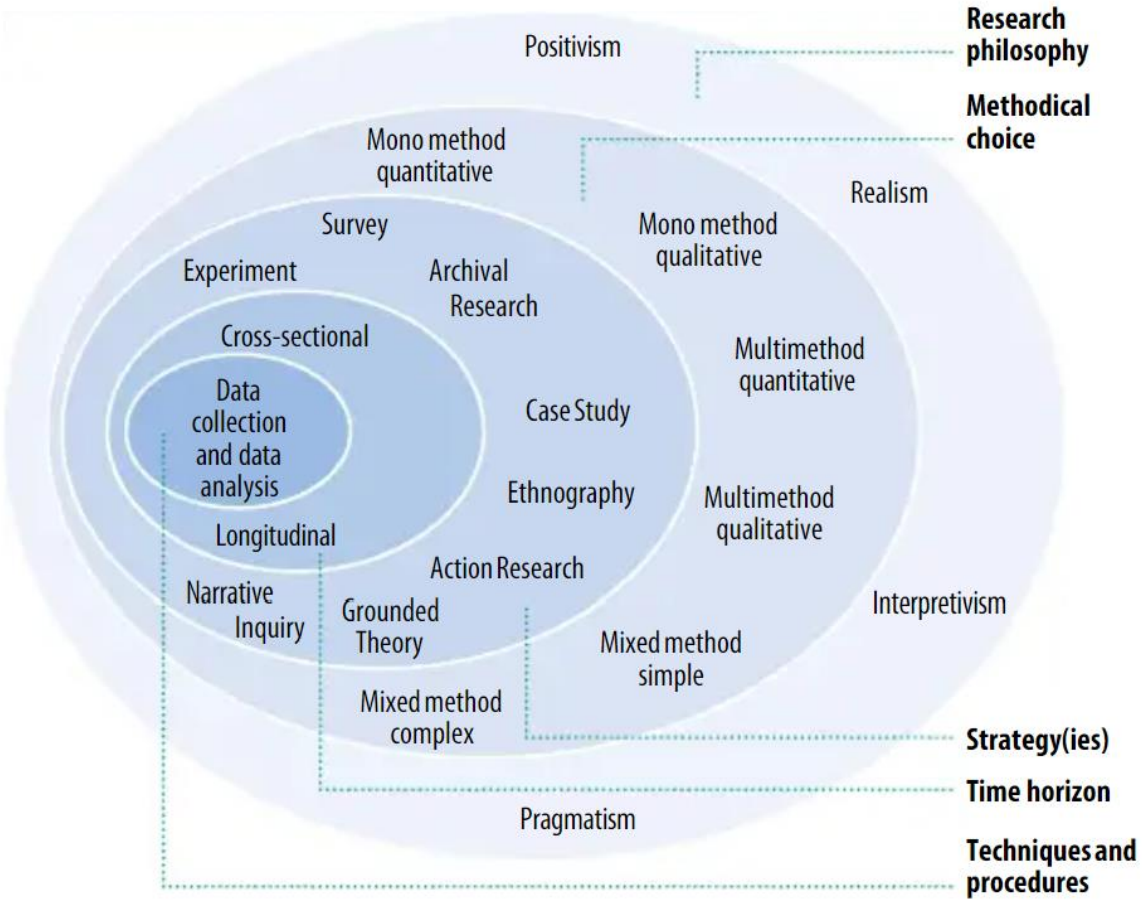


Figure 3-1 The research onion (Saunders et al. 2018).

3.2.1. Positivist Philosophy

The research philosophy of positivism presumes that reality is objective i.e. there is an observable social reality (Saunders et al., 2018). Positivism assumes the researcher is independent of the research i.e. two or more independent researchers – observing reality with the same hypothesis, would deduce similar results and conclusions. Moreover, research can be described as positivist research if the research applies formal propositions, quantifiable measures of variables, hypotheses testing, and draw inferences regarding such phenomenon; depending on the population sample. Normally, this leads to utilisation of quantitative data collection methods, often involving a large number of respondents; utilising statistical analysis, which is particularly useful when producing predictive knowledge concerning the phenomena under investigation (Niehaves, 2005). A brief comparison between these paradigms is shown in Table 3-1.

Table 3-1 Comparison of research philosophies (adapted from Saunders et al., 2018).

Research Belief	Positivism	Interpretivism	Pragmatism
Ontology	Objective and independent of social actors	Socially constructed and subjective	Multiple views chosen to best enable answering of research question
Epistemology	Only observable phenomena can provide credible data / facts. Focus on causal connections and reducing phenomena to its simplest elements	Subjective meanings and social phenomena. Focus upon the details of situation and a reality behind these details.	Either or both observable phenomena and subjective meanings can provide acceptable knowledge dependent upon the research questions. Focus on practical applied research, integrating different perspectives to help interpret the data.
Axiology	Research is undertaken in a value-free way, the researcher is independent of the data and maintains an objective stance	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.
Methodology	Highly structured, large samples, measurement, quantitative, but can use qualitative	Smaller contextual samples, in-depth investigations, qualitative	Mixed or multiple method designs, quantitative and qualitative.

3.2.2. Interpretivist Philosophy

Interpretivism is subject in nature and involves researcher's understanding differences amongst human's roles as social actors (Saunders et al., 2018). Interpretivism concentrates on understanding human behaviour from the participant's own frame of reference (Collis and Hussey, 2013). In comparison to positivist, researchers using interpretivism believe that reality can be evaluated subjectively through social constructs, such as meaning, languages and culture. Researchers using interpretivism usually link contextual meaning to the data that they get from people. Unlike positivist, interpretivism concentrates on meaning rather than measurements to understand phenomena (Niehaves, 2005). Interpretivist research commonly utilises qualitative data as the main data collection method.

3.2.3. Pragmatist Philosophy

Pragmatism emphasises the importance of the research question in determining the research philosophy underlying the research. Pragmatism states that it is possible to work within both positivism and interpretivism paradigms by integrating different approaches; in order to collect and interpret data (Saunders et al., 2018). Pragmatism is the basis of the mixed methods approach, where both quantitative and qualitative data can be used to answer the research question, as long as data is deemed appropriate to have a positive impact on the value system (Tashakkori and Teddlie, 1998). A growing number of researchers are applying pragmatism in information systems studies, in part due to its ability to balance and interplay subjective and objective contributions (Goles and Hirschheim, 2000). Pragmatism recognises the significance of theory as a means for illustrating, describing, and predicting phenomena, yet at the same time aims to subject theory to the test to determine its usefulness or practical value (Saunders et al., 2018). Due to this balance, pragmatism offers an appropriate practical basis for research in the information systems field, which is characterised as an applied field (Goles and Hirschheim, 2000).

3.2.4. Selected Research Philosophy

The selection of the research paradigm should be based on the research questions. Interpretivism relies on social interaction to explain phenomena. Positivism relies on quantifying the social reality using proposition and hypothesis testing. Pragmatism is the most appropriate paradigm for use in this research since it can justify the use of a mixed- or multi-method approach (Tashakkori and Teddlie, 1998). It is the thesis author's belief that there can exist multiple realities in interpreting the world and undertaking research, that no single point of view can provide a holistic view. Moreover, the meaning of 'pragmatic' surfaces from abductive logic which equates to commonsensical, down-to-earth thinking process to observations. The rhetoric may integrate both formal and informal styles of writing since pragmatism also establishes a business-oriented focus. While this approach is compatible with qualitative-dominant interpretivist understandings of socially constructed reality, the emphasis is on interrogating the value and meaning of research data through examination of its practical consequences (Kelly and Cordeiro, 2020). This is particularly useful in organisational environments where practice is intertwined with 'ways of working' through which knowledge is produced. Therefore, pragmatism, offers researchers a focus in organisational settings that can move beyond objectivist conceptualizations and which have dominated research in the organisational sciences, to exploring and understanding the connections between knowledge and action in context. 'Knowing' and 'Learning' in this sense, has the potential to transform practice.

Methodologically, the implication is that researchers are better equipped to deal with complex, dynamic organisational processes where action, even if carefully planned, can have varied spatial or temporal

qualities. Pragmatic inquiry recognizes that individuals within social settings (including organisations) can experience action and change differently, and this encourages them to be flexible in their investigative techniques (Kelly and Cordeiro, 2020).

There are various methods each one with its own limitations which when acting in isolation can protrude certain methodological constraints. However, when there are multiple methods combined under a study they can provide outcomes in complimentary manner. The pragmatic philosophy was coupled with mixed method which is a methodology to conduct research that involves collecting, analysing and integrating qualitative e.g. interviews, focus groups and quantitative e.g. surveys, experiments, research. The mixed method combines methods to create a single dataset. The multi method approach provides structure to research that is divided into segments with each producing a specific data set (Flick, 2020).

The dual perspective of how pragmatism establishes a unified philosophical approach for both; the quantitative and qualitative studies (interview, survey, focus group) that formulate part of this thesis, allows the researcher to design and produce a validated conceptual model from exploratory and confirmatory research. The philosophical approach will extend on the evaluation of the validated model to highlight the degree of support for the research aim and questions posed in section 1.7. Therefore, the adoption of a 'pragmatic' mindset is most appropriate for the research acc

3.3. Research Design and Methods

3.3.1. Literature Review Search Design

The pre-study research method used for the research was a thorough review of the literature, see Figure 3-4, concerning agile product development, lean software development, and DevOps practices and principles including service management and project management disciplines. The purpose of the review was to identify among worldwide published research papers all relevant studies that consider agile product development, lean software development, or DevOps as its main or secondary subject. Moreover, the approach adopted for the research takes into consideration established review process for identifying, assessing, and interpreting all available research evidence about agile, lean and DevOps practices in IT and more specifically software and product development (Webster and Watson, 2002). There are distinct stages which regard identification of inclusion and exclusion criteria, identification of relevant literature by conducting comprehensive and exhaustive search, selection of primary studies based on inclusive/exclusive criteria, data extraction and synthesis of evidence and interpretation of results.

Inclusion and Exclusion criteria

To select the literature review sources the following inclusion criteria were considered Saunders et al., 2018):

- Studies had to provide empirical data on agile product development, lean software development and DevOps;
- Studies could originate from both Academia and Industry;
- Quantitative and qualitative research studies should be published up to 2020;
- Studies should be written in English;
- Relevance to the research study;
- Peer reviewed;
- Published in conference proceedings or in scientific journals or reports from reputable publishers or books.

Exclusion criteria for literature review sources were the following:

- Studies did not focus in agile product development, lean software development and DevOps practices and principles;
- Studies should not present only the opinion of the researcher(s), "lessons learned" studies (papers without a research question and research design) and simulation studies (computer experiments that involve creating data by pseudo-random sampling);
- Should clearly discuss the agile product development, lean software development and DevOps practices and principles of software intensive product / systems / services (IEEE 2000);
- Should be related to the software domain e.g., not related to medicine, biology, physics, etc.
- Should be peer-reviewed scientific articles e.g. not related to presentations, call for papers, keynote speeches, prefaces;
- Studies should not be short papers;
- Studies should not be duplicate articles.

3.3.2. Phase One (Exploratory study) – Qualitative Method (Interviews)

Following the establishment of the main concepts from literature review, the main aim of the research focuses on understanding the current state of structured, agile, lean and DevOps approaches in the context of why and how organisations and organisational teams are adopting or have adopted certain practices and principles. This section describes the techniques and procedures for collecting and analysing the qualitative data.

Data Collection

The primary qualitative data collection design is attributed the one-to-one semi-structured interviews approach targeting industry practitioners who have had considerable experience with structure, agile, lean and/or DevOps practice and principle adoption, see Figure 3-2. The set of interviews to be scheduled will provide the basis of an improved understanding of why and how organisations and organisational teams are adopting or have adopted certain practices and principles.

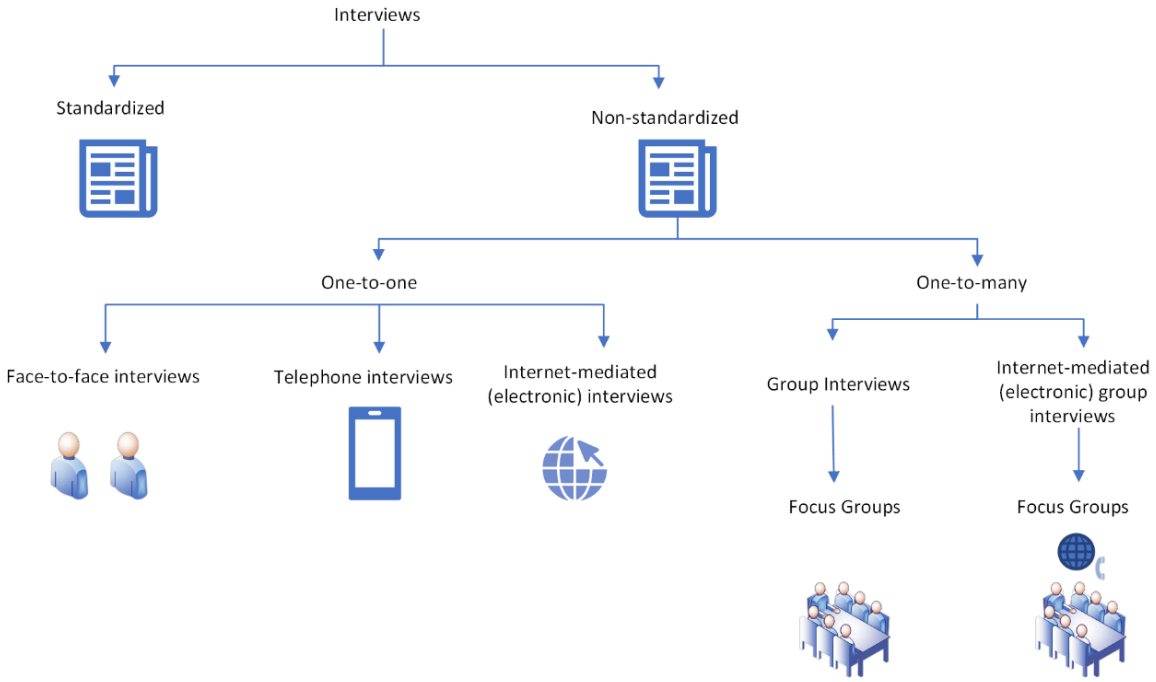


Figure 3-2 Qualitative research - forms of interview (adapted from Saunders et al 2018).

Moreover, the series of semi-structured interviews will enable the fulfilment of the exploratory study to reach meaningful and insightful outcomes. In semi-structured interviews, there is normally a list of themes and high-level questions to guide the interview discussion, ensuring that all essential aspects and perspectives are covered (Saunders et al., 2018). However, it is important to state that discussion intensity,

theme interpretation, and interview time length may vary from one interview to another. For example, some prompting questions may arise during a particular interview to fully understand the participant perspective or to respond, or further description of the questions may be require. Pilot interviews should, however, be conducted to ensure clarity and identify any issues before commencement of interviews.

Question Sequence and Phraseology

It is essential to consider the sequence of the questions to prevent ambiguity and biased responses. Furthermore, the sequence of questions within the same theme could also be changed depending on the flow of the conversation (Saunders et al., 2018). Deciding the order of questions is as important as the wording of questions (Bryman and Bell, 2011). The interview design will start with a cover page that will briefly introduce the research purpose, researcher affiliation and contact information, provide reassurance terms of confidentiality and data privacy with reference to the European Union (EU) General Data Privacy Regulation (EU, 2018) and reiterate voluntarily participation of the survey, see Appendices B and C. The cover letter will also state that the required time of interview participant commitment is the soft mark of one hour. Additionally, after the interview participant reads the consent and agrees to eligibility to participate, the interview is initiated either in physical or online virtual format. The questions will consist of short phrases with relatively accessible vocabulary, which should be designed to be both clear and easy to understand.

Sampling

The targeted participants, for inclusion in exploratory interviews, are to be individuals who have had previous experience and are knowledgeable to a certain degree with structured, agile, lean, and DevOps practices and principles adoption. The choice of a wide sample approach can be due to the need to understand and capture key aspects related to all potential practitioners' perspectives. There are not any particular demographic or individual differences that will be considered. Moreover, sampling is the process of selecting elements e.g., people, that can represent the population under study (Saunders et al., 2018). Based on that, the exploratory study relied on non-probability sampling. Additionally, non-probability sampling is the usual case for exploratory studies (Saunders et al., 2018). Moreover, snowball sampling method will be used in this study to recruit participants, i.e. where identified participants refer other potential interested participants (Saunders et al., 2018). However, the main events to be utilised to locate, acquaint with, and invite participants regard IT Service Management conferences, CIO Forums, and IT societies such as British Computer Society – Hellenic Section and the IT Service Management Forum – Greek chapter (itSMF Hellas). At the end of each interview, the researcher poses the questions to the interviewee to nominate further potential participants. Appendix B provides some background information

about the research thematology, describes the aim of the research, and requests involvement of the identified participant in the interview process. Appendix C requests consent outlining the full details of it.

Commonly, qualitative studies focus on understanding the specific domain, instead of determining generalised ability for the general populous. Accordingly, the issue of sample size is ambiguous, and it is arguably acceptable to have a relatively small sample size; assuming that conceptual saturation is achieved. According to Green Thorogood (2009), new insights can come having interviewed at least twenty (20) individuals. Furthermore, there are academic publications to support that interviewing twelve (12) people is likely to be sufficient for a series of interviews (Guest et al., 2006) (Saunders et al., 2018). However, given that our literature review has indicated lack of exploratory efforts in DevOps adoption and its leadership, the logical requirement would be to focus on inviting, at least, thirty (30) participants to the interview series to formulate a good mix of industry practitioners. In fact, when a point is reached during the interview series, where there are no repetitive themes or new emerging information, that will practically mean that the full range of ideas, concepts has reached saturation.

Data Analysis

There are several techniques used to analyse qualitative data like content analysis, grounded theory, discourse analysis and thematic analysis (Saunders et al., 2018). Thematic analysis is the basic analysis method for qualitative data, and is used to subjectively interpret, identify, analyse, and report themes from the collected data (Braun and Clarke, 2006). Furthermore, thematic analysis seems to be suitable for the aim of this study and will facilitate the research data analysis process attaining a broader description and understanding of structured, agile, lean, and DevOps adoption. The main objective of this, the exploratory study is to elucidate suggestions generated from literature review sources and identify a set of determinants that influence the usage of DevOps adoption. These determinants can then be used to develop the constituents of a model of DevOps adoption leadership. This model can then be used to support additional research to gain a quantitative understanding of the topic. To analyse interview data, this study will follow the guidelines developed by Braun and Clarke (2006), described in more detail in Table 3-2.

Identifying patterns using thematic analysis could be either deductive or inductive in nature (Braun and Clarke, 2021), however this study employs a mix of both types. This mixed approach is essential to explain how the predefined factors affect individuals' adoption of practices and principles and allow new factors to emerge from the data. An expanded illustration of emerged themes from the interview data are indicated in Chapter Four. Figure 3-3 shows the flow of research stages including qualitative research.

Table 3-2 Qualitative research - forms of interview (adapted from Saunders et al. 2018).

Phase	Process
Familiarisation with the data	Listen to every recording many times and read through the transcripts, make margin notes.
Generating initial codes	Coding interviews in a deductive manner, using the pre-defined factors from the literature. Further inductive coding was also conducted to find new issues from the data, collating data relevant to each code.
Searching for themes	Categorise the initial codes into broader themes; merge similar codes, delete unrelated codes and aggregate codes with hierarchical relations. Gather all data relevant to each potential theme.
Reviewing themes	Review the interview transcribe and extract related parts that support each theme.
Defining and naming themes	Refine the specifics of each theme, and the overall story the analysis tells, generating clear definitions and names for each theme.
Producing the report	Produce the final analysis outcomes

3.3.3. Phase Two (Confirmatory study) – Quantitative Method (Survey)

The results that will be yielded following Phase One primary qualitative analysis and evaluation is accomplished will serve as direct input to the definition of Phase Two which forms a considerable part of the research as the second quantitative empirical study. In this phase, a quantitative approach will be used applying numerical methods for data collection and statistical tools for data analysis. The aim of this study is to refine factors affecting DevOps practice and principles adoption as well as its leadership and to empirically validate deductions made and generated hypotheses using a large-scale survey.

Data Collection

It is crucial to ensure that the collected data is in a format that allows and facilitates analysis. All questions included in the surveys are closed-ended questions. The use of closed-ended questions in the surveys facilitates the collection of a higher number of responses in a shorter period of time – due to ease of completion. Furthermore, closed-ended questions are recognised as being the most effective tool for

capturing/measuring individuals' perceptions, which is needed to validate proposed research conceptual model (Johnson and Christensen, 2008). The use of closed-ended questions in a survey enables the identification and examination of relationships between constructs such as measuring the strength of the relationships and determine their direction (Saunders et al., 2018). Therefore, the use of closed-ended questions seems the most suitable method to enable and facilitate the validation of the relationships within the proposed research models. The use of this method is compatible with the methods used in almost all core behavioural models' research; such as Bhattacherjee, 2001, Venkatesh et al., 2012, Venkatesh et al., 2016.

Structured response formats are divided into five different types, i.e. binary, nominal, ordinal, interval, and continuous. Binary response formats propose to participants only two possible nonordered values such as 'yes/no'. Nominal response formats are characterised by more than two nonordered response categories for example country of resident - 'UK, USA, KSA …'. Ordinal response format is similar to nominal except that the choices are ordered or ranked such as '1-100 years, 101-1000 years, and 1001-10000 years'. Interval variables are ordinal variables that are at equally spaced intervals based on the interval variable, such as temperature.

In technology adoption research, construct's item data is often collected using the form of a Likert scale. Likert scales are one of the most reliable ways to measure opinions, perceptions, and behaviour, and enable participants to express both the direction and strength of their opinion (Garland, 1991). Some statistical techniques, such as SEM (Structural Equation Modelling), which is used to test complicated relationships between constructs, require interval scale data (Hair et al., 2017). Furthermore, there is argumentation that the use of a Likert scale (with interval data) is accepted practice that in random sampling. Likert scale is therefore considered, especially when it combined with the summated scale (Hair et al., 2010). This study aims to use 4-point Likert scale to measure items related to participants' perceptions. A 5-point or 7-point Likert scale approach could present concentrated distribution of answers towards the middle point and current research requires a more definitive approach to DevOps adoption research questions (posed in section 1.5). It is also vital to consider the number of anchors used in the 4-point scale approach of this study and the same number of anchors should be used on all questions in order conduct SEM successfully (Hair et al., 2010).

Question Sequence and Phraseology

It is important to consider the sequence of the questions to prevent ambiguity and order bias from respondents. The survey starts with a cover page that briefly introduced the research purpose, ensures confidentiality, and reiterates voluntarily participation of the survey, see Appendices E and F. The cover

letter also states (as a guide) the estimated time required to complete the questionnaire. After respondents read the consent and agree to their eligibility to participate, they proceed to answer the survey. Deciding the order of questions is as important as the wording of questions (Bryman and Bell, 2018). The questions consist of short phrases with relatively accessible vocabulary, which should be designed to be both clear and easy to understand. The order of the main questions are to be randomised to reduce question order bias and improve overall data quality. Additionally, negatively worded questions were included to flag unengaged participants. A pre-test and pilot study should be conducted to ensure the clarity and quality of the survey contents.

Sampling

Basic steps were followed to select the research sample; identifying the population of interest, specifying a sampling frame and selecting the sampling technique. Selection of the statistical data analysis tool can impact the sample size selection. PLS-SEM is the statistical method used to validate the research models. Although PLS-SEM does not require a large sample size like the covariance-based SEM, the expected complexity of the proposed research model requires a large sample to successfully utilise this analysis method. According to Hair et al. (2017) the required sample size should be determined by means of power analyses based on the construct. They provide a rule of thumb that the minimum sample size should be 10-times the maximum number of indicators pointing at a construct (manifest variable) anywhere in the model. However, in general, larger sample sizes tend to produce more reliable results and support generalisation of the findings.

According to Saunders et al. (2018), probability and non-probability are the fundamental techniques used to select an appropriate sample size. The key differences between probability and nonprobability is that every member of the population has an equal chance of selection in a probability sample, and an unknown chance of selection in non-probability sample (Bryman and Bell, 2011). Due to the lack of comprehensive sampling frame, non-probability sampling was deemed most suitable for this research. The current research utilises snowball sampling technique, i.e. where initial participants distribute the survey to other potential participants in their social networks until a sufficient sample size is reached. With this technique the potential human and selection bias is reduced. However, the diversity of samples generated via use of this method has repeatedly been questioned as respondents are most likely to identify other potential respondents who are similar to themselves, resulting in a homogeneous sample (Saunders et al., 2018). As identifying cases for this study is difficult based on time and resource constraints, snowball sampling technique seems to be suitable to provide enough sample for statistical analysis (Bryman and Bell, 2011). To enhance diversity in our sample, different sources are used to collect data e.g. emails to Microsoft Worldwide Modern Service Management community, LinkedIn networking capabilities, European

DevOpsDays conferences (Norway, Sweden, Switzerland, Turkey) and IT Service Management Forum Hellas members. In accordance to University of Reading - Research Ethics Compliance Guidance, only individuals over the age of 18 are to be eligible to participate in this study.

Descriptive and Inferential Statistical Analysis

There are two distinctive approaches in SEM analysis; i.e. covariance-based SEM (CB-SEM) and partial least square SEM (PLS-SEM). According to Hair et al. (2017), the main philosophical differences between the two methods are related to the aims of their analyses, the basis of their statistical assumptions, and nature of the fit statistics they produce. CB-SEM minimises the differences between the observed covariance matrix and the estimated covariance matrix; i.e., to achieve Goodness of Fit without focusing on explained variance. While this method is extremely popular, it has been criticised due to the rigorous assumption that the data is normally distributed, and that the sample size is large. PLS-SEM maximizes the explained variance of the endogenous latent constructs (dependent variables). PLS-SEM is able to account for measurement error like covariance-based SEM, but unlike covariance analysis, PLS does not necessarily require normally distributed data, or large sample sizes, to gain valuable research insights. PLS-SEM enables researchers to estimate complex cause effect relationship models, with many indicator variables; often facilitating solutions with 50+ items, which is not viable in CB-SEM (Hair et al., 2017). PLS-SEM is stronger in predictor specification, while CB-SEM is better for model fit testing (Chin, 1998; Hair et al., 2011). Construct measurement properties are less restrictive with PLS-SEM, constructs with fewer indicators (e.g., one or two), can be used than those that CB-SEM requires. PLS-SEM surpasses CB-SEM in terms of path modelling that requires handling of both; formative and reflective measurement models. Within PLS-SEM, the reflective measurement model represents the relationship between the latent variables and the reflective model constructs. Similarly, the formative measurement model depicts the relationship between the manifest variables and their associated model constructs.

Recently, PLS-SEM has become popular in social sciences disciplines, and offers vast potential for SEM researchers - especially in marketing and Management Information System disciplines (Hair et al., 2011; Hair et al., 2018; Ringle et al., 2012).

Considering the differences between the two SEM techniques explained above, it is decided that PLS-SEM is used for this study; due to its capability to test complex relationships despite minimum demands regarding sample size, measurement items, and data distributions. Furthermore, PLS-SEM is distribution-free and achieves a higher statistical power with smaller samples. Many technology adoption studies have employed PLS and found it to be an effective technique of analysis (Bhattacherjee, 2001; Bhattacherjee and Premkumar, 2004; Venkatesh et al., 2012; Venkatesh et al., 2016). Specifically, SmartPLS v3.3 software

package will be used to analyse the data in this study. Before conducting PLS-SEM, data screening is required to ensure its cleanliness and appropriateness.

Examining and cleaning the data facilitates elimination of errors in data analysis and an increase in the reliability and validity of the results (Hair et al., 2010). All the preliminary data analysis techniques used to clean the data - such as the influence of outliers, missing data, normality assumptions and common method bias – will be explained in detail in the relevant chapter. Preliminary data analysis is conducted using SPSS software. After ensuring the collected data is clean, and ready for advanced analysis, PLS-SEM is conducted.

The PLS path model is composed of two sub-models, i.e. structural model (also called inner model) and measurement model (also called outer model). The structural model represents the relationships between the constructs, whilst the measurement model refers to the relationships between the constructs and their indicator variables. The PLS model assessment follows a two-step process. Initially, PLS evaluates the measurement model to check the reliability and validity of the construct measures. If the assessment of the measurement model provides evidence of the measures' quality, then the structural model is examined. The primary evaluation criteria for the structural model are collinearity among constructs, size and statistical significance of path coefficients, and criteria to assess the model's predictive capabilities (Hair et al., 2017). PLS-SEM procedures and assessment will be explained in detail in the relevant chapter.

3.3.4. Phase Three (Model Evaluation) – Qualitative Method (Focus Groups)

Focus groups emerged as a research method in the 1950's in the social research as researchers expanded the open-ended interview format to a group discussion (Morgan 2003). The Focus Group approach is currently widely used in sociological studies, market research, product planning, and system usability studies (Kontio et al., 2004). Focus groups are carefully planned discussions, designed to obtain the perceptions of the group members on a defined area of interest. Moreover, focus group sessions are a research technique that collect data through group interaction on a specific topic determined by the researcher (Hasni et al., 2020).

There are typically between 5 to 10 participants (Morgan 1996) and the discussion is guided and facilitated by a moderator, who follows a predefined structure so that the discussion stays focused. The members are selected based on their individual characteristics as related to the session topic (so-called purposive sampling). The group setting enables the participants to build on the responses and ideas of the others, which increases the richness of the information gained (Mishra and Ramesh 2013). There are several textbooks and detailed guidelines available on how to plan and run focus groups making it a method that is relatively easy to adopt and use consistently (Kontio et al., 2004).

Focus groups data collection method is most suitable for types of studies where multiple perspectives needed to be obtained regarding the same problem.

Focus groups are led by a moderator who is responsible to ensure that group discussions remain focused on the research area. Advantages of focus groups include the possibility of obtaining primary data through non-verbal channels, as well as verbal channels and approaching the research area from various perspectives.

Additionally, focus groups have some disadvantages as well. Group discussions may be heavily influenced by one or two dominant individuals in the group. Also, some members of focus group may be discouraged from participating in discussions due to lack of confidence or not articulate communication skills. The nature of primary data obtained through focus groups are greatly influenced by environmental factors such as design of the room, room temperature, time of the day, etc and that can have an extension in the virtual setting as well as the physical meeting space. Moreover, data collection and data analysis using focus groups is much more challenging, compared to online surveys and interviews, due to the additional skillset required by the moderator/facilitator to manage time appropriately, be inclusive of every participant's voice during the focus group interview session. It is imperative to ensure that these difficulties are fully understood before making a final choice of primary data collection method. Figure 3-3 shows the flow of research stages including model evaluation.

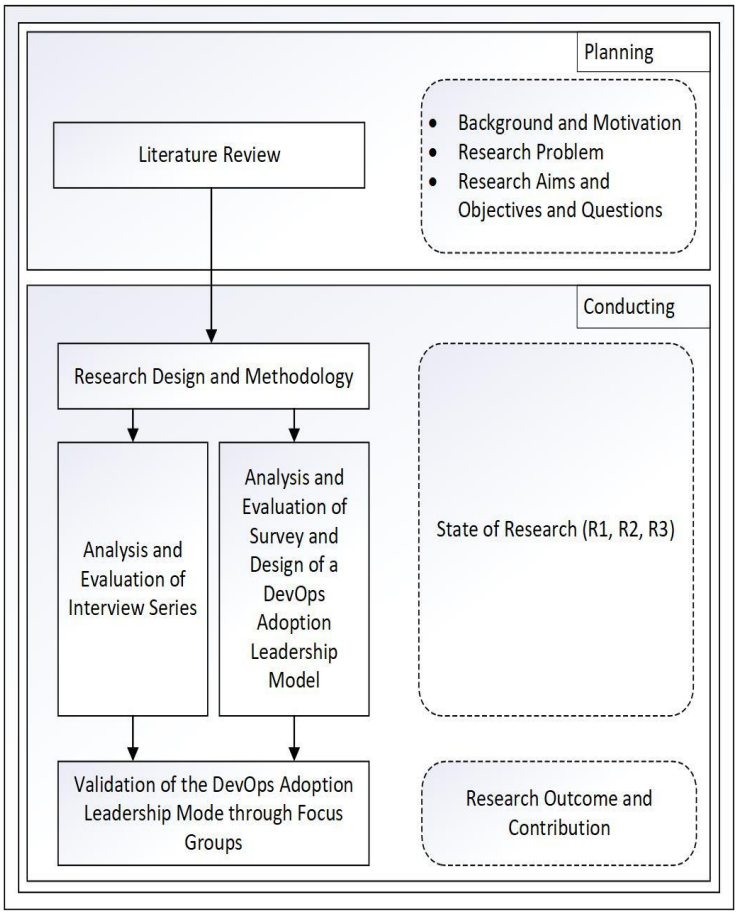


Figure 3-3 Model Validation of Research Stage.

Pre-Focus Group Activities

Before proceeding with focus group session scheduling there are certain preliminary steps that must be accomplished (Klagge, 2018), which include:

- Develop a clear and concise purpose statement of what needs to be known and why.
- Develop the discussion questions – ensure they are clear, unambiguous, and bias-free.
- Develop the script to frame the purpose of the focus group. This script should be used when making invitations and when opening the session. A closing script should also be developed thanking the participants and reiterating how the information participants provide will be used.
- Schedule the time, the place, the necessary equipment, and set the ground rules and agenda for the focus group.
- Invite potential participants and get commitments to attend from them – remind them a day or two before the focus group.

Facilitating the Session

The following are the general steps that will comprise each focus group session.

- Opening the session by the facilitator using the written script.
- Reading of the ground rules.
- Setting of the agenda.
- Reading of the questions.
- Asking for clarifications when needed.
- Making sure that everyone has the opportunity to participate on each question posed.
- Concluding the session is carried out by the facilitator.

The Role of the Facilitator

The following guidelines are provided (Klagge, 2018) for the role of the facilitator.

- Focus group facilitation is an art not a science, so care should be taken to follow the spirit and intent of the following guidelines and not to make them iron-clad rules.
- Set a friendly and informal tone.
- Make all the participants feel welcomed and relaxed.
- Get full answers by gently probing.
- Make sure everyone has the chance to participate. Ask those who have been quiet if they have anything to add.
- Head-off any arguments or disagreements that might hinder communication.
- Be neutral, open, and non-judgmental in all verbal and nonverbal responses.
- Keep the conversation flowing and on track.
- Monitor the time and remain on schedule with pre-agreed agenda.

Participation Standards

There is a specific set of standards for validating focus group findings. The requirements for the optimal number of Focus Group participants needed to adequately address research questions should be met. These requirements, based on combined researcher-based recommendations (Morgan,1996; Bader and Rossi, 1999; Beyea and Nicoll, 2000; Krueger and Casey, 2015; Klagge, 2018), are: the preferred number of confirmed participants scheduled for a single focus group session is 5-15; the minimum number of confirmed participants needed to schedule a single focus group session will be 5-8; the preferred number of actual participants needed to conduct a single focus group session is 5-8; the minimum number of actual participants needed to conduct a single focus group session will be 5 individuals.

3.4. Research Stages Flow

In this research an exploratory study (interviews), confirmatory study (survey), and model validation (focus-groups) will be conducted. The aggregate flow of research stages is shown in Figure 3-4.

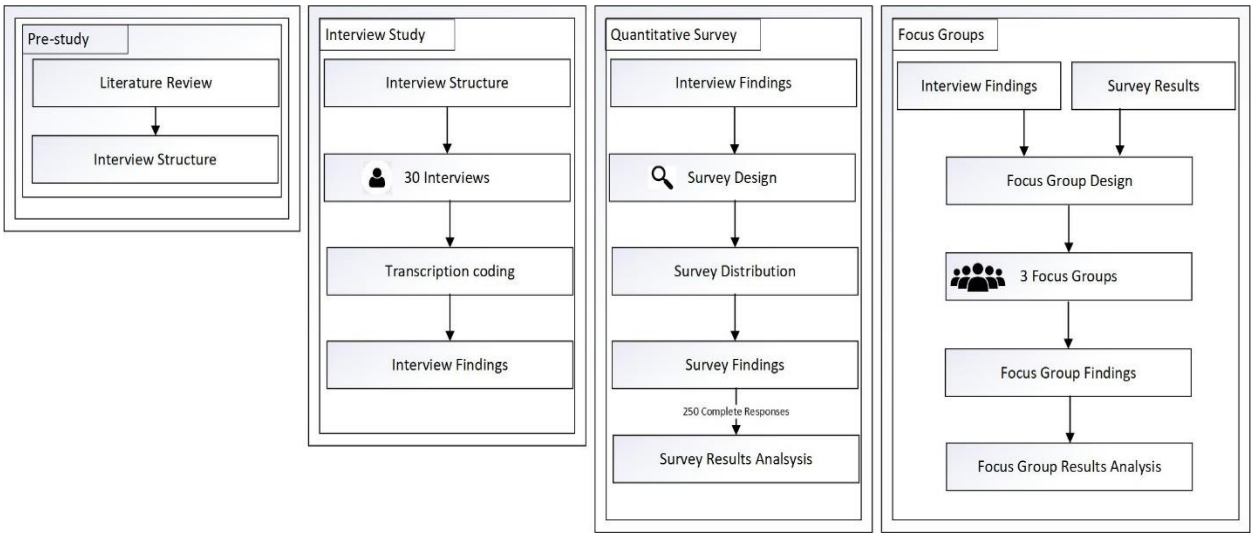


Figure 3-4 Flow of research stages.

3.5. Ethical Considerations

During the research, it is essential to ensure that there are no negative ethical concerns. All aspects of research requesting participants to provide personally identifiable information or otherwise will always week for consent prior to gathering and registering responses. Moreover, appropriate safeguarding should be required to ensure that no sensitive or inappropriate information is extracted from interviewees, online survey participants and focus group participants. Additionally, a statement of confidentiality will be presented prior to an interview and provided on-screen prior to attempting the online survey to i) ensure that respondents understand how data will be used by the researcher, ii) encourage respondents to provide a non-biased response rather than a response that they believe to be socially acceptable (Cacciattolo, 2015). During each data collection method, it should be specified that respondent participation is anonymous, and that privacy will be maintained according to *The University of Reading - Ethics Compliance Guidelines*. Moreover, participation to either the interview series or the online questionnaire must be explicitly stated as a voluntary act of commitment. Moreover, to address any concerns and/or clarifications sought by participants the researcher's email address will be shared to facilitate all necessary communication.

Payment will not be required from participants of the study. In any case, participants will be provided the researcher's email address to initiate communication, if required, for any issues or concerns that might arise in accordance with research interviews or survey e.g., withdrawal from the interview process. Furthermore, research that is conducted in settings where participants are non-native speakers of English can involve additional ethical reflections for researchers. In this context, it is required to properly segment native speakers from non-native speakers of English. The non-native English speakers shall be categorised as speakers of English as an additional language (EAL) (Cacciattolo, 2015). For interviews specifically, EAL speakers will be asked whether they feel comfortable answering questions in English as part of the aforementioned data retention and usage scheme consent.

The focus groups and survey participants will share industry knowledge for the sole purposes of current research. Therefore, prior to participating to the survey they will have to provide informed consent on a Microsoft Form checkbox to 'opt in', which will activate the branch for the survey questions to be answered.

3.6. Validation

3.6.1 Potential Risk to Internal Validity

In the context of this chapter, there are certain known threats to internal validity that relate to possible bias in the participant selection process for both; the qualitative (interviews) and quantitative (online survey) empirical studies. The communication channels, utilised to invite interview participants, were European conferences in the context of DevOps, CIO Forum, and IT service management. In addition, the majority of interview participants related their work to closed-sourced software products. The semi-structured interview series and online asynchronous survey approach undertaken offer rigorous procedures for data gathering and analysis but with a certain degree of research bias. It is probable, that other researchers might deduce different findings and outcomes looking at the same set of data, but the authors believe the main perceptions would be preserved. This is a typical threat related to similar studies, which do not claim to generate definitive findings.

3.6.2 Potential Risk to External validity

External validity is considered under the lense of the qualitative study. There is heavy reliance on each of the interviewed practitioners' subjective perception. However, currently there is no objective approach to measure whether a DevOps transition journey, in the context, of practice and principle adoption within organisations can be associated to successful outcomes. Although the viewpoint of the interviewed practitioners is considered with different backgrounds, working in organisations from nine (9) different industry domains and ten (10) different countries the researcher does not claim that research results from

this contribution are valid to other scenarios. The varying degrees of background and experience of the interviewees can potentially lead to biased results in terms of the industry role or the industry itself from which the interviewee was sourced. The same could apply for the online asynchronous survey to which participants will provide responses based on their existing knowledge of topics.

3.7. Chapter Summary

This chapter provided a clear explanation of the research philosophy, methods, and techniques to be applied and used in this research. A detailed description and outline of the research approach and tools utilised in this research has been also provided, expressing consideration of contextual benefits and limitations. Additionally, research method validity is also discussed. A mixed method approach (series of thirty interviews followed by an online asynchronous survey) will be used to triangulate the results between methods to clearly understand the research phenomena and attain a greater understanding of the obtained results. Moreover, the interview transcripts will be thoroughly examined in NVivo in terms of thematic analysis. The validation of the results will be examined using PLS-SEM through which model validation will be attained. The evaluation of the validated model will undergo a further confirmatory study of focus groups whereby industry leaders will provide necessary feedback to the model. The next chapter presents details of the exploratory study and empirical research conducted as part of the thesis.

Chapter Four
Exploratory Study Analysis and Outcomes

4.1. Chapter Overview

This chapter aims to explore the following research questions, see section 1.5, building on the baseline of acquired knowledge from Chapter Two literature review and executing on the design and planning of research methods of Chapter Three:

- (*RQ1) Which agile, lean and DevOps practices and principles can improve productivity for software product development teams in software-intensive organisations that have adopted a structured service management approach?*
- *(RQ2) Are DevOps-oriented environments and their teams an extension or a replacement of process-driven structured service management approaches?*

A deeper understanding of structured, agile, lean and DevOps practices and principles, that echo among industry practitioners, will contribute towards designing and developing a clearer picture of where DevOps adoption and its leadership characteristics currently stand in software intensive organisations. Additionally, the outcomes of the analysis of interview transcripts are explicitly explained, forming a baseline for use in the research conducted and described in Chapter Five.

4.2. Introduction

Several DevOps practices, principles, frameworks, models exist in the context of what needs to be in place for a DevOps team, its mindset, behaviours, technology-specific skillset, etc. to achieve their target metrics and ultimately, what is most meaningful to the team's success. However, limited research has investigated how distinct leadership traits and styles can co-exist, interact, and result to DevOps adoption due to that kind of interplay.

4.3. Interview Process and Structure

To capture contextually relevant data, semi-structured interviews were conducted with thirty (30) practitioners in companies working within a wide range of countries (i.e. Czech Republic, Estonia, Italy, Georgia, Greece, The Netherlands, Saudi Arabia, South Africa, UAE, UK). All interviewees contributed to

DevOps adoption processes in their respective companies. Participants were recruited using two approaches: 1) through direct contact at an ITSM / DevOps event in Europe, and 2) via a general call for participation posted on professional social media networks; including LinkedIn and IT societies such as IT Service Management Forum (itSMF) and British Computer Society (BCS) – The Chartered Institute for IT. Moreover, to achieve a heterogeneous perspective, and to increase the wealth of information, practitioners from a variety of organisations were invited and consulted. All of the interview participants provided consent, see Appendix C, to the Terms of Interview Participation, refer to Appendix B. Although face-to-face interviews were preferred, a number of web interviews were conducted using a range of conferencing technology platforms (Microsoft Skype for Business and Zoom). Table 4-1 presents the characteristics of the participants. The tabulated data can be viewed by referring to the following acronyms:

- PX - participant experience in years;
- CN - country of work;
- CS - company size (Micro - MC < 10, Small < 50, Medium - M < 250, Large > 251) (European Commission, 2003).

To maintain anonymity, in conformance with the human ethics guidelines, the participants are referred as P1–P30. At the beginning of each interview the interviewee consented to: I) an audio recording, and ii) the produced transcript being used solely for the purposes and in the context of the current research. Instructions were clear to state that no personal names nor organisation brand names would be disclosed as part of research documentation or in case of publication of results.

Table 4-1 Interview participant profile (PX = participant experience, CN = country of work, CS = company size).

P#	Job Title	PX	CN	Domain	CS
P1	PMO Director	14	Saudi Arabia	Aviation	L
P2	Principal Consultant, IT Service Management	13	Italy	IT Consulting Services	L
P3	CIO	26	Greece	Insurance	L
P4	Principal Consultant, IT Service Management	11	UK	IT Consulting Services	MC
P5	Managing Director, IT Service Management	32	UK	IT Consulting Services	S
P6	Smart Systems Manager	23	Greece	IT Consulting Services	L
P7	Senior Digital Transformation Technologist & Solution Practice Lead	30	UAE	IT Consulting Services	L
P8	Principal Consultant, IT Service Management	34	UK	IT Consulting Services	L
P9	Founding Consultant, IT Service Management	19	UK	IT Consulting Services	S
P10	Managing Director	29	UK	IT Consulting Services	S
P11	Head of Remote Transactions	16	Greece	Banking	L
P12	Consultant	34	Netherlands	IT Consulting Services	M
P13	Deputy CIO	22	Greece	Construction Management	L
P14	Head of Applications	18	Greece	Lottery	L
P15	Principal Consultant, IT Service Management	21	South Africa	IT Consulting Services	MC
P16	Founding Consultant, IT Service Management	34	UK	IT Consulting Services	MC
P17	Managing Director, IT Service Management	19	UK	IT Consulting Services	MC
P18	Managing Director and Lead Consultant	14	UK	IT Consulting Services	MC
P19	IT Operations Manager	13	Greece	Lottery	L
P20	IT Operations Manager	15	UK	Government	M
P21	Founding Consultant, IT Service Management	34	UK	IT Consulting Services	MC
P22	Assistant General Manager, IT Operations	28	Greece	Banking	L
P23	CDO	13	Estonia	Government	L
P24	CIO	20	Greece	Insurance	L
P25	CIO	27	Greece	Aviation	L
P26	Development Team Lead	11	Greece	Lottery	L
P27	IT Operations Lead	12	Georgia	Government	M
P28	Business Development Director	18	Greece	IT Consulting Services	L
P29	Operations and Innovation Lead, IT Services	11	Czech Republic	Courier Services	L
P30	CIO	28	Greece	Automotive	M

Interviews were conducted between September 2018 and January 2019. The interviews lasted a minimum of 34 minutes, a maximum of 67 min, and an average of 50 min. Data collection and analysis was aggregated in order to answer the research questions which were mapped to interview questions - see Table 4-2. The mapping represents the segmentation of research undertaken for each particular research question (RQ) entailed in this thesis. Effectively, the obtained results per question could be categorised so that an aggregate dataset is produced from responses received which can be analysed per research question.

Table 4-2 Research to interview questions mapping.

Research Question	**Interview Question**
Data collection for segmentation purposes	1, 2, 3
RQ1) Which agile, lean and DevOps practices and principles can improve productivity for software product development teams in software-intensive organisations that have adopted a structured service management approach?	4, 5, 6, 7, 8, 9, 10, 11, 12, 15, 16, 20
RQ2) Are DevOps-oriented environments and their teams an extension or a replacement of process-driven structured service management approaches?	13, 14, 15, 20
RQ3) Can Leadership affect DevOps adoption within a software-intensive organisation? What is the leadership style that can be attributed to the DevOps adoption leader role?	19, 20 17, 18, 20

The whole set of interview questions is available in Appendix A.

4.4. Analysis and Evaluation

4.4.1. Interviewee background segmentation

The semi-structured interview consisted of twenty (20) interview questions as seen in Appendix A. The first three questions aimed to collect data on interviewee demographics i.e. job role, industry domain, and working country, see Figures 4-1 – 4-3 for a demographic breakdown). The interview series consisted of thirty (30) participants from nine countries Greece (11), UK, (10), Saudi Arabia (2), Czech Republic (1), Estonia (1), Georgia (1), Italy (1), Netherlands (1), South Africa (1), United Arab Emirates (1). Fifteen (15) participants were IT consultants and fifteen (15) were employed at customer organisations - characterised

as “service providers” according to ITIL® (AXELOS 2019) - see Figure 4-1. Moreover, service consumers of IT consultants can be service providers or other IT consultants. The service consumers for the service provider organisation can be either internal or external entities. All Greek interview participants were “service providers”. Furthermore, UK interview participants consisted of nine (9) consultants and one (1) service provider.

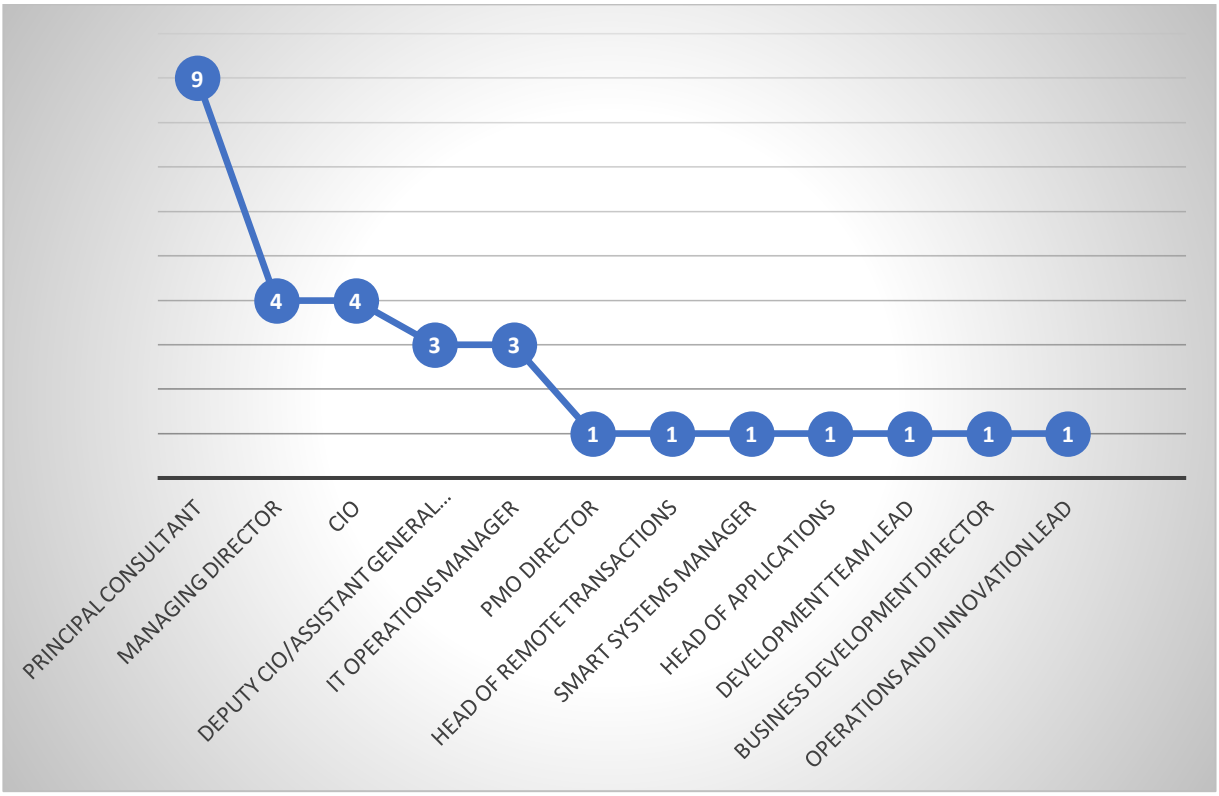


Figure 4-1 Job role of interview participants (interviewee count: 30).

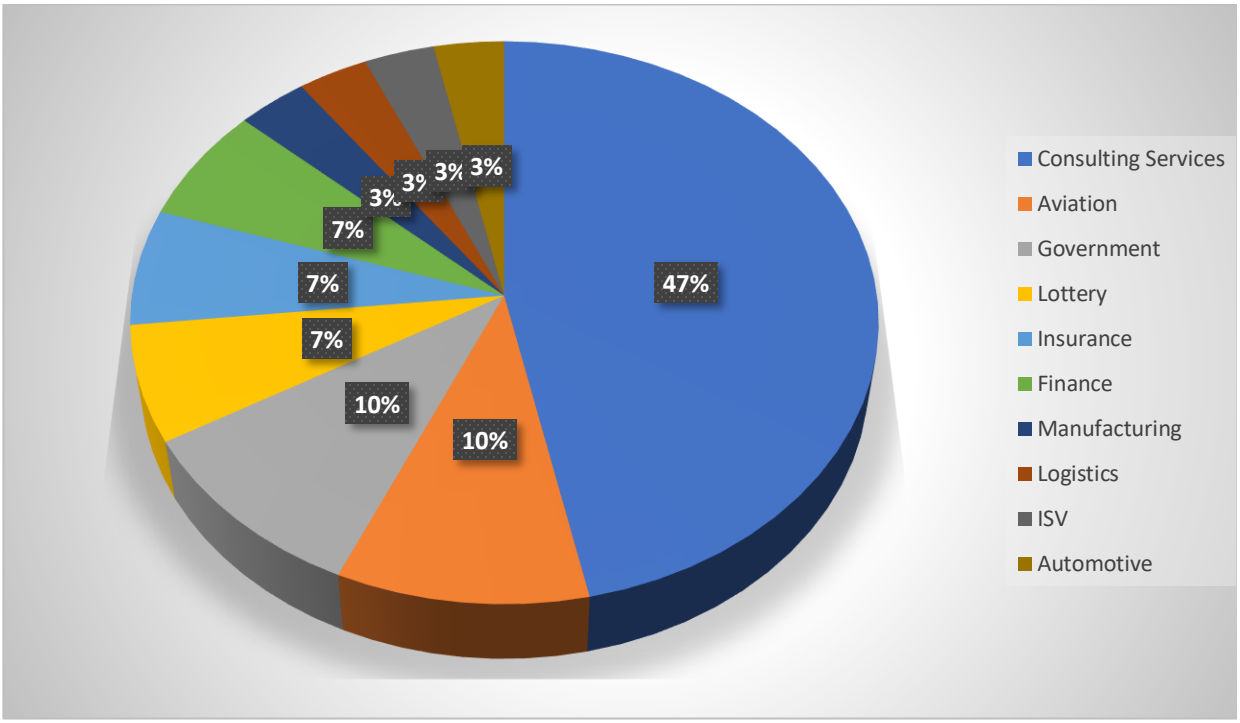


Figure 4-2 Industry of interview participants (interviewee count: 30).

There was a distinct diversity of participant roles, e.g., Principal Consultant (10), Managing Director (4), Chief Information Officer / Chief Digital Officer (6), IT Operations Manager (3), PMO Director (1), Head of Remote Transactions (1), Smart Systems Manager (1), Head of Applications (1), Development Team Lead (1), Business Development Director (1), Operations and Innovation Lead (1). Furthermore, the industries of participants were Consulting Services (14), Aviation (3), Government (3), Lottery (2), Insurance (2), Finance (2), Manufacturing (1), Logistics (1), ISV (1), Automotive (1) - see Figure 4-2.

4.4.2. Exploring and depicting the current research state

Following the demographics presented in the previous subsection the interview structure proceeded with questions related to the research questions as indicated in Table 4-2.

The interview participants were aware of, and had considerable previously experience applying a range of frameworks, international standards, methodologies, practices, and principles; such as ITIL (87%), SCRUM (73%), DevOps (63%), Lean IT (50%), PMBOK® (33%), ISO20000 (27%), PRINCE2® (27%), XP (13%), SAFe (10%) - see Figure 4-3.

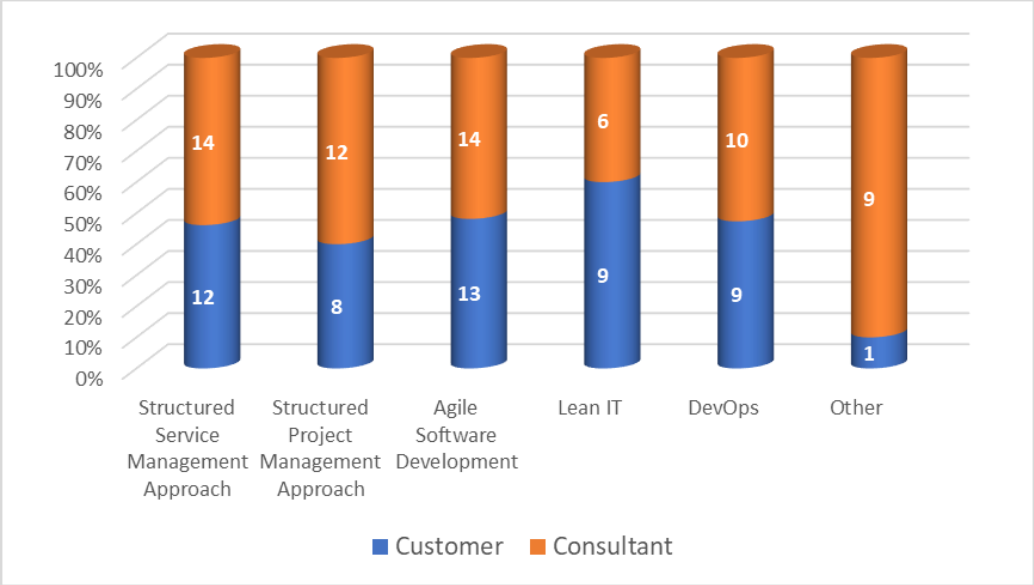


Figure 4-3 Experience level relevant to interview context in customer and consultant categorisation (interviewee count: 30).

Figure 4-1 through 4-3, indicate that the interviewee population is deemed appropriate for the type of exploratory research study pursued since there is a balance of fifteen consultants (owner, senior and middle level) and fifteen customer organisation representatives (owner, senior and middle level). Moreover, there is a good balance of knowledge for service management, project management disciplines as well as agile software development, Lean IT and DevOps and practices and principles.

Interview participants indicated that they were confident (according to preference) with specific agile, lean and DevOps practices such as Scrum (63%), Kanban (63%), Continuous Delivery (60%), Continuous Integration (53%) and Lean management (47%) - see Figure 4-4.

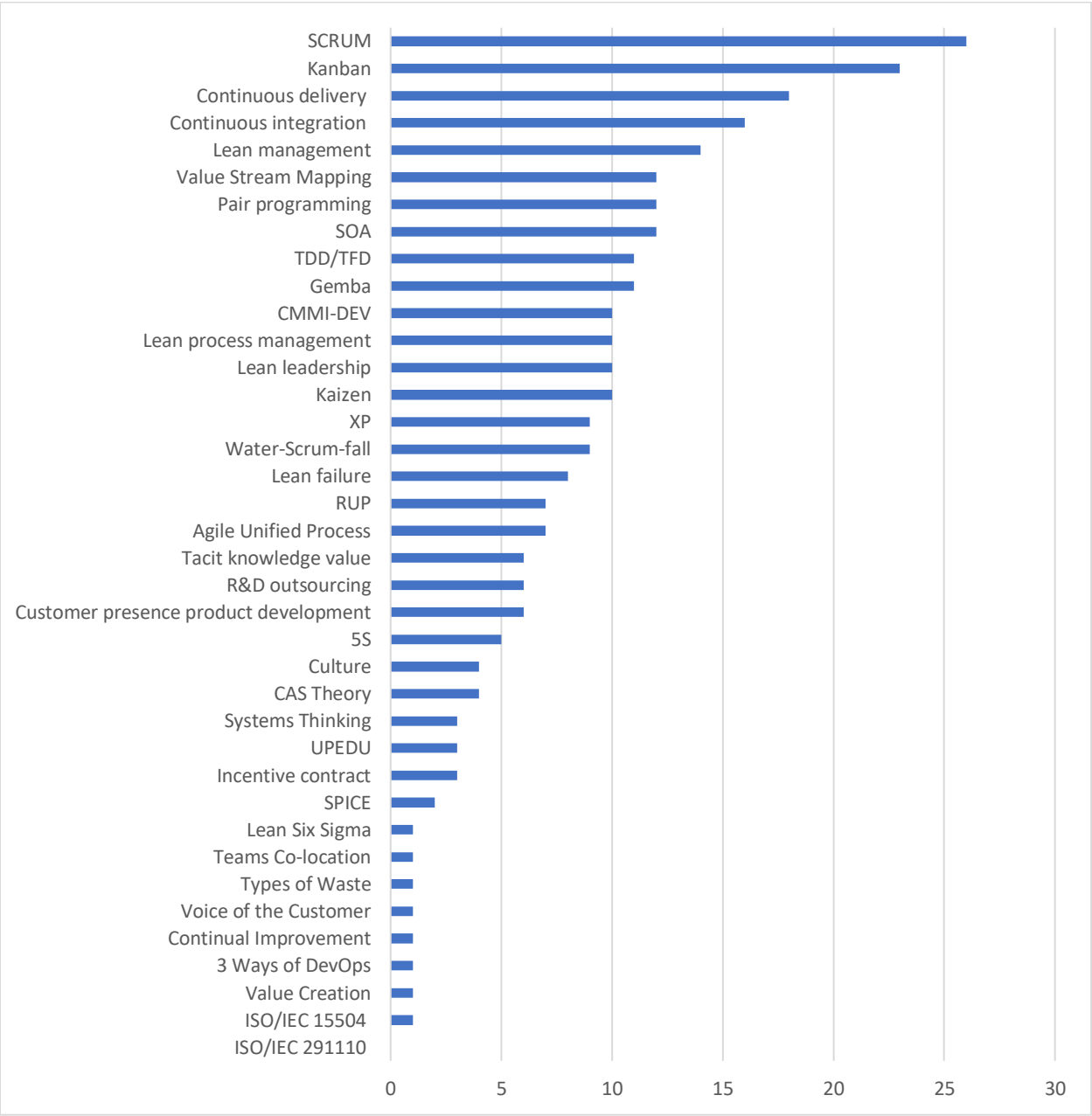


Figure 4-4 Competence of interviewee according to practices (interviewee count: 30).

Interview participants indicated that they were most competent in the areas of Scrum (87%), Kanban (77%), Continuous Delivery (60%), Continuous Integration (53%) and Value Stream Mapping (40%), Pair Programming (40%) and Lean management (40%) - see Figure 4-5.

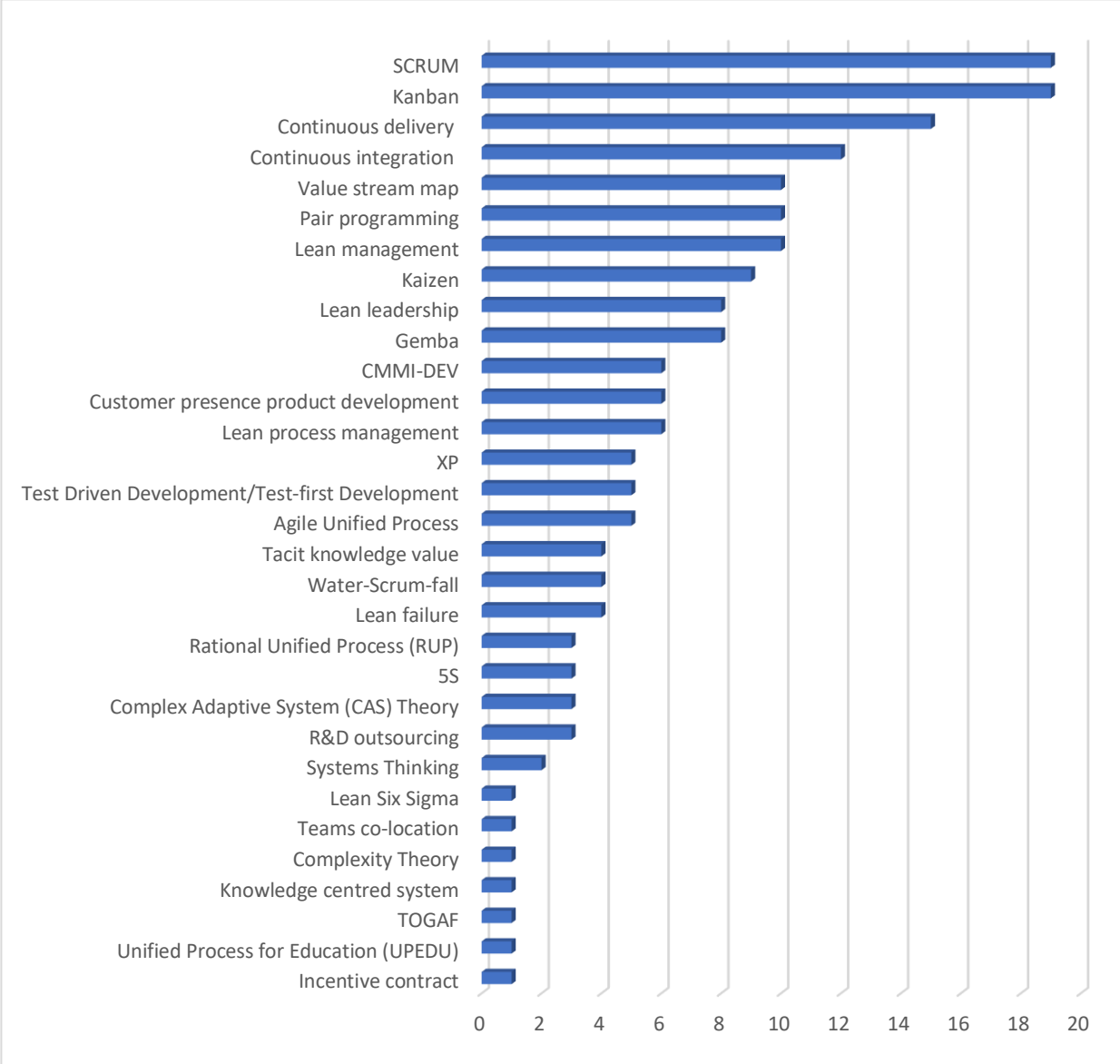


Figure 4-5 Most beneficial practices in order of preference (interviewee count: 30).

According to user preference, i.e. perception of the most beneficial approach in an organisational setting, interview participants indicated that they could confidently provide a definition for the set of agile, lean and DevOps principles such as Monitoring (83%), Automation (77%), Measurement (70%), knowledge sharing (70%) and agile software development (66%) - see Figure 4-6.

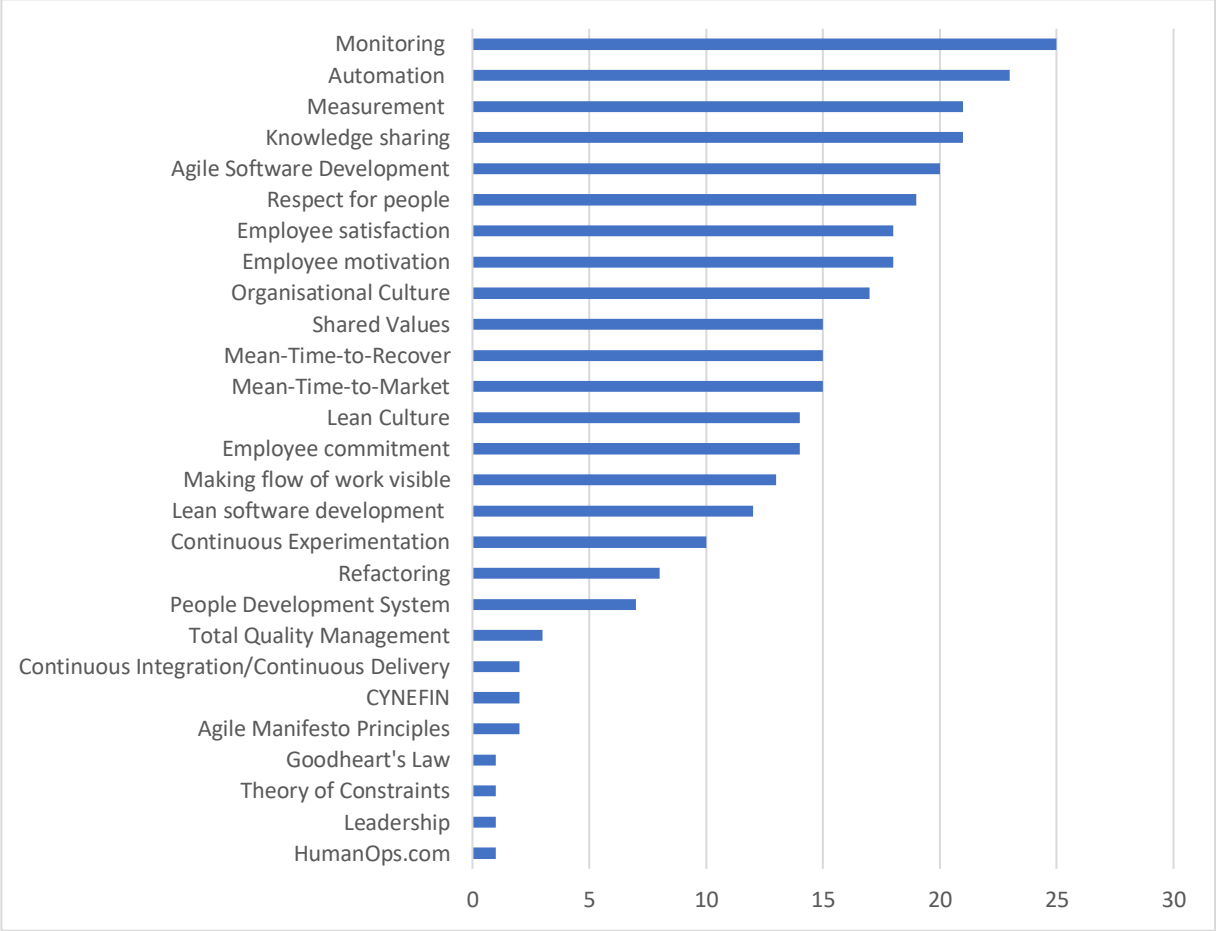


Figure 4-6 Competence of interviewee according to principles (interviewee count: 30).

Interview participants indicated that their most preferred agile, lean, and DevOps principles. These are organisational culture (60%), monitoring (53%), automation (47%%), measurement (47%), employee satisfaction (40%) and shared values (33%), see Figure 4-7. Evidently, results produced for the five most preferred principles, constitute part of CALMS (Culture-Automation-Lean-Measurement-Shared Values) model.

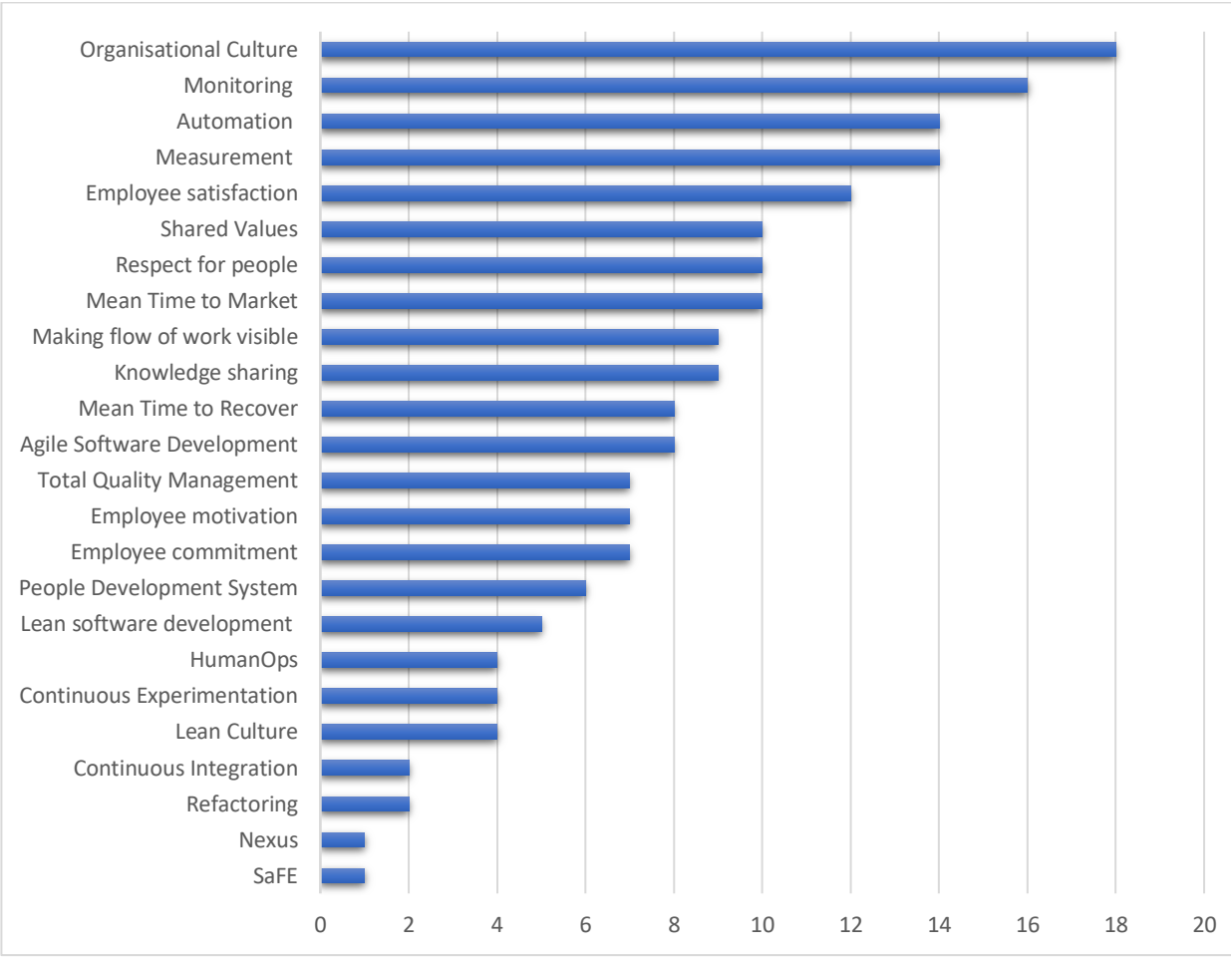


Figure 4-7 Most beneficial principles in order of preference (interviewee count: 30).

Considering structured IT service management frameworks, such as ITIL® and standards such as ISO20000, can agility and leanness, including DevOps practices and principles, become an extension to those approaches. Figure 4-8 indicates that two thirds of the interviewees agree that agility and leanness can form an extension to currently adopted structured frameworks, standards, methodologies, practices, and principles.

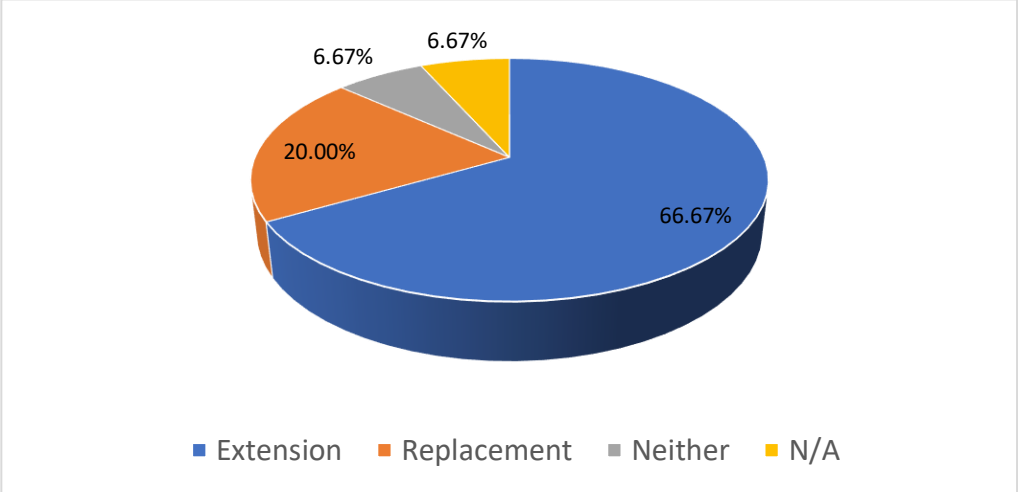


Figure 4-8 Responses on whether agility and leanness can become an extension or replacement to highly structured approaches to software development (interviewee count: 30).

Figure 4-9 indicates findings from the investigation of the that can lead to a significant need for a transformation of organisational capabilities. In fact, 80% of interviewees indicated poor communication and information flow as the main challenge in the transformation path towards DevOps. Moreover, around half of the interview participants indicated that a deep-seated company culture, operations not being involved in the requirements specifications, and a lack of knowledge sharing and transfer are key issues. The following serves as example and legend to the coded interview focus – see Figure 4-9.

14.1. Operations not being involved in the requirements specifications;
14.2. Poor communication and information flow;
14.3. Unsatisfactory test environment(s);
14.4. Lack of knowledge sharing and transfer;
14.5. Systems released to production before they are complete;
14.6. Operational routines not being established prior to deployment;
14.7. Deep-seated company culture;
14.8. Industry constraints and feasibility;
14.9. Heterogeneous DEV/Test/Production environments;
14.10. DevOps is unclear;
14.11. Industry constraints and feasibility;
14.12. A lot of operational responsibilities are not known-covered in development culture such as security, capacity management, incident response management.
14.13. Private Data Constraints (encrypted – masked data make test more difficult);
14.14. Dev and Ops responsibilities.

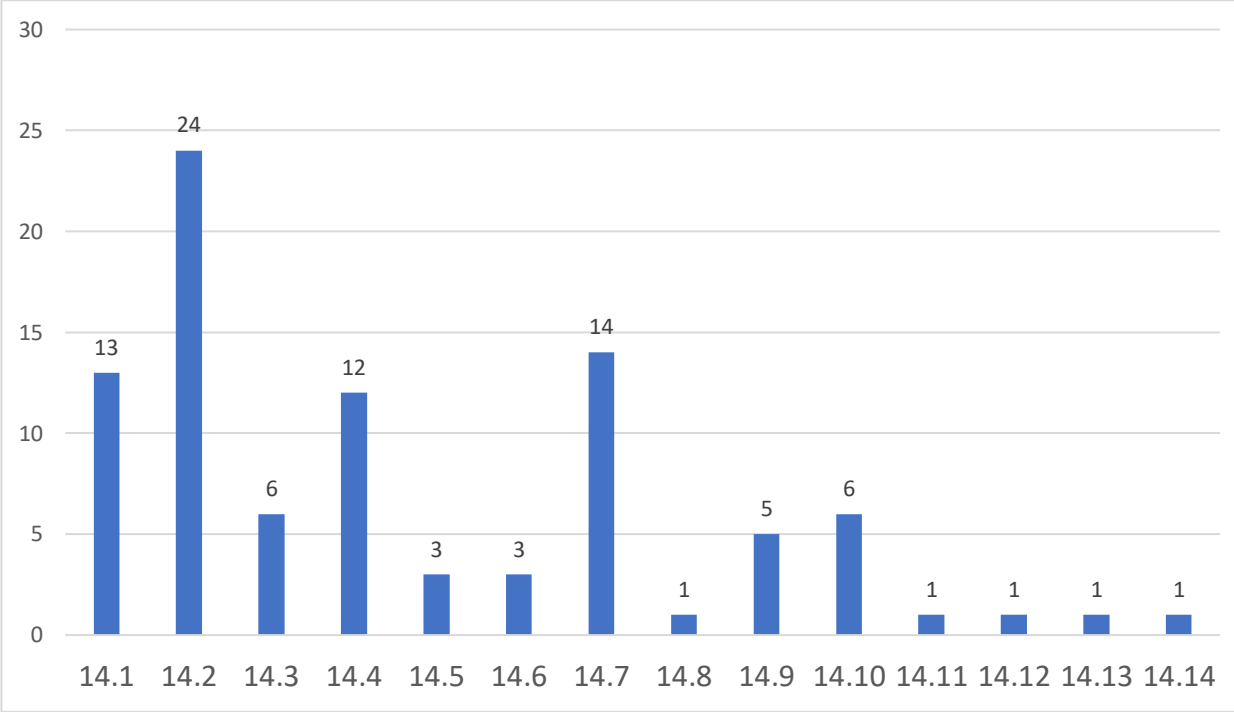


Figure 4-9 Areas to address in a DevOps transformation initiative (interviewee count: 30).

Interview participants indicated their preference as to 'what DevOps is' with highest preference concentrating on shift of mindset (57%), enhance collaboration and communication (57%), continuous deployment (37%), automated testing process (37%), frequent releases (33%), rapid feedback (33%) and improved service performance monitoring (33%) - see Figure 4-10. The findings are in agreement with a recent systematic literature review which also indicated that collaboration and communication is still regarded as the major challenge in DevOps adoption (Khan et al., 2022).

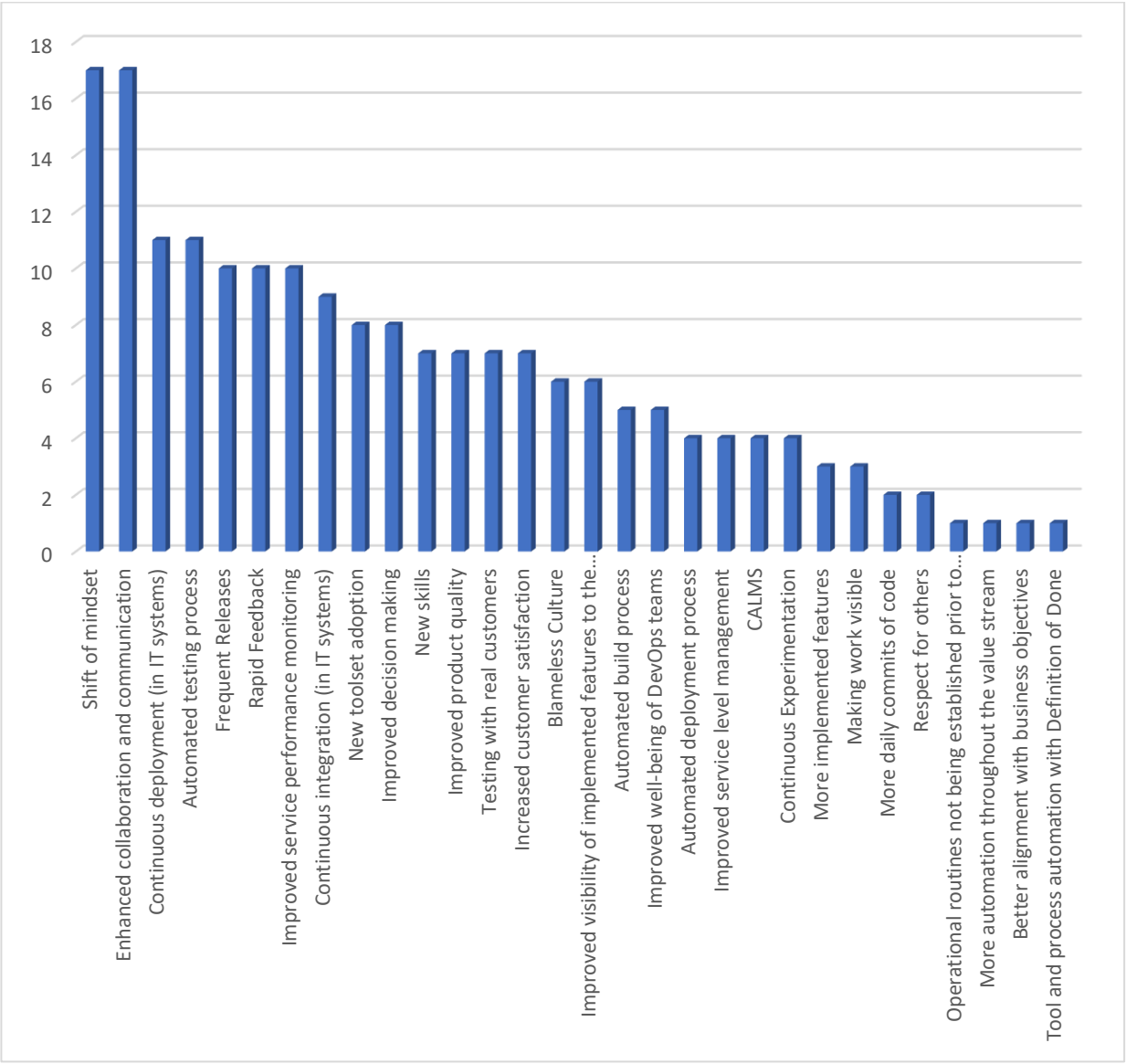


Figure 4-10 DevOps definition according to interviewees (interviewee count: 30).

Interview participants indicated their preference as to the skillset required for DevOps adoption leadership with highest preference concentrating on technical background (57%), negotiation skills (50%). communication and collaboration skills (47%), previous experience on transformation (47%), and holistic systems thinking (27%) - see Figure 4-11. In the context of the research questions results indicate that, the acquisition and adoption of new practices and principles such as agile, lean and DevOps are mostly considered as a 'shift-of-mindset' first and then as a shift of skillset (communication and collaboration) and ultimately a shift of toolset (continuous deployment, automated testing, etc.). The emergent theme of preference on practices and principles, skills and metrics is also supported by responses registered on Figure 4-7.

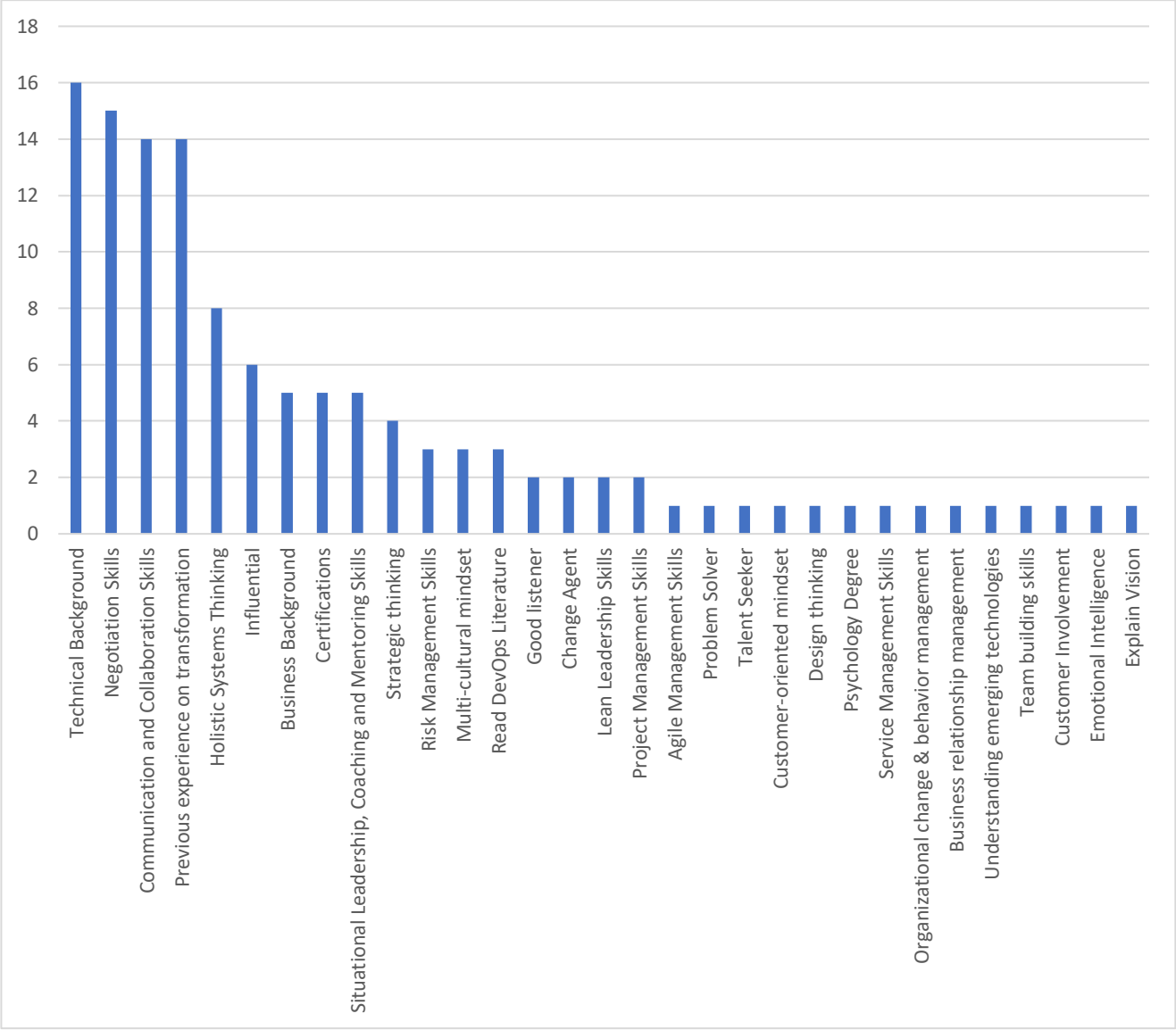


Figure 4-11 Skills and capabilities required for DevOps adoption leadership (interviewee count: 30).

Interview participants indicated their preference as to the organisational teams that should be part of the decision-making process of a DevOps adoption transitional journey and these include IT development teams (97%), IT operations teams (97%), quality assurance teams (93%), information security teams (80%), and the Board of Directors (73%) - see Figure 4-12. In the context of DevOps adoption leadership, the results obtained indicate a strong linkage to the preference of maintaining product-specific acumen (technical background), be able to navigate through corporate challenges and issue resolution (value negotiation) and communicate effectively (communication and collaboration skills).

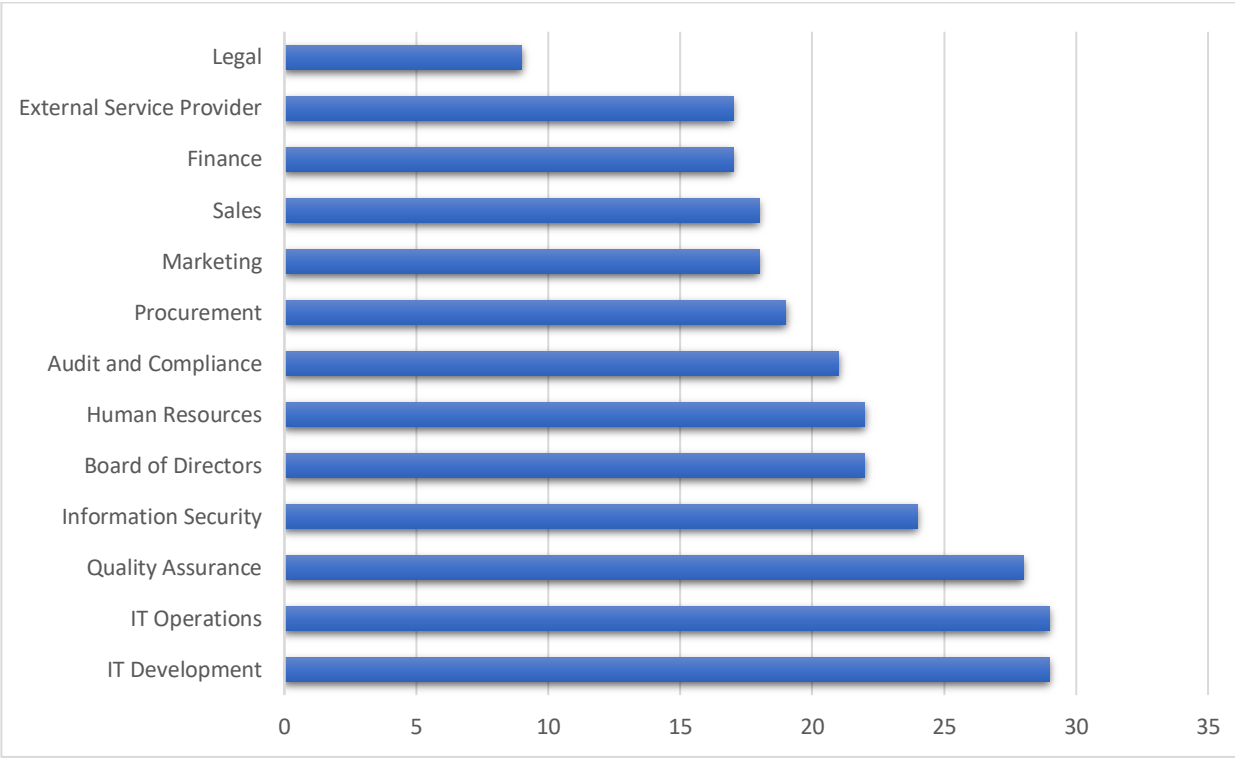


Figure 4-12 Organisational teams that should be part of a DevOps adoption transformational process (interviewee count: 30).

Having thoroughly accounted for descriptive statistical analysis, the baseline opinion registered through the interview manuscripts will also be analysed in section 4.4.3, i.e. to provide an extension to the synthesised aspects of the thirty (30) interviews conducted in the exploratory study.

4.4.3 Thematic Analysis

The recorded interviews were transcribed using Temi and reviewed word for word to establish the require consistency prior to importing and analysing the produced transcripts in NVivo 12. Temi is not a transcription analysis tool. It is only used for transposing audio into its textual equivalent. Figure 4-13 indicates the results of a recorded interview session from two online transcription services, i.e. Temi and NVivo. In fact, Temi, proved to analyse and produce better content, i.e. including timestamps for speaker's phrases and identifying explicitly numbering different speakers in the discussion. On the contrary, NVivo's functionality did not support the aforementioned two tasks.

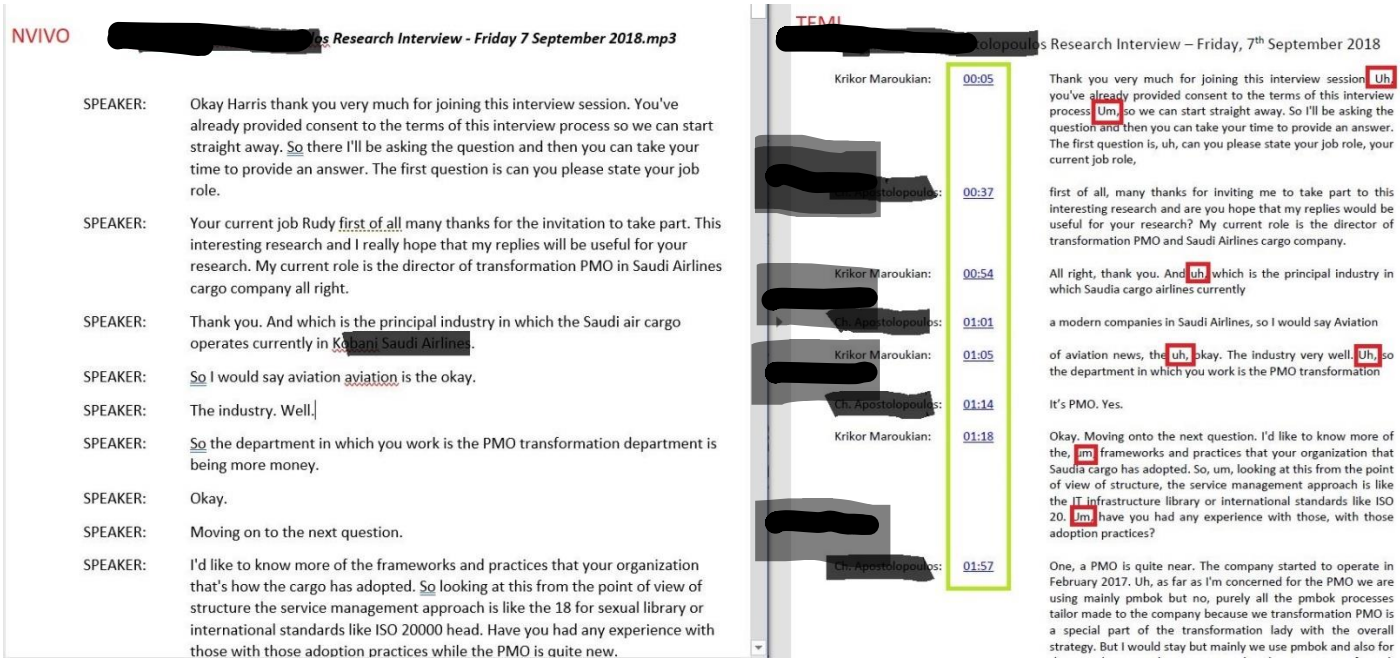


Figure 4-13 Comparison of online interview transcription results between Temi and NVivo.

The transcriptions produced were then imported into NVivo12 for thematic analysis whereby nodes and cases are assigned and mapped to the transcription of each interview. The coded themes per interviewee were generated and compared for discrepancies. The holistic approach which the interview transcriptions generated a coded thematic map is depicted in Figure 4-14. The thorough account of interview extracts is described in section 4.4.4.

Figure 4-14 Coded themes generated in NVivo12.

Most frequently used terms in the interview transcriptions (results qualified if 10 or more mentions) are: adoption; agile; approaches; change; culture; customer; development; DevOps; leadership; lean; level; management; practice; process; product; quality; role; service; service management; skills; software; software development; structured; team; transformation; and value.

Interview participants who identified themselves as *Consultants*, most frequently mentioned (results qualifying if term mentioned at least 15 times) during interviews: approaches; customer; development; lean; management; organisation; process; service; and team.

Interview participants who identified themselves as part of *Customer organisations*, most frequently mentioned (results qualifying if term mentioned at least 15) during interviews: development; management; service; software; and team.

4.4.4. Analysis of Interview Discussions

Throughout the series of interviews there was a focus on **DevOps adoption and the leadership role**. In fact, P5 (Managing Director, UK) and P19 (IT Operations Manager, Greece) stated that:

> Leadership skillset is the most important thing to adoption barrier breakdown.

P7 (Consultant, UAE) stated that:

> In the beginning of an adoption initiative there is a constant link to fear of people losing power, loss of position, etc. There is also a lack of leadership (walk-the-talk, lead-by-example, a need to confront undesirable behaviours, and a need to reward new behaviours).

In addition, P23 (CDO, Estonia), P28 (Business Development Director, Greece) and P30 (CIO, Greece) added that:

> End-to-end ownership of the leadership role is required in terms of cross-functional team leadership.

The prominence of **resistance to change in adopting new practices and principles** was repeated many times throughout the course of interviews with P27 (Georgia, IT Operations Lead) stating that:

> Any change can bring resistance and hinder adoption practices. Moving away from any already established approach generates resistance.

Moreover, P24 (Greece, CIO) added to that:

> Resistance happens because all the teams are getting out of their comfort zone. We are talking about different methodology, different structure, different KPIs, different roles, different rewarding scheme, different working location since the team is now collocated - everything is different.

Whereas P20 (UK, IT Operations Manager) states that:

> Change management is not generally well understood within organisations.

In terms of factors slowing down DevOps adoption, P15 (Principal Consultant, South Africa) mentioned that:

> Extremely hierarchical organisational structures pose as a communication barrier to DevOps adoption. Another failure point for DevOps adoption can be that DevOps practice adoption has to be at a wider enterprise scale for it to be labelled successful.

In addition, P27 (IT Operations Manager, Georgia) stated that:

> Top management is not interested in agile and DevOps practice adoption. They rely solely on customer satisfaction levels which can mean a reactive attitude towards the number of complaints received.

In disagreement P18 (Managing Director and Lead Consultant, UK) argued that:

> Rather than adopting every new framework, methodology, set of practices, organisations should look into identifying the current bottlenecks and improvement areas.

Notably, P3 (CIO, Greece) mentioned that:

> We identified the bottlenecks that we adopted while adopting these structured approaches.

However, P8 (Principal Consultant, UK) argued that:

> Senior management and team members should not blame the person who introduced the new practice since continuous experimentation is crucial to the success of DevOps adoption and any new practice adoption.

It is vital to establish the right organisational culture when it comes to the shift of mindset that DevOps adoption requires. To that extent P10 (Managing Director, UK) stated that:

> The team leading the adoption of the new way of working has to have the right skills and cultural drivers to succeed.

One of the interview questions focused on producing valuable results on preference of ITSM practice significance to value delivery of software development. The findings indicate a top preference for change management, release and deployment management, and Incident and problem management - see Table 4-3.

Table 4-3 ITSM process significance to value delivery of software development (interviewee count: 30).

IT Service Management Process	Adds Value to Software Development (%)
Change Management	24
Release and Deployment Management	15
Incident and Problem Management	10
Service Level Management	9
Availability Management	7

In addition, 66.67% of interviewees agree that agile and lean principle and practice adoption is an extension of established structured ITSM approaches - such as ITIL®. Only 20% stated that a complete replacement of those is required. However, concerns on ITIL adoption were mentioned by P6 (Greece, Smart Systems Manager):

> ITIL is only used for IT operations and too many roles and responsibilities are defined within ITIL, which means, that poor adoption, leads to increased confusion of the workforce adopting it.

In fact, the extension of principles and practices signals the transition an organisation has to pursue in order to achieve the desired adoption level. However, the top three challenges identified concerning DevOps practice and principle adoption journey were: Poor communication and information flow; Deep-seated company culture; and Operations not participating in the requirements specifications.

Additionally, a number of interviewees registered that blameful culture and time-consuming bureaucratic processes do not promote a sense of change in behaviour to adopt new practices and principles, but maintains a collective cultural complacency among IT teams. P7 (UAE, Senior Digital Transformation Technologist and Solution Practice Lead) mentioned that:

> A blame 'game' exists between IT teams which breads increased blameful culture, especially between Dev and Ops teams. By bringing these two teams together to code, test, deploy - the blame game stops. So now a blame-free culture starts to be promoted and gradually becomes evident as change emerges in behavioural patterns.

P11 (Greece, Head of Remote Transactions) adds that:

> Bureaucratic approach leads to informal ways of complete disregard of approval points. Senior management is keen to use this kind of approach to get things done quicker.

DevOps is highly regarded as a group of practices and principles that characterise collaborative culture (Luz et al., 2019), and these top three challenges (i.e. Poor communication and information flow, deep-seated company culture, and operations not participating in the requirements specifications) indicate a need to address the organisational culture perspective. According to answers from Question 4, 66% of participants are aware of DevOps and its associated practices and principles. Therefore, naturally the participants were asked to define DevOps. The four most popular phrases used were "a shift of mindset", "enhanced collaboration and communication", "continuous deployment", and "automated testing process".

The shift of mindset was pointing to established organisational cultural behaviours such as the one referred to by P3 (Greece, CIO):

> There is a mindset to "never outshine the master".

P11 (Greece, Head of Remote Transactions) mentioned that:

> The 'email culture' on which business units heavily rely is detrimental to DevOps adoption aspirations.

To that extent P18 (UK, Managing Director, and Lead Consultant) mentioned that:

> Culture is a very wide term. So if the incentives are in conflict with team expectations than there is going to be a situation of complaining about tool usage. Enterprise-wide incentives alignment is strongly required under such circumstances.

Moreover, 53% believe that the DevOps leader role should be an individual professional, whereas 33% would trust the role to a team. People suggested that it was best to have an individual lead DevOps adoption, and organisational transformation efforts initially, but that subsequent transition to a team effort was also good. Note that the adoption efforts should be continuous in nature, and not be conducted in a project-based manner as temporary endeavour. In this context P18 (UK, Managing Director, and Lead Consultant) stated that:

> DevOps adoption practices and principles should not be viewed as a project under the context of a transformation with a beginning and an end rather a continuous aspiration for improvement of the current state of adopted practices and principles.

In addition, P8 (UK, Principal Consultant, IT Service Management) added that a common pitfall is that:

> Overestimation of DevOps practice adoption is common.

P10 (UK, Managing Director) mentioned one area that requires particular attention:

> Uneven experience gives birth to assumptions. For instance, if not everyone in the same team has the same level of knowledge and understanding on ITIL then different people would assume different definition for IT service management. HR (Human Resources) plays a big role in recruiting people with uneven skills. This is an unrecognised cost to the IT organisation.

Furthermore, P21 (UK, Founding Consultant, IT Service Management) stated that:

> The transformation of Waterfall-to-Agile-to-DevOps in an IT organisation has to be an enterprise-wide endeavour. The missing link is HR not being on the same page with the efforts to change towards agility.

P1 (Saudi Arabia, PMO Director) added that:

The human resources department is an enabler leading the change.

Whereas P14 (Greece, Head of Applications) commented that:

Lack of continuous commitment to DevOps adoption by organisation-internal IT customers inhibits the adoption itself.

The leadership skills that were mentioned by 50% of interview participants included: 1) technical background; 2) negotiation skills; 3) communication and collaboration skills; and 4) previous experience on transformation. Holistic systems thinking was mentioned by 27% of interviewees. Business background by 17%. Strategic thinking by 13%. Furthermore, there was a lot of iteration around the influential skills, holistic systems thinking, a multi-cultural mindset, and increased awareness around dealing with suboptimal productivity.

When considering DevOps leadership objectives, a remarkable 87% of interview participants agreed that DevOps practice adoption should be extended in an enterprise-wide fashion, and should include external service providers in its scope. This result strongly links to research question RQ2 *, i.e., "Are DevOps-oriented environments and their teams an extension or a replacement of process-driven structured service management approaches?"*.

To overcome DevOps adoption inhibitors P19 (Greece, IT Operations Manager) stated that:

Leadership skillset is the most important thing to adoption barrier breakdown.

In addition, P23 (Estonia, CDO) added that:

A cross-functional leadership role with end-to-end ownership of DevOps adoption is imperative.

Additionally, the interview series responses identified that organisational teams that should be part of a DevOps practice adoption journey are as follows with corresponding approval rates:

- IT Development (97%);
- IT Operations (97%);
- Quality Assurance (93%);
- Information Security (80%); and
- Board of Directors (73%).

The data collected from a series of interviews and participating practitioners, indicate a clear list of specific agile, lean, and DevOps practices and principles that regarded an extension to structured service management approaches and are relevant to DevOps adoption theory. The main findings associated to the research questions are that:

1. Specific agile, lean, and DevOps practices such as 1) **organisational culture**, 2) **monitoring / measurement**, 3) **automation** are crucial in the software development lifecycle (RQ1).
2. Specific agile, lean, and DevOps principles such as 1) **SCRUM** 2), **Kanban** 3) **Continuous Delivery** are crucial in the software development lifecycle (RQ1).
3. The set of service management processes that continue to form a strong part of DevOps-oriented structures are **Change Management**, **Service Portfolio Management (including Service Catalogue Management)**, **Release and Deployment Management,** and **Service Level Management** (RQ2).
4. There is overwhelming consensus that a **DevOps leadership role should exist** (86%) and that the role should carry a continuous effect not a project-based (RQ3).
5. DevOps practices and principles adoption are challenged due to **poor communication and information flow**, **deep-seated company culture**, and **operations not being involved in the requirements specifications** (RQ3).
6. DevOps practice adoption should be extended in an enterprise-wide fashion (87%), with team structure based on existing Development (97%), Operations (97%), Quality Assurance (93%), and Information Security (80%) (RQ3).

4.6 Chapter Summary

This chapter considers the practical steps to the first empirical research approach - framed, described, and associated, earlier in Chapter Three - in the context of an exploratory study to evaluate the current state of research in structured service management frameworks, methodologies including agile and lean practices through a series of thirty interviews carried out with practitioners from software-intensive organisations. A transcript analysis of the interviews transposed to thematic analysis is also presented produced from NVivo. In the next chapter, the second empirical study is presented in terms of quantitative research, analysis, and evaluation of results.

Chapter Five

Quantitative Research Analysis and Outcomes

5.1 Chapter Overview

This chapter aims to continue building on top of the results, evaluation, and findings from Chapter Four. The chapter considers the design and development of a survey questionnaire considering the analysis and evaluation that preceded it in Chapter Four. Following the exploratory study, this chapter will provide deeper insights into the types of software product development practices and principles that are relevant to industry practitioner experiences in the context of DevOps adoption and its leadership traits, style, measurements, and required skillset. The research design and method selected was part of the orchestration of empirical studies thoroughly accounted for in section 3.3. Additionally, the outcomes of the analysis of the data collection process will form the baseline of a leadership model in DevOps adoption, which is subsequently evaluated under the lens of structural equation modelling (SEM); more specifically partial least squares (PLS).

5.2 Introduction

Several DevOps practices and principles were deduced from practitioner interviews (see chapter 4); i.e. software development product team practices and principles that establish the right mindset, behaviours, skillsets, and desired target metrics. The notion of adopting the right practices and principles in the software product development lifecycle, has been coupled with the leadership style, which was introduced in team efforts through a transition process to DevOps, to improve the velocity with which a team delivers on the customer promise.

5.3 Survey Data Collection Process

The main efforts to achieve a dataset from a rich diversity of participants required a specific approach in terms of the utilisation of different communication and survey announcement channels. Participants were recruited using two approaches: 1) through direct contact at ITSM / DevOps events in Europe, and 2) via a general call for participation posted on professional social media networks - including LinkedIn and IT societies, such as i) IT Service Management Forum (itSMF), ii) Microsoft Worldwide Application Lifecycle

Management and DevOps community, and iii) Microsoft Worldwide Modern Service Management community. Moreover, to achieve a heterogeneous perspective, and to increase the wealth of information, practitioners from a variety of organisations were invited to participate. All of the survey participants provided consent to the Terms of Survey Participation see Appendix E and F.

The participants of the survey (n=250) answered four sections, see below, which were structured based on the findings of exploratory conducted interviews (see chapter 4):

Section 1. Professional experience related demographics (n=250);
Section 2. Questions about the participant's experience with agile, Lean IT and DevOps (n=250);
Section 3. DevOps practices and principles adopted (n=250);
Section 4. Leadership style related to DevOps, including its skillset and metrics (n=169).

The target audience of the survey is defined mainly as Consultant, Product/Software Developer, C-Suite, Operations engineer, IT Architect. Data collection and analysis was mapped to the research questions posed at the end of the Introduction section for survey questions. The online survey collected a rich dataset on a sequence of twenty questions to be posed. The first six questions are focused on collecting demographic data including: participant age; years of professional experience; job role; industry; associated region; and size of the organisation. There are also questions to gain an understanding around DevOps practice adoption, skillset, metrics, and leadership related to DevOps. The target audience of the survey is defined mainly as Consultants, Product/Software Developers, C-Suite, Operations engineers, and IT Architects.

For most of the questions, a 4-point Likert scale was chosen to record opinion for the set of survey questions in an effort to add clarity to the distribution of positive or negative opinion.

From the 250 survey participants, 81% have held 10+ years of professional experience, and 78% have held a leadership position. Furthermore, the survey participants indicated by 67% that practice adoption leadership role should exist for the transitional path to DevOps adoption.

Data collection and analysis was aggregated in order to answer the research questions, which were mapped to survey questions - see Table 5-1. The mapping represents the segmentation of research undertaken for each particular research question (RQ) entailed in this thesis. Effectively, the obtained results per question could be categorised so that an aggregate dataset is produced from responses received which can be analysed per research question.

Table 5-1 Research Questions to Survey Questions Mapping.

Research Question	**Interview Question**
Data collection for segmentation purposes	1, 2, 3, 4, 5, 6,
RQ1) Which agile, lean and DevOps practices and principles can improve productivity for software product development teams in software-intensive organisations that have adopted a structured service management approach?	8, 9, 10, 12, 13, 14, 20
RQ2) Are DevOps-oriented environments and their teams an extension or a replacement of process-driven structured service management approaches?	20
RQ3) Can Leadership affect DevOps adoption within a software-intensive organisation? What is the leadership style that can be attributed to the DevOps adoption leader role?	7, 11, 20 15, 16, 17, 18, 19, 20

The whole list of survey questions is available in Appendix D.

5.4. Analysis and Evaluation

5.4.1. Survey Background Segmentation

A segmentation of the analysis of participant roles includes the following levels of participation of Consultant (59), Product/Software Development (45), C-Suite (36), Operations (28), PMO (26), IT Architect (21), Business Development (13), Information Security (9), Director (3), Head of Infrastructure (2), Service Management (2), Head of Legal Department (2), Support (1), IT Manager (1), Database administration (1), Deputy Manager IT (1). Moreover, the industries in which the survey participants worked in are IT Services / Consulting (83), Government (55), Financial Services (32), Technology/Telecommunications (19), Manufacturing (11), Financial Services/Consulting (8), Aviation (7), Construction (7), Retail/Consumer Services (6), Healthcare (5), Education (5), Recycling (3), Insurance (3), Energy/Utilities (3), Leisure & Hospitality (3). The aforementioned representation of industry practitioner roles translates to Consultants (23.6%), Product/Software Development (18%), C-Suite (14.4%), Operations (11.2%), PMO (10.4%), forming the top 5 segments of participation to the online survey.

The results obtained from the survey participants shed more light on our first research question (RQ1). Firstly, the list of most preferred practices were (1) version control, (2) trunk-based development, (3) issue tracking, (4) automated provisioning, (5) value stream map, (6) Scrum, (7) test automation, (8) static code

analysis, (9) Kanban, (10) automated deployment, (11) code coverage, (12) continuous integration, (13) performance monitoring, (14) containerised environments, (15) continuous delivery, (16) Infrastructure-as-Code.

Furthermore, the skills identified are (1) communication and collaboration, (2) active listening, (3) customer-centric mindset, (4) technical background, (5) problem solving, (6) multi-cultural mindset, (7) influential, (8) agile management, (9) strategic thinking, (10) project management, (11) business background, (12) talent seeker, (13) negotiation, (14) change agent, (15) risk management, (16) previous experience on transformation, (17) design thinking, (18) certifications, (19) leadership, and (20) holistic systems thinking.

Moreover, the list of identified metrics regards (1) time to market, (2) critical success factors, (3) key performance indicators (KPI), (4) deployment frequency, (5) deployment duration (time), (6) time to detect (defect), (7) time to recovery, (8) behavioural Metrics, (9) feature usage, (10) deployment size, (11) knowledge article creation frequency, (12) releases frequency per developer per day, (13) knowledge article read frequency, (14) lead time between code commit and code deploy, and (15) % of revenue impacted. The DevOps adoption leader roles was recognised in numerous roles namely, (1) C-Level executive, (2) Product Owner, (3) Architect, (4) System/Network/Database Administrator, (5) Development Lead, (6) Business Representative, (7) Operations Lead, (8) DevOps Engineer, (9) Team Leader, (10) Analyst, (11) Executive Committee, and (12) Developer.

5.4.2. Confirming Exploratory Study Findings

DevOps Adoption of Practices and Principles

In the survey of 250 participants, 25% of respondents do not plan to adopt DevOps in the future, 30% adopted across some parts of the IT organisation, 18% adopted across the IT organisation, 12% adopted across the enterprise, and 5% have not adopted nor have plans to adopt DevOps (see figure 5.1). In addition, the role in decision making process for DevOps adoption falls from 34% for C-level executives (i.e. members of the Board of Directors), to 21% for development team leads, to 16% for product owner, and to 10% for architects. The high concentration of responses to C-level executive and development team leads suggests that the development teams themselves, must shift from a highly hierarchical organisational structure to more autonomous self-organising team behaviours which characterise DevOps teams.

Lack of commitment by customer is recognised as the top inhibitor and resistance factor of DevOps adoption, followed by a lack of organisational practice adoption capability. These results represent the overall opinion expressed during the interviews, and indicate that there is overwhelming agreement on these type of factors slowing down DevOps adoption.

In the survey of 250 participants, there were certain close-ended questions that aimed to unravel the stage what DevOps adoption of practices and principles are presently used. Findings indicated that 5% of participants have no plans to adopt DevOps, 35% of have future plans for DevOps adoption, for 30% for participants DevOps is adopted across some parts of the IT organisation, for 18% of participants DevOps is adopted across the IT organisation, and finally 12% of participants DevOps is adopted across the entire enterprise. Figure 5.1 depicts the DevOps adoption stage expressed by survey participants per industry vertical. Notably, the Government and IT Services/Consulting industries have been embracing DevOps planning across parts or whole IT organisation whereas in terms of the Financial Services there is a two-state approach i.e. plan to adopt in the future or adopted across some parts of the IT organisation. This gap indicates that there could be a chasm in the support of some DevOps adoption initiatives with lower priority i.e. planned for the future as opposed to an already established DevOps transitional journey.

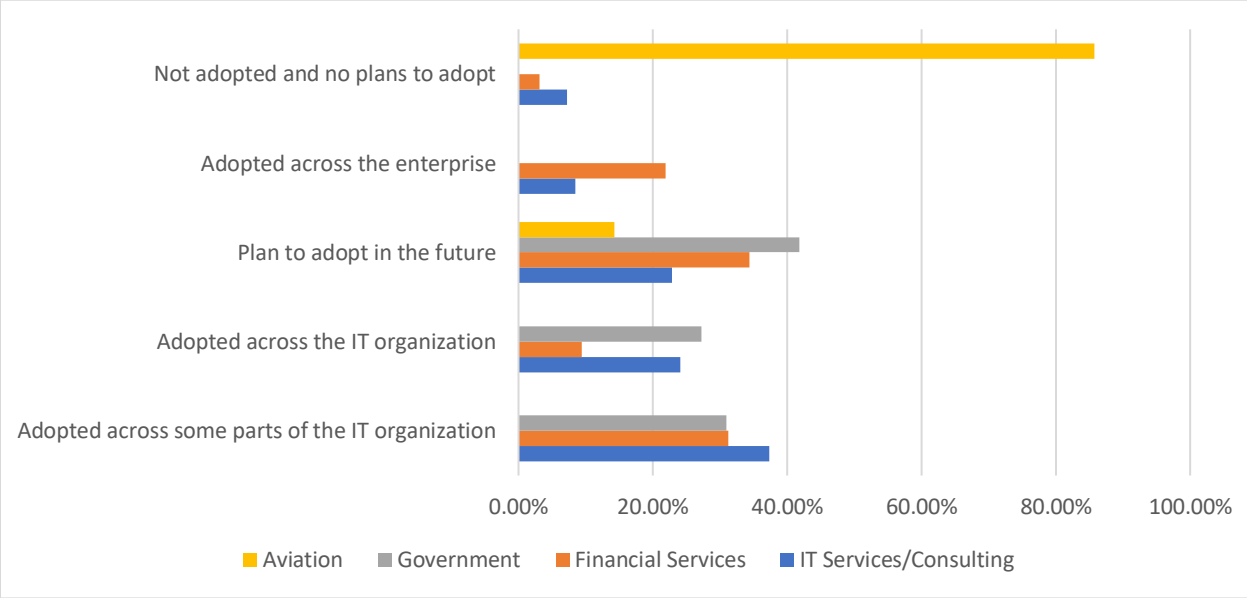


Figure 5-1 DevOps adoption stage of survey participants (n=250).

In addition, the roles responsible for the decision-making process in DevOps adoption initiatives are shown in Table 5-2. Notably, Information Security and DevOps Engineer roles were perceived as being two of the least contributing roles in the decision-making process steps in the DevOps adoption process. In addition, there seems to be low involvement of the business domain in DevOps adoption initiative. On the other hand, the high concentration of responses to C-level executive (Chief Information Officer, Chief Digital Officer, etc.) and development team lead could suggest that the development teams themselves have to shift from a highly hierarchical organisational structure to more

autonomous self-organising team behaviours, which characterise DevOps teams.

Table 5-2 Decision-making role in DevOps adoption process.

Role responsible for decision making in DevOps adoption process	Participant Preference (%)
C-Level (Chief Information Officer, Chief Digital Officer, etc.)	33.6
Development Lead	20.8
Product Owner	16
Architect	10.4
Operations Lead	6
Business Domain	3.6
DevOps Engineer	3.2
Developer	3.2
System/Network/Database Administrator	1.2
Executive Committee	0.8
Team Leader	0.4
Analyst	0.4
Not Sure	0.4

A lack of commitment by customers is recognised as the top inhibitor and resistance factor of DevOps adoption followed by a lack of organisational practice adoption capability. A 4-point Likert scale was chosen for this question to record opinions. These results are similar to the opinions expressed during the interviews and indicated that there is overwhelming agreement on these types of inhibitors to DevOps adoption. Having identified the set of most frequently adopted DevOps practice and principles, the next section attempts to provide clarity on DevOps adoption leadership.

DevOps Adoption Leadership Metrics

The interview series revealed that version control and issue tracking have been adopted widely by the respondents i.e., used by 95% of participants. Additionally, performance monitoring, test automation, and automated deployment seem to have strong penetration in software product development practices. Infrastructure-as-Code, code coverage, static code analysis, trunk-based development, automated provisioning of IT resources, and containerised environments, are still areas that have not been widely adopted by survey participants.

The main aim of this survey section is to uncover more around the metrics related to DevOps adoption and the leadership role. DevOps adoption practices and principles adoption levels can be measured with the

traditional approach of critical success factors (65%) and Key Performance Indicators (63%). However, DevOps oriented metrics also gained high agreement such as mean-time-to-market (75%), deployment frequency (58%), deployment duration (53%), behavioural metrics (52%), time to detect defect (52%), and mean-time-to-recover (50%). Feature usage (41%) seems to be an emerging practice for DevOps adoption. Moreover, 91% of respondents agreed that the leadership role should be associated and have ownership of the aforementioned metrics in order to facilitate the DevOps teams efforts in the adoption of practices and principles. Lastly, regarding the software development-oriented metrics there was negligible mention in the interviews and the survey.

The survey also showed that 76% of participants have held or hold a leadership position, and 91% claimed i) that the DevOps leadership role is required, and ii) that it should be an individual role (67%). These results are similar to the results produced from the thirty (30) interview participants. It is worth looking into the level of acceptance of a leadership role being an individual or team role and the influential effect it can have on team performance in the context of software product development and coding pipeline health. Nine (9) service providers and six (6) consultants agreed that the leadership role should be an individual role whereas five (5) service providers and five (5) consultants agreed that the leadership role should be a team role. Lastly, one (1) service provider and three (3) consultants stated that both approaches are required interchangeably throughout the course of a transitioning initiative towards DevOps practice and principle adoption. The main aim of this survey section was to uncover more around the metrics related to DevOps adoption and its leadership role.

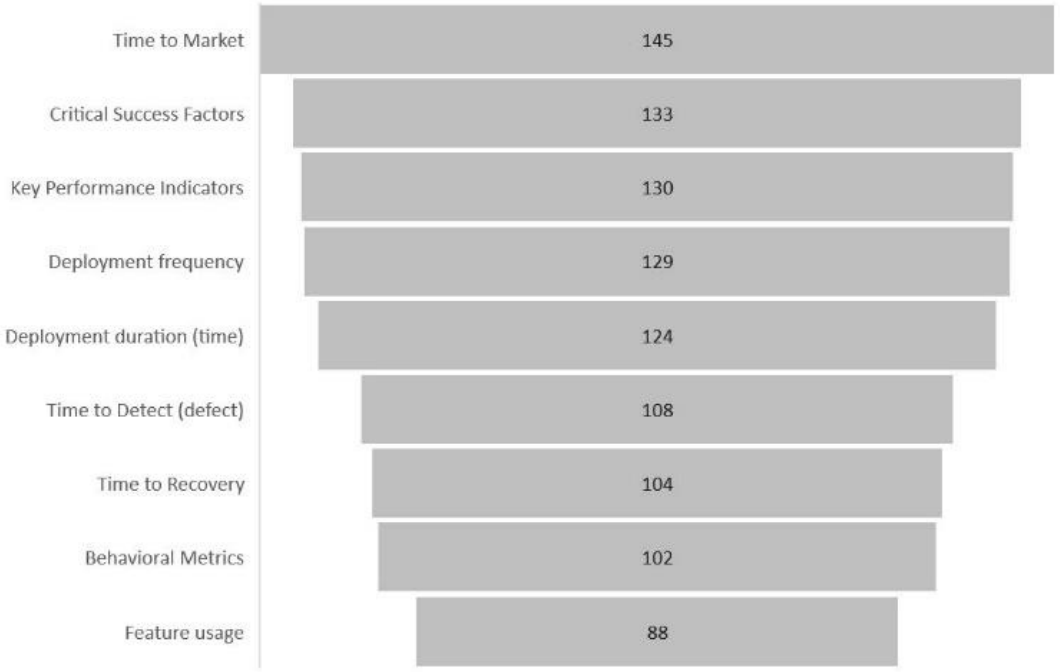


Figure 5-2 DevOps adoption metrics indicated by survey participants.

The traditional approach to measuring adoption in software development is presented in Figure 5-2; with results putting emphasis on time to market, Key Performance Indicators (KPI) and Critical Success Factors (CSF). The most prominent DevOps oriented metrics were deployment frequency, deployment duration, time to detect a defect, time to recovery and behavioural metrics. Feature usage seems to be an emerging practice for DevOps adoption. In fact, 91% of respondents agreed that the leadership role should be associated and have ownership of the metrics shown in Figure 5-2, to facilitate the DevOps teams efforts in the adoption of practices and principles.

Link between Transformational and Servant leadership in DevOps

The unison of DevOps adoption, its leadership, resistance factors and the way DevOps adoption leadership can be measured, provide insights to the leadership style that they resemble. Gaining an improved understanding of DevOps adoption leadership first requires mapping characteristics of servant and transformational leadership styles to DevOps adoption leadership, see Table 5-3. The servant leader commits time and effort to understand each follower's background, core values, beliefs, and behavioural patterns not only in the professional but also in the personal domain (Eva et al. 2019). Additionally, charismatic, and transformational leaders attempt to communicate their leadership qualifications by appealing to follower values, communicating in symbolic ways that are clear and vivid, and displaying emotional conviction and passion for the mission (Yukl, 2020). Moreover, transformational leaders inspire and transform followers by making them more aware of task outcome importance, motivated to expose their motivation to the benefits of the team or organisation, and activated to work towards higher-order needs (Yukl, 2020). Furthermore, the primary focus in the value-based leadership styles is that a leader who has power should use that power wisely and ethically. The distinction of ethical consideration is the fundamental difference between transactional leadership and value-based leadership styles. Value based leadership is based on a relationship of value and emotions exchanged for a specific set of benefits e.g. financial gains, increased influence over subordinates, increased authoritative right (Yukl, 2020). There are three periods through which servant leadership has been progressing. Firstly, the period that focused on the conceptual development of servant leadership. Secondly, the period that focused on producing the measures of servant leadership. We are currently living the third period, which regards the model development phase. Table 5.3. shows the DevOps adoption leadership characteristics identified during both; the thirty (30) interviews and the 250-participant survey. The practical contribution of the thesis is highlighted on the column of DevOps Adoption Leadership in Table 5-3.

Table 5-3 Characteristics of Leadership Styles.

Transformational Leadership (Greenleaf, 2002; Bass and Riggio, 2006)	Servant Leadership (Greenleaf, 2002)	DevOps Adoption Leadership (Maroukian and Gulliver, 2020)
• Idealised influence (realistically self-confident, determined, unconventional) • Inspirational motivation (articulate, flexible, emotional, perspicacious) • Individualised consideration (caring, empathetic, relations-oriented) • Intellectual stimulation (rational, unconventional, perspicacious)	1. Empathy 2. Active Listening 3. Emotional Healing 4. Awareness 5. Persuasion 6. Conceptualisation 7. Foresight 8. Stewardship 9. Commitment to the growth of people 10. Building community	1. Communication and collaboration 2. Active Listening 3. Customer-centric mindset 4. Technical background 5. Problem solving 6. Multi-cultural mindset 7. Influential 8. Agile management 9. Strategic thinking 10. Project management skills

Lastly, similar to the general leadership field, servant leadership, transformational leadership and DevOps adoption leadership are focused on the leader-follower dyad. The dyadic relationship in all three leadership approaches can give birth to opportunities to non-traditional facets of the relationship. Additionally, another common denominator of DevOps, transformational and servant leadership styles, is the form of influence they have, based, not on tradition or formal authority but rather on follower perceptions that the leader is endowed with exceptional qualities.

5.4.3 Design of a Conceptual Model of DevOps Adoption Leadership

The quantitative analysis conducted in section 5.6 validates the conceptual model (depicted in Figure 5-3) and more specifically the model constructs of Practices and Principles, DevOps Adoption Metrics, DevOps Adoption Skillset - which support the control construct of the model - and DevOps Adoption Leader Role.

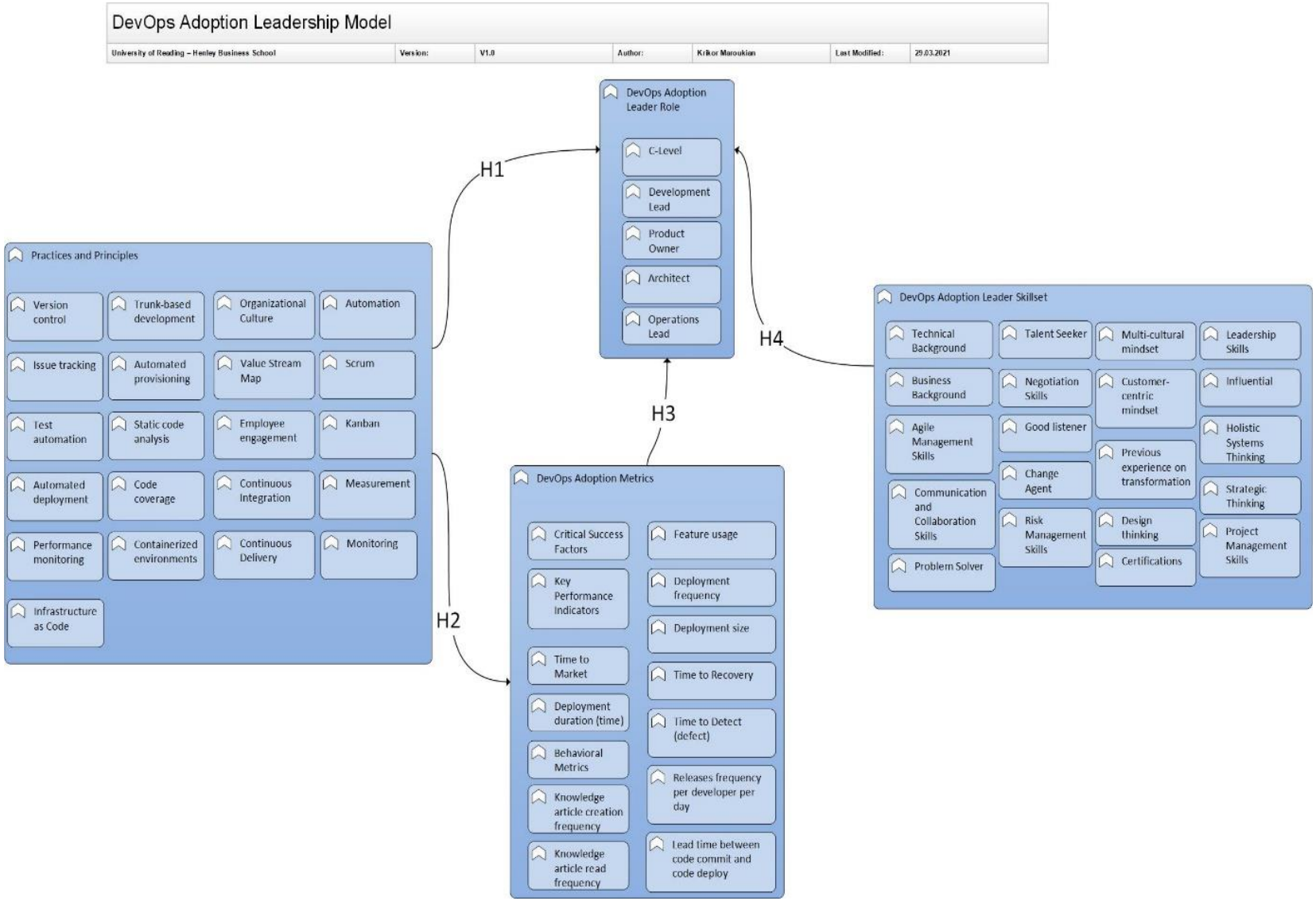


Figure 5-3 Conceptual Model for DevOps Adoption Leadership.

DevOps Practice and Principles Model Construct

There is a specific set of practices and principles included in the DevOps practice and principle model construct in Figure 5-3 and these regard both; responses from the exploratory qualitative study performed with thirty participants as well as the online asynchronous survey which collected the registered opinion of 250 individuals. Practices are defined under nine knowledge areas, which are: 1) software engineering management, 2) software construction, 3) software configuration management, 4) software testing, 5) software process, 6) software quality, 7) software engineering tools and methods, 8) software requirements, and 9) software design (Stahl et al. 2017). Additionally, continuous software engineering includes pertinent references to the practices (Fitzgerald et al., 2015).

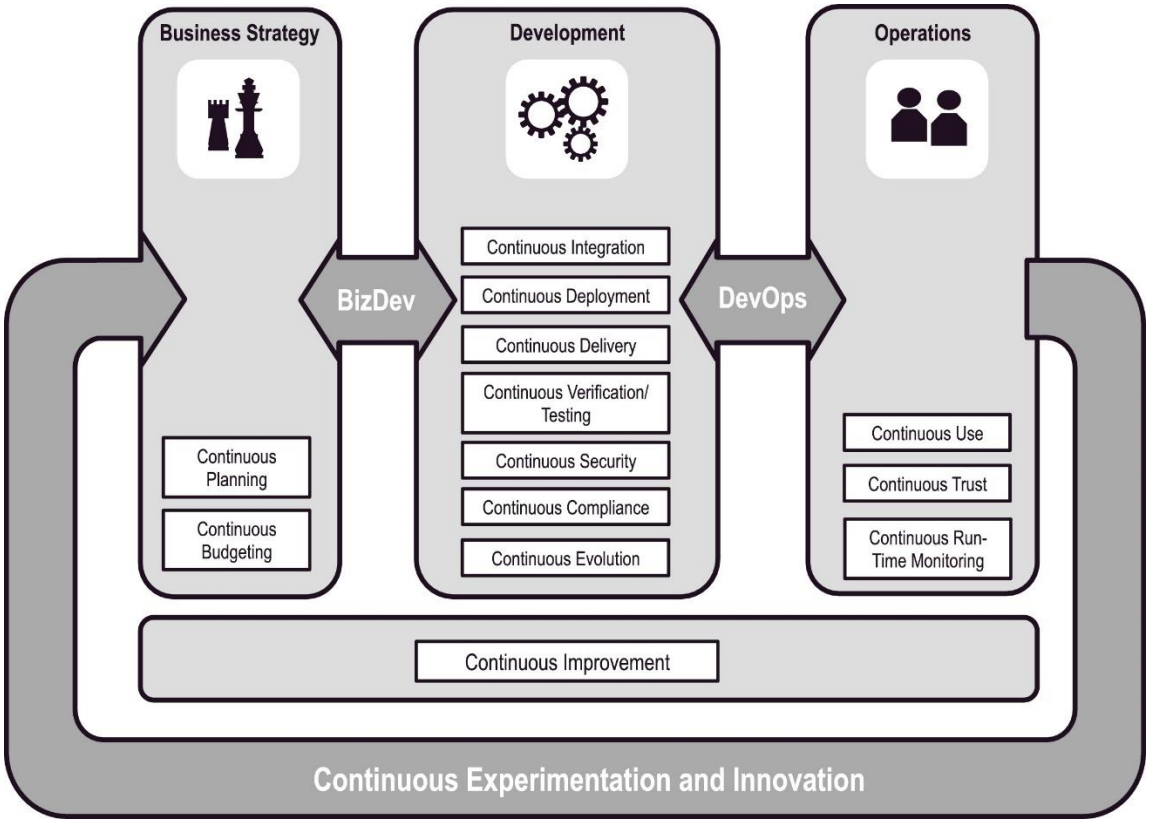


Figure 5-4 Holistic view of continuous practices in business, development, operations, and innovation.

The mixed method studies highlighted preference of industry practitioners on a selected set of agile, lean and DevOps practices and principles, skills and metrics. The emergent theme described in Chapter Four regarding the trinity of practices and principles, skills and metrics, see section 4.4.2. The grouping of emergent practices included: 1) version control, 2) trunk-based development, 3) issue tracking, 4) Value Stream Map, 5) Scrum, 6) static code analysis, 7) Kanban, 8) continuous integration, 9) containerised environments, 10) continuous delivery, 11) monitoring, 12) Infrastructure-as-Code.

The grouping of emergent principles included: 1) organisational culture, 2) automation, 3) automated provisioning, 4) test automation, 5) employee engagement, 6) automated deployment, 7) code coverage, 8) measurement, 9) performance monitoring.

DevOps Adoption Metrics Model Construct

There is a specific set of metrics that have been included in Figure 5-3 and these take into account the exploratory study (thirty interviews), confirmatory study (online survey n=250), the DevOps Research and Assessment (DORA) metrics (Puppet, 2019) which relate to deployment frequency, the Culture-Automation-Monitoring-Measurement metrics - as described in Lwakatare et al. (2016).

The grouping of included metrics included: 1) key performance indicator, 2) critical success factor, 3) time-to-market (aka lead-time-for-changes), 4) deployment frequency (DORA metric), 5) deployment duration, 6) deployment size, 7) knowledge article creation frequency, 8) knowledge article read frequency, 9) feature usage, 10) behavioural metrics, 11) time-to-recovery (aka mean-time-to-recover - MTTR), 12) releases frequency per developer per day, 13) lead time between code commit-to-code-deploy.

DevOps Adoption Leader Skillset Model Construct

The DevOps adoption leader metrics model construct shown in Figure 5-3 have taken into account the exploratory study (thirty interviews), confirmatory study (online survey n=250) and the DevOps Agile Skills Association (DASA) skill areas (DASA, 2022). These skill areas are as follows:

- Courage: Evangelism, coaching, self-confidence, proactivity, reflection, trust, open discussions, experimentation, fail fast, courage to change.
- Teambuilding: Understand the other's point of view, collaboration, mutual accountability, common purpose, ability to integrally support the service/product.
- DevOps Leadership: Facilitating teams to high performance, humility, transparency, Service lifecycle mindset, Stakeholder management.
- Continuous improvement: Today we do our work better than yesterday, kaizen mindset, quality at the source, first time right, knowledge-sharing, ability to adapt.

The grouping of included skills included: 1) technical background, 2) business background, 3) agile management, 4) communication and collaboration, 5) problem solver, 6) talent seeker, 7) negotiation skills, 8) good listener, 9) change agent, 10) risk management, 11) multi-cultural mindset, 12) customer-centric mindset, 13) previous experience on transformation, 14) design thinking, 15) certifications, 16), leadership skills, 17) influential, 18) holistic system thinking, 19) strategic thinking and 20) project management.

DevOps Adoption Leader Role Model Construct

The DevOps Adoption Leader Role model contract shown in Figure 5-3 is constituted by the variables of DevOps Adoption Leader Role Should Exist and DevOps Adoption Leader Role which maybe: 1) C-level, 2) development lead, 3) product owner, 4) architect, 5) operations lead.

5.4.4 Conceptual Model Hypotheses

The conceptual model in Figure 5-3 presents the sequence of presented model constructs - *DevOps Practice and Principles, DevOps Adoption Metrics, DevOps Adoption Leader Skillset* and *DevOps Adoption Leader Role*. It also includes the flow of hypotheses that have been developed based on previous literature, the interview series (exploratory study) and the definitive dataset of the online survey (confirmatory study). The specific hypotheses of the study are established as follows:

H1. *DevOps Practice and Principle* adoption *positively* affects *DevOps Adoption Leader Role*.
The reasoning behind the investigation and exploration of whether such a hypotheses can be validated, is purely based on whether there is a link between specific practices and principles that should be taken into account by the DevOps adoption leader role (individual or team role) in the transitional journey towards DevOps in an organisational setting.

H2. *DevOps Adoption Metrics* adoption *positively* affects *DevOps Adoption Leader Role*.
The reasoning behind the investigation and exploration of whether such a hypotheses can be validated, is purely based on whether there is a link between which specific adoption measurements and the DevOps adoption leader role (individual or team role) for the transitional journey towards DevOps in an organisational setting.

H3. *DevOps Practice and Principle* adoption *positively* affects *DevOps Adoption Metrics.*
The reasoning behind the investigation and exploration of whether such a hypotheses can be validated, is purely based on whether there is a link between which specific practices and principles and DevOps adoption measurements in the transitional journey towards DevOps in an organisational setting.

H4. *DevOps Adoption Skills positively* affects *DevOps Adoption Leader Role.*
The reasoning behind the investigation and exploration of whether such a hypotheses can be validated is purely based on whether there is a link between which specific DevOps adoption leader skills should the DevOps adoption leader role (individual or team role) learn or enhance, in the transitional journey towards DevOps in an organisational setting.

5.5. Multivariate Data Analysis

Many statistical techniques focus on just one or two variables. Multivariate analysis (MVA) techniques allow more than two variables to be analysed at once. Therefore, the main advantage of multivariate analysis is that since it considers more than one factor of independent variables that influence the variability of dependent variables, the conclusion drawn is more accurate. Selection of the appropriate multivariate technique depends upon certain decisions that needs to be reached based on Figure 5-5.

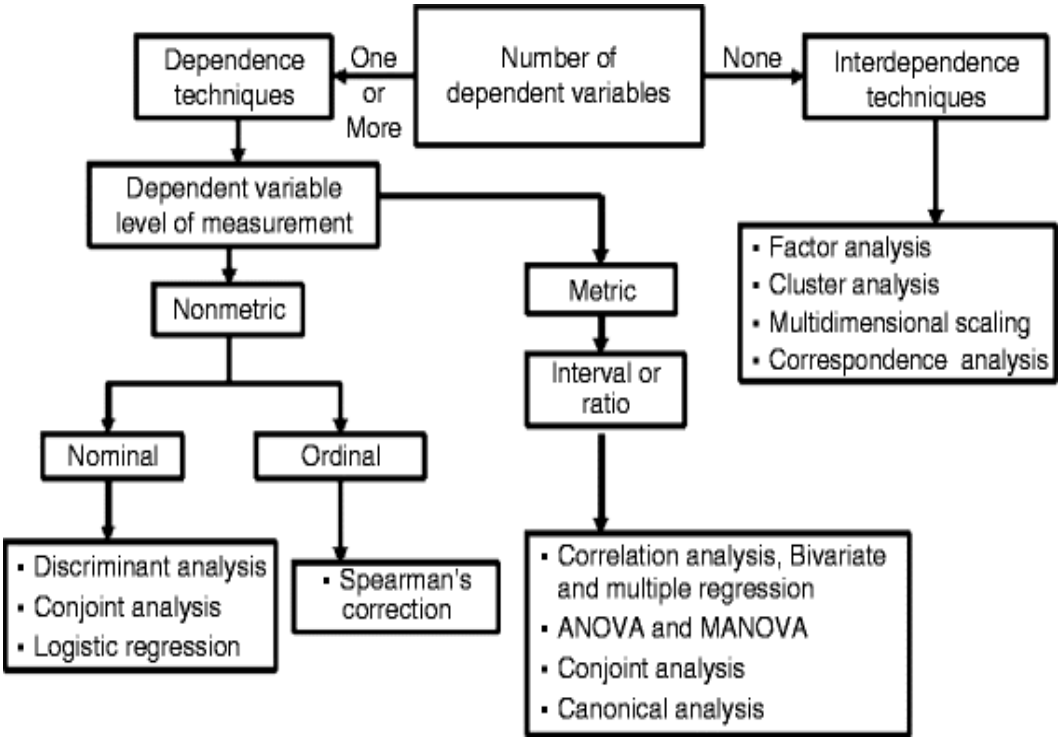


Figure 5-5 Classification Chart of Multivariate Techniques.

Dependence Techniques

Multiple regression is an extension of simple linear regression. It is used when we want to predict the value of a variable based on the value of two or more other variables. The variable we want to predict is called the dependent variable (or sometimes, the outcome, target, or criterion variable). Table 5-4. depicts the type of non-parametric quantitative analysis approach pertinent to each survey questions dataset extracted from the submitted survey results.

Table 5-4 Survey questions structure by type mapped to statistical analysis technique.

Question No.	Description	Model Construct	Statistical Analysis technique
1	Ratio	N/A	Chi-Square tests (categorical variables)
2	Ratio	N/A	Chi-Square tests (categorical variables)
3	Nominal	N/A	Chi-Square tests (categorical variables)
4	Nominal	N/A	Chi-Square tests (categorical variables)
5	Nominal	N/A	Chi-Square tests (categorical variables)
6	Ratio	N/A	Chi-Square tests (categorical variables)
7	Nominal	N/A	Wilcoxon signed-rank test/Mann-Whitney U
8	Ordinal	N/A	Spearman's Rho (Likert-type)
9	Nominal	DevOps Adoption Planning	Spearman's Rho(Likert-type)
10	Ordinal	Practices and Principles	Kruskal–Wallis
11	Nominal	N/A	Spearman's Rho (Likert-type)
12	Nominal	N/A	Spearman's Rho (Likert-type)
13	Ordinal	N/A	Spearman's Rho (Likert-type)
14	Ordinal	Practices and Principles	Spearman's Rho (Likert-type)
15	Nominal	DevOps Adoption Metrics	Spearman's Rho (Likert-type)
16	Nominal	N/A	Spearman's Rho (Likert-type)
17	Nominal	DevOps Adoption Leader Role Should Exist	Spearman's Rho (Likert-type)
18	Ordinal	DevOps Adoption Skills	Spearman's Rho (Likert-type)
19	Nominal	N/A	Spearman's Rho (Likert-type)

A set of questions that yielded a specific dataset and its variables will be introduced to the non-parametric statistical analysis which is thoroughly analysed in the next section. The section will aim to produce a valid set of variables that will formulate the baseline constructs of a DevOps model with correlated manifest variables which will be further validated in section 5.6 for PLS-SEM.

Non-Parametric Statistical Analysis

The symbols for Spearman's rho are ρ for the population coefficient and R_s for the sample coefficient. The bivariate Spearman Rho and Pearson correlation produces a sample correlation coefficient, r ($-1 \leq r \leq 1$), which measures the strength and direction of linear relationships between pairs of continuous variables. For p-value any coefficients <0.05 indicate a significant relationship.

Survey Question 9 (Q9) and Question 17 (Q17) regard the adoption stage that participants indicated that their organisations had achieved, i.e. concerning creation of DevOps Adoption Leader Roles. The tabulated results are shown in Table 5-5, which assesses the comparison between the selected bivariate variables of Q9 and Q17 for the Spearman Rho and Pearson rank-order correlation analysis. Reject the null hypothesis and conclude that the R_s correlation coefficient is significantly different from zero.

Table 5-5 Spearman Rho and Pearson correlation for the model construct 'DevOps Adoption Leader'.

DevOps Adoption Planning and DevOps Adoption Leader Role Should Exist	
Spearman Rho (R_s)	Pearson
0.066292178	0.094733432
p-value	
0.0296454233	

Survey Question 10 (Q10) regards the DevOps practices and principles at which survey participants indicated their preference. The tabulated results for the Spearman Rho and Pearson rank-order correlation analysis for Infrastructure-as-Code and Issue Tracking are shown in Table 5-6. Accept the null hypothesis and conclude that the R_s correlation coefficient is not significantly different from zero. Thus, 'Infrastructure-as-Code 'and 'Issue Tracking' variables are not correlated.

Table 5-6 Spearman Rho and Pearson correlation for the model construct 'Practices and Principles'.

Infrastructure-as-Code and Issue Tracking	
Spearman Rho (R_s)	Pearson
0.25528533	0.264093259
p-value	
4.42555E-05	

Additionally, for DevOps practices and principles survey Question 14 (Q14) was also analysed for the participants who indicated their preference. The tabulated results for the Spearman Rho and Pearson rank-order correlation analysis are shown on Tables 5-7 – 5-9.

Table 5-7 Spearman Rho and Pearson correlation for the model construct 'Practices and Principles'.

Automation and Continuous Integration	
Spearman Rho (R_s)	Pearson
0.437203449	0.413009317
p-value	
4.28911E-13	

Accept the null hypothesis and conclude that the R_s correlation coefficient is not significantly different from zero. Thus, 'automation' and 'continuous integration' variables are positively correlated.

Table 5-8 Spearman Rho and Pearson correlation for the model construct 'Practices and Principles'.

Automation and Monitoring	
Spearman Rho (R_s)	Pearson
0.120479929	0.098630414
p-value	
0.047126291	

Reject the null hypothesis and conclude that the R_s correlation coefficient is significantly different from zero. Thus, 'automation' and 'monitoring' variables are positively correlated and have a significant relationship.

Table 5-9 Spearman Rho and Pearson correlation for the model construct 'Practices and Principles'

Monitoring and Continuous Integration	
Spearman Rho (R_s)	Pearson
0.057195765	0.085062997
p-value	
0.367827767	

Accept the null hypothesis and conclude that the R_s correlation coefficient is not significantly different from zero. Thus, 'monitoring' and 'continuous integration' variables are positively correlated.

Survey Question 15 (Q15) regard the DevOps Adoption Leader Metrics at which survey participants indicated their preference to the type of metrics that the DevOps Adoption Leader Role should be measured against. The tabulated results for the Spearman Rho and Pearson rank-order correlation analysis is shown in Tables 5-10 – 5-26.

Table 5-10 Spearman Rho and Pearson correlation for the model construct 'DevOps Adoption Leader Metrics'.

Key Performance Indicator and Critical Success Factor	
Spearman Rho (R_s)	Pearson
0.414621165	0.414621165
p-value	
8.32132E-12	

Accept the null hypothesis and conclude that the R_s correlation coefficient is not significantly different from zero. Thus, 'key performance indicator' and 'critical success factor' variables are positively correlated.

Table 5-11 Spearman Rho and Pearson correlation for the model construct 'DevOps Adoption Leader Metrics'.

Key Performance Indicator and Deployment Duration	
Spearman Rho (R_s)	Pearson
-0.007686397	-0.007686397
p-value	
0.903750176	

Accept the null hypothesis and conclude that the R_s correlation coefficient is not significantly different from zero. Thus, 'key performance indicator' and 'deployment duration' variables are positively correlated.

Table 5-12 Spearman Rho and Pearson correlation for the model construct 'DevOps Adoption Leader Metrics'.

Key Performance Indicator and Deployment Frequency	
Spearman Rho (R_s)	Pearson
-0.065365763	-0.065365763
p-value	
0.303270486	

Accept the null hypothesis and conclude that the R_s correlation coefficient is not significantly different from zero. Thus, 'key performance indicator' and 'deployment frequency' variables are positively correlated.

Table 5-13 Spearman Rho and Pearson correlation for the model construct 'DevOps Adoption Leader Metrics'.

Key Performance Indicator and Time-to-Recovery	
Spearman Rho (R_s)	Pearson
0.096162627	0.096162627
p-value	
0.1294281	

Accept the null hypothesis and conclude that the R_s correlation coefficient is not significantly different from zero. Thus, 'key performance indicator' and 'time-to-recovery' variables are positively correlated.

Table 5-14 Spearman Rho and Pearson correlation for the model construct 'DevOps Adoption Leader Metrics'.

Deployment Duration and Critical Success Factor	
Spearman Rho (R_s)	Pearson
0.048613214	0.048613214
p-value	
0.444126918	

Accept the null hypothesis and conclude that the R_s correlation coefficient is not significantly different from zero. Thus, 'deployment duration' and 'critical success factor' variables are positively correlated.

Table 5-15 Spearman Rho and Pearson correlation for the model construct 'DevOps Adoption Leader Metrics'.

Deployment Frequency and Critical Success Factor	
Spearman Rho (R_s)	Pearson
-0.090279222	-0.090279222
p-value	
0.154679946	

Accept the null hypothesis and conclude that the R_s correlation coefficient is not significantly different from zero. Thus, 'deployment frequency' and 'critical success factor' variables are positively correlated.

Table 5-16 Spearman Rho and Pearson correlation for the model construct 'DevOps Adoption Leader Metrics'.

Time-to-Recovery and Critical Success Factor	
Spearman Rho (R_s)	Pearson
-0.11918281	-0.11918281
p-value	
0.049874848	

Reject the null hypothesis and conclude that the R_s correlation coefficient is significantly different from zero. Thus, 'time-to-recovery' and 'critical success factor' variables are positively correlated and the relationship is significant.

Table 5-17 Spearman Rho and Pearson correlation for the model construct 'DevOps Adoption Leader Metrics'.

Deployment Frequency and Deployment Duration	
Spearman Rho (R_s)	Pearson
0.320430351	0.320430351
p-value	
2.2399E-07	

Accept the null hypothesis and conclude that the R_s correlation coefficient is not significantly different from zero. Thus, 'deployment frequency' and 'deployment duration' variables are positively correlated.

Table 5-18 Spearman Rho and Pearson correlation for the model construct 'DevOps Adoption Leader Metrics'.

Time-to-Recovery and Deployment Duration	
Spearman Rho (R_s)	Pearson
0.055445801	0.055445801
p-value	
0.382688518	

Accept the null hypothesis and conclude that the R_s correlation coefficient is not significantly different from zero. Thus, 'time-to-recovery' and 'deployment duration' variables are positively correlated.

Table 5-19 Spearman Rho and Pearson correlation for the model construct 'DevOps Adoption Leader Metrics'.

Time-to-Recovery and Deployment Frequency	
Spearman Rho (R_s)	Pearson
0.330236372	0.330236372
p-value	
8.98887E-08	

Accept the null hypothesis and conclude that the R_s correlation coefficient is not significantly different from zero. Thus, 'time-to-recovery' and 'deployment frequency' variables are positively correlated.

Survey Question 18 (Q18) regard the DevOps adoption leader skillset at which survey participants indicated their preference to the type of skills that the DevOps Adoption Leader Role should have. The tabulated results for the Spearman Rho and Pearson rank-order correlation analysis. The tabulated results are shown in Tables 5-20 – 5-25.

Table 5-20 Spearman Rho and Pearson correlation for the model construct 'DevOps Adoption Leader Skillset'.

Technical Background and Business Background	
Spearman Rho (R_s)	Pearson
0.29307913	0.260095834
p-value	
0.000110175	

Reject the null hypothesis and conclude that the R_s correlation coefficient is significantly different from zero. Thus, 'technical background' and 'business background' variables are positively correlated, and the relationship is very significant.

Table 5-21 Spearman Rho and Pearson correlation for the model construct 'DevOps Adoption Leader Skillset'

Communication and Collaboration and Project Management	
Spearman Rho (R_s)	Pearson
0.232419227	0.238993887
p-value	
0.002359682	

Reject the null hypothesis and conclude that the R_s correlation coefficient is significantly different from zero. Thus, ‘communication and collaboration’ and ‘project management' variables are positively correlated and the relationship is highly significant.

Table 5-22 Spearman Rho and Pearson correlation for the model construct ‘DevOps Adoption Leader Skillset’.

Technical Background and Communication & Collaboration	
Spearman Rho (R_s)	Pearson
0.263132124	0.241958284
p-value	
0.00054735	

Reject the null hypothesis and conclude that the R_s correlation coefficient is significantly different from zero. Thus, ‘technical background’ and ‘communication and collaboration’ variables are positively correlated.

Table 5-23 Spearman Rho and Pearson correlation for the model construct ‘DevOps Adoption Leader Skillset’.

Technical Background and Project Management	
Spearman Rho (R_s)	Pearson
0.517011645	0.223676708
p-value	
6.17923E-13	

Accept the null hypothesis and conclude that the R_s correlation coefficient is not significantly different from zero. Thus, ‘technical background’ and ‘project management' variables are positively correlated.

Table 5-24 Spearman Rho and Pearson correlation for the model construct ‘DevOps Adoption Leader Skillset’.

Business Background and Communication & Collaboration	
Spearman Rho (R_s)	Pearson
0.242892201	0.26029647
p-value	
0.001462857	

Reject the null hypothesis and conclude that the R_s correlation coefficient is significantly different from zero. Thus, ‘business background' and ‘communication and collaboration’ variable

s are positively correlated.

Table 5-25 Spearman Rho and Pearson correlation for the model construct 'DevOps Adoption Leader Skillset'.

Business Background and Project Management	
Spearman Rho (R_s)	Pearson
0.266875972	0.230565284
p-value	
0.000452385	

Reject the null hypothesis and conclude that the R_s correlation coefficient is significantly different from zero. Thus, 'business background' and 'project management' variables are positively correlated. For all Spearman's Rank tests depicted in Tables 5-5 – 5-25, the null hypothesis is: there is no significant correlation between the two variables. Table 5-26a to Table 5-26c depicts the results of the Spearman Rank hypotheses testing.

Table 5-26a Spearman Rank Hypothesis Testing.

Construct	**Variables**	**Correlation**	**Significance**
DevOps Adoption Leader Role (Survey question no. 9 DevOps Adoption Planning and no. 17)	DevOps Adoption Planning and DevOps Adoption Leader Role Should Exist	Positive	significant
DevOps Practices and Principles (Survey Question No. 10 and No. 14)	Infrastructure-as-Code and Issue Tracking	Null hypothesis	non-significant
	Automation and Continuous Integration	Null hypothesis	non-significant
	Automation and Monitoring	Positive	significant
	Monitoring and Continuous Integration	Null hypothesis	non-significant

Table 5-26b Spearman Rank Hypothesis Testing.

Construct	Variables	Correlation	Significance
DevOps Adoption Metrics (Survey Question No. 15)	Key Performance Indicator and Critical Success Factor	Null hypothesis	non-significant
	Key Performance Indicator and Deployment Duration	Null hypothesis	non-significant
	Key Performance Indicator and Deployment Frequency	Null hypothesis	non-significant
	Key Performance Indicator and Time-to-Recovery	Null hypothesis	non-significant
	Deployment Duration and Critical Success Factor	Null hypothesis	non-significant
	Deployment Frequency and Critical Success Factor	Null hypothesis	non-significant
	Time-to-Recovery and Critical Success Factor	Positive	significant
	Deployment Frequency and Deployment Duration	Null hypothesis	non-significant
	Time-to-Recovery and Deployment Duration	Null hypothesis	non-significant
	Time-to-Recovery and Deployment Frequency	Null hypothesis	non-significant

Table 5-26c Spearman Rank Hypothesis Testing.

Construct	Variables	Correlation	Significance

DevOps Adoption Leader Skills (Survey Question No. 18)	Technical Background and Business Background	Positive	highly significant
	Communication and Collaboration and Project Management	Positive	highly significant
	Technical Background and Communication & Collaboration	Positive	highly significant
	Business Background and Communication & Collaboration	Positive	non-significant
	Business Background and Project Management	Positive	highly significant

The next section builds on the bivariate analysis concluded in this section, and extrapolates a model consisting of the manifest variables of the four model constructs; i.e. 1) DevOps Adoption Leader, 2) DevOps Adoption Practices and Principles, 3) DevOps Adoption Leader Metrics and 4) DevOps Adoption Leader Skillset.

5.6. Designing the PLS-SEM Model

5.6.1 Partial Least Squares – Structural Equation Modelling (PLS-SEM)

Structural equation modelling is a multivariate statistical analysis technique that is used to analyse structural relationships. In other words, Structural Equation Modelling (SEM), is a multivariate method used to test hypotheses regarding the influences among interacting variables. Moreover, the purpose of structural equation modelling is twofold. First, it aims to obtain estimates of the parameters of the model, i.e. the factor loadings or weights, the variances and covariances of the factor, and the residual error variances of the observed variables (Hair et al., 2014). SEM in a single analysis can assess the assumed causation among a set of dependent and independent constructs, i.e., validation of the structural model and the loadings of observed items (measurements) on their expected manifest variables (constructs) to achieve validation of the measurement model. The combined analysis of the measurement and the structural model enables the measurement errors of the observed variables to be analysed as an integral part of the model, and factor analysis combined in one operation with hypotheses testing. The PLS-SEM characteristic of higher

statistical power is quite useful for exploratory research that examines less developed or still developing theory. The theory around DevOps adoption leadership is still being enriched in the research community. Moreover, PLS-SEM is not only appropriate for exploratory research but also for confirmatory research (Hair et al., 2019).

The data were collected form 250 respondents through a survey method from professional selected randomly. Moreover, the '10 times rule' can be followed for the determination of sample size while analysis of data uses PLS-SEM technique (Hair et al., 2014). By this rule the sample size will be the ten times the maximum number of arrows pointing at a manifest construct. Thus the analysis is good enough with only 20 samples for this study (Hair et al., 2014) although (Cornish, 2002) suggested additional representation to receive more acceptable results. In this study the items of the survey have been adapted from the interview series. responses were keyed in using IBM SPSS software. Then, the data file was converted into a csv (comma-separated value) file making the file useable for PLS analysis. Subsequently, the data were analysed through Structural Equation Modelling (SEM) - with the use of SmartPLS v3.3. The produced responses analysis and evaluation aims to form the baseline of the PLS-SEM model design and validation process. The validation process of the formative model will be carried out based on a specific sequence of steps as follows below (Hair et al. 2018):

- Redundancy analysis of model constructs, see section 5.6.2;
- Measurement model assessment, see section 5.6.3;
- Structural model assessment, see section 5.6.4.

The conceptual model of the study has been analysed by considering development of equation path model, assessment of measurement model, assessment of structural model, and measurement of mediating effects.

The survey questions examined for correlation relationships regard:

- Survey Question No. 9 - DevOps Adoption Planning and No.17 - DevOps Adoption Leader Role Should Exist for the model construct 'DevOps Adoption Leader', see Table 5-5.
- Survey Question No. 10 and No. 14 - Practices and Principles for the model construct 'DevOps Adoption Practices and Principles', see Table 5-6 – 5-9.
- Survey Question No. 18 - Skills for the model construct 'DevOps Adoption Leader Metrics', see Table 5-10 – 5-19.
- Survey Question No. 15 - Metrics for the model construct 'DevOps Adoption Leader Skillset', see Table 5-20 – 5-25.

5.6.2 Develop Equation Path Model

The first step in evaluating PLS-SEM results involves development of the path model by undertaking redundancy analysis of model constructs. If the measurement models meet all the required criteria, it is then needed to assess the structural model (Hair et al., 2018). The redundancy analysis shown in Figure 5-6 yielded the refined set of metrics manifest variable that has a significant relationship to the 'metrics' latent construct.

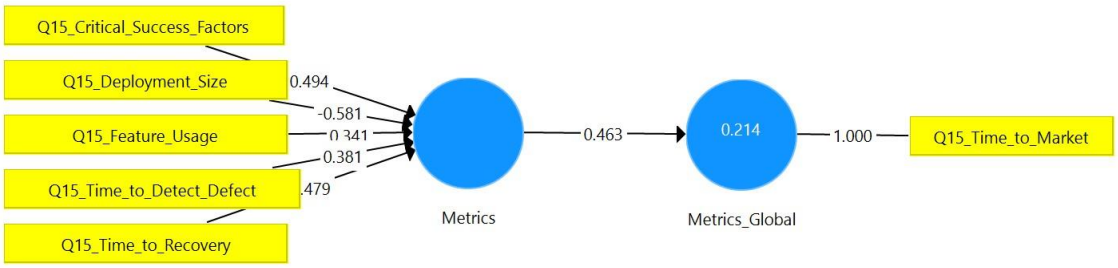


Figure 5-6 Redundancy analysis of 'Metrics' Manifest Variable.

The redundancy analysis shown in Figure 5-7 yielded the refined set of DevOps practices manifest variable that has a significant relationship to the 'practices and principles' latent construct.

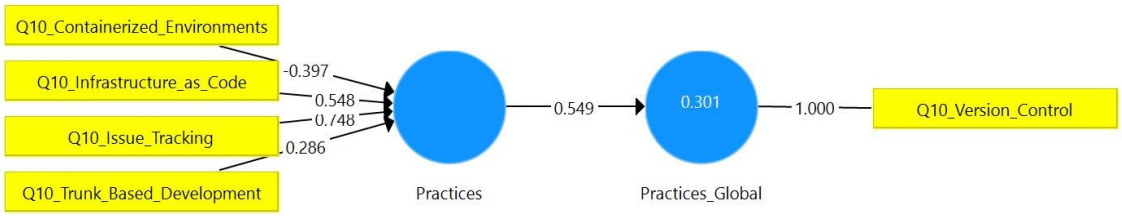


Figure 5-7 Redundancy analysis of 'Practices' Manifest Variable.

The redundancy analysis shown in Figure 5-8 yielded the refined set of DevOps principles manifest variable that has a significant relationship to the 'practices and principles' latent construct.

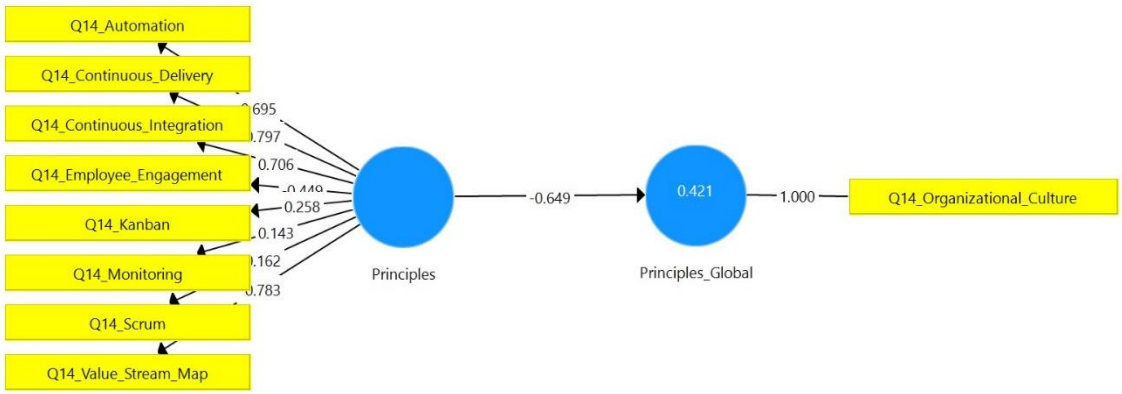


Figure 5-8 Redundancy analysis of 'Principles' Manifest Variable.

The redundancy analysis shown in Figure 5-9 yielded the set of skills manifest variables that have a significant relationship to the 'Skills' latent construct.

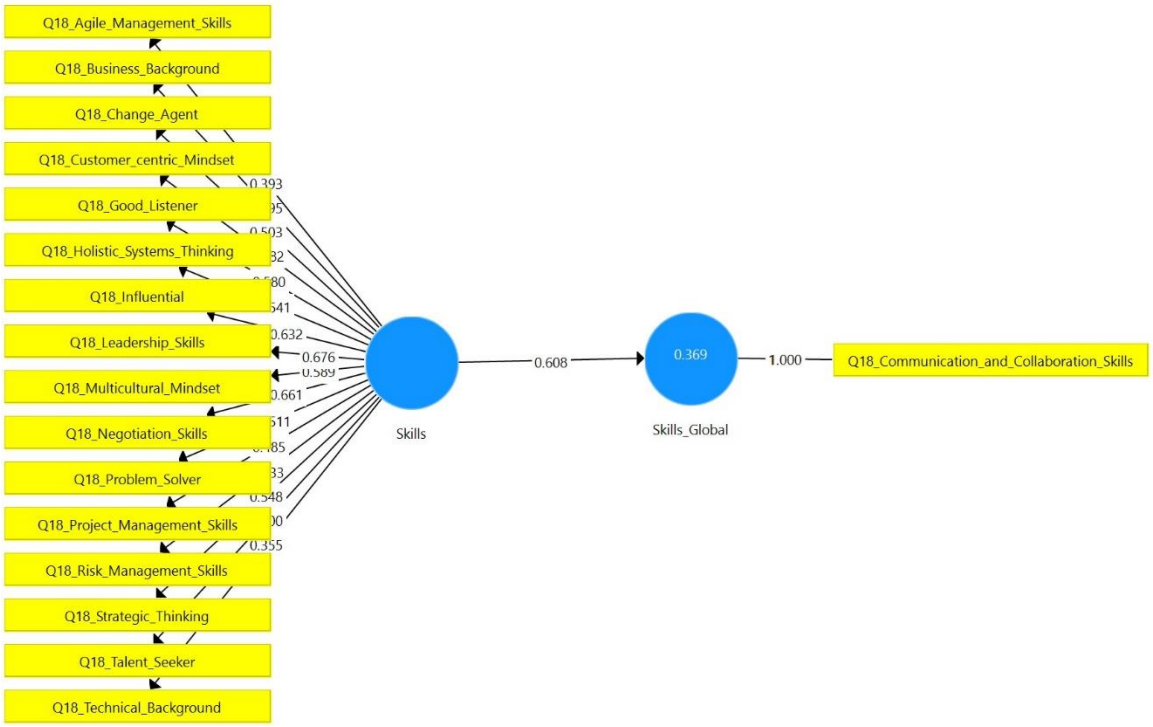


Figure 5-9 Redundancy analysis of 'Skills' Manifest Variable.

5.6.3. Assess Measurement Model

The survey analysis and evaluation generated a set of constructs (i.e. practices, skills, metrics, DevOps adoption leads) that can used to define a baseline model, which when applied to results, facilitates the answering of RQ3, see section 1.5. The authors proceeded with an initial evaluation of the survey dataset in CB-SEM (using IBM SPSS Amos 26), which indicated a Standardised Root Mean Squared Residual (SRMR) of 1.20 much higher than the validity threshold coefficient of 0.05. Therefore, the grouped construct indicators should be assessed in PLS-SEM in the context of redundancy analysis, i.e., to understand whether all constructs show convergent validity.

In a reflective model, the underlying construct drives the indicators, which have positive and, desirably, high intercorrelations. In a formative model, the indicators do not necessarily share the same theme and hence have no preconceived pattern of intercorrelation (Hair et al., 2014). In other words, in a reflective model, constructs cause measures, while in a formative model, measures cause constructs. Thus, reflective models describe measures as imperfect indicators of underlying phenomena while formative models describe measures as an indistinguishable part of the constructs they are tied to (Hair et al., 2018).

Following redundancy analysis, for each of the formative measurement model constructs, the analysis yielded path coefficients above 0.70, which indicates that all formatively measured constructs exhibit convergent validity (Hair et al., 2014). Convergent validity refers to how closely the new scale is related to other variables and other measures of the same construct. Not only should the construct correlate with related variables but it should not correlate with dissimilar, unrelated ones (Hair et al., 2014). The produced formative measurement model is depicted in Figure 5-10 and the description of the formative indicators in Table 5-27.

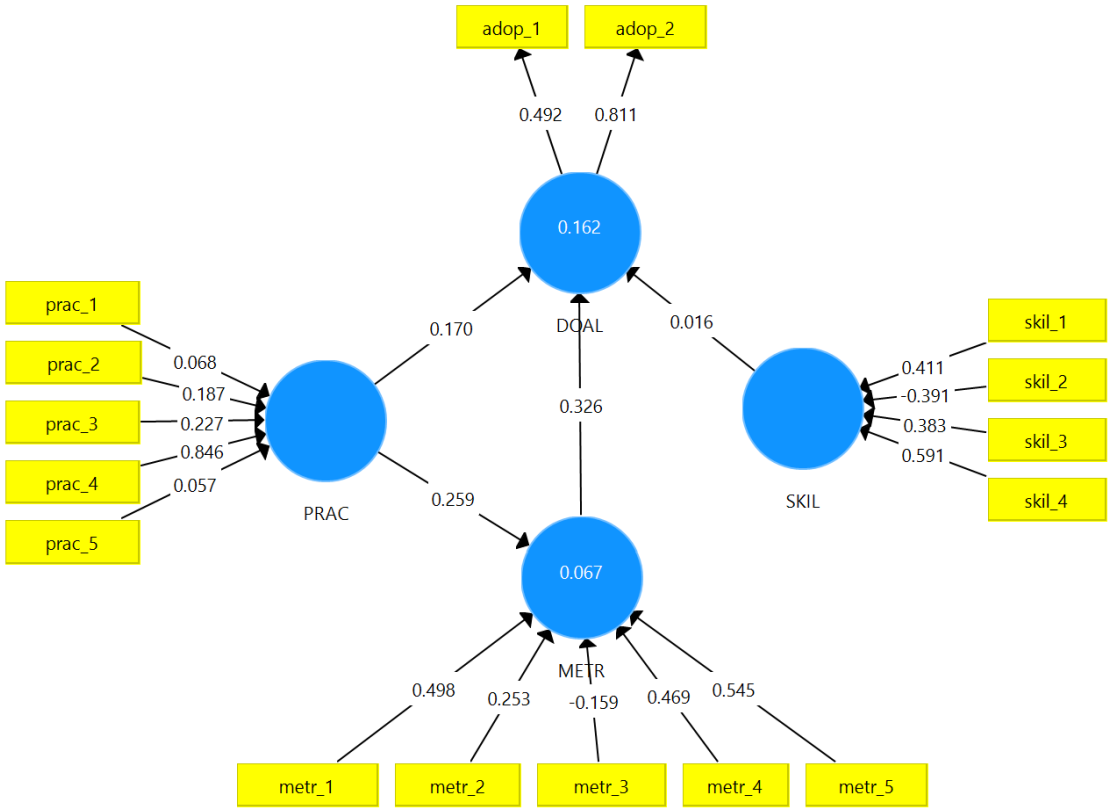


Figure 5-10 Formative Measurement Model.

Conceptualised model constructs, see Figure 5-3, are subsequently transposed to the formative measurement model, in SmartPLS. The formative constructs are PRAC (practice), METR (metric), SKIL (skill) and DOAL (DevOps Adoption Lead). The formative measurement model describes the relationship among the manifest variables in the PLS-SEM path model, see Figure 5-10. The path model equation measures the T-value between the constructs and associated indicators of the manifest variable. Therefore, the measurement model signifies the level of the constructs as well as the manifest variables.

Table 5-27 Model Constructs and Indicator List and Descriptions.

Model Construct	Model Indicator Description	Description
PRAC	Infrastructure as Code (IaC)	The practice IaC is adopted for DevOps
	Issue Tracking	The practice Issue Tracking is adopted for DevOps
	Automation	The practice Automation is adopted for DevOps
	Continuous Integration (CI)	The practice CI is adopted for DevOps
	Monitoring	The practice Monitoring is adopted for DevOps
METR	Critical Success Factors (CSF)	The metric CSF is used to measure DevOps adoption
	Deployment Duration	The metric Deployment Duration is used to measure DevOps adoption
	Deployment Frequency	The metric Deployment Frequency is used to measure DevOps adoption
	Key Performance Indicator (KPI)	The metric KPI is used to measure DevOps adoption
	Time to Recovery	The metric Time to Recovery is used to measure DevOps adoption
SKIL	Business Background	The skill Business Background is acquired by the DevOps adoption leader
	Communication and Collaboration	The skill Communication and Collaboration is acquired by the DevOps adoption leader
	Project Management	The skill Project Management is acquired by the DevOps adoption leader
	Technical Background	The skill Technical Background is acquired by the DevOps adoption leader
DOAL	DevOps Adoption Planning	There is a decision on DevOps adoption planning by my organisation
	DevOps Leader Role Should Exist	A DevOps adoption leader exists for the transition to DevOps

The Standardised Root Mean Square (SRMR) value of 0.078 indicates a good fit for the model for PLS-SEM (Hair et al., 2014). Since we have confirmed that the formative measurement model constructs are reliable and valid, the next steps of the process assess the interpretation of outer Variance Inflation Factor (VIF) values. Table 5-28 indicates that formative indicator 'Infrastructure-as-Code' has the highest VIF

value (3.046). Hence, VIF values are uniformly below the threshold value of 5. The collinearity issue of the indicators also measured by the calculation of VIF of each observation in the study and found no collinearity issue exists among the manifest variables. Therefore, the constructs and manifest variables of the study can be used for analysis and interpretation. The model assessment included the interpretation of the formative model constructs. Therefore, multicollinearity was assessed among the indicators of the constructs of the formative model.

Table 5-28 Outer VIF values of model.

Model Code	Manifest Variable	VIF
prac_1	Q10_Infrastructure-as-Code	1.113
prac_2	Q10_Issue_Tracking	1.093
prac_3	Q14_Automation	1.232
prac_4	Q14_Continuous_Integration	1.217
prac_5	Q14_Monitoring	1.034
metr_1	Q15_Critical_Success_Factors	1.253
metr_2	Q15_Deployment_Duration	1.125
metr_3	Q15_Deployment_Frequency	1.266
metr_4	Q15_KPI	1.250
metr_5	Q15_Time_to_Recovery	1.173
adop_1	Q9 DevOps Adoption Planning	1.012
adop_2	Q17_DevOps_Leader_Role_Should_Exist	1.012
skil_1	Q18_Business_Background	1.143
skil_2	Q18_Communication_and_Collaboration_Skills	1.111
skil_3	Q18_Project_Management_Skills	1.404
skil_4	Q18_Technical_Background	1.369

Results suggest that collinearity does not reach critical levels in any of the formative constructs and is not an issue for the estimation of the PLS path model.

5.6.4. Assess Structural Model for Significance

Following confirmation of validity and reliability of the formative measurement model constructs, the next steps of the process is to assess the interpretation of the structural model, see Figure 5-11.

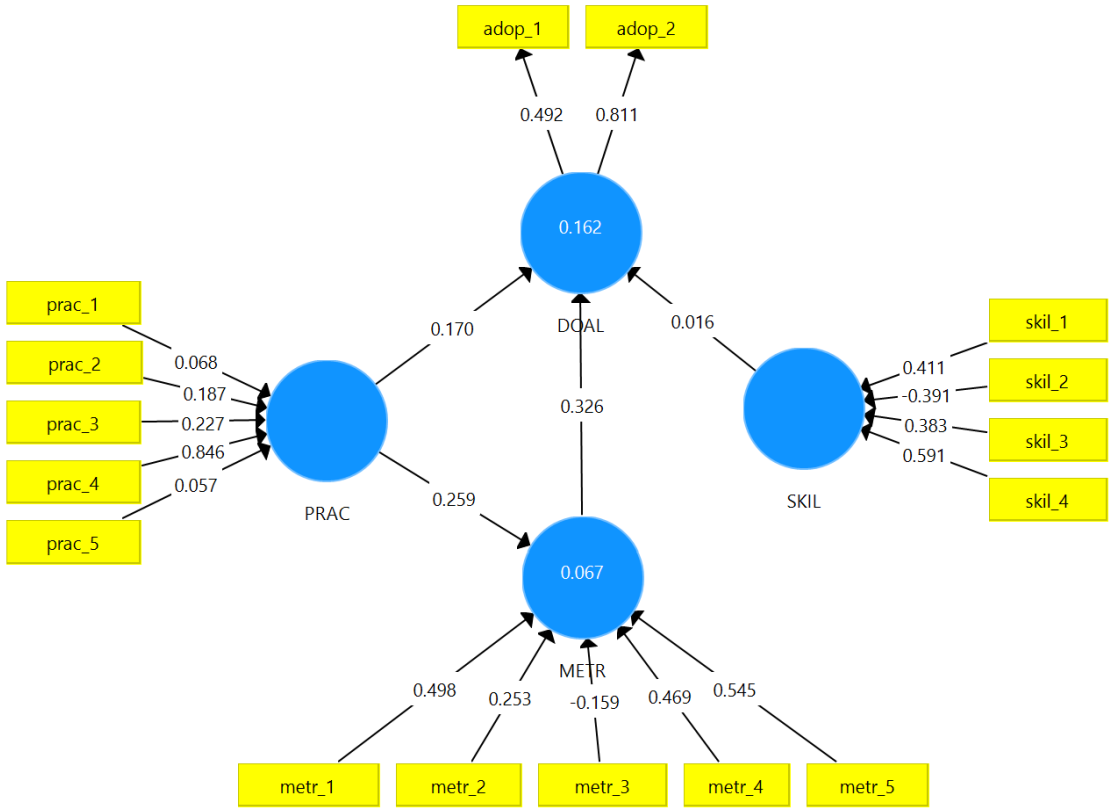


Figure 5-11 Formative Structural Model.

Firstly, multicollinearity was assessed among the indicators of the constructs in terms of outer VIF values, which regards indicators and constructs of the model (shown in Table 5-28). Secondly, measuring significance levels of formative structural model ultimately depends on its standard error; obtained by means of bootstrapping. Therefore, for the purposes of deducing significance levels per model construct, two-tailed tests were used to conclude which of the coefficients are statistically significant at the 5% error probability, i.e., significance level. It was observed that continuous integration, critical success factors, key performance indicators (KPI), time to recovery, DevOps adoption planning, and DevOps adoption leader role existence, were all significant at a 5% level. Path coefficients are standardised versions of linear regression weights which can be used in examining the possible causal linkage between statistical variables in the structural equation modelling approach (Hair et al., 2014). Significance levels coupled with path coefficient results indicate that (H2) DevOps Adoption Metrics *positively* affects DevOps Adoption Leadership, can be supported. The path coefficient of the SKIL model construct has a high significance relationship to DOAL model construct. Therefore, that can be an indication that (H4) DevOps Adoption Leader Skills *positively* affects DevOps Adoption Leader Role., can be supported.

Measurement model, i.e. the outer model, deals with the measurement of manifest variables of the PLS-SEM path modelling. Each latent constructs of the model comprises of with multiple reflective

observations. Figure 5-10 depicted that Practices have highest effect on the level of adopted metrics for the leader role in the organisation. PLS-SEM algorithm provides model relationships (path coefficient) among the constructs that represents hypothesised relationship of the constructs. The standardised values provided by path coefficient are higher than zero, which signifies positive relationship between the constructs; whereas the t-value or p-value signifies the level of relationships (see table 5-29).

Table 5-29 Formative Constructs PLS-SEM Testing Results.

Formative Construct	Formative Indicator Description	Formative Indicators	VIF	t Value	p Value	95% BCa Confidence Interval		Significance (p < 0.05)?
PRAC	Infrastructure as Code	prac_1	1.113	0.198	0.843	-0.445	0.582	No
	Issue Tracking	prac_2	1.093	0.853	0.394	-0.246	0.647	No
	Automation	prac_3	1.232	0.82	0.413	-0.318	0.752	No
	Continuous Integration	prac_4	1.217	3.345	**0.001**	0.557	1.079	Yes
	Monitoring	prac_5	1.034	0.086	0.931	-0.512	0.61	No
METR	Critical Success Factors	metr_1	1.253	2.492	**0.013**	0.147	0.781	Yes
	Deployment Duration	metr_2	1.125	1.148	0.251	-0.161	0.622	No
	Deployment Frequency	metr_3	1.266	0.721	0.471	-0.543	0.348	No
	KPI	metr_4	1.25	2.167	**0.031**	0.019	0.83	Yes
	Time to Recovery	metr_5	1.173	2.554	**0.011**	0.201	0.809	Yes
SKIL	Business Background	skil_1	1.143	0.974	0.33	-0.469	1.026	No
	Communication and Collaboration	skil_2	1.111	0.744	0.457	-0.993	0.542	No
	Project Management	skil_3	1.404	0.474	0.636	-0.884	1.155	No
	Technical Background	skil_4	1.369	1.243	0.214	-0.102	1.192	No
DOAL	DevOps Adoption Planning	adop_1	1.012	2.237	**0.026**	0.114	0.999	Yes
	DevOps Leader Role Should Exist	adop_2	1.012	4.339	**0**	0.387	1.01	Yes

Table 5-30 Assessment of Formative Structural Model Relevance of Path Relationships (Total Effects).

Path Relationship	Original Sample (O)	Sample Mean (M)	Standard Deviation (STDEV)	t Value	p Value
METR -> DOAL	0.331	0.335	0.067	4.906	**0**
PRAC -> DOAL	0.249	0.264	0.109	2.274	**0.023**
PRAC -> METR	0.26	0.287	0.106	2.447	**0.015**
SKIL -> DOAL	0.015	-0.035	0.081	0.184	0.854

Thirdly, in addition to significance level interpretation of the formative structural model, relevance of significant relationships was also measured to assess, besides significance, the size of the model constructs by examining Total Effects, see Table 5-30.

Assess the model construct predictive power through level of R^2, f^2, predictive relevance Q^2, q^2 effect size

The formative structural model assessment include a few more steps as follows:

1) Assessment of levels of R^2 (see Table 5-31)
2) Assessment of levels of f^2 (see Table 5-32)
3) Assessment of predictive relevance Q^2 (see Table 5-33)
4) Assessment of q^2 effect size (see Table 5-34)

Table 5-31 Formative Path Relationship R^2.

Formative Construct	Original Sample (O)	Sample Mean (M)	Standard Deviation (STDEV)	t Value	p Value	95% BCa Confidence Interval		Predictive power?
DOAL	0.152	0.189	0.055	2.765	0.006	0.048	0.225	Weak
METR	0.064	0.09	0.051	1.251	0.211	-0.004	0.142	None

The Coefficient of Determination R^2 value is a measurement of the model's predictive power. Currently, we considered the Adjusted Coefficient of Determination R^2 value, which indicated that 'DevOps Adoption Lead' construct possesses weak predictive power. Moreover, the effect size f^2 is also measured (see Table 5-32) to evaluate the R^2 values of all endogenous constructs; i.e. the chance in the R^2 value when a specified exogenous construct is omitted from the model and can be used to evaluate whether the omitted construct has a substantive impact on the endogenous constructs (Hair et al., 2014). Table 5-32 indicates that the relationship between 'Metrics' and 'DevOps Adoption Lead' constructs has a large predictive power.

Table 5-32 Formative Constructs f^2.

Formative Path Relationship	Original Sample (O)	Sample Mean (M)	Standard Deviation (STDEV)	*t* Value	*p* Value	95% BCa Confidence Interval		Predictive power?
METR -> DOAL	0.12	0.133	0.057	2.113	0.035	0.039	0.039	Large
PRAC -> DOAL	0.029	0.051	0.044	0.665	0.506	-0.355	0.150	None
PRAC -> METR	0.072	0.107	0.064	1.126	0.261	-0.183	0.103	None
SKIL -> DOAL	0	0.009	0.012	0.022	0.983	-0.099	0.186	None

The Q^2 value is an indicator of the model's out-of-sample predictive power or predictive relevance (see Table 5-33) and can be coupled with interpretations of R^2 values. Moreover, cross-validated redundancy builds on the path model estimates of both the structural model and the measurement model; and is therefore the preferred measurement method for Predictive Relevance Q^2.

Table 5-33 Blindfolding and Predictive Relevance Q^2 (Cross-validated Redundancy).

Formative Indicator	SSO	SSE	Q^2	Predictive power?
DOAL	500	479.875	0.04	Large
METR	1250	1239.902	0.008	Small
PRAC	1250	1250	-	None
SKIL	1000	1000	-	None

On the other hand, cross-validated communality uses only the construct scores estimated for the target endogenous construct, without including structural model information, to predict the omitted data points. Moreover, the Effect Size q^2 (MV Prediction Summary), aims to compare the PLS (Partial Least Squared) and LM (Linear model) results. The PLS results should be higher that the LM results in order to establish the predictive power of the formative structural model (see Table 5.34). Since only three formative indicators possess predictive powers, the formative structural model can be attributed as having an overall weak predictive power.

Table 5-34 Effect Size q^2 (MV Prediction Summary).

	PLS			LM			
Formative Indicator	RMSE	MAE	Q^2_predict	RMSE	MAE	Q^2_predict	Predictive power?
adop_2	0.468	0.433	0.027	0.462	0.413	0.051	Yes
adop_1	1.207	0.989	-0.013	1.211	1.018	-0.02	No
metr_4	0.492	0.481	0.013	0.503	0.485	-0.034	No
metr_2	0.502	0.5	0.003	0.502	0.481	-0.001	Yes
metr_1	0.505	0.496	-0.013	0.505	0.488	-0.016	Yes
metr_5	0.499	0.489	0.011	0.491	0.461	0.044	No
metr_3	0.503	0.503	-0.005	0.516	0.508	-0.056	No

The list of initial hypotheses is provided below:

(H1) DevOps Practice and Principle adoption *positively* affects DevOps Adoption Leader Role.

(H2) DevOps Adoption Metrics *positively* affects DevOps Adoption Leader Role.

(H3) DevOps Practice and Principle adoption *positively* affects DevOps Adoption Metrics.

(H4) DevOps Adoption Leader Skills *positively* affects DevOps Adoption Leader Role.

Following a structure approach to PLS-SEM analysis and evaluation, the results add further support to the formative model; since constructs 'Critical Success Factors', 'Deployment Duration' and 'DevOps Leader Role Should Exist' are significant, and as such possess predictive power. Therefore, these are indications that hypothesis (H1), can be supported by the formative model's DOAL and METR constructs which also possess large predictive power.

The PLS-SEM analysis, and evaluation results, indicate that the hypothesis (see below) can be supported by the formative model, and hence: (H2) DevOps Adoption Metrics positively affects DevOps Adoption Leadership. The significance levels of the formative structural model indicators did not support (H1) DevOps Practice and Principle adoption *positively* affects DevOps Adoption Leader Role and (H3) DevOps Practice and Principle adoption *positively* affects DevOps Adoption Metrics.. However, the assessment of formative structural model relevance of path relationships (Total Effects) showed that all three hypotheses are supported.

5.7 Chapter Summary

This chapter has considered the practical steps to the second empirical research approach framed, described, and associated, earlier in Chapter Three. A detailed account of the quantitative research techniques and tools utilised in this research has been also provided which leads from conducting a survey with 250 participants right through to a leadership model for DevOps adoption. A conceptual model design was baselined on the dataset of 250 responses which produced a set of hypotheses (H1) through (H4). The dataset was imported in SmartPLS and validated using PLS-SEM. The validation results and examine findings in terms of significance and predictive power indicate that RQ3A. 'Can Leadership affect DevOps adoption within a software-intensive organisation?' and RQ3B. 'What is the leadership style that can be attributed to the DevOps adoption leader role?, are supported. The next chapter validates the DevOps Adoption Leadership Model conducted in this research.

Chapter Six
Model Evaluation

6.1. Chapter Overview

This chapter aims to evaluate through a series of online focus groups, the conceptual model designed and thoroughly described in Chapter Five, see Figure 5-3, including a set of hypotheses linked to the conceptual model's onstructs. The conceptual model's dataset led to PLS-SEM model validation which produced the corresponding formative measurement model, see Figure 5-10 and formative structural model, see Figure 5-11, which in turn, supported the set of hypotheses. The model evaluation will establish key aspects of the model's applicability and generalisability in the industry and deepen the already acquired understanding. A clearer picture of the effects of the relationship between model constructs will be examined in the context of DevOps adoption and its leadership characteristics within software intensive organisations. The degree of consensus by focus group participants on presented key concepts, and the relationship of model constructs, are also described with pertinent results and validation outcomes explained.

6.2. Introduction

The focus group design, as described in section 3.3.4, assumes that there is no single qualitative perspective (Krueger and Casey, 2015). The focus group approach allows the researcher to gather opinions, ideas, and definitive answers from different vantage points; and has the potential to bring focus on the model analysis. The confirmatory nature for which the recruiting, focus group session scheduling, analysis and evaluation of results and outcomes was performed is specific to the research aim, questions, and objectives of the thesis at hand.

6.3. Analytic Frameworks in Focus Groups

The focus group method is suitable for gathering multiple perspectives on a specific set of topics. Five focus group analytic frameworks were defined by Kruger and Casey (2015):

1. Constant comparative – compare one segment of data with another to identify similarities and difference;

2. Identifying individual change – monitor an individual's comments throughout the focus group session;
3. Critical incidents – identify events, actions or situations that were influential to individuals, organisations or society;
4. Key concepts – identify a limited number of important ideas, experiences or preferences that illuminate the study;
5. Testing alternatives – show participants examples, descriptions or actual products, and ask them to choose.

From the five established focus group analytic frameworks (Kruger and Casey, 2015), *key concepts and constant comparative* were most relevant to the current confirmatory study. The primary analytic framework used to meet the needs of the confirmatory study, was *key concepts*; i.e. considering firstly the model constructs, and then subsequently the model in its entirety. *Constant comparative* approach was used, as a complementary method, to identify any alternative preferences to the relationships of model construct, as presented in section 5.4.

The aim of the focus was, amongst others, to: i) obtaining practitioner feedback on research questions; ii) recognise past experience that can be studied in more detail by other methods; iii) provide initial evaluation of potential solutions, based on practitioner feedback; iv) collect lessons learned recommendations; v) identify potential root causes of phenomena; and vi) investigate and identify further research questions in the scope of future work.

Extending the structure of the focus groups design, a five-phased approach was adopted which had been shown to be relevant to all main phases of research life cycle (Langford and McDonaugh, 2003). In the informational phase the focus group method can be used to collect characterising information about the current practices, experience, or problems. Moreover, in the propositional phase the initial constructs, i.e., models, theories, or prototypes, can be subjected to practitioner and user opinions to provide early feedback. In the analytical phase the user feedback can be used to evaluate the operationalisation of constructs, or to test initial feasibility of them. In the evaluative phase focus groups can be used to refine research questions, provide some of the empirical feedback, and support the interpretation of empirical data. Finally, in the technology transfer phase the focus group can help researchers to package their contributions into a form that is more easily deployable by users. In addition, a focus group session can also act as a "sales session" for such research results. Research questions for each area are included in Table 6-1.

Table 6-1 Model Evaluation for all Research Lifecycle Phases.

Information Phase	IP1. What DevOps practices currently exist in the industry?	RQ1) Which agile, lean and DevOps practices and principles can improve productivity for software product development teams in software-intensive organisations that have adopted a structured service management approach?
	IP2. What DevOps principles currently exist in the industry?	
	IP3. What risk factors currently exist in the industry for DevOps adoption?	
	IP4. What DevOps metrics currently exist in the industry?	
Propositional Phase	PP1. What are possible hypotheses?	RQ2) Are DevOps-oriented environments and their teams an extension or a replacement of process-driven structured service management approaches?
	PP2. What similar models exist in the industry?	
	PP3. Are the assumptions of the model realistic?	
Analytical Phase	AP1. Is the model understandable?	RQ3) Can Leadership affect DevOps adoption within a software-intensive organisation? What is the leadership style that can be attributed to the DevOps adoption leader role?
	AP2. How can it be practically employed	
	AP3. What are the potential problems in using or understanding the model?	
	AP4. Are there any omissions or gaps in the model?	
Evaluative Phase	EP1. Are the model constituents sound and practical?	
	EP2. What do the model constituents mean?	
Technology Transfer Phase	TTP1. Is the model fit for reuse purposes?	
	TTP2. How could it be improved?	

The objective of all focus group sessions was to provide insights to the motivation behind the need to gain an improved understanding of the inner workings of organisations aiming to embark on a transitional journey. The goals of the focus groups aimed to uncover more information about DevOps practice and principal adoption, the adoption risk factors, and obtain feedback on the specific characteristics, style, and traits of how DevOps adoption leaders are perceived; both as an individual, and as team member undertaking a role, based on a previous study (Maroukian and Gulliver, 2020).

6.4. Conducting Focus Groups and Model Evaluation

6.4.1. Planning and Conducting Focus Groups

The qualification and identification process of focus group participants was based on three primary selection criteria.

- Firstly, individuals were selected from companies that were not competitors with each other to avoid some of the weaknesses of focus groups such as hidden agendas of potential business relationships of participants.
- Secondly, it was paramount to include a high diversity of participants from differing company sizes, i.e. to trigger as many shared insights as possible in each one of the scheduled three focus groups sessions.

  Thirdly, all the participants should have possessed either prior or current leadership roles in software-intensive organisations of differing company sizes - based on the European Commission's definition of company sizes (EC, 2003).

The focus group participant selection approach was crucial to secure as much constructive criticism as possible. See Table 6-2 for a detailed description of focus group participants. PX = Participant Experience (years), Role (C-Level - CIO, CTO, CDO, etc., Con – Consulting Services, BD – Business Development, PS Dev – Product/Software Development, Info Sec – Information Security), Region of work (ME - MEA, E-Europe, NA - North America), CS = Company Size (Micro- MC < 10, Small- S < 50, Medium- M < 250, Large- L > 251) based on European Commission definition of company size (EC, 2003).

Table 6-2 Focus Groups Participant List (PX = participant experience, RG = region, CS = company size).

ID	Focus Group	Participant ID	Age	PX	Role	Principal Industry	RG	CS
1	FS1	FSP1	41-50	16-20	C-Level	IT Services/Consulting	ME	5 - 50
2		FSP2	51-60	21+	C-Level	IT Services/Consulting	E	5 - 50
3		FSP3	61-70	21+	Con	IT Services/Consulting	E	1 - 4
4		FSP4	61-70	21+	BD	IT Services/Consulting	E	5 - 50
5		FSP5	31-40	11-15	PS Dev	IT Services/Consulting	E	251 - 1000
6		FSP6	61-70	21+	Con	IT Services/Consulting	NA	1 - 4
7		FSP7	41-50	16-20	Con	IT Services/Consulting	E	More than 5000
8	FS2	FSP8	31-40	16-20	Con	IT Services/Consulting	E	More than 5000
9		FSP9	41-50	21+	Con	IT Services/Consulting	E	1 - 4
10		FSP10	61-70	21+	Con	IT Services/Consulting	NA	1 - 4
11		FSP11	51-60	21+	C-Level	Manufacturing	E	1001 - 5000
12		FSP12	51-60	21+	Con	Aviation	E	5 - 50
13	FS3	FSP13	61-70	21+	Con	IT Services/Consulting	E	1 - 4
14		FSP14	51-60	21+	Info Sec	Financial Services	E	More than 5000
15		FSP15	41-50	21+	PS Dev	IT Services/Consulting	E	5 - 50
16		FSP16	51-60	21+	Con	IT Services/Consulting	E	1 - 4
17		FSP17	61-70	21+	Con	IT Services/Consulting	E	1 - 4
18		FSP18	51-60	21+	C-Level	Government	E	1001 - 5000
19		FSP19	51-60	21+	C-Level	IT Services/Consulting	E	5 - 50

The electronic invites for the virtual focus groups were sent two weeks prior to the scheduled date of the focus groups, so that participants could join the session having already read the sequence of topics to be covered by the focus group moderator. The participants were requested to submit a pre-session survey, see Appendix G which consisted of seven (7) demographic questions. Moreover, the first and second focus groups were held in March and April 2021. The focus group agenda was included in the focus group participant invite. The moderator and facilitator for all three focus groups was the researcher. The focus group sessions lasted between 50-65 minutes (depending on the session), and the agenda for them was carefully planned and prepared in advance. Each focus group session started with an overview of the objectives of the study, and with a discussion on how participants should discuss and act during the session. Special emphasis was given to participants ensuring that the participants' opinions should represent the real situation and opinions from their organisational perspective, and that the study organisers guaranteed the confidentiality and anonymity of the discussions. Participants in this study were not anonymous. All the

participants introduced themselves briefly stating their name, current role, and company where they are currently employed. The sessions were audio and video recorded so that transcripts of the sessions could be generated to document all points that were raised. Moreover, the researcher worked as the sessions' moderator.

Lastly, it is important to state that a presentation deck was consistently shared throughout the entire series of the three focus groups with the following agenda: Introductions, Research Motivation, Model Constructs, Research Hypotheses, and DevOps Adoption Leadership (DAL) Model. Notably the research hypotheses that were depicted for evaluation purposes to all focus group participants were as follows:

H1. Practice and Principle adoption positively affects DevOps Adoption Leader Role.
H2. DevOps Metrics adoption positively affects DevOps Adoption Leader Role.
H3. Practice and Principle adoption positively affects DevOps metrics.
H4. DevOps Skills positively affects DevOps Adoption Leader Role.

6.4.2. Model Evaluation

Three techniques were applied to ensure that the research construct in the focus group confirmatory study was valid and in line with research aim, questions, and objectives. Firstly, the content and format of the sessions, and the presentations, was consistently maintained throughout the focus group sessions. Secondly, instrumentation errors were reduced by using the audio and video recordings described earlier. Thirdly, the potential bias in interpreting the results was reduced by having another person review all interpretations made during the analysis.

Informational Phase

There was extensive discussion to encourage knowledge sharing, in terms of which DevOps practices and principles focus groups participants witness as the more prominent ones in their field of work. The questions posed for Informational purposes are IP.1 – IP.5, see Table 6-1. According to the informational phase, the list of DevOps practices include: version control; trunk-based development; issue tracking; automated provisioning; value stream map; Scrum; test automation; static code analysis; Kanban; automated deployment; code coverage; continuous integration; performance monitoring; containerised environments; continuous delivery; and Infrastructure-as-Code.

FSP1 (Chief Transformation Officer, UAE) added that: 'continuous deployment' should be added to the practices whereas FSP3 (Consultant, Netherlands) mentioned that 'there are more practices than principles'. Moreover, the list of DevOps principles mentioned by focus group participants were: organisational culture; employee engagement; monitoring; measurement; automation, which aligns to the CALMS model. FSP2

(C-Level executive, UK) added that: all construct 'definitions should accompany the construct constituents', so there needs to be a common understanding of terms, acronyms, and abbreviations.

Additionally, the list of metrics to DevOps adoption included: critical success factors; key performance indicators; time to market; deployment duration (time); behavioural metrics; knowledge article creation frequency; knowledge article read frequency; feature usage; deployment frequency; deployment size; time to recovery; time to detect (defect); releases frequency per developer per day; lead time between code commit and code deploy. In fact, FSP8 (Consultant, UK) mentioned that 'pirate' metrics should be meticulously considered so that they are not attributed to model constructs. A list of DevOps adoption skills included: technical background; business background; agile management; communication and collaboration; problem solver; talent seeker; negotiator; active listener; change agent; risk manager; multi-cultural mindset; customer-centric mindset; previous experience on transformation; design thinker; certifications; leadership; influencer; holistic systems thinker; strategic thinker; and project manager. FSP7 (Consultant, Greece) mentioned that 'informed influencer' should is an additional skill that should be included.

Lastly, the list of DevOps adoption leader roles, regarded either an individual or team role, which included: C-Level executive; development lead; product owner; architect; and operations lead. FSP10 (C-Level executive, Switzerland) stated that the role of a 'coach' and corresponding skills should also be considered.

Propositional Phase

In terms of questions PP.1 to PP.3, the research hypotheses that were uncovered as part of the design and development process were also established as follows according to Propositional Phase questions, see Table 6-1:

H1. Practice and Principle adoption *positively* affects DevOps Adoption Leader Role.
H2. DevOps Metrics adoption *positively* affects DevOps Adoption Leader Role.
H3. Practice and Principle adoption *positively* affects DevOps metrics.
H4. DevOps Skills *positively* affects DevOps Adoption Leader Role.

Focus Group 1 - Discussion and Findings

FSP7 (Consultant, UK) mentioned that the model construct of 'Practices and Principles' should include Theory of Constraints (TOC). FSP7 also expressed that the lack of a defined Body of Knowledge and what DevOps means can pose a huge challenge in the industry when deciding to adopt DevOps practices and principles. The point that DevOps has a confusing definition introduces unnecessary difficulty for non-IT organisational units to be involved with DevOps adoption initiatives. FSP5 (Consultant, UK) commented that one of the model constructs could be inclusive of challenges faced in terms of attitude-behaviour-

culture (ABC) and that there is an identified willingness by people to contribute towards DevOps practices and principles adoption. FSP4 agreed that ABC is an area where leaders need to be focusing, including the introduction of behavioural metrics that regard enablers for habitual change. FSP4 (Consultant, Netherlands) added that the 'Practices and Principles' model construct could also welcome 'Patterns' and 'Anti-patterns' identified as individual and team behaviour. FSP4 (Consultant, Netherlands) also mentioned that the global industry has become 'framework-fixated' with adoption initiatives focusing on frameworks, e.g. agile, lean and DevOps adoption, without always understanding the fundamental changes that are required in the trinity of shift of mindset-skillset-toolset.

FSP5 (Consultant, UK) commented that 'Business Background' in the DevOps Adoption Leader Skillset model construct, should not infer that the leader should have a business-oriented background per se rather business intelligence to improve expert judgement on the selective nature of how DevOps adoption could work best for the IT organisation initially and, at a later stage, for the entire organisation. FSP2 (Managing Director, UK) added that '*Acumen'* rather than '*Business Intelligence*' could be more appropriate. Moreover, dealing with complexity in the industry, focuses mostly on the 'why' rather than a specific set of skills. In that aspect, '*Coaching*' is an area that is growing in popularity because industry practitioners are sensing its effectiveness on adoption initiatives. On another note, FSP2 (Managing Director, UK) also mentioned that selling ideas in terms of marketing the 'why' an organisation needs to change "*how they are doing what they are doing*". FSP4 (Consultant, Netherlands) agreed to that point and added that being 'Influential' is part of selling the 'why an organisation needs to change'. FSP3 (Consultant, Netherlands) agreed with what the approach of selling the 'why' and added that DevOps Agile Skills Association (DASA) could also help better understand the model and that is also worth noting what the target audience of the model is and what is the sense of purpose behind the model.

FSP2 (Managing Director, UK) stated that a Body of Knowledge (BoK) is still missing for DevOps. FSP10 (Consultant, USA) also mentioned the lack of a DevOps-oriented BoK in the second Focus Group session. FSP4 (Consultant, Netherlands) added that "*everybody has their interpretation of what DevOps is and so developing a BoK can be a huge challenge, if not impossible*". FSP2 (Managing Director, UK) argued that a common direction is now to refer to DevOps as 'Sooner-Safer-Happier'. In this context, 'sooner' refers to the frequency of newly shipped designed and developed features to production environments. Moreover, 'safer' links to the concept of promoting new features, and thus new experiences to customers, in a more predictable, reliable, and stable manner. Lastly, 'happier' relates to the individual and team wellbeing when there is high velocity of features being deployed in production environments without causing major incidents and with customers experiencing promised value (Smart, 2020).

FSP1 (CTO, UAE) commented that 'Technical Background' in the DevOps Adoption Leader Skillset model construct, should be inclusive of different degrees of technical background e.g. strong technical background or 50% business and 50% technical background, etc. Another approach could be to have a dimensional breakdown of capabilities such as technical and background capabilities. FSP3 (Consultant, Netherlands) agreed with what the meaning of words can mean for everyone, separately.

FSP4 (Consultant, Netherlands) stated that the model could be adopted in enterprise-wide fashion as it could benefit the whole software-intensive organisation and its constituent business units and not just the IT organisation unit. However, FSP1 (CTO, UAE) mentioned that the model should not attempt to contribute towards a 'one-size-fits-all' mentality rather it should focus on generalisability with certain constraints that allow flexibility in which manifest variables are selectively chosen for adoption purposes from the model constructs. At the closing parts of the focus group session, FSP2 (Managing Director, UK) added that one example from the industry is the existence of a Chief-Detail-Officer (CDO) whose responsibility is to identify low-cost opportunities with high business impact and promote their design and application to realise that potential high impact.

Focus Group 2 – Discussion and Findings

FSP10 (Consultant, USA) and FSP8 (Consultant, UK) agreed with the construct of 'DevOps Adoption Leader Skillset' being accurate. FSP8 (Consultant, UK) added that the model construct of 'DevOps Adoption Metrics' that indicates types of measurements for the DevOps Adoption Leader, are '*really powerful*' but it is important to know which metrics are relevant to each DevOps-oriented transitional journey within the context of corporate environments. Furthermore, FSP10 (Consultant, USA) added that "Value stream and flow metrics are key".

FSP11 (IT Director, Greece) noted that a DevOps Adoption Leadership model should also be inclusive of '*corporate culture'* and '*corporate environment*' but those two were not apparent anywhere in the current model. FSP11 (IT Director, Greece) added that "*the environment is for me one of the top 3 factors that need constant attention. It is the prime dictator of our behaviours and the key to unlock change*". Additionally, FSP8 (Consultant, UK) added that '*Continuous Deployment (CDep) as well as, or instead of, Continuous Delivery (CDel) is needed, as CDep forces teams to build robust test suites, whereas CDel can allow greater reliance on manual testing as the go-live decision has not been automated*". FSP10 (Consultant, USA) mentioned that in the Lean IT industry-oriented organisation engagements there is a new emerging trend – that of '*Coaching*' and coach-like behaviours by leaders. FSP12 (Aviation, Greece) added that '*Coaching*' is a skill that is "*more important even than leadership skills*". FSP10 (Consultant, USA) added that in a lean environment there are the '*Production*' and '*People*' systems, and agreed with the moderator that

People Development Systems (PDS) could be an appropriate addendum to the model constructs. FSP9 (Consultant, UK) agreed that "*the place DevOps adoption often falls down is when the Enterprise, as a whole, views DevOps as an "IT thing". In reality, DevOps only truly works when its practices are applied globally*".

FSP9 (Consultant, UK) agreed with the entire set of the model's hypothesis and mentioned that DesignOps, and its conference, is a new term as an attempt to enlarge the worldwide influence of DevOps. FSP10 (Consultant, USA) added that discussing about '*mindfulness*' and starting with a 3-minute guided meditation of mindfulness can help overcome silo mentality and communication barriers that cognitive overload is causing.

FSP10 (Consultant, USA) also noted that leaderships' behaviours are key to unlocking desired workforce attitudes, habits, and behaviours. They also signal the imitation and ultimately reinforcement of habitual change. FSP11 (IT Director, Greece) stated that the model in its entirety is truthful and possesses a strong linkage to adoption of change and organisational change management and characteristically mentioned that "*it is all about change - unless you are prepared for it there is no destination*". Moreover, FSP11 (IT Director, Greece) agreed to a point made by the group that leadership certifications cannot certify the ability to lead. FSP12 (Aviation, Greece) further commented that the dyadic relationship of leader-follower within organisations needs to promote bilateral communication and collaboration. Moreover, '*Psychological Safety'* also plays a significant role in how the DevOps Adoption Leader will nurture targeted individual and team wellbeing outcomes. In the closing remarks FSP10 (Consultant, USA) noted that "*a leader is both born and made - so ability to accept change and criticism goes hand in hand with practice and theories about transformation and lean*". Lastly, FSP9 noted that "*Finding good generalists for this kind of DevOps Adoption Leader position can be really hard*".

Focus Group 3 - Discussion and Findings

FSP16 (Consultant, UK) noted that the DevOps Practices and Principles model construct, could contribute to the challenges of DevOps adoption if imposed upon a workforce. FSP13 (Consultant, UK) was particularly inquisitive in terms of why does 'DevOps' need leadership in its transitional initiatives. The moderator explained that research-based data from the industry that the DevOps adoption role is required in its individual or team role existence and the annual State of DevOps report reinforces that to include transformational leadership traits. FSP13 (Consultant, UK) commented that a lot of the efforts focus on organisational change management and setting motivational factors to employ new practices and principles. FSP18 (CIO, UK) mentioned that ITIL and agile have introduced practices and principles evident on the model constructs including 'behavioural metrics' which are immensely significant for organisation-wide

DevOps adoption in the context of individual and team wellbeing; i.e., being safer-happier while improving individual and team productivity through the adoption of new set of practices and principles.

FSP17 (Consultant, Switzerland) commented that the model constructs which represent building blocks with one-directional correlation or causality should be bi-directional in order to better understand positive and negative relationships between model constructs and their manifest variables. The moderator provided guidance on PLS-SEM analysis that examines this bi-directional approach. FSP15 (Product Manager, Georgia) agreed that the DevOps Adoption Leader Role should exist and that a formalised model similar to the one depicted is required. FSP14 (Information Security Director, Greece) and FSP19 (CEO, Greece) agreed to the comments made on the model's structure and constituent model constructs and manifest variables. Additionally, there is a matrix that is required in terms of the job role responsibilities and accountabilities to show that the model can bring value within enterprise environments.

FSP16 (Consultant, UK) extended on previous thoughts to add that experience has shown that it is necessary to avoid a 'push-based' model being introduced in top-down fashion by the management to the workforce, as opposed to outcome-oriented, invitation-based and / or continuous approach in which solutions emerge from the process. As long as, the aforementioned two factors are accounted for the model is towards the right direction. Additionally, FSP16 (Consultant, UK) mentioned that common purpose and a 'north-star vision', and the 'why we're doing this' understanding, is more important than metrics or the behaviours instilled by metrics. FSP13 (Consultant, UK) agreed to the aforementioned points adding that "*anything that introduces structure should be inclusive of dependency reduction*" and 'organisational governance' is required regardless of labelling it 'DevOps', 'agile', etc.

Analytical Phase

In the context of questions AP.1 – AP.4 (see Table 6-1). The first focus group participants highlighted that without specific construct definitions the model is not understood on a common ground by all participants. Particular attention is required to achieve an acceptable level of common understanding of what the model is attempting to achieve and 'why'. Participants of the third focus groups mentioned that the type of practitioners to whom the conceptual model is most meaningful should also be identified. Moreover, the model's complexity level can be improved to add clarity to interdependencies between constructs.

Evaluative Phase

In regard to questions EP.1 and EP.2 (see Table 6-1), among the three focus group participants there was consensus on the converging notion that there are certain key factors that need to be addressed to increase the conceptual model's validity, namely 1) the definition of all construct constituents, 2) the definition of interferences and relationships between constructs to allow a rational flow, 3) avoidance of technology-specific terms, and 4) a focus on the human side of DevOps.

Technology Transfer Phase

The research study considered questions TP.1 and TP.2 (see Table 6-1), which aims to gain an improved understanding of whether the conceptual model can be transferable to practical situations. FSP9 (C-Level executive, UK) added that all the model constructs are valid and applicable to real life corporate environments. However there needs to be certain amendments to reflect a more accurate application with improved DevOps adoption leadership outcomes. The discussions yielded insightful perspectives for all model constructs. All in all, there was consensus on the inclusion of the conceptual model constructs presented during the focus groups and agreement levels ranked commonly above 70% in terms of total average, see Table 6-3.

Table 6-3 Focus Groups Participant Agreement Levels for Conceptual Model.

Focus Group	Participant ID	Practices and Principles	DevOps Adoption Leader Role	DevOps Adoption Metrics	DevOps Adoption Leader Skillset
First	FSP 1	**60%**	**100%**	**100%**	**100%**
	FSP 2	**80%**	**90%**	**80%**	**80%**
	FSP 3	**30%**	**30%**	**30%**	**30%**
	FSP 6	**80%**	**80%**	**60%**	**100%**
	FSP 7	**70%**	**60%**	**70%**	**60%**
	FSP 8	**50%**	**70%**	**30%**	**70%**
	FSP 9	**80%**	**80%**	**80%**	**100%**
Second	FSP 10	**90%**	**100%**	**90%**	**90%**
	FSP 11	**90%**	**100%**	**90%**	**100%**
	FSP 12	**90%**	**90%**	**90%**	**90%**
	FSP 13	**100%**	**100%**	**100%**	**100%**
	FSP 14	**50%**	**50%**	**50%**	**50%**
Third	FSP 15	**70%**	**70%**	**70%**	**80%**
	FSP 16	**80%**	**50%**	**30%**	**70%**
	FSP 17	**60%**	**70%**	**50%**	**70%**
	FSP 18	**100%**	**100%**	**100%**	**100%**
	FSP 19	**100%**	**100%**	**100%**	**100%**
Degree of Participant Agreement (mean value)		**75%**	**79%**	**72%**	**82%**

A majority agreement was also indicated concerning the value the model possesses for DevOps adoption guidance purposes. Evidently there is wide agreement with all the model constituent areas in the context of DevOps adoption leadership.

The model validation can be further evaluated in terms of frequency (number of times a phrase is repeated), specificity (looking to uncover specific details of liking or disliking e.g. specific model aspects, emotion (give weight on reactions of participants that show enthusiasm, passion or intensity in their answers) and extensiveness (the frequency with which a phrase is repeated by not only one but a wider set of individuals) - adapted from Krueger and Casey (2015).

Seventeen (17) out of nineteen (19) industry practitioners, all of whom have possessed leadership roles in the past, provided responses through a focus group in-session survey; allowing the researcher to capture detailed feedback on specific model constructs. The developed model's acceptance levels were : 75% for a specific set of practices and principles, 73% for DevOps adoption planning, 79% for the need of a DevOps adoption leader role, 72% for a set of DevOps adoption metrics and 82% for a specific attributed leader skillset. Moreover, the same level of acceptance was confirmed and reaffirmed for the following hypothesis that were identified following the exploratory study of thirty interviews.

H1. DevOps Practice and Principle adoption *positively* affects DevOps Adoption Leader Role.

H2. DevOps Adoption Metrics *positively* affects DevOps Adoption Leader Role.

H3. DevOps Practice and Principle adoption *positively* affects DevOps Adoption Metrics.

H4. DevOps Adoption Leader Skills *positively* affects DevOps Adoption Leader Role.

6.5. Focus Group Data Collection Process

Data collection and analysis was aggregated in order to answer the research questions which were mapped to focus group registration and in-session survey' questions, see Table 6-4. The whole list of registration survey questions is available in Appendix G, and focus group in-session questions are available in Appendix H.

Table 6-4 Research questions to focus group survey questions mapping.

Research Question	**Interview Question**
Data collection for segmentation purposes	All focus group registration survey questions i.e. 1 – 9
RQ1) Which agile, lean and DevOps practices and principles can improve productivity for software product development teams in software-intensive organisations that have adopted a structured service management approach?	No relevant questions
RQ2) Are DevOps-oriented environments and their teams an extension or a replacement of process-driven structured service management approaches?	No relevant questions

RQ3) A) Can Leadership affect DevOps adoption within a software-intensive organisation? B) What is the leadership style that can be attributed to the DevOps adoption leader role?	All questions of focus group in-session survey i.e. 1 – 9

6.6 Chapter Summary

This chapter has considered the practical steps to examine key concepts derived from Chapters Four and Five in a series of three focus groups which evaluated the constructs and manifest variables that constitute the DevOps Adoption Leadership model. The high degree of agreement on model constructs by the focus groups participants is discussed which reaffirms the significance of the set of hypotheses supported in Chapter Five. The consensus reached on model evaluation, serves as a baseline for a set of guidelines for software-intensive organisations willing not only adopt and embrace DevOps practices and principles, skills with specific measurements in place but to lead the adoption journey too.

Chapter Seven
Conclusions and Future Work

7.1. Research Contribution

The holistic set of seven chapters in this thesis present research undertaken to explore, analyse, deduce, and model aspects of the synopsis of research considerations presented in section 2.7. This closing chapter offers an overview of all covered research areas involved in the storyline described in all aforementioned chapters. Moreover, the following subsections outline the theoretical, methodological, and practical contributions of the current research, and consider future work aspirations, which the researcher hopes will provide guidance to DevOps community researchers.

7.1.1 Theoretical Contribution

The research has reviewed and analysed theories, models of legacy service management, agile software development, lean IT and DevOps practices and principles. The research reviews led to a thorough investigation of the relationship between software product development team culture, behaviours and habits, and the design and development of a novel adoption model to lead DevOps adoption of new practice and principles. Consequently, a key theoretical research contribution is the definition of a novel approach to DevOps adoption leadership. In a vast corporate landscape, which is increasingly deciding to embrace the transition towards a DevOps oriented environment, there is a very limited set of models to show how to support DevOps adoption efforts by software product development teams which are committed to be receptive of a leadership style. This thesis aimed to close that gap, in the attempt to produce a validated and evaluated model that synthesises an approach to adoption of practices and principles, appropriate metrics and pertinent skills for the DevOps adoption leader role.

7.1.2. Methodological Contribution

The thesis aimed to develop an approach to design, build and model the capability for software product development teams to adopt DevOps practices and principles and lead its adoption through a specific set of metrics and skillset. Such guidelines were extracted from a validated model and can be reused by industry practitioners and organisations - define the DevOps leadership role in the context of software product

development lifecycle. This approach can help software developers, software testers, information security officers and operational roles to focus on developing the growth mindset to be inclusive of new behaviours and habits that trains the workforce to transform a challenging right-brain task into a routine left-brain task - thus achieving the process of unlearning and relearning a software product development skillset. This consistent effort of skillset mutation thus leads to the shift of mindset from legacy and traditional highly structured approaches, in service management, to the enablement of agility and leanness in everyday software product development activities.

7.1.3. Practical Contribution

Currently, there is limited peer-reviewed and/or published validated guidance to support a leadership adoption model for software product development teams in the context of DevOps practice and principles. Moreover, there isn't any indication of the leadership style and traits that can be expected to be part of the DevOps adoption leader's character. This thesis aimed to support the DevOps adoption transitional efforts, and commitment of software-intensive organisations - to enhance the competence level of an organisation's adoption capability, guide DevOps adoption leadership through its upskilling journey and achieve the cultural shift of mindset to enable habitual change.

7.2. Research Aim and Contributions

The previous chapters have presented an in-depth research analysis, research design, empirical research-based data collection, model development and its validation/evaluation; having in mind consistently the research aim as outlined in Section 1.5, i.e. *To understand the effect that Agile, Lean and DevOps practice and principle adoption have on structured service management processes and how it is possible to lead adoption efforts through a specific model design for product development teams in software-intensive organisations.*

The data collected from a series of interviews and participating industry practitioners, indicate a clear list of specific agile, lean and DevOps practices and principles that regard an extension to structured service management approaches and are relevant to DevOps adoption theory. The main findings associated to the research questions are shown in Table 7-1.

Table 7-1 Research questions to findings mapping.

ID	Research Question	Findings
RQ1	Which agile, lean and DevOps practices and principles can improve productivity for software product development teams in software-intensive organisations that have adopted a structured service management approach?	Specific agile, lean and DevOps practices such as: 1) organisational culture; 2) monitoring / measurement; and 3) automation, are crucial in the software development lifecycle. Specific agile, lean and DevOps principles - such as: 1) SCRUM; 2), Kanban; and 3) Continuous Delivery - are crucial in the software development lifecycle. DevOps practices and principles adoption are challenged due to poor communication and information flow, deep-seated company culture and operations not being involved in the requirements specifications. DevOps practice adoption should be extended in an enterprise-wide fashion (87%), with team structure based on existing Development (97%), Operations (97%), Quality Assurance (93%), and Information Security (80%) teams.
RQ2	Are DevOps-oriented environments and their teams an extension or a replacement of process-driven structured service management approaches?	The set of service management processes that continue to form a strong part of DevOps-oriented structures are Change Management, Service Portfolio Management (including Service Catalogue Management), Release and Deployment Management and Service Level Management
RQ3	a) Can Leadership affect DevOps adoption within a software-intensive organisation? b) What is the leadership style that can be attributed to the DevOps adoption leader role?	There is overwhelming consensus that a DevOps leadership role should exist (86%) and that the role should carry a continuous effect not a project-based. There certain leadership styles, such as transformational and servant characteristics - that share commonalities with leadership traits witnessed in DevOps adoption transitional journeys within the software-intensive industry. A designed and developed DevOps Adoption Leadership Model (DOAL) analysed through PLS-SEM and validated through three Focus Groups also attempts to provide clarity and baselined on empirically collected research-based datasets.

7.3. Discussion on Research Aim and Contributions

The research aim, outlined in Section 1.5, states that this research aims to "*understand the effect that Agile, Lean, and DevOps practice and principle adoption have on structured service management processes and how it is possible to lead adoption efforts through a specific model design for product development teams in software-intensive organisation.*" Chapter Two provided a detailed account of agile software development, lean IT, and DevOps practices and principles, benefits, and challenges adoption. The detailed accounts of research-based evidence of agile software development, lean IT and DevOps practices and principles indicated that there is evidence to support productivity improvements for software product development teams willing to shift from structured service management or 'waterfall' product development approaches towards DevOps (RQ1). Additionally, evidence indicated that it was unclear whether a process-driven structured service management approach can deliver benefits to a DevOps-oriented environment and its teams (RQ2). There is also very limited evidence to indicate a relationship between how Leadership affects DevOps adoption within a software-intensive organisation, and which leadership style is more attributable to a potential DevOps adoption leader role (RQ3). Therefore, to establish the link between structured service management approaches with agile, lean and DevOps approaches an exploratory study was designed which carefully considered aspects of that relationship. In addition, the exploratory study was inclusive of leadership aspects present in DevOps adoption which is, currently, lacking research community publications.

The research-based data has served as evidence to indicate that DevOps practice and principle adoption maintained strong linkage to agile and lean practice and principle adoption; i.e. identified by conducting thirty (30) interview with participants from private and public sectors in the EMEA region. In addition, the evaluation of a survey completed by 250 participants, of which 76% have held previous leadership positions further enhanced the linkage of DevOps, agile and lean practices, and principles. Moreover, a mixed methods approach was used. The thirty (30) interviews generated coded themes to expand our understanding of relevant factors – from most to least recurring in interview transcripts: DevOps leadership, practice and principle adoption, employee culture, product development, skills.

The data was collected from a series of interviews and a survey indicate a clear list of specific agile, lean, and DevOps practices and principles; including leadership characteristics, which form a crucial part to DevOps adoption theory and is organised according to the study's research questions.

7.3.1 Challenges in Transitional Journeys of DevOps Adoption

The analysis and evaluation of interviews indicated several factors that slow down DevOps adoption such as: communication barriers; lack of cross-functional collaboration; lack of senior management buy-in; lack of leadership; lack of cross-functional leadership; lack of enterprise-wide DevOps adoption; plethora of IT systems coupled with numerous IT support roles; lack of cross-functional collaboration. A recent systematic literature review indicates that lack of communication and collaboration is still considered the most significant issue at hand with DevOps adoption transformational initiatives within organisations (Khan et al., 2022).

The interview participants established that the cultural behaviour that poor distinctions of responsibilities, especially in terms of "us" and "them", is immensely detrimental to what the cross-functional DevOps teams and cross-functional DevOps leaders aim to achieve. In essence, this inhibitor leads to DevOps enterprise-wide adoption facing failure from the off start of such an initiative, implying that it is important to first let the cultural character within the IT organisation take form and shape and then aim for adoption at a wider scale, outside the IT organisation. To that extent, the interviews showed that Human Resources departments can be a first step outside the IT organisation where DevOps adoption can contribute in terms of shift of culture-skillset-toolset. Simply put, as one interviewee stated, "Leadership skillset is the most important thing to adoption barrier breakdown". In addition, the set of inhibitors identified could have a direct cause of exacerbation from the perspective of the Human Resources department; utilising a rudimentary selection approach that qualifies new hires based on the right toolset experience without considering mindset and skillset-specific aspects falls short of DevOps-oriented team structure expectations. Thus, this selection process could insinuate that IT teams that fail or partially fail to adopt DevOps practices and principles are because the transition to the right mindset e.g., embrace continuous experimentation, cross-collaboration between development, operations, quality assurance and information security teams, etc. and skillset is simply, under-developed. There are findings in the survey to indicate that talent seeking is not considered an important characteristic of the DevOps leader, since this is a responsibility area normally covered by Human Resources. Therefore, the perception that DevOps teams and their leaders should not engage or engage minimally with talent seeking opportunities could affect the future staffing of those teams.

7.3.2 Leadership Styles that Enable DevOps Adoption

From the 250 survey participants, 81% have held 10+ years of professional experience and 76% have held a leadership position. Furthermore, the survey participants indicated by 67% that a new practice and principle adoption leadership role should exist for transformation initiatives; i.e. that the C-Suite should be the direct report of the DevOps leader. The top leadership skills identified are: Communication and

collaboration; Active listening; Customer-centric mindset; Technical background; Problem solver; Technical background; Multi-cultural mindset; Influential; Agile management; Strategic thinking; Project management. A recent systematic literature review indicates that lack of skills and knowledge is still considered the second most significant issue at hand with DevOps adoption transformational initiatives within organisations (Khan et al., 2022).

The results obtained from the survey participants shed more light on the already established beliefs extracted from the interview participants. For instance, there was strong indication by interview participants that a shift of skillset towards acquiring, developing and applying more soft skills is necessary to achieve new practice and principle adoption - in this case agile, lean, and DevOps. In fact, communication, and collaboration as well as customer-centric or even customer-obsessed mindset is an extension to that viewpoint. Another example shoots from the technical and / or business backgrounds that could play a role in DevOps adoption leadership. Ever since the term "DevOps" was coined back in 2009, the worldwide IT and business community have come to an assumingly obvious realisation; "DevOps" is associated to the IT organisation and that is where it stays. This belief seems to reflect in the survey findings where possessing a technical background is more important than a business background by as much as 15% in the "Strongly Agree" category. However, the survey findings also suggest that possessing a business background is beneficial to a certain extent; with interview participants stating that a balanced background is preferable to technical-only or business-only. The least important DevOps leadership characteristics were: gaining a relevant certification, design thinking, previous experience on transformation projects, and a talent seeking competence.

Survey results indicate that certification was, by a considerable degree, the least preferred characteristic for the DevOps leader. Although there is availability of DevOps leader certifications e.g., DevOps Leader (DOL) certification, by the DevOps Institute, it seems that the desire to become certified in DevOps leadership is not regarded as being an important characteristic or requirement. In addition, design thinking which entails observation, insights generation, ideation, prototype, and testing for product development purposes was clearly not considered a crucial characteristic or requirement. Furthermore, previous experience of transformation projects did not yield any connection to DevOps leadership. The authors' intent was to investigate a finding from the interview series, where there was an indication that constant coaching by an external entity, e.g., consultant, is required (although not always) to sustain transformation initiatives. However, most of the time, the IT organisation cannot sustain newly adopted practices in their structure, and default to the "old habits of working", which could suggest that an individual with previous experience on transformation projects would know how to avoid a similar situation in the transition process to DevOps practice and principle adoption.

Extending the findings, the characteristics that transformational and servant leadership in DevOps-oriented environments is highlighted in Table 7-2. There seems to be an intersection whereby the dimensions of active listening and empathy are highlighted in transformational and servant leadership as well as the results of this study.

Table 7-2 Characteristics of Leadership Styles.

Transformational Leadership (Greenleaf, 2002; Bass and Riggio, 2006)	Servant Leadership (Greenleaf, 2002)	DevOps Adoption Leadership (Maroukian and Gulliver, 2020)
• Idealised influence (realistically self-confident, determined, unconventional) • Inspirational motivation (articulate, flexible, emotional, perspicacious) • Individualised consideration (caring, empathetic, relations-oriented) • Intellectual stimulation (rational, unconventional, perspicacious)	1. Empathy 2. Active Listening 3. Healing 4. Awareness 5. Persuasion 6. Conceptualisation 7. Foresight 8. Stewardship 9. Commitment to the growth of people 10. Building community	1. Communication and collaboration 2. Active Listening 3. Customer-centric mindset 4. Technical background 5. Problem solving 6. Multi-cultural mindset 7. Influential 8. Agile management 9. Strategic thinking 10. Project management skills

7.3.3 Key metrics for DevOps adoption leadership

During the survey, participants indicated that DevOps adoption leadership practices should still be governed by traditional approaches, such as critical success factors and key performance indicators and time-to-market. However, agile and lean metrics formed a significant part of the wider picture with the most popular being: mean-time-to-market; deployment frequency; deployment duration; behavioural metrics; time-to-detect-defect; mean-time-to-recover; and feature usage.

Consideration of DevOps-oriented metrics – i.e., mean-time-to-market, deployment frequency, deployment duration, behavioural metrics, time-to-detect-defect; mean-time-to-recover; and feature usage - indicates software product development measurements that can be applied to a DevOps team structure and the DevOps leadership role. From the cultural perspective, time-to-detect-defect can refer to behaviour that aims to increase knowledge sharing in cross-functional fashion, the frequency that a leader performs one-to-ones with DevOps teams (and their members) to understand what is on top of mind.

Moreover, feature usage is an emerging practice for DevOps adoption, and it regards monitoring usage of a released product feature in a production system environment; and whether performance is as expected. Lastly, the vast majority of respondents agreed that the leadership role should be associated, and have ownership of the aforementioned metrics.

Presently, the conclusion is that DevOps adoption leadership is an interdisciplinary requiring a specific set of competencies and capabilities built on a set of DevOps practices and principles. The leadership approach of the organisational structure is vital to the level of resistance exhibited by IT professionals during the transitioning period from a highly structured software product development approach to DevOps. Additionally, the transitional phase of DevOps adoption requires an individual to lead DevOps teams which leads to the belief that DevOps has a substantial leadership component at its transitional level. Moreover, it although there is a distinction in how DevOps adoption leadership is conceptually overlapping with other forms of value-based leadership approaches such as authentic and inclusive leadership. The challenge is to empirically test how it differs from other forms of leadership as well as cross-culturally.

7.4 Future Work

DevOps adoption leadership, and its relationship to software product development teams, has become a vastly popular research topic. The researcher intent is to maintain focus on the analysis and evaluation of presently collected research data and to provide further insights relative to current findings in order to witness which leadership styles can become part of the transitional journey of organisations towards DevOps practice and principle adoption. The organisational change required to achieve a successful state of a DevOps-oriented environment in today's global market, raises challenges, and with them resistance factors, in terms of shift of mindset, skillset, and toolset that needs to be achieved. The effects of the change need to be continuously monitored to identify the link to the shift of the triage experienced. With this in mind, one of the future research aims could be to invite and/or select IT practitioners with prior and / or current Open Source Software (OSS) experience and attempt to identify whether similar patterns exist in the OSS developer community.

The outcomes of this thesis can be further evaluated and reused by practitioners in software-intensive organisations willing to introduce a DevOps orientation, i.e., in terms of practices and principles adoption in the product development lifecycle. The research can be extended in the future to explore more of the different facets of leadership style(s), capabilities, skills, and competencies required in the context of continuous DevOps adoption. In particular, a question that could be posed, is to explore 'why' specific

leadership styles have a higher degree of impact on the performance of DevOps teams within an organisation and whether there are variations in that impact degree for low-, medium-, and high-performing DevOps-oriented organisations; who are either now starting on their transitional journey to DevOps or are exploring ways to improve their already impactful experiences having adopting DevOps practices and principles.

It is becoming clearer that DevOps or its various branches of Enterprise DevOps or scaled DevOps adoption will receive deeper meaning when accommodating a 'developer velocity' mindset. Cognitive Load Theory (CLT) can be an research-topic to be closely examined in relevance to teams norming, forming, storming and performing in the software product development industry. Additionally, future research could focus in gaining more insights on the extent of influence posed on DevOps teams and their leadership role due to cognitive load.

Lastly, the global health crisis (covid-19), has shifted the working experience towards an increasingly virtual setting, whereby a 'phygital' (merging of words 'physical' and 'virtual') approach has become part of the 'new normal'. For instance, team member colocation; one of DevOps practices, for software product development, operations, quality assurance and information security teams is no longer occurring just in its pre-pandemic physical format. As long as the "remote work" paradigm is enforced in the global software product development community teams, even in a 'remote work' or 'work from anywhere' or 'hybrid' approach, the characteristics, traits, and styles of leadership relevant in that new setting can potentially effect DevOps adoption, and become part of future research consideration.

References

Agutter, C., 2017 “VeriSM™ - A service management approach for the digital age”, Van Haren Publishing, The Netherlands.

Adidas, (2020) DevOps Maturity Framework. https://github.com/adidas/adidas-devops-maturity-framework, Accessed March 25, 2022.

Åhlström, P., 1998. Sequences in the implementation of lean production, European Management Journal, 1998, vol. 16, issue 3, 327-334.

Alahyari, H., Gorschek T. and Svensson R. B., “An exploratory study of waste in software development organizations using agile or lean approaches: A multiple case study at 14 organizations,” Information and Software Technology, vol. 105, 78-94, Jan. 2019, doi.org/10.1016/j.infsof.2018.08.006.

Almeida, F. (2017). Challenges in Migration from Waterfall to Agile Environments.

Anderson, J., “Kanban: Successful Evolutionary Change for Your Technology Business”, 2010, Blue Hole Press, USA.

Antonakis, J., Atwater, L. (2002) “Leader distance: A review and a proposed theory”, The Leadership Quarterly. 13 (2002), 673-704.

Avolio, B. J., Bass, B. M. (2004) "Multifactor leadership questionnaire manual and sampler set", 3rd ed., Palo Alto, CA: Mind Garden.

AXELOS, "ITIL®4 Managing Professional High Velocity IT," The Stationery Office, London, UK, 2019, ISBN: 9780113316403.

AXELOS, “Managing Successful Projects with PRINCE2®, *TSO (The Stationery Office)*, London, UK. 2017, ISBN: 9780113315338.

Bader, G. E., & Rossi, C. A. (1999) "Focus groups: A step-by-step guide". San Diego, CA: The Bader Group.

Bass, B. M., Avoilio, B. J. (1994) “Improving organizational effectiveness through transformational leadership”, Thousand Oaks, CA: Sage Publications.

Bass. B. M. (1998) “Transformational Leadership: Industrial, Military, and Educational Impact”, Mahwah, NJ: Lawrence Erlbaum Associates.

Bass, B.M, Riggio, R.E., (2005) "Transformational Leadership: A Comprehensive Review of Theory and Research, Psychology Press, 2nd ed., ISBN:978-0805847628. Beck K., "Extreme programming explained: embrace change," Addison-Wesley, Don Mills, Ontario, Canada.

Bass, L., Weber, I.O., Zhu, L. (2015) “DevOps: A Software Architect's Perspective”, Addison Wesley, US.

Belhadi, A., Touriki, F.E., El Fezazi, S., (2016) “A Framework for Effective Implementation of Lean Production in Small and Medium-sized Enterprises”, Journal of Industrial Engineering and Management. http://dx.doi.org/10.3926/jiem.1907.

Bezemer, C.P., et al., (2019) "How is Performance Addressed in DevOps?" Proceedings of ACM/SPEC International Conference on Performance Engineering (ICPE), Association for Computing Machinery, New York, NY, USA, 45–50, doi.org/10.1145/3297663.3309672.

Beyea, S. C., & Nicoll, L. H. (2000) "Methods to conduct focus groups and the moderator's role". AORN Journal, 71(5).

Bhattacherjee, A. (2001) Understanding Information Systems Continuance: An Expectation-Confirmation Model. MIS Quarterly, 25, 351-370. https://doi.org/10.2307/3250921

Bhattacherjee, A. and Premkumar, G. (2004) Understanding Changes in Belief and Attitude toward Information Technology Usage: A Theoretical Model and Longitudinal Test. MIS Quarterly, 28, 229-254.

Braun V., Clarke, V. (2006) Using thematic analysis in psychology, Qualitative Research in Psychology, 3:2, 77-101, DOI: 10.1191/1478088706qp063oa

Braun V., 2021. Thematic Analysis: A Practical Guide. Sage

Bryman, A., Bell, E. (2018) Business Research Methods, 5th ed., OUP Oxford.

Cacciattolo M. (2015) Ethical Considerations in Research. In: Vicars M., Steinberg S., McKenna T., Cacciattolo M. (eds) The Praxis of English Language Teaching and Learning (PELT). Critical New Literacies (The Praxis of English Language Teaching and Learning (Pelt)). SensePublishers, Rotterdam. https://doi.org/10.1007/978-94-6300-112-0_4

Carvalho, B.V., Mello, C.H.P. 2011. "Scrum agile product development method -literature review, analysis and classification", DOI: 10.4322/pmd.2011.005.

Clayton, I., (2008) "Guide to the Universal Service Management Body of Knowledge", Service Management 101.

Crotty, M. (1998) "The Foundations of Social Research: Meaning and Perspective in the Research Process" London: SAGE Publications Inc.

DASA DevOps Competence Model – DevOps Agile Skills Association (DASA), https://www.devopsagileskills.org/dasa-competence-model/. Accessed March 25, 2022

De França, B.B.N., Jeronimo H., Travassos, G.H., "Characterizing DevOps by hearing multiple voices," Proceedings of the 30th Brazilian Symposium on Software Engineering (SBES), Association for Computing Machinery, New York, 53–62, 2016.

DevOps Institute. (2022) DevOps Collective Body of Knowledge. Available from: devopsinstitute.com/resources, Accessed March 25, 2022.

DevOps Podcast (2015) The Goat Farm - The Podcast for DevOps in the Enterprise. "We don't need no manifesto", https://goatcan.do/2015/01/16/we-dont-need-no-manifesto/ , Accessed March 25, 2022.

Dinibutun, S., R., 2020, Leadership: A Comprehensive Review of Literature, Research and Theoretical Framework, January 2020, Journal of Economics and Business 3(1):44-64 Follow journal, DOI: 10.31014/aior.1992.03.01.177.

Dingsøyr T., Lassenius, C., "Emerging themes in agile software development: Introduction to the special section on continuous value delivery," Information and Software Technology, 56-60, 2016.

Drews, P., Schirmer, I., Horlach, B., Tekaat, C. (2017) "Bimodal Enterprise Architecture Management: The Emergence of a New EAM Function for a BizDevOps-Based Fast IT", IEEE 21st International Enterprise Distributed Object Computing Workshop (EDOCW), 57-64.

Dyck A., Penners R., Lichter H. (2015) "Towards Definitions for Release Engineering and DevOps", IEEE/ACM 3rd International Workshop on Release Engineering.

Eva, N., Robin, M., Sendjaya, S., Dierendonck, D.V., Liden, R.C., 2019. Servant Leadership: A systematic review and call for future research. The Leadership Quarterly, 30, 1, 111-132.

European Commission, (2003) "Concerning the definition of micro, small and medium-sized enterprises", https://eur-lex.europa.eu/eli/reco/2003/361/oj, Accessed March 25, 2022.

European Union, (2018) "Complete Guide to EU GDPR Compliance", https://gdpr.eu/. Accessed March 26, 2022.

Feitelson, D., Frachtenberg E., Beck, K., "Development and Deployment at Facebook," IEEE 1089-7801/13, 2013.

Fitzgerald, B., Stol, K., (2014). Continuous software engineering and beyond: trends and challenges. *RCoSE 2014*.

Fitzgerald, B., Stol, K., (2017). Continuous software engineering: A roadmap and agenda. Journal of Systems and Software, Volume 123, pp. 176-189, ISSN 0164-1212,doi.org/10.1016/j.jss.2015.06.063.

Flick, U. (2020) "Introducing research methodology: A beginner's guide to doing a research project", SAGE Publications, 3rd ed.

Forsgren, N., Humble, J., (2018) "Accelerate: The Science of Lean Software and Devops: Building and Scaling High Performing Technology Organizations", Trade Select.

Fowler M., "Refactoring: improving the design of existing code," Addison-Wesley, Don Mills, Ontario, Canada, 1999.

Geurts W. J. W., "Faster is Better and Cheaper," Wiley Online, vol. 26, 1002-1015, Jul. 2016, doi.org/10.1002/j.2334-5837.2016.00207.x.

Goldratt, E., "Theory of Constraints and How it Should be Implemented," North River Press, 1994.

Goles, T., Hirschheim, R. 2000. The paradigm is dead, the paradigm is dead… long live the paradigm: the legacy of Burrell and Morgan. Omega, 28, 249-268.

Green, J. and Thorogood, N. (2009) Qualitative methods for health research. Sage, London.

Greenleaf, R.K., “Servant leadership: A journey into the nature of legitimate power and greatness”, Paulist Press, 2002, ISBN: 978-0809105540

Gruver, G., "Start and Scaling DevOps in the Enterprise," Bookbaby, 2016.

Gottesheim, W., (2015) “Challenges, Benefits and Best Practices of Performance Focused DevOps”. In Proceedings of the 4th International Workshop on Large-Scale Testing (LT '15). Association for Computing Machinery, New York, NY, USA, 3. DOI:https://doi.org/10.1145/2693182.2693187

Guest, G., Bunce, A., Johnson, L. (2003) How Many Interviews Are Enough?: An Experiment with Data Saturation and Variability, SAGE Publications, Inc.

Hair, J.F., Hult, G.T.M., Ringle, C., Sarstedt, M. (2014) A Primer on Partial Least Squares Structural Equation Modeling (PLS-SEM), 2nd ed., SAGE Publications, Inc.

Hair, J.F., Sarstedt, M., Ringle, C.M. et al. An assessment of the use of partial least squares structural equation modeling in marketing research. J. of the Acad. Mark. Sci. 40, 414–433 (2012). https://doi.org/10.1007/s11747-011-0261-6

Hair, J.F., Black, W., Anderson. R., Babin, B., (2019) Multivariate Data Analysis, 8th ed., Cengage Learning EMEA.

Hair, J.F., Sarstedt, M., Ringle, C., Ringle, M., 2018. When to use and how to report the results of PLS-SEM. European Business Review, Vol. 31:1.

Harvard Business Review. (2017) The Countries Most (and Least) Likely to be Affected by Automation, https://hbr.org/2017/04/the-countries-most-and-least-likely-to-be-affected-by-automation , Accessed March 25, 2022.

Harvard Business Review, M. Harris, B. Tayler. (2019) Don’t Let Metrics Undermine Your Business https://hbr.org/2019/09/dont-let-metrics-undermine-your-business, Accessed March 25, 2022

Herring, M., "DevOps for the Modern Enterprise," IT Revolution, Portland, Oregon, 2018.

Herring, M., DeGrandis, D., Forsgren N., Guckenheimer, S., "Measure efficiency, effectiveness and culture to optimize devops," IT Revolution, Portland, Oregon, 2015.

Horlach, B., Drews, P., Schirmer, I., and Boehmann, T. (2011) "Increasing the Agility of IT Delivery: Five Types of Bimodal IT Organization," Hawaii International Conference on System Sciences, USA.

Hoving, W., Bon, J.V., (2012) "Integrated service management", Academic service; 1st ed.

Humble J., Molesky, J., "Why enterprises must adopt devops to enable continuous delivery," Cutter IT Journal, vol. 24(8), 6-12, 2011.

Iden, J., Tessem B., Päivärinta, T., "Problems in the interplay of development and IT operations in system development projects: A Delphi study of Norwegian IT experts," Information and Software Technology, vol. 53(4), 394-406, 2011, DOI: 10.1016/j.infsof.2010.12.002.

IEEE (2000) Recommended Practice for Architectural Description for Software-Intensive Systems: https://standards.ieee.org/ieee/1471/2187/, Accessed June 12, 2022

Jabbari, R., Bin Ali, N., Petersen K., Tanveer, B., "What is DevOps? A Systematic Mapping Study on Definitions and Practices," Proceedings of the Scientific Workshop Proceedings (XP2016). Association for Computing Machinery, New York, NY, USA, Article 12, 1–11, doi.org/10.1145/2962695.2962707.

Jasti N.V.K. and Kodali R., Lean production: literature review and trends, International Journal of Production Research, 2016, 53:3, 867-885, DOI: 10.1080/00207543.2014.937508.

Julian B., Noble J. and Anslow C., Agile Practices in Practice: Towards a Theory of Agile Adoption and Process Evolution, Agile Processes in Software Engineering and Extreme Programming, 20th International Conference, XP 2019, Montréal, QC, Canada, May 21–25, 2019 Proceedings.

Jones, S., Noppen, J., Lettice, F., "Management challenges for DevOps adoption within UK SMEs." ACM, 2016, 978-1-4503-4411-1/16/17.

Joseph, N.M.C. and Santana, M.J., 2016. Agile software development and IT project performance in South Africa: a positive relationship. In The 25th International Conference for Management of Technology (pp. 335-355).

Kamuto, M.B., J.J. Langerman, "Factors inhibiting the adoption of DevOps in large organisations: South African context," 2nd IEEE International Conference on Recent Trends in Electronics, Information & Communication Technology (RTEICT), 2017, DOI: 10.1109/RTEICT.2017.8256556.

Karim, A., Arif-Uz-Zaman, K., (2013) "A methodology for effective implementation of Lean strategies and its performance evaluation in manufacturing organizations". Business Process Management Journal, Vol. 19(1), 169-196.

Kelly, L. M. and Cordeiro, M. (2020) 'Three principles of pragmatism for research on organizational processes', Methodological Innovations, 13(2). doi: 10.1177/2059799120937242.

Kersten, M. (2018) "What Flows through a Software Value Stream?," in IEEE Software, Vol.35-4, 8-11.

Kerzazi, N., B. Adams, B., "Who needs release and devops engineers, and why?," Proceedings of the International Workshop on Continuous Software Evolution and Delivery (CSED2016). Association for Computing Machinery, New York, NY, USA, pp. 77–83, 2016, doi.org/10.1145/2896941.2896957.Krafcik, J. F. (1988) "Triumph of the Lean Production System", Sloan Management Review, Vol. 30 (1), 41–52.

Khan, M. S. Khan, A. W. Khan, F. Khan M. A. and Whangbo, T. K. (2022). Critical Challenges to Adopt DevOps Culture in Software Organizations: A Systematic Review, in IEEE Access, vol. 10, pp. 14339-14349, doi: 10.1109/ACCESS.2022.3145970.

Kiv S., Heng S., Kolp M. and Wautelet Y., (2019) "Agile Methods Knowledge Representation for Systematic Practices Adoption", Julian B., Noble J. and Anslow C., Agile Processes in Software

Engineering and Extreme Programming, In Proceedings of 20th International Conference, XP 2019, Montréal, QC, Canada, May 21–25.

Klagge, J., (2018) "Guidelines for Conducting Focus Groups", DOI: 10.13140/RG.2.2.33817.47201.

Kontio, J., Lehtola, L., Bragge, J., (2004) Using the Focus Group Method in Software Engineering: Obtaining Practitioner and User Experiences. In Proceedings of the 2004 International Symposium on Empirical Software Engineering (ISESE '04). IEEE Computer Society, USA, 271–280.

Krafcik, J. F. 1988. "Triumph of the Lean Production System." Sloan Management Review 30 (1): 41–52.

Kropp, M., Meier, A., Anslow, C., Biddle, R. (2018) "Satisfaction, Practices, and Influences in Agile Software Development", Proceedings of the 22nd International Conference on Evaluation and Assessment in Software Engineering (EASE), Association for Computing Machinery, USA, 112–121.

Kruchten P., "The rational unified process: an introduction," Addison-Wesley Longman Publishing Co., Inc., USA, 1999.

Krueger, R. A., & Casey, M. A. (2014) "Focus groups: A practical guide for applied research". 5th ed., Thousand Oaks, CA: Sage.

Larjovuori, R.L., Bordi, L., Heikkilä-Tammi, K. (2018) "Leadership in the digital business transformation" Proceedings of the 22nd International Academic Mindtrek Conference (Mindtrek), Association for Computing Machinery, USA

Langford J., McDonaugh, D. (2003) Focus Groups. Supporting Effective Product Development, Taylor and Francis, 2003.

Levinson, W.A. (2002) "Henry Ford's Lean Vision: Endurng Principles from the First Ford Motor Plant", Productivity Press, New York.

Liker, J. K. (1996) Becoming Lean. New York: Free Press.

Liker, J. K. (2004) The Toyota Way: 14 Management Principles from the World's Greatest Manufacturer. New York: McGraw-Hill.

Lipińska-Grobelny A., Papieska E., (2012) "Readiness for change and job satisfaction in a case of lean management application — A comparative study". International Journal of Occupational Medicine and Environmental Health 25:4.

Loukides M., What is DevOps?, O'Reilly Media, Sebastopol, CA, 2012.

Luz, P.W., Pinto, G., Bonifácio, R. (2019) "Adopting DevOps in the real world: A theory, a model, and a case study", Journal of Systems and Software, 157.

Lwakatare[a], L. E., Kuvaja P., Oivo, M (2016) "An exploratory study of DevOps: extending the dimensions of DevOps with practices," 11th International Conference on Software Engineering Advances, pp. 91–99, IARIA, Rome.

Lwakatare[b], L. E., Kuvaja P., Oivo, M., (2016) "Relationship of DevOps to Agile, Lean and Continuous Deployment," 17th International Conference on Product-Focused Software Process Improvement (PROFES), 399-415, doi:10.1007/978-3-319-49094-6.

Maroukian K. and Gulliver S. R. (2020) "Leading DevOps practice and principle adoption," Proceedings of the 9th International Conference on Information Technology Convergence and Services (ITCSE2020), AIRCC, Computer Science and Information Technology, 41-56, 2020, ISBN13: 978-1-925953-19-0.

Maroukian K. and Gulliver S. R. (2020) "The Link Between Transformational and Servant Leadership in DevOps-Oriented Organizations". In Proceedings of the 2020 European Symposium on Software Engineering, Association for Computing Machinery, New York, NY, USA, 21–29. DOI:https://doi.org/10.1145/3393822.3432340.

Martin, K., Osterling, M., (2014) "Value Stream Mapping: How to Visualize Work and Align Leadership for Organizational Transformation", McGraw-Hill Education, UK.

Masombuka, T., Mnkandla, E. (2018) "A DevOps collaboration culture acceptance model", Proceedings of the Annual Conference of the South African Institute of Computer Scientists and Information Technologists (SAICSIT), Association for Computing Machinery, USA, 279–285.

McManus, J.,Wood-Harper, T., (2008) "A study in project failure", British Computer Society, Project Management PM: LSR PARADIGM, DOI: 10.13140/RG.2.2.28178.20161.

Microsoft Corp., Microsoft Operations Framework (MOF), (2008), https://docs.microsoft.com/en-us/previous-versions/tn-archive/cc506049(v=technet.10), Accessed March 25, 2022.

Microsoft Corp., Azure DevOps Microsoft Documentation, (2022) *DevOps at Microsoft*. Available from: https://docs.microsoft.com/en-us/devops/, Accessed March 25, 2022.

Microsoft Corp., Microsoft DevOps Dojo, (2021) Available from: https://devblogs.microsoft.com/devops/intro-of-devops-dojo/, Accessed March 25, 2022.

Microsoft Docs. (2022) Scrum and best practices, https://docs.microsoft.com/en-us/azure/devops/boards/sprints/best-practices-scrum. Accessed March 25, 2022.

Mishra, S., & Ramesh, R., (2013) :A Software Solution to Facilitate Moderation, Observation and Analysis in a Focused Group Interview (FGI)". IEEE Fifth International Conference on Technology for Education, 9-12.

Morgan[a], D.L., (1996) "Focus Groups as Qualitative Research: 16 (Qualitative Research Methods)". SAGE Publications, Inc; 1st ed.

Morganb, D.L., (2003) "Focus Groups". Annu. Rev. Sociol. 22 (1), 129–152.

Mujtaba, S., Feldt, R., Petersen, K. (2010) "Waste and Lead Time Reduction in a Software Product Customization Process with Value Stream Maps", 21st Australian Software Engineering Conference, 139-148.

Mueller, R.O., Hancock, G.R. (2001) “Factor Analysis and Latent Structure, Confirmatory”, in International Encyclopedia of the Social & Behavioral Sciences.

Myrbakken H., Colomo-Palacios R. (2017) “DevSecOps: A Multivocal Literature Review”, Software Process Improvement and Capability Determination, SPICE, Communications in Computer and Information Science, Vol.770, Springer, Cham.

Niehaves, B., (2005) Epistemological perspectives on multi-method information systems research. 13th European Conference on Information Systems (ECIS 2005). Regensburg.

Nygard, M., Pal, T., Magill, S., Guckenheimer S., Willis, J., "DevOps Governance Architecture," IT Revolution, Portland, Oregon, 2019.

Office of Government Commerce (2011) “ITILv3 Core Publications”, *TSO (The Stationery Office)*, UK.

Ohno, T. (1988) “Toyota Production System: Beyond Large- Scale Production”, Productivity Press, New York.

Pankratz, O., Basten, D., “Eliminating Failure by Learning from It – Systematic Review of IS Project Failure”, International Conference on Information Systems (ICIS 2013): Reshaping Society Through Information Systems Design. 2., 2013.

Papadopoulou T.C., Özbayrak M., Leanness: experiences from the journey to date, Journal of Manufacturing Technology Management, 16 (7), 784-807, 2005.

Poppendieck M. and Poppendieck T., “Lean Software Development: An Agile Toolkit”, Addison-Wesley Professional, 2003, Boston, USA.

Project Management Institute (PMI) (2017) “A guide to the project management body of knowledge (PMBOK® Guide), 6th ed.”, *Project Management Institute*, Inc, Pennsylvania, USA.

Puppet, DORA. *State of DevOps Report 2021*. https://puppet.com/resources/report/2021-state-of-devops-report/ , Accessed March 25, 2022.

Puvanasvaran P., Megat H., Hong T.S., Razali M.M., Magid S.H.A., Lean process management implementation through enhanced problem solving capabilities, Journal of Industrial Engineering and Management, 447-493, 2010.

Rajkumar, M., Pole, A.K., Adige, V.S., Mahanta, P. (2016) "DevOps culture and its impact on cloud delivery and software development" International Conference on Advances in Computing, Communication, & Automation (ICACCA), 1-6.

Ravichandran, A., Taylor, K., Waterhouse, P. (2017) “DevOps for Digital Leaders, Reignite Business with a Modern DevOps-Enabled Software Factory”, Winchester.

Reinertsen, D. G., (2009) "The Principles of Product Development Flow: Second Generation Lean Product Development", Celeritas Pub; 1st ed.

Ringle, Christian M. and Sarstedt, Marko and Straub, Detmar, A Critical Look at the Use of PLS-SEM in MIS Quarterly (2012). MIS Quarterly (MISQ), Volume 36, No. 1, iii-xiv, March 2012.

Riungu-Kalliosaari, L., Mäkinen, S., Lwakatare, L.E., Tiihonen, J., Männistö T., (2016) “DevOps Adoption Benefits and Challenges in Practice: A Case Study”. In: Abrahamsson P., Jedlitschka A., Nguyen Duc A., Felderer M., Amasaki S., Mikkonen T. (eds) Product-Focused Software Process Improvement. PROFES 2016. Lecture Notes in Computer Science, vol 10027. Springer, Cham. https://doi.org/10.1007/978-3-319-49094-6_44.

Rodríguez, P., Partanen, J., Kuvaja, P., Oivo, M. (2014) “Combining Lean Thinking and Agile Methods for Software Development A Case Study of a Finnish Provider of Wireless Embedded Systems”, Proceedings of the Annual Hawaii International Conference on System Sciences, 4770-4779.0.

Rodríguez, P., et al., (2017) "Continuous deployment of software intensive products and services: A systematic mapping study," Journal of Systems and Software, vol. 123, 263-291, doi.org/10.1016/j.jss.2015.12.015.

Rodríguez P., et al., (2019) "Chapter Four - Advances in Using Agile and Lean Processes for Software Development," Advances in Computers, Elsevier, vol. 113, 135-224, doi.org/10.1016/bs.adcom.2018.03.014.

Rother, M., (2016) “Toyota Kata: Managing People for Improvement, Adaptiveness and Superior Results”, 1st ed., McGraw Hill.

Sahu, S., Pathardikar, A. and Kumar, A. (2018), "Transformational leadership and turnover: Mediating effects of employee engagement, employer branding, and psychological attachment", Leadership & Organization Development Journal, Vol. 39 No. 1, 82-99. https://doi.org/10.1108/LODJ-12-2014-0243

Sánchez-Gordón, M., Colomo-Palacios, R. (2018) “Characterizing DevOps Culture: A Systematic Literature Review”, Software Process Improvement and Capability Determination (SPICE), Communications in Computer and Information Science, Vol.918, Springer.

Saunders, M.N.K., Lewis, P., Thornhill, A., (2018) “Research Methods for Business Students”, 8th ed., Pearson

Schlossnagle T., "Monitoring in a DevOps world," ACM Queue, 2017, dl.acm.org/doi/pdf/10.1145/3178368.3178371.

Sfetsos P., Stamelos, I., (2010) "Empirical Studies on Quality in Agile Practices: A Systematic Literature Review," Seventh International Conference on the Quality of Information and Communications Technology, Porto, 44-53, doi: 10.1109/QUATIC.2010.17.

Scrum.org™ , (2022) The Scrum Framework, https://www.scrum.org/resources/scrum-framework-poster. Accessed March 25, 2022.

Shingo, S., Dillon, A.P. (1988) “A study of the Toyota production system”, Productivity Press, New York.

Sims, C., Carter, A., Moore de Peralta, A. (2020). Do servant, transformational, transactional, and passive avoidant leadership styles influence mentoring competencies for faculty? A study of a gender equity

leadership development program. Human Resource Development Quarterly. 32. 10.1002/hrdq.21408.

Skelton, M., Pais, M., “Team Topologies”, IT Revolution, Portland, USA.

Smart J., (2020), Sooner Safer Happier - Patterns and Antipatterns for Organizational Agility: Antipatterns and Patterns for Business Agility, IT Revolution Press.

Smeds, J., Nybom K., and Porres, I., (2015) "DevOps: A definition and perceived adoption impediments," Agile Processes in Software Engineering and Extreme Programming (XP 2015), Lecture Notes in Business Information Processing, vol. 212, Springer, Cham.

Špundak, M., (2014) “Mixed Agile/Traditional Project Management Methodology – Reality or Illusion?”. Procedia - Social and Behavioral Sciences, Vol. 119, 939-948, ISSN 1877-0428, https://doi.org/10.1016/j.sbspro.2014.03.105.

Stahl, D., Martensson T., Bosch, J., (2017) "Continuous practices and devops: beyond the buzz, what does it all mean?”.43rd Euromicro Conference on Software Engineering and Advanced Applications (SEAA), pp. 440-448, doi: 10.1109/SEAA.2017.8114695.

Standish Group – CHAOS Report 2020, https://www.standishgroup.com/news/45, Accessed March 25, 2022.

Straub, E. T., (2009) “Understanding technology adoption: Theory and future directions for informal learning”. Review of Educational Research, 79, 625-649.

Sutherland J. and Schwaber K., “The Definitive Guide to Scrum: The Rules of the Game”, 2017, USA.

Takeuchi, H. and Nonaka, I. (1986) “The New New Product Development Game”, Harvard Business Review, Vol. 64, 137-146.

Tashakkori, A., Teddlie, C., (1998) “Mixed methodology: Combining qualitative and quantitative approaches”. Sage Publications, Thousand Oaks, CA, US.

Techtarget, (2018) “Are DevOps responsibilities in production unfair to developers?” https://devopsagenda.techtarget.com/feature/Are-DevOps-responsibilities-in-production-unfair-to-developers, Accessed March 25, 2022.

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

Venkatesh, V., Thong, J., & Xu, 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. DOI: 10.17705/1jais.00428.

Webster, J., Watson, R.T., (2002) “Analyzing the past to prepare for the future: writing a literature review”. MIS Q. 26, 2.

Wiedemann, A., Wiesche, M. (2018) “Are You Ready for DevOps? Required Skill Set for DevOps Teams”, Proceedings of the European Conference on Information Systems (ECIS), Portsmouth.

Willis, J., 2010. What DevOps means to me, https://blog.chef.io/what-devops-means-to-me , Accessed March 25, 2022.

Womack, J.P. and Jones, D.T. (1990) “The Machine That Changed the World”, Rawson Associates, New York.

Womack, J. P., Jones, D.T., (1994) “From Lean Production to the Lean Enterprise.” Harvard Business Review 72 (2): 93–103.

Womack, J.P., Jones, D.T., Roos, D., (1996) “Lean Thinking: Banish Waste and Create Wealth in Your Corporation”. New York, Simon & Schuster.

World Economic Forum - Future of Jobs 2020, https://www3.weforum.org/docs/WEF_Future_of_Jobs_2020.pdf?msclkid=b1ec7c01a21611eca98d6eb30c50a999 , Accessed March 25, 2022.

Wright, S. (1921) Correlation and Causation. Journal of Agricultural Research, 20, 557-585.

Xiaofeng, W., Conboy, K., Cawley, O., (2012) "Leagile" software development: An experience report analysis of the application of lean approaches in agile software development," J. Syst. Softw. Vol. 85.

Yukl, G.A., Gardner, III, W.L., (2020) "Leadership in organizations", Pearson Education, Essex, ISBN13:978-1-292-31440-2.

Yusof, S.M., Aspinwall, E., (2000) “TQM implementation issues: review and case study". International Journal of Operations & Production Management, Vol. 20 No. 6, 634-655. https://doi.org/10.1108/01443570010321595.

Zarbo R.J., D’Angelo R., (2007) “The Henry Ford Production System Effective Reduction of Process Defects and Waste in Surgical Pathology”, American Journal of Clinical Pathology, Volume 128, Issue 6, 1015–1022.

Appendix A – Interview Questions

Q1. Please state your job role

_____

Q2. Which is the principal industry in which your organisation currently operates?

☐ Accounting

☐ Administrative

☐ Aviation

☐ Consulting Services

☐ Customer Service

☐ Business Intelligence

☐ Engineering

☐ Finance

☐ Human Resources

☐ IT

☐ Insurance

☐ Legal

☐ Manufacturing

☐ Marketing

☐ Operations

☐ Lottery

☐ Research & Development

☐ Sales

☐ Other (please specify):

_____

Q3. Which department of your organisation do you work for?

_____

Q4. Has your organisation adopted any of the following frameworks or practices? Choose any that apply. **(RQ1)**

☐ Structured Service Management Approach (e.g. IT Infrastructure Library (ITIL®), ISO20000, etc.)

☐ Structured Project Management Approach (e.g. PMBOK®, PRINCE2®, etc.)

☐ Agile Software Development

☐ Lean IT

☐ DevOps

☐ Other (please specify): _____

Q5. Can you please share examples of any benefits, in terms of human resource management, productivity, processes, tools and culture that you have experienced while adopting this approach(es). State as many as you like. **(RQ1)**

_____

Q6. Can you please share examples of any impediments, in terms of human resource management, productivity, processes, tools and culture that you encountered while adopting this approach(es). State as many as you like. **(RQ1)**

_____

Q7. Considering the following *practices and standards* which ones are you aware of so that you would feel comfortable to provide a definition for? **(RQ1)**

☐ Agile Unified Process

☐ Value Stream Map

☐ ISO/IEC 15504 Software Process Improvement and Capability Determination (SPICE) (Maturity Model)

☐ ISO/IEC 291110 Systems and Software Life Cycle Profiles and Guidelines for Very Small Entities (VSEs)

☐ Lean failure

☐ Lean leadership

☐ Lean management

☐ Lean process management

☐ Culture

☐ Customer presence product development

☐ CMMI-DEV

☐ R&D outsourcing

☐ Complex Adaptive Systems (CAS) Theory

☐ Water-Scrum-fall

☐ Incentive contract

☐ Tacit knowledge value

☐ SCRUM

☐ XP

☐ Test Driven Development (TDD) or Test-first Development (TFD)

☐ Unified Process for Education (UPEDU)

☐ 5S

☐ Kaizen

☐ Kanban

☐ IT Infrastructure Library (ITIL®)

☐ Rational Unified Process (RUP)

☐ Service Oriented Architecture

☐ Continuous delivery

☐ Continuous integration

☐ Pair programming

Q8. Considering the following *practices and standards* which ones do you consider most beneficial to an organisation's ability to maximise value delivered to the customer via a software product? **(RQ1)**

☐ Agile Unified Process

☐ Value Stream Map

☐ ISO/IEC 15504 Software Process Improvement and Capability Determination (SPICE) (Maturity Model)

☐ ISO/IEC 291110 Systems and Software Life Cycle Profiles and Guidelines for Very Small Entities (VSEs)

☐ Lean failure

☐ Lean leadership

☐ Lean management

☐ Lean process management

☐ Culture

☐ Customer presence product development

☐ CMMI-DEV

☐ R&D outsourcing

☐ Complex Adaptive Systems (CAS) Theory

☐ Water-Scrum-fall

☐ Incentive contract

☐ Tacit knowledge value

☐ SCRUM

☐ XP

☐ Test Driven Development (TDD) or Test-first Development (TFD)

☐ Unified Process for Education (UPEDU)

☐ 5S

☐ Kaizen

☐ Kanban

☐ IT Infrastructure Library (ITIL®)

☐ Rational Unified Process (RUP)

☐ Service Oriented Architecture

☐ Continuous delivery (in IT production systems)

☐ Continuous integration (in IT production systems)

☐ Pair programming

Q9. Considering the following *principles* which ones are you aware of so that you would feel comfortable to provide a definition for? **(RQ1)**

☐ Employee commitment

☐ Employee motivation

☐ Employee satisfaction

☐ Knowledge sharing

☐ Culture

☐ Lean Culture

☐ Agile software development (ASD)

☐ Lean software development

☐ People Development System (PDS)

☐ Refactoring

☐ Respect for people

☐ Automation (process, tool, etc.)

☐ Measurement (performance metrics)

☐ Monitoring (IT infrastructure, resources, systems, etc.)

☐ Shared Values

Q10. sider most beneficial to an organisation's ability to maximise value delivered to the customer via a software product? **(RQ1)**

☐ Employee commitment

☐ Employee motivation

☐ Employee satisfaction

☐ Knowledge sharing

☐ Culture

☐ Lean Culture

☐ Agile software development (ASD)

☐ Lean software development

☐ People Development System (PDS)

☐ Refactoring

☐ Respect for people

☐ Automation (process, tool, etc.)

☐ Measurement (performance metrics)

☐ Monitoring (IT infrastructure, resources, systems, etc.)

☐ Shared Values

Q11. Considering structured IT service management processes such as Service Portfolio Management, Change Management, Release and Deployment Management, Incident & Problem Management, Service Level Management, Availability Management, Configuration Management, etc. which practices, principles, and tools can contribute to value delivered to the customer as an outcome of the software development process? Please provide as many examples as you wish. **(RQ2)**

_____

Q12. Considering structured IT service management processes such as Change Management, Release and Deployment Management, Incident & Problem Management, Service Level Management, Availability Management, Configuration Management, etc. what sort of impediments can there be to value delivered to the customer in the software development process? Please provide as many examples as you wish. **(RQ2)**

_____

Q13. Considering structured IT service management frameworks such as ITIL and standards such as ISO20000, can agility and leanness become an extension to those approaches or a replacement? Please explain. **(RQ2)**

_____

Q14. DevOps practice adoption can lead to a significant need for a transformation of organisational capabilities. Which of the following should be addressed by a DevOps transformation initiative? Choose any three (3) that apply. **(RQ2)**

☐ Operations not being involved in the requirements specifications

☐ Poor communication and information flow

☐ Unsatisfactory test environment(s)

☐ Lack of knowledge transfer

☐ Systems released to production before they are complete

☐ Operational routines not being established prior to deployment

☐ Deep-seated company culture

☐ Industry constraints and feasibility

☐ Heterogeneous DEV/Test/Production environments

☐ DevOps is unclear

Q15. Considering the aforementioned definition of DevOps or your own experience, what does DevOps mean to you? Choose any six (6) that apply. **(RQ3)**

☐ Shift of mindset

☐ Blameless Culture

☐ New skills

☐ Continuous Integration

☐ Continuous Deployment

☐ New toolset adoption

☐ More implemented features

☐ Frequent Releases

☐ Automated testing process

☐ Automated build process

☐ Automated deployment process

☐ More daily commits of code

☐ Improved product quality

☐ Enhanced collaboration and communication

☐ Testing with real customers

☐ Improved visibility of implemented features to the customer

☐ Improved well-being of DevOps teams

☐ Rapid Feedback

☐ Increased customer satisfaction

☐ Respect for others

☐ Improved decision making

☐ Improved service performance monitoring

☐ Improved service level management

Q16. Quality assurance is an important aspect of producing shippable products and services by accurately satisfying customer requirements. There is evidence to suggest that customer involvement during QA increases product quality. In your view, in which cases should customer involvement be considered? Choose any two (2) that apply. **(RQ3)**

☐ Requirements elicitation

☐ Requirements prioritisation

☐ Definition of features/user stories

☐ Definition of Done (DoD)

☐ User Acceptance Testing

☐ Other (please specify): _____

Q17. Considering transformational leadership for DevOps practice adoption purposes, what kind of skills, training and capabilities should a DevOps leader role possess within an organisation? Please explain. **(RQ3)**

_____

Q18. Should DevOps leadership be designated as an individual or a team role? **(RQ3)**

_____

Q19. Considering DevOps leadership objectives, should DevOps practice adoption be extended in an enterprise-wide fashion and also aim to include service providers of the organisation in its scope? Why? **(RQ3)**

_____

Q20. Considering your own experiences, which organisational teams should be part of a DevOps practice adoption journey? Choose any that apply. **(RQ3)**

☐ IT Development

☐ IT Operations

☐ Information Security

☐ Quality Assurance

☐ Audit and Compliance

☐ Board of Directors

☐ Marketing

☐ Finance

☐ Legal

☐ Procurement

☐ Sales

☐ Human Resources

☐ Other (please specify): _____

Appendix B – Terms of Interview Participation

This study is not anonymous and it is not the intention of the researcher to collect your name. However, you do have the option to participate anonymously. Please know that if you do, it may be linked to your responses in this study. Any consequences are outside the responsibility of the researcher, faculty supervisor or the University of Reading.

INTRODUCTION

My name is Krikor Maroukian. I am a doctoral student at the University of Reading, UK. I am also an employee of Microsoft. I am conducting a research study on a multifaceted strategic model of DevOps adoption practices and value based product development. I am completing this research as part of my doctoral degree. Your participation is completely voluntary. I am seeking your consent to involve you and your information in this study. Reasons you might ***not*** want to participate in the study include not wanting to disclose information about your professional experiences with customers. Reasons you might want to participate in the study include helping to advance research around DevOps practice adoption and assisting an IT professional in completing his educational journey. An alternative to this study is simply not participating. I am here to address your questions or concerns during the informed consent process.

PRIVATE INFORMATION

This interview is intended to be anonymous. If you insist on giving your name, I will make the following effort to protect your private information, including not bringing your name out of the interview process, making the response confidential. Even with this effort, there is a chance that your private information may be accidentally released. The chance is small but does exist. You should consider this when deciding whether to participate. I encourage you to participate anonymously.

ACTIVITIES

If you participate in this research, you will be asked to:

1. Answer to a set of open and close-ended interview questions.
2. The interview process should not take longer than 60 minutes to complete.

RISKS

There are minimal risks in this study. Some possible risks include: slight anxiety in answering the questions. To decrease the impact of these risks, you can: skip any question or stop participation at any time.

BENEFITS

If you decide to participate, there are no direct benefits to you. The potential benefits to others are: finding new ways to improve DevOps practice adoption for organisations and professionals.

CONFIDENTIALITY

The information you provide will be kept confidential to the extent allowable by law. Some steps taken to keep your identity confidential are: There will not be use of your name. The answers will be anonymised. Individuals who will have access to your information are: myself, and/or, my supervisor and/or research assessors. To prevent this exposure, you can choose to participate anonymously.
Your information with be kept secure with the following steps: your name will not be extracted from the interview results.

Your data will be stored for seven (7) years after which all electronic and/or paper copies will be destroyed.

CONTACT INFORMATION

If you have questions for me, you can contact me at:
K.Maroukian@pgr.reading.ac.uk

My supervisor's name is Dr. Stephen Gulliver. He works at the University of Reading and is supervising me on the research. You can contact him at: s.r.gulliver@henley.ac.uk

VOLUNTARY PARTICIPATION

Your participation is voluntary. If you decide not to participate, or if you stop participation after you start, there will be no penalty to you. You will not lose any benefit to which you are otherwise entitled.

FUTURE RESEARCH

Any information or specimens collected from you during this research may **not** be used for other research in the future, even if identifying information is removed.

Appendix C – Consent to Interview Participation

Certain information collected as part of this interview may fall under articles of the General Data Protection Regulation (GDPR) for European Union member countries. The interviewee maintains the right to exercise GDPR rights by informing the interviewer (K.Maroukian@pgr.reading.ac.uk).

You understand that any data or information provided by me as part of this interview may be used by The University of Reading in connection with this interview, other studies, or analyses performed by The University of Reading.

You understand that this interview and the interview results are the property of the University of Reading, UK.

You understand that any such data or information may be disclosed by The University of Reading to related entities or other third parties, including, without limitation, in publications, in connection with this interview or such studies, analyses, or services, provided that such data or information does not contain any information that identifies me or associates me with the responses I have provided to this interview.

You understand disclosure of such data or information may be required by law, in which case The University of Reading will endeavor to notify me.

The University of Reading is not, by means of this interview or the interview results, rendering professional advice or services to me or my company. Neither this interview nor the interview results is a substitute for such professional advice or services. The University of Reading is not responsible for any loss sustained by any person who relies on this interview or the interview results.

Appendix D – Survey Questions

1.Please state your age (years) to help us improve segmentation of obtained survey outcomes.

2.How many years of professional experience do you have?

3.Please state your role area within the organisation you are currently employed? Please select only one option.

4.Which is the principal industry in which your organisation currently operates? Please select only one option.

5.Which region are you based in? Please select only one option.

6.Approximately how many employees are there in your organisation (across all locations)? Please select only one option.

7.Do you currently or have you in the past held any organisational or team leadership position?

 Yes
 No

8.Please indicate the level of experience you possess in the following areas.

	1-12 months	1-2 years	3 - 5 years	5+ years	No experience	Not Sure
Scrum						
Kanban						
DevOps						
Continuous Delivery						
Continuous Integration						

9.Has your organisation adopted or planning to adopt DevOps practices? Please select only one option.

10.Which, if any, of the following techniques are already in place in your organisation?

	Yes	No	Not sure
Version control			

	Yes	No	Not sure
Issue tracking	○	○	○
Test automation	○	○	○
Automated deployment	○	○	○
Performance monitoring	○	○	○
Trunk-based development	○	○	○
Automated provisioning	○	○	○
Static code analysis	○	○	○
Code coverage	○	○	○
Containerised environments	○	○	○
Infrastructure as Code	○	○	○

11.Who is primarily involved in the decision-making process of DevOps practice adoption? Please select only one option.

12.When do you primarily witness higher resistance to change in terms of DevOps practice adoption? Please select only one option.

13.State the level of agreement to various identified barriers to organisational capabilities and traits to allow the adoption of new practices such as DevOps.

	Strongly Agree	Agree	Disagree	Strongly disagree
Lack of organisational practice adoption capability	○	○	○	○
Lack of commitment by customers of the IT organisation e.g. in feedback loop mechanism, UAT, etc.	○	○	○	○

	Strongly Agree	Agree	Disagree	Strongly disagree
Designing and throwing over the wall, not engaging end-to-end stakeholders in improving their own process	○	○	○	○
Lack of organisational change management capabilities	○	○	○	○
Leadership skillset	○	○	○	○

14.Considering the following practices and principles which ones do you consider most beneficial to an organisation's ability to maximise value delivered to the customer via DevOps?

Automation

Employee engagement

Continuous Delivery

Continuous Integration

Kanban

Monitoring

Measurement

Organisational Culture

Scrum

Value Stream Map

15.State in your opinion how should DevOps practice and principles adoption levels be measured. Please select as many options as you deem appropriate.

☐ Behavioural metrics
☐ Critical Success Factors
☐ Deployment duration (time)
☐ Deployment frequency
☐ Deployment size
☐ Feature usage
☐ Key Performance Indicators
☐ Knowledge article creation frequency
☐ Knowledge article read frequency
☐ Releases frequency per developer per day
☐ Time to Market

☐ Time to Recovery

☐ Time to Detect (defect)

☐

16.In your opinion should the DevOps leadership role be associated to any of the aforementioned metrics (see Question 15)?

◉ Yes

◉ No

◉

17.Should a new practice adoption leadership role exist such as DevOps Leader role?

◉ Yes

◉ No

◉

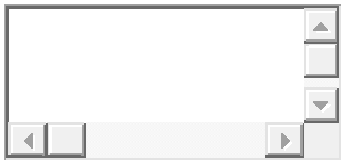
18.Please state whether you agree or disagree to the following list of skills, which a leader's role could possess, to enable new practice adoption such as DevOps.

	Strongly Agree	Agree	Disagree	Strongly disagree
Technical Background	◉	◉	◉	◉
Business Background	◉	◉	◉	◉
Agile Management Skills	◉	◉	◉	◉
Communication and Collaboration Skills	◉	◉	◉	◉
Problem Solver	◉	◉	◉	◉
Talent Seeker	◉	◉	◉	◉
Negotiation Skills	◉	◉	◉	◉
Good listener	◉	◉	◉	◉
Change Agent	◉	◉	◉	◉
Risk Management Skills	◉	◉	◉	◉

	Strongly Agree	Agree	Disagree	Strongly disagree
Multi-cultural mindset				
Customer-centric mindset				
Previous experience on transformation				
Design thinking				
Certifications				
Leadership Skills				
Influential				
Holistic Systems Thinking				
Strategic Thinking				
Project Management Skills				

19.Where should the DevOps leadership role directly report to? Please select only one option.

20.If you wish to add any special comments/opinions for the purposes of this survey please provide them below.



Appendix E – Terms of Survey Participation

This study is not anonymous and it is not the intention of the researcher to collect your name. However, you do have the option to participate anonymously. Please know that if you do, it may be linked to your responses in this study. Any consequences are outside the responsibility of the researcher, faculty supervisor or the University of Reading.

INTRODUCTION

My name is Krikor Maroukian. I am a doctoral student at the University of Reading, UK. I am also a full-time employee of Microsoft. I am conducting a research study on a multifaceted strategic model of DevOps adoption practices and value based product development. I am completing this research as part of my doctoral degree. Your participation is completely voluntary. I am seeking your consent to involve you and your information in this study. Reasons you might ***not*** want to participate in the study include not wanting to disclose information about your professional experiences with customers. Reasons you might want to participate in the study include helping to advance research around DevOps practice adoption and assisting an IT professional in completing his educational journey. An alternative to this study is simply not participating. I am here to address your questions or concerns during the informed consent process.

PRIVATE INFORMATION

This survey is intended to be anonymous. If you insist on giving your name, I will make the following effort to protect your private information, including not bringing your name out of the survey process, making the response confidential. Even with this effort, there is a chance that your private information may be accidentally released. The chance is small but does exist. You should consider this when deciding whether to participate. I encourage you to participate anonymously.

ACTIVITIES

If you participate in this research, you will be asked to:

1. Answer to a set of open and close-ended survey questions.

2. The survey process should not take longer than 15 minutes to complete.

RISKS

There are minimal risks in this study. Some possible risks include: slight anxiety in answering the questions. To decrease the impact of these risks, you can: skip any question or stop participation at any time.

BENEFITS

If you decide to participate, there are no direct benefits to you. The potential benefits to others are: finding new ways to improve DevOps practice adoption for organisations and professionals.

CONFIDENTIALITY

The information you provide will be kept confidential to the extent allowable by law. Some steps taken to keep your identity confidential are: There will not be use of your name. The answers will be anonymised. Individuals who will have access to your information are: myself, and/or, my supervisor and/or research assessors. To prevent this exposure, you can choose to participate anonymously.
Your information with be kept secure with the following steps: your name will not be extracted from the survey results.

Your data will be stored for seven (7) years after which all electronic and/or paper copies will be destroyed.

CONTACT INFORMATION

If you have questions for me, you can contact me at:

K.Maroukian@pgr.reading.ac.uk

My supervisor's name is Dr. Stephen Gulliver. He works at the University of Reading and is supervising me on the research. You can contact him at: s.r.gulliver@henley.ac.uk

VOLUNTARY PARTICIPATION

Your participation is voluntary. If you decide not to participate, or if you stop participation after you start, there will be no penalty to you. You will not lose any benefit to which you are otherwise entitled.

FUTURE RESEARCH

Any information or specimens collected from you during this research may **not** be used for other research in the future, even if identifying information is removed.

Appendix F –Consent to Survey Participation

Dear respondent,

My name is Krikor Maroukian. I am a doctoral student at Henley Business School, University of Reading, UK. I am also a full-time employee of Microsoft. Currently, I am conducting a research study on a model of new practice adoption in terms of value-based product development which will allow us to understand the affects a leadership role can have in a business environment.

I would like to thank you for participating to the survey research and for providing consent to answer the questions which follow below. Information collected will be kept and used solely for the research period and purposes. The survey should take less than ten minutes to complete.

Your participation in this research study is strictly voluntary, and you may choose not to participate without fear of penalty or any negative consequences. You will be able to withdraw from the survey at any time and all survey responses will be deleted.

There will be no individually identifiable information, remarks, comments, or other identification of you as an individual participant. All results will be presented as aggregate, summary data. If you wish, you may request a copy of the results of this research study by writing to the researcher at: K.Maroukian@pgr.reading.ac.uk

Appendix G – Focus Group Registration Questions

1.Do you agree to the aforementioned research study terms and provide consent? Single choice.

- I agree
- I do not agree

2.Please state your age (years) to help us improve segmentation of obtained survey outcomes. Single choice.

- Less than 21
- 21-30
- 31-40
- 41-50
- 51-60
- 61-70
- 71+

3.How many years of professional experience do you have? Single choice.

- less than 2
- 2-5
- 6-10
- 11-15
- 16-20
- 21+

4.Please state your role area within the organisation you are currently employed? Please select only one option. Single choice.

5.Which is the principal industry in which your organisation currently operates? Please select only one option. Single choice.

6.Which region are you based in? Please select only one option. Single choice.

7.Approximately how many employees are there in your organisation (across all locations)? Please select only one option. Single choice.

8.Please state the annual turnover of your organisation. Single choice.

9.Do you currently or have you in the past held any organisational or team leadership position? Single choice.

- Yes
- No

Appendix H – In-session Focus Group Questions

1.Do you agree with the constituents of 'Practices and Principles'?

Strongly Disagree Strongly Agree

2.Please provide additional comments for "Practices and Principles" influencing "DevOps Adoption Planning".

3.Do you agree with the constituents of 'DevOps Adoption Leader Role'?

Strongly Disagree Strongly Agree

4.Please provide additional comments for "DevOps Adoption Leader Role" influencing "DevOps Adoption Leader Skillset".

5.Do you agree with the constituents of 'DevOps Adoption Metrics'?

Strongly Disagree Strongly Agree

6.Please provide additional comments for "DevOps Adoption Metrics" influencing "DevOps Adoption Leader Skillset".

7.Do you agree with the constituents of 'DevOps Adoption Leader Skillset'?

Strongly Disagree Strongly Agree

8.Do you agree with the overall structure of the model?

	Strongly Disagree	Disagree	Neutral	Agree	Strongly agree
Practices and Principles					
DevOps Adoption Leader Role					

	Strongly Disagree	Disagree	Neutral	Agree	Strongly agree
DevOps Adoption Metrics					
DevOps Adoption Leader Skillset					

9.Please state which other areas, in your opinion, are important to be added to the model.

10.If you wish to add any special comments/opinions for the purposes of this survey please provide them below.

Appendix I –Consent to Focus Group Participation

Dear focus group participant,

I am a doctoral student at Henley Business School, University of Reading, UK. I am also a full-time employee of Microsoft. Currently, I am conducting a research study on a leadership model of new practice adoption having 'DevOps' as a main research topic.

I would like to extend an invite to participate and actively contribute to a scheduled Research Focus Group #2 due to be conducted under the auspices of *Informatics Research Center (IRC), School of Business Informatics, Systems & Accounting (BISA), Henley Business School, University of Reading, UK* and its *Ethics Compliance* requirements, on Monday, 29th of March, 2021 at 19.00 (EET).

Should you wish to accept the invite, I would like to thank you for participating to the research study's focus group.

Please visit the following link **to provide consent to the terms of the research study** (described below for reference purposes) prior to the commencement of the scheduled focus group.
https://forms.office.com/r/U4eJ4ccgq5

RESEARCH STUDY TERMS

RISKS

There are minimal risks in your participation in this study. Some possible risks include: slight anxiety in answering the questions. To decrease the impact of these risks, you can opt for one of the following steps:

- Skip moment in time,
- stop participation to the focus group.

BENEFITS

If you decide to participate, there are no direct benefits to you. The potential benefits to others are:

- Finding new ways to improve DevOps practice adoption for organisations and professionals.
- The researcher is provided with an improved understanding of the effects leadership and its styles, traits and characteristics can have in the context of the research study.

CONFIDENTIALITY

The information you provide will be kept confidential to the extent allowable by law. Information collected will be kept and used solely for the research period and purposes. Your data will be stored for seven (7) years after which all electronic and/or paper copies will be destroyed.

Some steps taken to keep your identity confidential are:

- There will not be use of your name.
- The answers will be anonymised.
- Individuals who will have access to your information are:
  - Myself, and/or;
  - My supervisor and/or;
  - Research assessors.

This study is not anonymous and it is not the intention of the researcher to collect your name. However, you do have the option to participate anonymously. Please know that if you do, it may be linked to your responses in this study. Any consequences are outside the responsibility of the researcher, faculty supervisor or the University of Reading.

There will be no individually identifiable information, remarks, comments, or other identification of you as an individual participant. All results will be presented as aggregate, summary data. If you wish, you may request a copy of the results of this research study by writing to the researcher at: K.Maroukian@pgr.reading.ac.uk

Best regards,

Krikor Maroukian
Informatics Research Center (IRC)
School of Business Informatics, Systems & Accounting (BISA)
Henley Business School
University Of Reading, Whiteknights, Reading RG6 6UD