

MASTER'S THESIS

Measures of Critical Success Factors of CI/CD-processes

de Vries, N.J.

Award date:
2021

[Link to publication](#)

General rights

Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain.
- You may freely distribute the URL identifying the publication in the public portal.

Take down policy

If you believe that this document breaches copyright please contact us at:

pure-support@ou.nl

providing details and we will investigate your claim.

Downloaded from <https://research.ou.nl/> on date: 12. Dec. 2021

Open Universiteit
www.ou.nl



Measures of Critical Success Factors of CI/CD-processes

Degree programme: Open University of the Netherlands, Faculty of Management, Science & Technology
Business Process Management & IT master's programme

Course: IM0602 BPMIT Graduation Assignment Preparation
IM9806 Business Process Management and IT Graduation Assignment

Student: Nancy J. de Vries

Date: 25-04-2021

Thesis supervisor: Prof.dr.ir. Jos J.M. Trienekens

Second reader: Michiel van Belzen MSc

Third assessor: Prof. dr. R.J. Kusters

Version number: 1

Status: final version

Prologue

In front of you is my thesis 'Measures of Critical Success Factors of CI/CD processes'. I have written this thesis as a part of my Master's degree in Business Process Management & IT at the Open University.

A long-cherished opportunity came to reality when I started studying for a master's degree. Because of my employer, I was given the opportunity to develop myself in various functions that ultimately led to following this study.

Pursuing an education, however, alongside work and everyday life proved to be a fierce challenge. Writing this thesis was during one of the most difficult periods I have been through, and there have been times when I have wondered if I had started this with the knowledge of today. Nevertheless, I am grateful for the opportunity I was given and all the things I was allowed to learn during the past few years culminating in writing this thesis.

I would like to thank my supervisors Michiel van Belzen and Jos Trienekens for their patience, critical eye and wise advice. Despite the Covid-19 measures, we fortunately had good Skype conversations. I would also like to thank my colleagues for the enjoyable and educational interviews I was able to conduct with them. Especially the post-interview conversations with them have increased my enthusiasm to continue with my newfound knowledge in my attempt to improve a little part of the organization. Finally, I would like to thank my family and especially my boyfriend for the understanding, support and the space they gave me during the study and especially during the writing of this thesis.

Nancy de Vries

April 25, 2021

Abstract

With the increasing expectation of faster and higher quality applications, Continuous processes can be a solution for organizations. However, the adoption of these continuous processes is not an effortless task for organizations. A previous research provided a framework of the Critical Success Factors for the adoption of Continuous Processes. The purpose of this research was to examine what measures organizations can take for the Critical Success Factors that can support the implementation of Continuous Integration and Continuous Delivery. Through a systematic literature review, a search was conducted to find the measures that can contribute to the Critical Success Factors for the adoption of Continuous Integration and Continuous Delivery. These measures together provide a theoretical framework. This theoretical framework was then validated during an empirical research by interviewing experts in the field of Continuous Integration and Continuous Delivery and finally resulted in a practical framework that can be used by organizations as a guide before, during and after the implementation of Continuous Integration and Continuous Delivery.

Key terms

Critical Success Factors, Continuous Integration, Continuous Delivery, Continuous Deployment, Measures, Continuous Processes

Summary

With the fast development of information technology, there are always high expectations for rapid development but despite the increase in software complexity, good quality must not be compromised. Continuous Integration and Continuous Delivery are continuous processes that can be used by organizations to ensure that the expectations of fast delivery and good quality are nevertheless achieved. However, implementing these continuous processes are not without challenges.

This research is a follow-up to an earlier study in which van Belzen developed a framework with the Critical Success Factors for the adoption of continuous processes. This report builds on that with the objective of examining what measures organizations can take for the Critical Success Factors that can support the implementation of CI/CD.

The main question of this research is: What are measures of Critical Success Factors of the CI/CD process? For this research, the Critical Success Factors as found by van Belzen are divided into three sections: Process control/management/governance, Customer and CI/CD process. During the first part of this report, the central question is "What are measures of Critical Success Factors of the CI/CD process from a process management perspective?". During the second part of this report, the central question is: 'What are measures of Critical Success Factors from a customer perspective of the CI/CD processes?'

For the purpose of forming a framework with measures for each Critical Success Factor, the first part of this report involved a Systematic Literature Review to find measures that could contribute to a successful implementation from a process management perspective. These measures were then merged with the measures with a Customer perspective found by a fellow student to form a theoretical framework. The theoretical framework contains as many as 32 measures and for the purpose of proper further analysis it was decided to individually empirically validate only the measures related to the Customer.

During the empirical study, six interviews were conducted with experts in the field of CI/CD within a government organization to validate the measures. It emerged that of the twenty measures found in the literature, eighteen measures were confirmed. However, it should be noted that one measure has been modified and that the measures relating to "Resistance to Change" are not directed towards the customer according to the interviewees. It should be noted that the measures have been validated in a government organization that is not yet mature in the application of CI/CD so the result regarding the applicability of the measures in another organization may lead to a different result.

The result of this report provides a framework of measures for Critical Success Factors that an organization can apply before, during and after the implementation of CI/CD processes.

Table of contents

Prologue	ii
Abstract	iii
Key terms.....	iii
Summary	iv
1. Introduction	1
1.1. Background.....	1
1.2. Exploration of the topic.....	1
1.3. Problem statement.....	2
1.4. Research objective and questions	2
1.5. Motivation/relevance	3
1.6. Main lines of approach	3
2. Theoretical framework.....	4
2.1. Research approach	4
2.1.1. Search process.....	4
2.1.2. Data collection	5
2.1.3. Data analysis.....	5
2.2. Implementation	6
2.2.1. Search process.....	6
2.2.2. Inclusion and exclusion criteria	7
2.2.3. Quality assessment	8
2.2.4. Data collection	8
2.2.5. Data analysis.....	8
2.2.6. Deviations from protocol	8
2.2.7. Results and conclusions	8
3. Methodology.....	13
3.1. Conceptual design: select the research method(s)	13
3.2. Technical design: elaboration of the method.....	14
3.3. Data analysis	16
3.4. Reflection w.r.t. validity, reliability and ethical aspects	17
3.4.1. Internal validity	17
3.4.2. External Validity	18
3.4.3. Internal & External Reliability.....	18
3.4.4. Ethical aspects	18
4. Results	20
4.1. Research implementation	20
4.2. Contextual information on research topic	21
4.2.1. Information about the organization.....	21

4.2.2.	Information about the interviewees.....	21
4.3.	Results interviews.....	23
4.3.1.	Acceptance by customer.....	23
4.3.2	Communication.....	25
4.3.3	Complexity across customer organization boundary.....	26
4.3.4	Customer involvement.....	29
4.3.5	Quality.....	31
4.3.6	Resistance to change.....	35
4.3.7	Sales and intermediaries.....	38
4.4.	General conclusion interviews.....	38
5.	Discussion, conclusions and recommendations.....	39
5.1.	Discussion – reflection.....	39
5.2.	Conclusions.....	40
5.3.	Recommendations for practice.....	42
5.4.	Recommendations for further research.....	42
References.....		43
Appendix A.....		45
Appendix B.....		48
Appendix C.....		49
Appendix D.....		52
Appendix E.....		53
Appendix F.....		54
Appendix G.....		65
Appendix H.....		66
Appendix I.....		73
Appendix J.....		77
Appendix K.....		79
Appendix L.....		81
Appendix M.....		82
Appendix N.....		83
Appendix O.....		84
Appendix P.....		86
Appendix Q.....		87
Appendix R.....		88
Appendix S.....		89

1. Introduction

1.1. Background

Information Technology is developing rapidly and we are not even making use of all the possibilities that the internet, for example, has to offer us. These developments also influence the way software is developed and deployed (Rodríguez et al., 2017). Regardless of the type of organization (profit, non-profit or government organizations) when millions of users are using large-scale websites or mobile applications, a good and stable release pipeline is very important for organizations to meet a good release cadence according to Adams et al. (2015). DevOps is an integration of development ('dev') and operations ('ops') and aims, through collaboration, to ensure that changes to a system or application can be implemented and released much faster, without losing quality (Bass, Weber, & Zhu, 2015; Wettinger, Andrikopoulos, & Leymann, 2015).

Continuous Integration, Continuous Delivery and Continuous Deployment are DevOps practices that can provide a faster software development cycle (Bass et al., 2015). Continuous integration is a software development method that enables developers to integrate automatic tested changes frequently up to several times a day (Fowler & Foemmel, 2006). Continuous delivery is an extension of Continuous integration and enables developers to release the developed changes quickly and automatically while remaining reliable (L. Chen, 2015; Wettinger et al., 2015). Continuous deployment takes software implementation a step further by deploying every change automatically to production when it passes the automated tests (Humble & Farley, 2010).

1.2. Exploration of the topic

Continuous Integration, Continuous Deliver and Continuous Deployment are practices that belong to Continuous Software Engineering (Fitzgerald & Stol, 2017) . The literature mentions several advantages why organizations should want to apply Continuous Software Engineering. For example, a faster release pipeline ensures more frequent releases, higher customer satisfaction, better product quality and cost savings. (L. Chen, 2015; Leppänen et al., 2015; Toh, Sahibuddin, & Mahrin, 2019).

Claps (2015) describes Continuous Integration and Continuous Delivery as emerging software development processes in which developers can immediately integrate newly developed code into production. While he recognizes the advantages of CI/CD, in his research he focusses on the challenge's organizations face while implementing CI/CD.

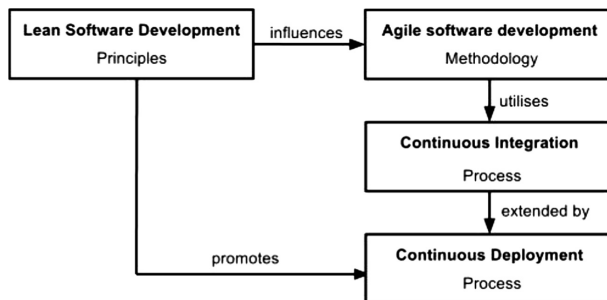


FIGURE 1 CONCEPTUAL MAP OF THE CD PROCESS. CLAPS ET AL (CLAPS ET AL.)

Claps' research (2015) has shown that when an organization is well prepared and has the right mitigation strategy, with its existing expertise, processes and tooling, it can meet the various technical and social adoption challenges. While Chen (2015) distinguishes between 3 types of challenges when adopting Continuous Delivery; organizational challenges, technical challenges and process challenges, he indicates in his article that the biggest challenges is organizational. He states that different departments with different interests require good cooperation or restructuring to avoid tensions.

For a successful implementation of Continuous Integration and Continuous Delivery, van Belzen (van Belzen, Trienekens, & Kusters, 2019) has put together an initial framework with the Critical Success Factors that play a role in this by means of a systematic literature review. This framework will be used as a basis for this research. This research is part of a broader and subdivided study. The Critical Success Factors found are divided into three

clusters; Process control/management/governance, Customer and CI/CD process. This sub-research will focus on the biggest challenge mentioned by Claps (2015) and Chen (2015) , namely the organization. More specifically, this research will focus on Process Control, Management and Governance of the CI/CD process.

The Critical Success Factors for Process Control, Management and Governance for the implementation of the CI/CD process on which this research will focus are: "Preconditions", "Objectives", "Strategy and approach", "Architecture", "Process design", "Motivation" and "Resistance to change".

1.3. Problem statement

"The software intensive industry is moving towards the adoption of a value-driven and adaptive real-time business paradigm." (Rodríguez et al., 2017)

The article by Rodríguez (2017) shows that the transformation to CD has great advantages for organizations to stay ahead of their competition, such as reducing time-to-market and obtaining faster feedback so that any problems can be found sooner. But he also mentions that the transition to CD poses a challenge for organisations.

In their articles, Chen (2015), Debbiche (2014), Ilyas & Khan (2012) and Laukkanen (2015), among others, have explored the various challenges that organizations face when adopting the CI/CD process. In his article Van Belzen (2019) has drawn up an initial framework to identify the critical success factors for the adoption of continuous processes. However, there is still a lack of measures that can ensure that these critical success factors can be optimally achieved.

The problem can therefore be described as follows:

At the moment there is no practical overview in the literature of measures to successfully implement CI/CD processes. Such a framework enables organizations to evaluate themselves so that they can prepare themselves optimally, and that the transition to the use of CI/CD processes does not become a daunting challenge.

1.4. Research objective and questions

In his article Van Belzen (2019) developed an initial framework to identify critical success factors for the implementation of continuous processes. The aim of this research is to extend this framework on the basis of existing theory by identifying and validating organizational measures for critical success factors of the CI/CD process. The main research question is as follows:

What are measures of Critical Success Factors of the CI/CD process from a process management perspective?

In order to adequately identify measures of Critical Success Factors in the literature, it is important to properly define the term 'measures'. Therefore, the following sub-question is compiled:

- *SQ1: What are measures of Critical Success Factors?*

To take a more specific and more targeted look on the Critical Success Factors that this research focuses on the following sub-questions were formulated:

- *SQ2: Which measure(s) contribute to CSFs on "Preconditions"?*
- *SQ3: Which measure(s) contribute to CSFs on "Goals"?*
- *SQ4: Which measure(s) contribute to CSFs on "Strategy and approach"?*
- *SQ5: Which measure(s) contribute to CSFs on "Architecture"?*
- *SQ6: Which measure(s) contribute to CSFs on "Process Design"?*
- *SQ7: Which measure(s) contribute to CSFs on "Motivation"?*
- *SQ8: Which measure(s) contribute to CSFs on "Resistance to change"?*

1.5. Motivation/relevance

Software development has increased in complexity in recent years. Higher expectations in the area of quality that must also be delivered rapidly offers software companies a major challenge (Debbiche et al., 2014). The continuous processes of software development make it possible for companies to deploy the new software straight away after construction and thus offer many advantages to a company (Claps et al., 2015). However, in addition to the many benefits, it also presents major challenges (L. Chen, 2015).

Kahn and Shameem (2020) indicated in their research that in the future a maturity matrix could be developed on the basis of their research to assess, for example, the capabilities of an organization in the field of DevOps. Chen (2015) puts more emphasis on architecture in his article and that in the future an extensive compiled list of ASRs (Architecturally Significant Requirements) can contribute to the implementation of CD. Bolscher and Daneva (2019) indicate, based on their systematic literature research, that a work of reference, set of guidelines or a framework composed based on their identified issues and characters can support practitioners in resolving software related issues during the adoption of Continuous practices and DevOps.

And where van Belzen (2019) set up an initial Framework with the Critical Success Factors for the adoption of continuous processes, he expressed the hope that the Framework can serve as a checklist for the successful implementation of Continuous Practices.

This research is the next step towards the development of a Framework with measures for Critical Success Factors that offers organizations a guideline for the adoption of CI/CD processes. This research will contribute to the empirical knowledge about the implementation of CI/CD processes and which measures can support organizations in this implementation. It will also provide recommendations for future research.

1.6. Main lines of approach

This report is structured as follows. In the next chapter, the theoretical framework will be formed by a literature study. In chapter three the focus will be on the methodology for the empirical research. In this chapter we will look at the conceptual and technical design, but also at the method of analysis. In the fourth chapter the results of the analysis of this research will be presented and in chapter five the findings will be discussed and the conclusions and recommendations for practice and further research will be presented.

2. Theoretical framework

This section will provide the theoretical framework. This framework will be realised by conducting a literature study. The aim of this literature study is to answer the research questions. Step one towards a theoretical framework is the preparation of the research approach. This research approach describes how the literature study will be conducted. The second section of this chapter will describe the course of the research, what the search result is and which search results were used for further investigation. The third part describes the theoretical framework derived from the systematic literature review and provides a first answer to the research questions and in the final section the Objective of the follow-up study will be explained.

2.1. Research approach

This research includes two different aspects. The first aspect is to search for additional literature related to the research of Belzen and the second aspect is to extract measures related to the Critical Success Factors from the available literature.

This research builds on the earlier research conducted by van Belzen and his report; “Critical Success Factors of Continuous Practices in a DevOps Context” (van Belzen et al., 2019). For his research the search process of Kitchenham (2010) has been applied to find the Critical Success Factors of Continuous Practices in the existing literature. A literature list was received as a first foundation. An additional study was carried out to add more recent literature related to Critical Success Factors. Appendix A provides a detailed overview of this search process and Appendix B shows the list of 53 articles that were used for the next step of this research.

In conformity with Van Belzen's approach to systematic literature review, this research will also be carried out according to the steps Kitchenham (2010) applies for the Systematic Literature Review. Systematic review is a specific methodology that locates existing studies, selects and evaluates contributions, analyses and synthesizes data, and reports the evidence in such a way that allows reasonably clear conclusions to be reached about what is and is not known. (Denyer & Tranfield, 2009) The SLR methodology aims to be as unbiased as possible by being auditable and repeatable. (Kitchenham, Pretorius, Budgen, Brereton, et al., 2010).

Kitchenham (2010) applies the following steps when conducting a Systematic Literature Review.

- Search Process
- Inclusion and Exclusion Criteria
- Quality assessment
- Data collection
- Data analysis

2.1.1. Search process

The literature list as shown in Appendix B is used to manually search whether the authors in their articles also mentioned measures that could contribute to the Critical Success Factors. The following steps will be taken:

1. Selecting the literature on the basis of CSFs
2. Reviewing the selected literature
3. Searching for a definition of the concept 'Measures'
4. Re-read through the articles to select measures that meet the concept of 'Measures'.
5. Adding the found measures to the overview

Inclusion and exclusion criteria

The measures found in the search process will then be tested against inclusion and exclusion criteria. The objective is to remove measures that may resemble measures but are not measures at all. The following criteria will be used:

Inclusion Criteria:	
•	A measure must to some extent meet the definition of measure (an action or practice that contributes to a CSF, or mitigates risk or challenges)
•	It must be clear on which part(s) of the CSF the measure has an impact and how the measure contributes to the CSF in question. (i.e. mitigate challenges, minimize risks, improve performance, address concerns, enable, support, play a role, needed/required, action, dealing with etc.)
•	A measure must be measurable qualitatively or quantitatively.
Exclusion Criteria:	
•	The identified measure is ambiguous (Based on the information from the article it is not possible to define a unambiguous definition)
•	It is very difficult to make the measure measurable

TABLE 1 INCLUSION AND EXCLUSION CRITERIA METAPLANSESSION

Quality assessment

After assessing the found measures on the basis of the inclusion and exclusion rules, Kitchenham (2010) applies the Quality assessment which allows for a refinement in order to obtain an even better result. A sound definition of quality does not seem to exist according to Kitchenham (2010) but she herself refers to the CRD Guidelines and to the Cochrane Reviewers' Handbook in which it is indicated that quality refers to minimizing bias and maximizing internal and external validity.

Term	Synonyms	Definition
Bias	Systematic error	A tendency to produce results that depart systematically from the 'true' results. Unbiased results are internally valid
Internal validity	Validity	The extent to which the design and conduct of the study are likely to prevent systematic error. Internal validity is a prerequisite for external validity.
External validity	Generalisability, Applicability	The extent to which the effects observed in the study are applicable outside of the study.

TABLE 2 QUALITY CONCEPT DEFINITIONS ACCORDING TO KITCHENHAM & CHARTERS 2007

The intention is to take a more in-depth look at the measures mentioned by the authors to see how they have reached these measures. Hereby it will be examined whether they mention these measures on the basis of previous studies or whether these are conclusions that they themselves have found on the basis of, for example, interviews and have they validated these results?

2.1.2. Data collection

The data that will be selected from the articles are the measures as described by the author in his article with the reference and the context in which it is placed. This in order to have as clear as possible during the data analysis why exactly this measure has been selected. It will also be examined whether the articles indicate how these measures could be measured. The data is recorded as follows:

CRITICAL SUCCES FACTOR			
Measure	Tekst from article	Measurability	References

TABLE 3 EXAMPLE DATA RECORDING

2.1.3. Data analysis

With the remaining measures found in the literature, which are against the inclusion and exclusion criteria, and for which a quality assessment has taken place, it will be checked during a Metaplan session whether double

measures have been found and then the measures will be categorized. Together with the fellow students and the thesis supervisors the measures found will be examined.

During the Session it will be examined under which topic a measure can be categorized. This means that for the first measure a topic is created and for the next measure it is determined whether it falls under the same topic or whether a new topic should be created and so on. This Metaplan method was developed by Wolfgang and Eberhard Schiele and will be used to reduce the measures found to 5 measures per Critical Success Factor and thus a minimum of 60 to 90 measures in total for the 19 different Critical Success Factors.

2.2. Implementation

This section describes the execution of the research on measures in the literature that was carried out on the basis of section 2.1 Search Process.

2.2.1. Search process

The framework of van Belzen (2019) contains a large number of identified Critical Success Factors. Because of, among other things, the limited time in which this follow-up research has to be carried out, the CSF's found have been divided into 3 clusters: Process control/management/governance, Customer and CI/CD process. The CI/CD process receives input and delivers output. Process control/management/governance manages the CI/CD process and delivers output to the customers. For a visual representation of the relations between the different clusters, the SADT method was selected (Ross, 1977). This is a method for Structured Analysis and Design Technique.

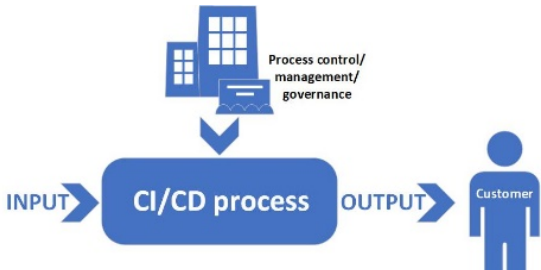


FIGURE 2 VISUAL REPRESENTATION REPRESENTATION RELATIONSHIPS CLUSTERS

Potential CSF	Cluster
Preconditions	Process control/management/governance
Goals	
Strategy and Approach	
Architecture	
Process design	
Motivation	
Resistance to Change	Customer
Complexity across customer organization boundary	
Acceptance by customer	
Sales and Intermediars	
Quality	
Customer Involvement	
Test complexity & source code control	CI/CD process
Coordination	
Communication	
Knowledge and training	
Tooling	
Pace	
Pressure	

TABLE 4 GROUPING POTENTIAL CSF'S PER CLUSTER

Appendix C gives an overview per Critical Success Factor with a description and the examples from the literature with the corresponding references. This is the starting point of this sub research.

SELECTING THE LITERATURE ON THE BASIS OF CSFS

The full literature overview provided for this study is presented in Appendix B. This is a total overview of articles for which a selection on CSF's has not yet been made.

For the provided literature it is indicated per Critical Success Factor in which articles these are mentioned. The provided literature overview is therefore reduced to an overview with articles in which "Preconditions", "Objectives", "Strategy and approach", "Architecture", "Process design", "Motivation" or "Resistance to change" are mentioned. In appendix D an overview can be found of articles from the perspective of Process Control, Management and Governance which have been studied for this sub research.

REVIEWING THE SELECTED LITERATURE

The list of 26 articles listed in appendix D has been completely read through and analysed for indicated measures for Critical Success Factors.

SEARCHING FOR A DEFINITION OF THE CONCEPT 'MEASURES'

An initial analysis of the literature shows that searching for the term 'Measures' is not a sufficient approach. When reviewing the literature, it was examined whether there is a definition of 'Measures' in the articles. Not only was this definition not found, but it also appeared that authors often use other terms when they indicate in their articles how an organization can improve itself for a successful implementation of the CI/CD process. In order to obtain a comprehensive overview of measures, it was therefore examined which terms are often used by authors. Table 5 presents the various terms. Appendix E provides a list of the context in which these terms are found in the literature. For the purpose of this study, a preliminary definition of the term Measures has been defined:

Measures of Critical Success Factors are Prerequisites, Mitigation strategies, Requirements, Adoption actions, Guidelines, Principles, Operationalized constructs, Solutions and attributes that reduce risks and contribute to an organization's successful implementation of CI/CD process.

Term	Reference(s)
Prerequisite	Debbiche et al (2014)
Prerequisite	Mårtensson et al (2017)
Mitigation strategies	Claps et al. (2015)
Architecturally Significant Requirements / Requirements	Chen (2015) (IFIP)
Adoption actions	Laukkanen (2017)
Guidelines	Laukkanen (2015)
Principles	Sahin et al (2016)
Operationalized constructs	Eck et al (2014)
Solutions	Laukkanen (2017)
Attributes	Stahl & Bosch, (2014)
Required	Debbiche (2014)

TABLE 5 ALTERNATIVE TERMS FOR 'MEASURES'

RE-READ THROUGH THE ARTICLES TO SELECT MEASURES THAT MEET THE CONCEPT OF 'MEASURES'.

With the various terms found for 'Measures' in mind, the literature has been re-examined. This resulted in 65 found measures. With the various terms found for 'Measures' in mind, the literature has been re-examined. It appeared that identifying these measures sometimes proved to be a challenge because articles often pointed out the challenges an organization might face, but not necessarily how they could solve them. Sometimes they indicated the respondents' reactions as a potential solution. Nevertheless, 65 measures were identified. This overview with the identified measures can be found in Appendix F.

2.2.2. Inclusion and exclusion criteria

The measures found were then tested against the inclusion and exclusion criteria. Appendix G contains an overview of the Inclusion criteria that the measure meets and the method of measurability. In this step, two measures were eliminated:

Measure	Elimination reason
The mindset is an important factor in the success of implementing CI. (The mindset is an important factor in the success of implementing CI. Scepticism towards the introduction of a new process needs be considered in order to win over non-believers. Debbiche (2014)	The identified measure is ambiguous (Based on the information from the article it is not possible to define an unambiguous definition)
Integrate the automated deployment of software using CD into the existing CI workflow of developers to ensure there is no, or a low learning curve (Team experience – having an experienced team is critical in the successful adoption of CD Claps et al. (2015)	The identified measure is ambiguous (Based on the information from the article it is not possible to define an unambiguous definition)

TABLE 6 ELIMINATED MEASURES AGAINST INCLUSION AND EXCLUSION CRITERIA

2.2.3. Quality assessment

In this section the intention was to apply a refinement to the inclusion and exclusion rules. Consideration was given to the basis on which authors found and presented the measures but it became apparent that this generally was not clear from the article. Because it was not uniformly assessable to determine a further depth of quality it was decided not to include this in the results and not to reduce the list of measures found in the present section.

2.2.4. Data collection

In appendix F an overview is given of the measures that were found. The measures found are listed, with the context in which it is mentioned, the possible measurability found and the reference to the author who wrote the concerning article. This list of raw data will be taken to the next step, the data analysis.

2.2.5. Data analysis

As indicated in section 2.1.5 a Metaplan session was used to perform the data analysis. Due to the precautions related to Covid-19 the Metaplan session took place through a Skype session. With two supervisors and two students all the measures that were found were reviewed, duplicated, assessed again for actually meeting the definition that was used for the measures and then categorized. These sessions took a total of 6 hours and resulted in a list of notes. This list of notes is included in Appendix H.

During this session, the 60 measures found for Process control/management/governance were grouped under 17 newly composed topics. During the session a number of measures were found to have the same scope, a number of measures were found to be more related to another Critical Success Factor and were therefore relocated and a few interrelationships became visible. The table below shows the new topics that were formed per Critical Success Factor. The extended table is shown in Appendix I.

Critical Success Factors	Topics
SQ2: Preconditions	Knowledge management
	Infrastructuur
	Strategy
SQ3: Goals	Goals
SQ4: Strategy and approach	Process
	Architecture
	Knowledge management
	Management
SQ5: Architecture	Standards
	Product architecture
	Infrastructure supporting CI/CD process
SQ6: Process Design	Organizing teams
	Process conditions
	Planning
SQ7: Motivation	Motivation
Q8: Resistance to change	Mindset and process
	Process

TABLE 7 NEW FORMED TOPICS PER CRITICAL SUCCESS FACTOR

2.2.6. Deviations from protocol

The subdivision of the CSF's was divided into 3 clusters: Process management/administration, Customer and CI/CD process. Each cluster was assigned to a student to use as a basis and to search for measures. However, due to the loss of one of the students, one component (CI/CD process) was not discussed during the Metaplan sessions because it was not completed. This will have to be addressed and discussed at a later time during a new scheduled Metaplan session.

2.2.7. Results and conclusions

This section contains the results of the research up to now and the conclusions to which they lead. A theoretical framework has been constructed on the basis of the measures found by conducting a systematic literature study.

SQ1: What are measures of Critical Success Factors?

The first sub-question is answered in section 2.2.1. Search process: SEARCHING FOR A DEFINITION OF THE CONCEPT 'MEASURES'. In the literature there is no definition of the term 'Measures' found and therefore it has been decided to formulate a preliminary definition for the purpose of the research. This definition, though, has not been empirically validated.

Measures of Critical Success Factors are Prerequisites, Mitigation strategies, Requirements, Adoption actions, Guidelines, Principles, Operationalized constructs, Solutions and attributes that reduce risks and contribute to an organization's successful implementation of CI/CD process.

SQ2 to SQ8

For sub-questions 2 to 8 the answers are listed in the table below. The Critical Success Factors are included, along with the measures that have been compiled on the basis of the measures found in the literature. Also, a description is included and the examples that were found. The composite framework of the clusters Process control/management/governance and Customer is included in Appendix J.

Process control/management/governance Measures			
CSF	Measure	Description	Examples
Preconditions	Implement Knowledge management strategy	A good shared knowledge in advance of CI and Lean	<ul style="list-style-type: none"> - Knowledge of the continuous deployment pipeline (Leppänen et al.) - Single and united organisational culture (Debbiche et al.) - A "lean" mind-set (Claps et al.) - well prepared to handle technical and social adoption challenges with their existing expertise, processes and tools (Claps et al.)
	Select a good infrastructure and resource strategy	A mature infrastructure with good resources available	<ul style="list-style-type: none"> - Mature infrastructure (Debbiche et al.) - Agile development methods are considered a prerequisite for CI (Debbiche et al.) - Hardware resources for CI servers (Claps et al.) - Adjust the architecture by changing technologies or components if needed. (E. Laukkanen et al.)
	Select an appropriate management strategy	Selecting a suitable strategy and implementing this step by step	<ul style="list-style-type: none"> - Bottom up approach (Debbiche et al.) - Step-by-step implementation (Debbiche et al.) (Kotter)
Goals	Establishing proper and well-defined goals	Making goals as clear as possible so that everyone understands what the goal is	<ul style="list-style-type: none"> - Set clear goals for teams (Debbiche et al.)
Strategy and approach	Choose a well-organized process strategy	A well-organized process with space for resilience (use of small batches, parallel systems, well-organized incident process)	<ul style="list-style-type: none"> - Well organized incident process (Claps et al.) - Adopting the practice of small batches (Claps et al.; Mårtensson, Ståhl, & Bosch) - Use parallel running systems for deployment (Claps et al.) - Resilience (Shahin, Zahedi, Babar, & Zhu) - Use document software products (Claps et al.)
	Select the right branch strategy	Choosing the industry strategy with the availability of multiple branches can be beneficial.	<ul style="list-style-type: none"> - Devising a Branching Strategy (Eck, Uebernickel, & Brenner) - Availability of several branches per software product (Claps et al.)

	Maintain a knowledge management strategy	Giving developers the time during the implementation/learning phase	<ul style="list-style-type: none"> - Overcoming Initial Learning Phase (Eck et al.) - Adopt 'social rules' which must be adhered to when deploying software. (Claps et al.)
	Implement an appropriate management strategy	Applying a suitable management strategy	<ul style="list-style-type: none"> - Management support (Shahin, Babar, Zahedi, & Zhu) (Shahin et al.) - Good overview on organization structure (L. Chen) - invest in the communication (E. Laukkanen et al.) - Clarifying Division of Labor (Eck et al.) - Devising an Assimilation Path (Eck et al.) - CI and Distributed Development (Eck et al.)
Architecture	Establishing well-defined Architecture guidelines	Provide well-described guidelines where it is made clear what the architecture must comply with	<ul style="list-style-type: none"> - Implement architecture principles (Lianping Chen) - Focusing too much on reusability can be a huge bottleneck to continuously deploying software (Shahin, Babar, & Zhu) - Isolate changes and minimize the impact of changes (Shahin et al.) - Have an architecture that supports continuous practices (flexible and modular architecture). In example loosely coupled architecture. (Eero Laukkanen, Itkonen, & Lassenius) Shahin et al. (Shahin et al.) - small and independent deployment units (Shahin et al.) - delaying (architectural) design decisions (Shahin et al.)
	Establish clear guidelines for product architecture	Provide clear guidelines for the architecture (various aspects using agile principles, such as controllability, modifiability, logability)	<ul style="list-style-type: none"> - Monitorable software application (Lianping Chen) - Modifiable architecture (Lianping Chen) - Micro-services Architectures: small and independently deployable units (Shahin et al.) - Proper logging (Shahin et al.) - Testability inside the architecture (Shahin et al.) - Choose a good system design solution. (Eero Laukkanen et al.) - Developers must think about the complete system. (Mårtensson et al.) - Implement rollback and redundancy properties. (Eero Laukkanen et al.)
	Use a good defined architecture strategy	Select the appropriate infrastructure to support the CI/CD process	<ul style="list-style-type: none"> - Use a common service bus architecture. (Eck et al.)

Process Design	Establish a collaboration strategy	Improve the (internal) collaboration possibilities with transparent communication	<ul style="list-style-type: none"> - Improved collaboration among teams and team members (Shahin et al.) - Institutionalizing CI (Eck et al.) - Teams and responsibilities (Mårtensson et al.) - Activity sequencing (Mårtensson et al.) - Clarity, Visibility and awareness of a project status to the team (Shahin et al.) - Promote a collaborative culture. (L. Chen) - Strong and proper communication and coordination between multiple teams (Shahin et al.)
	Establish the right process conditions from the start	When initiating the adoption of the CI/CD process, it is important to be well equipped and prepared.	<ul style="list-style-type: none"> - Providing CI with Project Start (Eck et al.) - Extending CI Beyond Source Code (Eck et al.) - Flexible organizational structure (Shahin et al.)
	Create a proper adoption planning	Take the start-up phase into account during the planning phase	<ul style="list-style-type: none"> - Interdependent deployment planning. (Claps et al.) - Team coordination (Claps et al.) - Give the team time to adopt CI (E. Laukkanen et al.)
Motivation	Adopt the CI/CD process with the entire company	Adopting must be a company-wide approach	<ul style="list-style-type: none"> - Company-wide effort (Claps et al.)
Resistance to change	Guide organisational aspects of change	Provide guidance in the organisational aspects of change	<ul style="list-style-type: none"> - Use a CI driver to implement CI (Debbiche et al.) - Improve communication among developers and managers. (Claps et al.)
	Guide process aspects of change	Provide guidance in the process aspects of change	<ul style="list-style-type: none"> - Work with more experienced teams (Debbiche et al.) - Step by step increase integration frequency (Debbiche et al.) - Sharing knowledge among team members (Shahin et al.)

TABLE 8 THEORETICAL FRAMEWORK MEASURES PROCESS CONTROL/MANAGEMENT/GOVERNANCE

During the literature review a number of things were observed. Contrary to expectations, only one measure was found for the Critical Success Factor 'Goals' and 'Motivation'. During the empirical phase of this research it will be necessary to investigate whether more measures can be taken in practice with regard to 'Goals' and 'Motivation'. Also, for the Critical Success Factor 'Resistance to change' not many measures have been found. For these measures, it is a good idea to talk to both management and software developers to make it clear to both parties how this Resistance can best be tackled. For the Critical Success Factor 'Preconditions' a conscious choice was made to leave 'Tooling' out of this research since a separate Critical Success Factor has been included in the overall research.

Conceptual model

Based on the complete theoretical framework, a visual representation has been made in which the Critical Success Factors and their associated measures are included. All three clusters Process control/management/governance, Customer and CI/CD process are included here. However, for the cluster CI/CD process the measures have not yet been composed, this will be completed at a later moment.

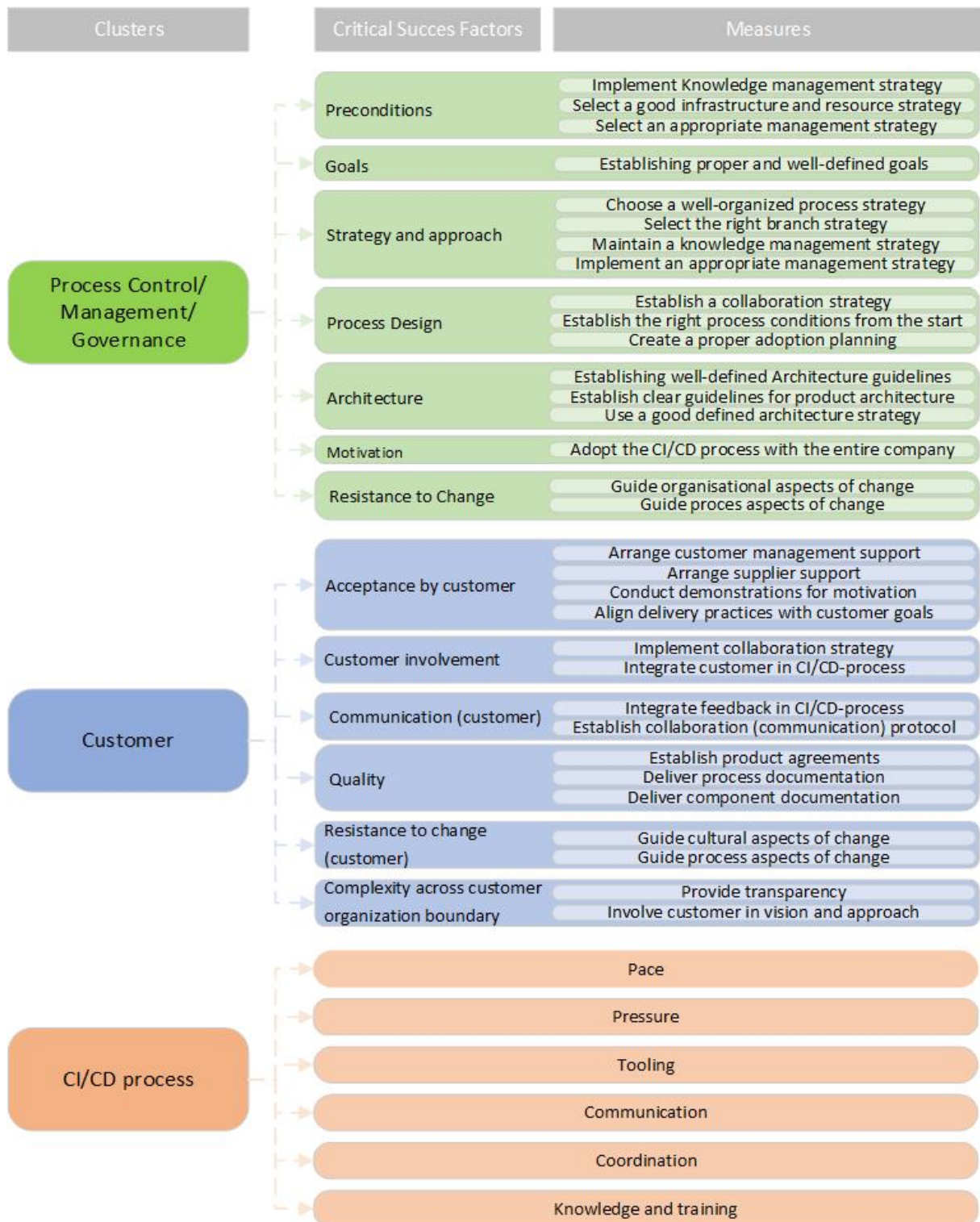


FIGURE 3 VISUAL CONCEPTUEL FRAMEWORK

3. Methodology

This section describes how the empirical research was conducted. Section 3.1 describes the conceptual design and discusses the choice of research method. In section 3.2 a description is given of the technical design in which the elaboration of the research method is explained. In section 3.3 it is described how collected data will be analysed. Finally, in section 3.4 a reflection is given on the justification of the methodical design of the research.

3.1. Conceptual design: select the research method(s)

Continuous Integration and Continuous Delivery is a process that we have studied in three separate aspects (Process Control/Management/Governance, Customer, CI/CD process). In the first part of this study, through a systematic literature review, a theoretical framework was established with measures that can contribute to Critical Success Factors in the adoption of the CI/CD process. This theoretical framework consists of the 3 different clusters with their associated Critical Success Factors and the measures found in the literature.

For the following part of the research, it is decided to focus on the measures for the customer in relation to Continuous Integration and Continuous Delivery. A reason for this decision is that from the time technical point of view it has been found more appropriate to use the available time to do a more thorough research on the customer aspect of the Critical Success Factors avoiding the risk of not gaining enough in-depth knowledge.

In order to validate and possibly supplement the measures, we will need information from the practitioners. The experiences of experts who work with CI/CD, customers who receive products through the implementation of CI/CD, documents available in the organization related to working with CI/CD could provide answers whether the measures found in the literature are indeed applied and are contributing to the Critical Success Factor.

In Saunders (Saunders, Lewis, & Thornhill, 2019), the distinction is made between a qualitative research, quantitative research and mixed method research (see Figure 4). A quantitative research refers to a research in which numerical information is collected and analysed, and a qualitative research in which non-numerical data is collected and analysed. The information that needs to be collected to validate and supplement the found measures of the technical framework are not measurable in numbers but of a descriptive nature, that makes this research a Qualitative research.

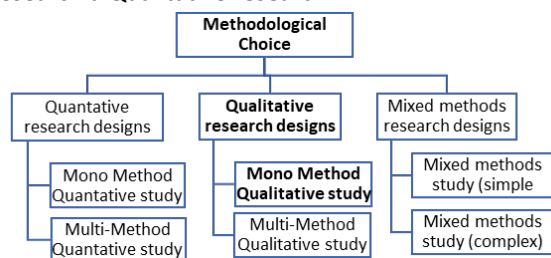


FIGURE 4 METHODOLOGICAL CHOICE (ACQUIRED FROM RESEARCH METHODS FOR BUSINESS STUDENTS (P. 176) BY MARK N.K. SAUNDERS, PHILIP LEWIS AND ADRIAN THORNHILL, 2011, HARLOW: FINANCIAL TIMES/PRENTICE HALL. COPYRIGHT 2019, PEARSON

In qualitative research, an inductive and deductive approach can be taken. In the case of this research, the first step was to examine existing theory in the literature and the next step was to empirically validate the constructed theoretical framework in practice, this means that there is an inductive approach.

There are two methods outlined in Saunders (Saunders et al., 2019) to conduct a qualitative study, the Mono Method and the Multi-Method. The Mono method involves collecting data through a single data collection strategy and the Multi method involves using more than one data collection strategies though only quantitative data collection strategies are used. Examples of mono strategies are surveys, experiments and case studies.

In order to validate the theoretical framework, a case study was chosen as a research strategy. The definition of a case study given by Yin (2018) in Saunders (Saunders et al., 2019) is “an indept inquiry into a topic or phenomenon within its real-life setting”.

By using the Case Study, we can gather information regarding the Framework in two different ways:

- Conducting interviews with Stakeholders related to the CI/CD process
- Collecting documentation related to the CI/CD process

One of the reasons for selecting a case study is the possibility to go into more depth on the different measures of the theoretical framework during the interviews. This is not achievable with a standardized questionnaire (survey) or a structured interview. An additional reason is that the research will be conducted within a limited time frame. Conducting a case study within one organization can provide more in-depth information than depositing questionnaires with multiple organizations.

3.2. Technical design: elaboration of the method

The framework that is based on the theory contains 7 different Critical Success Factors for the cluster "customer" with 20 different measures. In order to test these 20 different measures with experts in practice, the case study, as indicated in section 3.1, consists of two elements: conducting interviews with stakeholders of the CI/CD process and collecting documentation related to the CI/CD process. The objective is to conduct this research within a division within an organization which has already adopted CI/CD.

Conducting interviews with Stakeholders related to the CI/CD process

According to Saunders (Saunders et al., 2019) types of interviews can be distinguished on the basis of structure as standardised or non-standardised. The non-standardised can again be divided into 3 types of structures of interviews namely: Structured interviews, semi-structured interviews and unstructured interviews. A semi-structured interview is chosen for this research study because it fits well with the exploratory nature of the research. With this method of interviewing, there is a guideline to conduct the interview but it is possible to deviate from this in order to, for example, explore in more depth the provided answers. This also allows the focus to be on the specific area of interest to the particular stakeholder for the interview.

For this study, the approach was chosen to have a one-on-one interview so that there can be a genuine conversation with the person and his or her experiences so that the person can speak out freely and without influence from other individuals.

At the present time, the Covid-19 measures are applicable. For the organization where the interviews will take place, this means that no physical interviews can be conducted. The choice of an Internet-mediated interview over a telephone survey is to provide as close to a face-to-face interview as possible.

According to Saunders (Saunders et al., 2019) and Hanna (Hanna, 2012), synchronized electronic interviews in which interviews take place via a VoIP or web conference such as Skype, Microsoft Teams and Webex have many advantages. For example, this form of interviewing offers both the interviewer and the interviewee the possibility to choose an environment in which they feel comfortable, it is also easier to plan because the time and place are independent. However, this method of interviewing does bring a number of other important aspects, such as, for example, the privacy of the interviewee, which must remain intact. It is therefore important to make clear agreements about this. (Saunders et al., 2019).

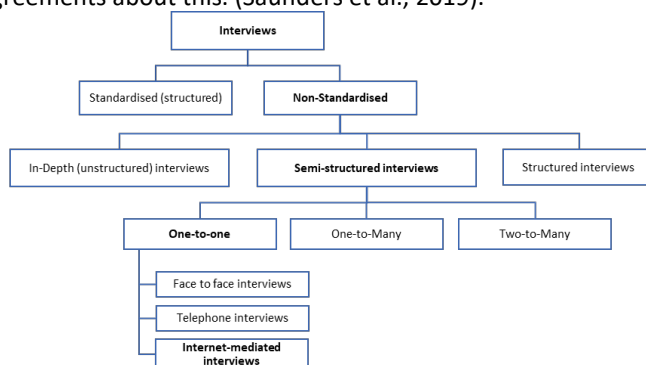


FIGURE 5 INTERVIEW STRUCTURE (ADAPTED ACQUIRED FROM RESEARCH METHODS FOR BUSINESS STUDENTS (P. 437 & 443) BY MARK N.K. SAUNDERS, PHILIP LEWIS AND ADRIAN THORNHILL, 2011, HARLOW: FINANCIAL TIMES/PRENTICE HALL. COPYRIGHT 2019, PEARSON

Collecting documentation related to the CI/CD process

By means of a desk research we will search for documents within the organization that are related to the CI/CD process. Attention will be paid to process descriptions, work descriptions, manuals etc. These are documents where practices and working arrangements are normally formulated.

Developing the interview structure

The theoretical framework is the basis of the semi-structured interview. The framework with the measures can be used as a kind of topic list, but it is important to ensure that bias is avoided and that it is clear to the interviewees what is being asked of them. Therefore, based on the framework, a questionnaire will also be prepared to be used as a guide during the interviews. Open-ended and probing questions will be used. Through the open-ended questions, the interviewee can be given the space to give an example from the practice and through the probing question, more specific attention can be given to the answer given. (Saunders et al., 2019). The interview guideline with the topics and interview questions is included in Appendix N.

Selection criteria interview participation.

Because it is not possible to interview the entire population of the government organization that works with CI/CD, a selection will be chosen. Non-probability sampling is a sampling technique often used for qualitative research and will also be used in this study. A selection is chosen based on the subjectivity of the respondent. One risk here, however, is that it relies heavily on the expertise of the researchers (Saunders et al., 2019). In order to achieve maximum variation, a heterogeneous selection was chosen, where interviewees with different roles within the CI/CD process are selected. This was chosen in order to validate measures at different ends of the process. Experts in the field of CI/CD will be interviewed which means that they have practical experience with the implementation and application of CI/CD. Because experts are chosen and there is a limited time frame in which to conduct this research, it was decided to conduct approximately 6 to 8 interviews with different "roles" within the use of the CI/CD process within an organization to look at the measures related to the customer with as broad a view as possible. This will include at least looking at employees in the functions below with experience with CI/CD.

Developers

A developer deals with the development of a software product. They should build a product based on the customer's requirements and therefore have a dependency on the customer. The Developers should therefore be able to indicate how they work with the customer to achieve a good end product.

Product Owner

The product owner is a link between the customer and the developers and represents the interests of the customer. The Product Owner should therefore be able to give a good representation of the way they work together with the customer.

Agile Coach / Consultant

An Agile Coach supports a scrum team in different areas like mindset and development. A Consultant also provides support by solving problems within a scrum team. A coach or consultant should be able to provide insight into the collaborative processes.

Functional Manager

A functional manager makes sure that an application works and when there are disruptions he tries to solve them. In doing so, he supports the users and maintains contact with the developers.

IT Architect

The IT Architect designs together with the Developers an application where he takes care of the total architecture with a robust framework. This IT Architect should be able to provide insight into involving the customer in the development of an application.

Program Manager

A program manager has a broad view to ensure that projects run smoothly. Because of his overall view he should be able to see how collaborations work and how customers are involved in a project.

End user

Within the organization, the end user is a customer. Since the focus is on the user, this user should be able to demonstrate how the measures are applied from his perspective.

In order to select the right candidates for the interviews, a number of selection criteria have been established that they must meet as a minimum.

- The interviewee must have been employed by the organization for more than 3 years

This so that the candidates are well aware of the general functioning of the organization.

- The interviewee must have been working with CI/CD for more than 2 years.

This is important so that the candidate has a solid experience of working with CI/CD.

- The interviewee must be working in IT development or be involved in IT development as a customer

The interview is conducted to empirically validate the measures of Critical Success Factors for CI/CD implementation. CI/CD is a method of IT product development. Therefore, it is important that there is a distinct relationship between the interviewee and IT.

- The interviewee generally has a positive attitude towards CI/CD

The purpose of the study is to create a framework that can support an organization in implementing CI/CD. If the interviewee has a negative attitude towards the concept of CI/CD, this can have a negative impact on the outcome of the framework. This does not mean, however, that candidates may be critical.

Pilot-interview

The purpose of this preliminary interview is to verify that all interview technologies work as required and to see if the prepared interview guide provides enough guidance to cover the various topics. It is also important to test whether the prepared questions are clear enough and whether they provide the answers that are useful for further analysis. By checking this thoroughly beforehand, problems during further interviews can be avoided and a smooth interview can be conducted with a good guarantee of construct validity (Saunders et al., 2019).

Preparing the interview

In order to inform the interviewee well in advance about the topic at stake in the interview, an information sheet (Appendix M) will be provided in advance containing important information about the research, a brief summary of the direction of the interview and information given per topic. A confidentiality agreement (Appendix L) will also be drawn up to guarantee privacy and confidentiality. These will be sent to the interviewee per email at least 2 days before the interview.

The step-by-step outline of the interview protocol is included in Appendix K.

3.3. Data analysis

After conducting the interviews, the next step is to analyse the data results. In order to do this in a proper and structured way, it was chosen to use a thematic analysis.

Thematic analysis

The thematic analysis is considered by Saunders (Saunders et al., 2019) as a general approach for conducting a qualitative analysis. This is emphasized by Braun & Clarke (Braun & Clarke, 2006) who even indicate that this systematic yet flexible approach is the basic method for conducting qualitative analysis. The main purpose of this approach is to look for themes and/or patterns in the data collection. In the case of this study, the deductive approach is followed, as the goal is to test the measures found in the literature in practice. The analysis is divided into 4 different phases that are therefore used during this research as a guideline: Familiarization with the data, coding the data, searching for relationships and/ or themes, refining the themes found and testing the propositions.

Transcribing

In order to become well known with the data collected through the interviews and to prepare the data for analysis (Saunders et al., 2019), the data must be transcribed. A copy of the audio recordings made will be saved first to avoid data loss.

The choice was made to transcribe the interviews on a verbatim basis, not including sounds other than spoken words. There are several ways to get from an audio file to a textual file but for this study it was chosen to transcribe the audio recording itself. While this is a time-consuming activity, it has the advantages that no extra money is involved and you can reflect the feel of the conversation while transcribing.

After transcribing, the transcribed file will be shared with the interviewee and asked again if they agree with the information given and whether they have any additions to the given information. When this is approved the coding can take place.

Coding

Coding is used to label the data found within the data collection by means of a word or a short phrase. With the deductive nature of this phase of research, codes are usually generated in advance based on the theoretical framework, also known as Theory Driven Codes. The use of only a deductive approach has the potential pitfall that the pre-created codes may not provide sufficient information to answer the research questions and additional codes may need to be added. By including data driven codes in an inductive way during coding, for example based on 'In Vivo', this can be avoided. (Saunders et al., 2019)

The coding is done in Dutch and initially based on the established framework. After that the transcripts will be looked at again and additional codes will be added. For coding, Atlas.ti will be used. This is a software program that can be used in qualitative research and as in this case will be used for coding the transcripts.

The coding is performed based on the theoretical framework and the questions asked during the interviews. Code groups can be created in Atlas.TI, which will be used to identify and group the measures per Critical Success Factor. There will be a unique code for each measure, so that per citation it is clear which measure is referred to. Subsequently, universal codes are distinguished to indicate which interview question of the text is related to. The table below shows the connections between the interview questions and the codes:

Interview question	Code	Application
What identified measures are applied?	MA Confirmation	This will code whether the measure is applied.
	MA Not applied	This will be coded if the measure is not applied.
	MA Substantiation	This will code the substantiation given by the interviewee regarding the confirmation or disconfirmation of the measures.
How is measure 1 (2, 3, etc.) applied? Can you cite examples?	MA Example	This will code examples given by the interviewee regarding the measures.
What are the results of measure 1 (2, 3, etc.) in the context of the associated CSF(s)? And what (e.g., measurement data) do those results show?	MA Measurability	This will code how the measures can be measured.
	MA Result Negative	This will show a negative result.
	MA Result Positive	This will show a positive result
Why are they the results of measure 1 (2, 3, etc.)?	MA Context	This will give extra information considering the measure mentioned by the interviewee.
What measures did you apply that were not in the overview?	MA New measure	This coding will be applied to new measures mentioned by the interviewee.
If you were allowed to put the measures in order of importance, how would you represent it?	NM Most Important Measure	This codes the most important measure in relation to the other measures per Critical Success Factor according to the interviewee.

TABLE 9 CONNECTION INTERVIEW QUESTIONS, CODES AND APPLICATIONS

3.4. Reflection w.r.t. validity, reliability and ethical aspects

In this section it is indicated how to ensure that the research will deliver a reliable result. In this section the Internal Validity, External Validity, Internal Reliability, External Reliability and the ethical aspects are discussed.

3.4.1. Internal validity

According to Saunders(Saunders et al., 2019), internal validity relates to the reliability of the quality of the research design. When applying semi-structured interviews, as used in this study, Saunders (Saunders et al., 2019) indicates that due to the lack of standardization, it can lead to unreliability as replication may be impossible. However, a high level of validity can still be achieved by asking clear questions, probing for meanings and selecting people for the interviews from different angles. Saunders indicates that if one manages to avoid three types of bias that good validity can be ensured. They are: 'interviewer bias', 'interviewee/respondent bias' and 'participant bias'.

In interviewer Bias, Saunders (Saunders et al., 2019) addresses that the interviewer's nonverbal behaviour, comments and tone can lead to a biased response from the interviewees. By Interviewee/Response Bias, Saunders (Saunders et al., 2019) is referring to the interviewees' perceptions of the interviewer and the possible uncertainty that interviewees have when conducting an unstructured interview which may cause them to be afraid not to share certain information because they are afraid that sensitive information will be discussed. Finally, Saunders (Saunders et al., 2019) mentions Participation Bias where the amount of time and data can take up so much time that the interviewee is less willing to participate or will be increasingly unwilling to speak during the interview. The table below shows by type of bias how this will be avoided:

Type of Bias	Measures to prevent bias
Interviewer Bias	<ul style="list-style-type: none"> An initial trial interview will be held to validate the clarity of the questions. The interview is partly scripted with standardized interview questions. During the interviews, the interviewer will have a neutral attitude regarding the topic. Each candidate will be addressed with respect and without prejudice.
Interviewee/Response Bias	<ul style="list-style-type: none"> Prior to the interview, the information will be sent to the interviewees about the purpose of the study and the measures to be discussed. Hereby, the examples will not be mentioned in order not to immediately give a direction on how the measures could be applied. It will be indicated that all information will be processed anonymously. Leading questions will be avoided. Clear language will be used. The topics will be discussed in the order they were delivered to the interviewee. If it is decided to deviate from this, this will be clearly indicated.
Participation Bias	<ul style="list-style-type: none"> This was taken into account by firstly splitting the measures into 3 different clusters, Process control/management/governance, Customer and CI/CD process. Here the choice has been made to limit the data for the interviews to the measures related to the Customer because 20 measures were found for testing in practice for this alone. Therefore a shorter list of measures will be the subject of the interview. A time frame of 60 minutes was chosen with a possible extension of 30 minutes so that the time in which the interview took place was also limited.

TABLE 10 OVERVIEW BIAS TO PREVENT AND MEASURES TO PREVENT THEM

3.4.2. External Validity

According to Saunders (Saunders et al., 2019), External validity is about the generalizability of research results. In this study, the interviews will be conducted at only one organization, so that the generalizability remains limited.

3.4.3. Internal & External Reliability

Saunders (Saunders et al., 2019) states that internal reliability is about replication and consistency but also that when using in-depth or semi-structured interviews the findings are generally not intended to be reproducible. Since the choice has been made to use semi-structured interviews, this will not be the case.

3.4.4. Ethical aspects

Saunders (Saunders et al., 2019) has compiled a table of ethical principles. This table has been used as a basis for the elaboration of the ethical justification of this research.

Ethical principle	Measures for guarantee
Integrity, fairness and open-mindedness of the researcher	To ensure this, there will be openness about the research and clarity on the subject. Because the research will be conducted within a government organization where the researcher herself works, there is no ambiguity about the integrity of the organization. Prior to the interview, a document containing information about the training, the research, and the conduct of the interview will be sent to the interviewee. (See Appendix M - Participant Information Sheet) The interview will be transcribed and sent first to the interviewee for acceptance. This will also allow them to read through again what was said and if they disagree with anything.
Respect for others	Stakeholders will be communicated with respect and openness. The rights of stakeholders will therefore be respected.

Privacy of those taking part & ensuring confidentiality of data	Since the interviews are conducted within a government organization, safeguarding privacy is of great importance. For this reason, personal data will not be mentioned in the report but will be identified on job function. During the interviews the focus will be on the content of the topics and not on the personal information of the participant. A privacy statement has been prepared for this purpose which will be signed by both the interviewer and the interviewee. Permission will also be requested in advance for the recording of the interview. (See Appendix L Privacy Statement interview)
Voluntary nature of participation and right of withdraw	Participation in the research is on a voluntary basis and the participant can withdraw from participation at any time. This right is also mentioned in the Privacy statement . (See Appendix L Privacy Statement interview)
Informed consent of those taking part	Participants will be well informed about the scope of the research in advance. The interview will be conducted using the conceptual model with the measures found per critical success factor. This has been translated into Dutch and provided to the interviewees prior to the interview. (See appendix O Conceptual Model Customer translated)
Responsibility in the analysis of data and reporting of findings	While analysing the data, no modifications will be made to the collected data and no information will be created by the researcher herself. Transcripts of the interviews will be produced so that all information given during the interviews will be traceable.
Compliance in the management of data	When conducting the interviews and analysing the found data, the privacy law as stated in the Netherlands will be taken into account.

TABLE 11 OVERVIEW ETHICAL PRINCIPLES AND MEASURES TO GUARANTEE THEM

4. Results

This chapter contains the research results. In 2.1 it is indicated how the research was conducted with the problems encountered during this execution. Section 2.2 presents the results of the case study.

4.1. Research implementation

Chapter 3 indicated that it was chosen for further research to focus on the customer perspective. The sub-research questions that are central to this process are as follows:

- RQ2: What measure(s) contribute to the CSF “Resistance to change” from a customer perspective?
- RQ3: What measure(s) contribute to the CSF “Complexity across customer organization boundary” from a customer perspective?
- RQ4: What measure(s) contribute to the CSF “Acceptance by customer” from a customer perspective?
- RQ5: What measure(s) contribute to the CSF “Sales and intermediaries” from a customer perspective?
- RQ6: What measure(s) contribute to the CSF “Quality” from a customer perspective?
- RQ7: What measure(s) contribute to the CSF “Customer involvement” from a customer perspective?
- RQ8: What measure(s) contribute to the CSF “Communication” from a customer perspective?

With these sub-research questions the literature study was conducted which led to a Theoretical Framework. This Framework was then given additional editing by the thesis supervisor that led to the Conceptual Model that was used as the basis for the interviews with the experts in the field to empirically validate the measures found. This model can be found in the Appendix P. For the purpose of the interview, the model was translated into Dutch. This can be found in Appendix O.

Conducting the interviews

A case study with interviews was chosen for this research. These interviews were conducted during the months of February and March in 2021. Due to Covid-19 measures, these interviews were conducted via Webex with video. These interviews lasted approximately 60 minutes. Six interviews were actually conducted with seven respondents in the end. These seven respondents are from different teams in order to get as comprehensive a representation of the application of the measures as possible. They all participated very willingly and, as far as technically possible, signed the privacy statement or gave their agreement by email.

The first interview also functioned as a pilot interview. It was indicated that this was the first interview so that it could be verified that the audio recording actually worked and that the measures were sufficiently clear to the respondents. During the interview, it became clear that it was sometimes difficult to remain with the questions and not give one's own opinion. Also, after the interview it became clear when listening back to the audio recording that the listening noise interfered with the audio quality of the interview.

This was taken into account during the follow-up interviews by indicating prior to the interview that I try to make as little listening noise as possible for good audio quality but so that they do not think during the interview that I am not actively listening. Also, I mentioned beforehand that during the interview I want to hear as much as possible their experiences and opinions rather than my opinion. In addition, I put the questionnaire in front of me to stay as close to the questions as possible.

During the interviews, the measures for each Critical Success Factor were discussed that, according to the interviewee, were applied. In some cases the examples were mentioned by the interviewer to get more information from the interviewee. During the interviews, the confirmations of applying the interviews and how the measures are applied were given but a result of applying these measures did not come out very well. During the literature search, no measures were found for the Critical Success Factor 'Sales and Intermediaries'. This topic was also not discussed during the interviews with the interviewees because in the government organization in question there are no sales and intermediaries.

At the conclusion of each interview, these were transcribed and sent back to the interviewees for review, along with the question whether they had any further additions. To this, responses were received back with agreement but no further additions were given.

4.2. Contextual information on research topic

In this section, in 2.1.1 there is a brief description of the organization where the data was collected and in 2.1.2 there is a representation of the respondents and to what extent they met the predefined participation conditions.

4.2.1. Information about the organization.

The organization where the case study was conducted is a government organization. This government organization is a large organization with over 30,000 employees of which approximately 3,000 people are employed in an IT function.

The organization uses some externally developed software, but develops and maintains most of the software for internal and external use itself. Because the organization has been around for many years, one of its biggest problems is that it still uses many outdated systems that require a lot of maintenance, but are also complex to revamp or even replace. This is mainly due to the high degree of interdependence and solving this is an intensive and accurate task. This is well recognized and an organization-wide program has been set up to improve ICT.

The ICT organization of this government organization is divided into several components that are often highly interdependent. In addition to maintaining and renewing applications, research is also being done on a small scale into new techniques such as Container hosting.

In terms of Continuous Integration and Continuous Delivery, the government organization is still in the early stages. There are teams that have been working with CI/CD for several years or are learning about it, but this is done with a bottom-up approach. The interviews revealed, among other things, the lack of an umbrella corporate vision in which it appears that managers and executives often have insufficient knowledge of CI/CD. There also appears to be a lack of transparency so that developers have to find out from fellow developers that and how they can get started with CI/CD. This makes it entirely impossible for non-developers to find out who is working with CI/CD within the organization.

4.2.2. Information about the interviewees.

Section 3.2 provided a number of conditions for selecting candidates for the interviews. Using these criteria, a search was conducted within the organization to find employees who met these criteria and were willing to participate in the interview.

A number of challenges emerged in the search for suitable candidates for the interview. First, networking during Covid-19 proved to be an additional challenge. The naturalness of networking at, for example, the coffee machine, during company events or by simply walking into a department does not apply now. Making contact via email, chat or phone proved impossible without the proper introduction.

It also quickly became clear that managers, who are normally excellent contacts because of their helicopter view of their processes, had insufficient knowledge of CI/CD and therefore could not appoint the right people. Candidates for the interviews therefore had to be found through known contacts. This created a certain limitation so that the chosen candidates did not always fully meet the criteria. However, it was carefully checked that they did not have insufficient experience with CI/CD to be able to give a reliable impression of the application of the measures found.

Position in the organization

In selecting the interviewees, a conscious decision was made to also look at different hierarchical roles. Below is an overview of the interviewees in a hierarchical view. By looking at the different layers in terms of position relative to the end-user, it is assumed that a complete overview with profound differences about the customer's involvement in the CI/CD process is obtained. The interviewees work at different departments/applications/projects within the government organization with the exception of the functional manager and the IT Architect, they both work at the same project.

During the search for candidates for the interviews, no end user was identified to participate in the interview, however, the functional manager is not a comprehensive member of a DevOps team yet so he was approached somewhat as an end user after all.

The hierarchical overview mainly reflects the position within the organization and thus the distance to the end user.

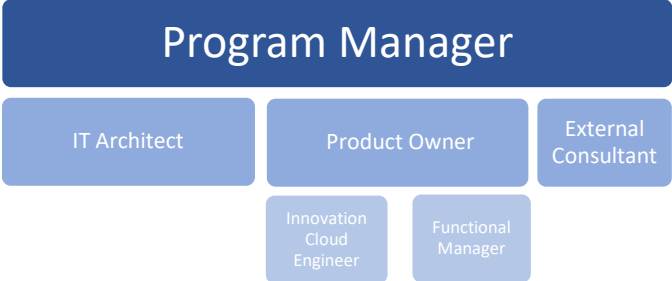


FIGURE 6 HIERARCHICAL OVERVIEW POSITIONS INTERVIEWEES

Innovation Cloud Engineer (Identified as ICE1 and ICE2 or #1)

For the first interview, the exception was made to speak to two interviewees. They are two collaborative cloud innovation engineers who have been working on innovative software development for several years now. ICE1 has been working at the government organization since 2004 and ICE2 has been working at the government organization since 2015. Both have been working as Cloud innovation engineers since August 2018. They are working on application development and research on cloud technology.

External Consultant (Identified as EC or #2)

The Consultant interviewed does not work internally at the government organization, but has been hired externally in addition to his work as a Solution Architect at another government organization. His job within the government organization is to provide support for the adoption of Continuous Integration and Continuous Delivery. The consultant has been working within the organization for two years and at the other government organization for thirteen years. Despite this, the decision was made to select him for the interview because he did work with CI/CD at another government organization for an extended period of time.

IT Architect (Identified as ITA or #3)

The IT Architect has been with the government organization for two years and has previously worked in various IT roles. As an architect, he is involved in the development of a new and replacement application developed via CI/CD. This architect also did not meet all the criteria because he has been working within the organization for less than two years but because he does have several years of experience with CI/CD, the choice was made to interview this IT Architect.

Functional Manager (Identified as FBI or #4)

The closest customer to IT is the functional manager until DevOps is chosen, at which point the functional manager is no longer a customer but a part of the team. The functional manager in question is involved in the development of a new application that has been in development for several months. This application is being developed based on CI/CD. The functional administrator has been employed by the government organization for five years and has two years of work experience as a functional administrator. The functional manager does not yet have two years of experience with CI/CD but has been selected to meet with someone who can be seen as a customer and yet also meets the other requirements.

Program Manager (Identified as PM or #5)

The Program Manager has been working within the Central Government for two years and has worked in various IT roles at different organizations since 1999. In a new role in a new department, he is engaged in renewed IT development with a strong focus on CI/CD. Because he has not yet worked for the government organization for three years, he does not meet the first criteria. However, due to his position and work experience with CI/CD, it was nevertheless decided to select him for the interview.

Product Owner (Identified as PO or #6)

The Product Owner is a product owner of the java development pipeline. He has been with the organization for three years and before that he worked as an IT Consultant for twenty years focusing on Continuous Delivery. In his role, he enables other development teams to use CI/CD by facilitating and supporting them.

Meeting selection criteria

In Chapter 3, a number of selection criteria were established in advance that the candidates for the interviews had to meet as a minimum. As indicated in the information per interviewee above, these criteria were not always met and conscious decisions were made to choose these candidates regardless. The table below is a summary of the criteria and which ones the interviewee met.

	ICE1/ ICE2	EC	ITA	FBI	PM	PO
The interviewee must have been employed by the organization for more than 3 years	yes	no	no	yes	no	yes
The interviewee must have been working with CI/CD for more than 2 years.	yes	yes	yes	no	yes	yes
The interviewee must be working in IT development or be involved in IT development as a customer	yes	yes	yes	yes	yes	yes
The interviewee generally has a positive attitude towards CI/CD	yes	yes	yes	yes	yes	yes

TABLE 12 MEETING SELECTION CRITERIA INTERVIEWEES

4.3. Results interviews

In this section, the measures per Critical Success Factor are presented with the results from the interviews. Each sub-paragraph is in the context of a Critical Success Factor and the corresponding measures are addressed in the corresponding sub-paragraphs. For each measure, the following interview questions were compiled:

- 1 What identified measures are applied?
- 2 How is measure 1 (2, 3, etc.) applied? Can you cite examples?
- 3 What are the results of measure 1 (2, 3, etc.) in the context of the associated CSF(s)? And what (e.g., measurement data) do those results show?
- 4 Why are they the results of measure 1 (2, 3, etc.)?

Each subsection begins with a table indicating which interviewee confirmed (+), denied (-), think it belongs more in the Agile base (A) or a combination in response to the first question of the interview. To answer the second interview question, a summary of how the interviewees apply the measures is then provided under the "Substantiation" heading and the cited examples of application are listed under the "Example" heading. Questions 3 and 4 regarding the results of the measures and their measurement were not adequately addressed during the interviews. Finally, if the interviewees gave a good quote regarding the measure because it indicated a situation well for example with this, it has also been added.

Responses to questions 5 and 6 are noted by critical success factors. In the end, no new measures were mentioned by the interviewees but some most important measures were mentioned by the interviewees, they are listed in the subsections.

- 5 What measures did you apply that were not in the overview?
- 6 If you were allowed to put the measures in order of importance, how would you represent it?

The list with confirmation per measure can be found in appendix R. The complete data including substantiation, examples and quotes can be found in appendix S.

4.3.1. Acceptance by customer

Description: Adopting the practice of continuous releases. Customer perception of their involvement in development and customer behaviour. Domain constraints. Feature discovery.

The most important measures for the Critical Success Factor 'Acceptance by customer' is according to the Functional Manager 'Consider customer goals and context' and according to the IT Architect 'convincing the customer'.

4.3.1.A Consider customer goals and context

Description: Consider customer goals and context when making trade-offs on speed, release frequency, security, learning curve of end-users etc.

Confirmation

	#1	#2	#3	#4	#5	#6
1A Consider customer goals and context	+	-	+	+	+	+

Substantiation

The Innovation Cloud Engineers indicated that the customer expects the application development to be carried out in a controlled and responsible manner but that in doing so the customer does not care how exactly this process is structured. They did listen a lot to the customers and collaborated with them to work out a problem. The IT Architect also indicated that they started the conversation with the customer and indicated that the customer could then also be helped well with CI/CD or with Container hosting. The Program Manager and the Product Owner put this down to the developers. The Program Manager indicated that the it is the first question they ask developers, what is the reason for wanting to implement Continuous Delivery. The Product Owner even stated that it is a very important objective and way that they deliver their facilities but ultimately the teams, the users, their customers are responsible for their own pipeline. The External Consultant explained particularly indicates that in the application of CI/CD there is a very large knowledge gap between the IT department and the customers but does not indicate how to deal with this. The Functional Manager also indicates this and says that it is very pleasant that knowledge sessions are planned to learn to deal with the new techniques. The Functional Manager also states that release frequencies are certainly being discussed but that this cannot be done optimally yet due to the structure of the government organization, which does not yet seem ready for continuous releasing.

Examples

- #4 FBI But we are always consulted on when a release is scheduled, when it is planned or if it would be better to do it on a Saturday, weekend or during the week.'
- #4 FBI knowledge sessions planned for those new techniques.

Quotes:

The best descriptive quote given during the interviews came from the functional manager:

- #4 FBI 'Ultimately, of course, they are building for us and not for themselves. They can come up with something really nice and communicate about it very openly and try to convince me, but if that totally ignores the objectives we have as a business then it's no use.'

4.3.1.B Convince the customer.

Description: Convince the customer of the benefits of continuous practices through effective communication (e.g. blogs, materials, workshops, demonstrations), high quality releases, right behaviour etc.

Confirmation

	#1	#2	#3	#4	#5	#6
1B Convince the customer.	-	+	+/-	-	-	+

Substantiation

The Innovation Cloud Engineers state very clearly that the customer does not care which way an application is developed as long as the product is delivered on time and working properly. The Product Manager does not even

want to convince but is willing to show the benefits of working with CI/CD. The IT Architect indicates that the customer does want to be convinced in a conceptual way but indicates that this is not so much about accepting CI/CD but about accepting the software. The functional manager agrees and indicates that he does not need to be convinced that she really understands that this is a more convenient way to develop. Both the IT Architect and the Product Owner indicate that it is mainly about giving insight, whereby the Product Owner indicates that the exchange of knowledge and experience could be much better within the government organization.

4.3.1.C Establish a culture of open communication.

Description: Establish a culture of open communication, for example seek permission to gather information.

Confirmation

	#1	#2	#3	#4	#5	#6
1C Establish a culture of open communication.	+	-	+/-	+		+

Substantiation

A culture of open communication is acknowledged by the interviewees but it is indicated that this is not specific about CI/CD because this is a technical domain. They mainly talk about the goals to be achieved and give explanations where necessary. The Product Owner even indicates that there should be a kind of open source culture where test reports and image jobs and even the code should be publicly available. The Functional Administrator indicates that she does experience the feeling of that open communication by being on a DevOps team where a lot is discussed. The Product Manager indicates that the open communication is a no-brainer but that it's actually already more with Agile.

Quotes

- #1 ICE: 'That really makes them very happy, that open communication.'
- #5 PM 'Yes it has to do with it but open communication is in the way you work together. Agile you need for Continuous Delivery and vice versa. So you can say this is a no-brainer, you work multidisciplinary with multidisciplinary teams. If you bring DevOps closer together then you need to have an open culture and open communication'
- #2 EC 'it's often a technical domain, a technical story to someone who just wants functionality and has no idea what it actually takes or what's going on there.'

4.3.2 Communication

Description: Intra and inter team communication, the right communication tools, awareness and transparency.

The Innovation Cloud Engineers and the IT Architect identify 'culture of open communication' as the most important measure for the Critical Success Factor 'Communication. The IT Architect states that when you take Functional Management in consideration that 'Agree on appropriate ways of working' could be seen as the most important measure for the Critical Success Factor 'Communication'. The Product Owner identifies 'Agree on appropriate ways of working' as the most important measure for the Critical Success Factor 'Communication.

4.3.2.A Establish a culture of open communication among stakeholders and create awareness.

Description: Share information and knowledge, make activities transparent, communicate frequent, involve stakeholders and create awareness.

Confirmation

	#1	#2	#3	#4	#5	#6
2A Establish a culture of open communication among stakeholders and create awareness	+	+/-	+	+	+	+

Substantiation

All interviewees agree that open communication among stakeholders is important, and this should be interactive and two-way. However, the External Consultant indicates that this is not necessarily CI/CD but Agile and that

Continuous delivery works best in a DevOps environment. The Innovation Cloud Engineers indicate that they let the customer have its say about what they need and that as long as you have an open communication where everyone dares to say what they want, you can always find a solution together. They do indicate that with a small group everyone can come into his own more and that this is for the benefit of the product. The External Consultant also indicates that you want to have a short feedback loop on what you develop. The Functional Manager indicates that the accessibility to Confluence and Jira contributes to this. The Product Owner does indicate that making everything as open as possible is not without its challenges.

Examples

- #4 FBI Knowledge sessions, but also information sessions in which an architect is also present.
- #4 FBI Use of communication tooling such as Confluence and Jira, virtual desktop via the Horizon Client

Quotes

- #1 ICE 'What we're saying is that if you make software and you have an end user then you have to show where you stand in the interim. And that just has to be a decent story with graphics to go with it.'
- #1 ICE 'The whole in between layer is not there for us. It's the tech guy talks to the customer.'
- #4 FBI 'Yes. This week I asked what error messages I could call the architect out of bed for at night. That's just a joke but you eventually go there of course. That with certain errors, that you have to be on standby on the weekends.'

4.3.2.B Agree on appropriate ways of working.

Description: Agree on appropriate ways of working among team members and with the customer, make right decisions and trade-offs during the design of the CI/CD process, and be transparent.

Confirmation

	#1	#2	#3	#4	#5	#6
2B Agree on appropriate ways of working.	+	-	+	+	+/-	

Substantiation

The interviewees indicate that there is agreement on appropriate practices but as the Innovation Cloud Engineers indicate, the customer is not a box-filling form but a user who wants well-functioning software. In this regard, the Product Manager also states that the application of Continuous Delivery has zero impact on the customer because made agreements basically do not change. Changes take place when shifting to DevOps because activities are then moved. The IT Architect emphasizes that with DevOps, the Functional Manager is also moved to the team and is therefore more involved in the delivery. The Product Owner indicates that here you would like to see an overarching business vision, but this is still missing. The Product Owner gives workshops and supports a team so that they set up their own pipeline and program their own CI/CD process and states that he wants to enable the developer much more and that this developer should be central.

Examples

- #1 ICE Meeting bi-weekly

Quotes

#5 PM 'With Continuous Delivery, there is always a need for an approval from the stakeholder or at least the one who is mandated to give the approval for going into production. And that in this case be the customer. And that process piece is not going to change overnight. If that's a process agreement then that will always stay with Continuous Delivery.'

4.3.3 Complexity across customer organization boundary

Description: No access to or control on a production environment or diversity and complexity of customer sites, which make it harder to fully automate the deployment process.

The most important measures for the Critical Success Factor 'Complexity across customer organization boundary' are according to the Innovation Cloud Engineers 'Ensure transparency on the status quo' and according to the IT Architect 'Align process dependencies towards continuous practices'.

4.3.3.A Align process dependencies towards continuous practices.

Description: Convince all actors in the process to adopt continuous practices and customize updating mechanisms of involved systems and devices.

Confirmation

	#1	#2	#3	#4	#5	#6
3A Align process dependencies towards continuous practices.			-	-	-	+

Substantiation

The IT Architect indicates with regard to this measure that there are always dependencies with other teams, and it does not matter whether they apply CI/CD or not. He also states that it is important to minimize these dependencies so that you have more freedom to configure CI/CD for yourself. Continuous Integration is very easy, Continuous Deployment is something where you have to manage all kinds of dependencies, both in the process, and we are also working on that and that is also pretty important measure to go further. The Product Owner seems to indicate this as well, that you don't need to automate the process dependencies but that you especially need to look together with the different teams at the steps each team needs to take to go from code to production. The Product Manager responds very firmly that you are not accomplishing anything with this. He indicates that at the moment the system of applications is a monolith and you cannot maintain integrity if you take one particle out and start tinkering with it. This can only be achieved if you change the application architecture and re-serviced the associated components. The External Consultant indicates that when it comes to the boundaries of an organization that you have to think mainly about DevOps and that CI/CD is on top of that but that the customer above that has much less to do with it.

Quotes

- #3 ITA 'Implicitly you can start forming teams, which is what we are doing now, but eventually you will also have to adjust your organization so that it is more in line with this, but at that point everybody then has to start working the same. Or is everyone going to work the same.'
- #4 FBI 'And I suppose that's a bit on the architects as well, tuning those dependencies.'
- #5 PM 'If one application doesn't work, then it immediately affects the other applications they are linked to. So with the answer you can be short, you can optimize processes until you weigh an ounce but that's not the marche.'

4.3.3.B Increase customer involvement.

Description: Involve organizational units which are the interface towards customers.

Confirmation

	#1	#2	#3	#4	#5	#6
3B Increase customer involvement.			+		+	

Substantiation

With this measure, the responses are straight up against each other. The Product Manager indicates you have a common goal and that this starts at the front end. The products that are created must provide a certain value for citizens and organizations. The product manager was triggered by the term "Customer Engagement" and states that this trust does not just happen, but that the customer must be involved continuously. But the involvement is according to the Product Manager not so much as an 'actor' but rather as a spectator.' The IT Architect says that the measure is mainly related to the change to DevOps by involving functional management more in the process. Which in practice is more of a Virtual DevOp team because the organizational culture is not easily transformed. The Functional Administrator agrees and says that this gap is going to be reduced in a DevOps

team. The External Consultant says that a customer doesn't have to feel anything about this except that it has to be faster and the Innovation Cloud Engineers also say that this involvement is a given when you make something that the customer is expecting. They therefore argue that if you have to make an effort to increase customer engagement whether you are making the right product.

Agile

#2 EC 'Yes that's so true but, the question is so are customer engagement and quality topics of Continuous Delivery and can you mention those there or is it better to say that a good Agile process is a success factor for Continuous Delivery'

Quotes

- #2 EC 'When I look at the government organization, there is an application there at Interaction that is for the help desk staff. Those employees who use this software have very little to do with whether that is Continuous Delivery or not. They just want functionality, so if you define that as the customer then they have very little to do with it.'
- #6 PO 'It's important that developers know that and that they are helped so you have to provide them in their development and build process reports on that, but you also have to make sure that so management, product owner or whatever, system teams that they see the overall picture a little bit.'

4.3.3.C Ensure transparency on the status quo.

Description: Be transparent on the status of development projects.

Confirmation

	#1	#2	#3	#4	#5	#6
3C Ensure transparency on the status quo	+	+/A	+		+	+

Substantiation

When it comes to transparency, the interviewees are more like-minded. The Innovation Cloud Engineers argue that you should always be transparent about the status of a development project. That this is actually an obligation towards the customer, because otherwise you quickly lose the customer's involvement and trust. The IT Architect indicates that this transparency is already being applied in the form of Agile and Safe, that it is not necessarily CI/CD but that it is necessary for this. The External Consultant also indicates that this transparency is a part of Agile and DevOps, just like trust and fast feedback. He states that Continuous Delivery works easiest when it is DevOps and DevOps doesn't really exist without Agile. The Product Owner states that when you are transparent you make sure that the relevant information is made publicly available. Yet he also indicates that the organization is not always transparent because of the island culture. It is often difficult to find all the required stakeholders.

Examples

- #6 PO dashboard or anything else to show insight and what they are working on (e.g. the time from code to production)
- #1 ICE: 'We actually had three demonstrations in a row where we didn't actually deliver anything functional for the customer because we just had way too much shit going on around it, well then you just explain what the fuss is about, what we've done with it and that may be a bit technical but I don't think we should be afraid to tell that to the customer. Plus, it also makes them think that there's more than just that screen that they get. And we explain to them what it means for them and for the organization as a whole that we are busy with other things. Because it bothers us, but then we take a bigger approach than just our own piece.'

Quotes

- #6 PO 'You have to be proud as a team as well, that's why I also think that we as a Government organization should make the code publicly available. Look how much better our code becomes when

you as a developer know that the whole of the Netherlands can see your code. If we have that transparency, I think that contributes enormously.'

- #1 ICE 'But it's not a problem to say that something didn't work out or that it's very difficult, that you don't know. What's going on just tell them, explain it to them and let them know that you're trying your hardest'
- #1 ICE 'And what's also another positive effect of really telling them what you're doing, so if you're very transparent, they have a right to know what's going on under the hood. That it is sometimes too much for them or that it is very technical then comes naturally to them. It's probably a technical story, but I think it's important to know that this and that is what's going on, and that creates confidence. That eliminates all kinds of fallacies and smokescreens that we also see teams throw up sometimes.'
- #2 EC 'I assume you have also watched the Spotify Culture videos, I keep coming back to them because they contain very good essentials. Because if a release is difficult then you tend to do it less often. If the release is easier then you just do it again. So if you come into the use, the headit of releasing is easy, then you tend to do that more often, thereby releasing becomes even easier and then you just come into the use of a lot. But that works within an agile context where you're based on trust.'

4.3.4 Customer involvement

Description: Preparing and receiving customer input, establishing a customer sample group, and delivering feature growth.

4.3.4.A Be aware of customers' situation.

Description: Use feedback mechanisms, be aware of possible barriers and even assume the role of a customer.

Confirmation

	#1	#2	#3	#4	#5	#6
4A Be aware of customers' situation.	+	+/A	+	+	+	+

Substantiation

It is clear that the measure concerning the awareness of the customer situation is applied. However, the External Consultant indicates that it is not specifically CI/CD but more of an Agile measure. Also the IT Architect thinks it is not directly related to CI.CD. The Innovative Cloud Engineers indicate that they have really looked at the employees to see how they do their work now and what goes well and what can be improved. The functional manager does indicate that, above all, something needs to be made that can be worked with and that being aware is only good if it is also acted upon.

Examples

- #1 ICE 'Actually, you should be doing their work. And if you want to do that and you're still happy or you've even become happier, then you're a good software developer. If you're not then you have to go back to the drawing board and then you have to give it to that customer first.'
- #5 PM 'So you involve them in the first session, the intakes also that the customer is asked what benefits are in it for them, if there are any.'
- #3 ITA 'We see it mostly in the form of error messages. For example, we will soon be invoked by a portal. And then the customer is with the citizen who provides his information himself without an office worker being involved. A number of our processes still have an office element in them, a kind of manual check before we can enter an appointment.'

4.3.4.B Involve the customer in the CI/CD-process.

Description: Involve the customer as actor in the CI/CD-process, take measures to get feedback, take into account customers' needs and prepare the receiving end.

Confirmation

	#1	#2	#3	#4	#5	#6
4B Involve the customer in the CI/CD-process.	+	+/A	+	+	+	+

Substantiation

Involving the customer as a measure is also not a point of discussion for all interviewees. The External Consultant emphasizes here that it belongs more to Agile than to CI/CD. The IT Architect indicates that this is only possible if the customer is close by. He mentions Functional Management as an example, but adds that when Functional Management is involved in DevOps, it is not actually a customer anymore. The Product Manager indicates that the customer is mainly involved indirectly. The Product Owner and the Functional Manager indicate that you should mainly talk to each other. The Innovative Cloud Engineers indicate that it is especially important that the right people are around the table. They often complain that only managers talk instead of the people who actually do the work. The IT Architect says that by taking the customers with him during testing and showing them the changes, you will get a great deal of involvement from the customer during the CI/CD process, which in turn will make the process run more smoothly. The Product Owner indicates that he would like to have much more feedback and input from the user community and would like to use a dashboard for this. Substantiation:

Examples

- #3 ITA So you have to set up a process to receive impulses
- #3 ITA Sit down with the client a lot more at the time when you're sparring about what a screen should look like. 'Look we now have this and is this good?' Then this is in production the same afternoon. And at that point you can also use the customer in your whole testing process by saying I'll just build something, you show it, there will be comments, you adjust it.
- #4 FBI "For example, I have all kinds of management features that I've submitted, that are picked up by the architect, who plans sessions with us to discuss exactly what we want, whether we want it in a screen or whether we want to run queries, how do you want to see your error messages, which error messages, error messages that you have to take immediate action on, do you want them in red, for example. That's all being gone through with us right now.'

Quotes

- #3 ITA 'I think if you can successfully use the CI/CD process to do stable successful deliveries then all you have is a happy customer. Then you don't have to involve it in the process but then you say it's there in two days, then he's happy too.'
- #1 ICE 'Well I think it's very important that you have the right people around the table. If you're with the wrong people, you won't get the result that's important for the party.'
- #3 ITA 'So when you talk here about "be aware of the customer's situation" then with CI/CD you actually want to respond to those customer requirements. But that doesn't just apply to CI/CD.' 4C Employ strategies to obtain accurate expectations on customers' need.
- Description: Apply different measures to get feedback quickly, develop an appropriate engagement model with customers and enable other continuous practices (e.g. continuous improvement, continuous planning).

Confirmation

	#1	#2	#3	#4	#5	#6
4C Employ strategies to obtain accurate expectations on customers' need.	-	+/A	+		+	-

Substantiation

Applying strategies to obtain accurate expectations about customer needs is confirmed by the IT Architect ("It's something you have to implement anyway because otherwise no project will go well"), the Product Manager ("part of the approach and the starting point before you start with CD") and the External Consultant. The Innovative Cloud Engineers indicate that it is part of the process and that if they were to work with a larger team, more formalization would be required than is currently the case. The Product Owner also indicates that this is not being applied well enough at the moment.

Example

- #4 FBI 'There are structurally scheduled weekly certain consultations.'
- #4 FBI 'And we then have Mattermost, which is actually a kind of chat online where you're assigned to a certain group, we have that for the pilot where we can communicate very quickly with each other and exchange information.'
- #4 FBI 'That does have the intention of building that, building dashboards for us.'

Quotes

- #6 PO 'In the past we also had some structures to also be in discussion with the developer, that has become a little less. We are now working very hard to improve our stakeholder management in order to actually apply Continuous improvement and vice versa, how do we get our users, our customers, our stakeholders optimally involved. So we are working very hard on that and I would rather have finished it yesterday than today but practice is a bit more obstinate in that respect.'

4.3.5 Quality

Description: Preserving quality and adequate documentation.

The most important measures for the Critical Success Factor 'Quality' are according to the Innovation Cloud Engineers 'Adopt social rules.' and 'Apply proper strategies and consider preconditions'.

4.3.5.A Employ appropriate strategies, approaches and guidelines on documentation.

Description: Use feedback mechanisms, be aware of possible barriers and even assume the role of a customer.

Confirmation

	#1	#2	#3	#4	#5	#6
5A Employ appropriate strategies, approaches and guidelines on documentation.	-	+		+	+	+

Substantiation

The measure regarding the use of appropriate strategies, approaches and guidelines regarding documentation is confirmed by five of the six interviewees. Only the Innovative Cloud Engineers indicate that they do not document for the customer but that the developed software should speak for itself, giving the example that you should build an application like Whatsapp, which also does not need a manual. They do this by evaluating whether they could work with the application in this way. The IT Architect says that documentation is important but that it is a challenge to do this in the most effective way. On the one hand you don't want to produce a complete user handbook every time but on the other hand the organization still expects this at the moment because otherwise employees would not understand how to work with the application according to them. The functional manager also indicates that Jira and Confluence are indeed used, where all user cases and services are written out and extensively maintained. The Product Owner indicates that they mainly use the book Continuous Delivery by Jez Humble and David Farley as a reference book because it contains good examples to ensure that everyone is speaking the same language.

Examples

- #3 ITA 'So you're going to change something in your process, have a somewhat larger adjustment in your screen and maybe even changing the name on a button from 'Submit' to 'Send'. ' That's when some departments say that they have to modify the documentation first before releasing. That is something that does clash with IC/CD, and that is indeed a challenge to get right.'
- #3 ITA 'Except that you have to be very deliberate, especially if you have an office-like application or a portal that takes your documentation into your screen. Putting those i's there and where they can click and get additional information and that you also want to have that feedback on that so that if something is not clear you can quickly adjust that. Those mechanisms are there but that's a very tricky one. What you might be able to take with you, you have those UX Designers (User Experience Designer), who will accompany you anyway during the CI/CD process to see whether everything is workable as you might have thought. So that's also a strategy, a feedback mechanism, that you can apply there. But it's a very

persistent thing and so that's where you have to add, a little bit depending on your situation, you have to think about those UX Designers or targeted documentation in your screens to make sure that this is not going to hold you up.'

- #5 PM 'So when we talk about appropriate strategy, that book is a useful starting point, as knowledge, for common language, for common framework of concepts. We use that, of course.'
- #5 PM 'We also have a solution architecture that actually kind of sets a direction, that really, say, sets the framework when we talk about implementing Continuous Delivery, that we're all doing the same thing when we get started with this. So those are actually the strategies that we use in it.'
- #6 PO 'That's also a difference between the v1 pipeline and the v2 pipeline, v2 is basically on the entire code, on all branches the teams are working on so they are informed early on of what they are doing. Again, that means you want to be able to address different aspects or different times. You are not going to bring a feature branch all the way to acceptance, you probably don't want to do static code analysis on it either. Because then the user gets "hey he's probably in that feature in that code doing some reorganizing" and then he gets hit around the ears continuously with all kinds of code issues. So we try to put our own experience into that, together with what customers want and what they come up against. So in that respect, yes.'

4.3.5.B Agree on trade-offs which affect quality.

Description: Consider trade-offs on integration frequency, security, certainty, size of increments, quality gates and involvement of stakeholders.

Confirmation

	#1	#2	#3	#4	#5	#6
5B Agree on trade-offs which affect quality.	-	+	+	+	+	+

Substantiation

Five of the six interviewees agree that reaching agreement on compromises that affect quality is applied. Here, the IT Architect indicates that you can do this when a certain maturity level of CI/CD is achieved. In particular, because of the customer confidence that is gained from a good CI/CD setup, it makes it possible to get things done faster to get this agreement. The Product Manager also refers to a maturity model and indicates that teams are self-managing and autonomous so they can determine the speed of CI/CD development themselves. He states that flexibility is one of the conditions but that you could actually see this as a condition for Agile. The External Consultant indicates that interaction with the customer is necessary to determine what is sufficient quality. The Product Owner also indicates this, that what is high quality for one person is not necessarily high quality for another. The Innovative Cloud Engineers mainly indicate that they are not currently a standard team where they do not have long-term planning and targets but they do recognize that when there is a higher release pressure there will also be a greater need for agreements.

Examples

- #3 ITA 'and one of those choices that you can make, what you can agree with the customer is that you apply something that we call FailForward. If we find a bug on a release we're going to fix that bug and re-release. That's a different choice than rollback. But that's it's a choice you have to make together with the customer.'
- #6 PO 'That's why a tool like Sonarqube is so tricky, it's very cool on the one hand but they are generic rules. And what if your application, if you all say we don't think that rule is important, do we as a company think that? Do we think we should all comply with that? So why do we think that? There are 400 rules in there. I find that a very tricky one because, as I said, those 400 rules, what do I think about them as a company, but as a department we are not involved in that at all. We do have an opinion on that but we don't go into that. We provide the facilities so that teams and management, architects can identify/define what they want to do and pursue. But in that, you have to enable and train and educate everybody and all that kind of stuff. Then you have to get the craftsmanship right. And that's also quite a tricky one. In a company where you can become senior by seniority rather than by craftsmanship.'

4.3.5.C Adopt social rules.

Description: Act cooperatively to fulfill customer expectations, such as apply customer feedback, react on warnings, broken builds and bugs, ensure compatibility and roll back if necessary. Think about the complete system.

Confirmation

	#1	#2	#3	#4	#5	#6
5C Adopt social rules.	+		+	+	+	

Substantiation

When it comes to adopting social rules, the IT Architect indicates that it is best to start with them immediately when you begin with CI/CD. The Innovative Cloud Engineers indicate that these social rules are applied implicitly, give people space to ask questions during demos and that treating each other with respect is paramount. The IT Architect indicates that by adopting social rules you take responsibility for your own toko and that you can do this through incident management and especially by proactively responding to alerts. The Product Owner critically indicates that within the organization it is not clear who will then educate the team. The Product Manager states that there is one Code Of Conduct: 'You have to want it, you have to be willing to put energy into it (i.e. you have to be enthusiastic), and you are responsible. In addition, the following applies: 'We are all people, it is okay to make mistakes, and we are direct and open but fair to each other'.

Example

- #1 ICE 'Exactly, and then give feedback that you don't enforce by making really hard agreements that is how you are put together and what you find important'
- #1 ICE 'And then you can also say what you did wrong, and they don't mind anymore. They can also make mistakes, we can also make mistakes. It doesn't matter anymore. As long as everyone knows that the mistake was made and that they are working to fix it.'
- #3 ITA 'If you see warnings or errors in your location so to speak then you have to be proactive about it. If you do that then the customer doesn't even come with bad feedback but at most with 'hey, I have to press a button three times, can't I press it once?' But that is more of a functional feedback, which is also important.'
- #1 ICE 'Being respectful of each other is above everything else of course. And inviting them to look at it, just let them know, they can always call, if necessary it can even be at night or on the weekend. You have to work on trust.'

4.3.5.D Implement measures in the CI/CD-process to preserve quality.

Description: Ensure rapid feedback and code reviews, manage artifacts and system configuration, integrate quality checks and fool proofing mechanisms.

Confirmation

	#1	#2	#3	#4	#5	#6
5D Implement measures in the CI/CD-process to preserve quality.	+		+	-	-	+

Substantiation

On the measure of implementing measures in the CI/CD process to maintain quality, the Innovative Cloud Engineers indicate that they do this mainly by using the software continuously. They do indicate that they would prefer to have an automated test story that produces reports. The IT Architect indicates that it starts with code reviews and that the beauty of container hosting is that the artifacts and system configuration are under version control. The Product Manager states that this measure is actually what Continuous Delivery does rather than a measure for implementing Continuous Delivery.

Examples

- #6 PO 'we just have standard modules for Sonarqube analysis, to send things to Sig and there are such a stack of them on the backlog, tollgates, those are the most important ones, artifact management'

- #1 ICE 'And continually looking at each other's code work'
- #3 ITA 'And we have those quality checks, sig measurements, sonarqube, the vulnerabilities, the security check, you want those all automated too. But that's more at a higher level that you don't go belly up on things like that. So we apply that and we should.'

Quotes

- #1 ICE 'You don't want your software to go down. So it does just have to be of such a high quality that you don't get called out of bed if it doesn't work or get pulled back from your vacation if it doesn't work. And you can take care of that with a CI/CD street by just testing every commit.'

4.3.5.E Apply continuous testing.

Description: Automatically test immediately after a code commit, test new features in real use, involve the customer in testing and assess changes in testing.

Confirmation

	#1	#2	#3	#4	#5	#6
5E Apply continuous testing.	-	+	+	+/-	+	+

Substantiation

The importance of continuous testing is recognized by the interviewees but in practice it is not always applied. The Product Owner facilitates continuous testing and collaborates with other teams for this purpose and also indicates that if you cannot do continuous testing you simply cannot do CI/CD. The External Consultant advocates extensive automation and indicates that customer involvement and transparency are part of this. According to him, proper and complete testing is part of Risk Management and this in turn has to do with trust. The Innovative Cloud Engineers say that if they had extra people available they would be very happy with this. The IT Architect indicates that this definitely still needs to be implemented and the Functional Administrator also indicates that it is not yet being applied but that they are working on this. The main problem here is obtaining the right authorizations and tooling, but as soon as there is a VDI available and the tool Postmen they can start with this.

Example

- #1 ICE 'We don't have monitoring on it, you would want that in addition to wanting that automated testing you also want to have monitoring on that which is automated in your development processes what components in my whole piece of software have new versions. And that's even automable nowadays in addition to your setup and some maintenance. To upgrade the components one by one to the latest version and see if it still passes the image and test if you have all that automated in your test lane you get the upgrades of your components automatically. And if that doesn't cause any problems in your submission from that test then that system itself can make a pull request to upgrade that version.'

4.3.5.F Apply proper strategies and consider preconditions.

Description: Apply continuous strategies on refactoring, improvement, monitoring, measurement, compliance, security, use, innovation etc. Consider preconditions, such as strategies to decrease technical debt, modularisation of development, reliable test environments. And take care with top-down imposition of a metric-based evaluation.

The Innovative Cloud Engineers identify 'Apply proper strategies and consider preconditions' also as the most important measure for the Critical Success Factor 'Quality'.

Confirmation

	#1	#2	#3	#4	#5	#6
5F Apply proper strategies and consider preconditions.	+		+		+	

Substantiation

The measure 'Apply proper strategies and consider preconditions' is applied by the IT architect, Innovative Cloud Engineers and the Product Manager. The Innovative Cloud Engineers do this by regularly upgrading the underlying components. And that close to applying strategies is because this is your skill as a software developer. Refactoring is mentioned not only by the Innovative Cloud Engineers but also the IT Architect, reducing your technical debt. The Product Manager uses Demming's ball here, "Plan-do-check-act. For example, how quickly is something secured and does this include quality improvement in one go.

4.3.6 Resistance to change

Description: Difficulty to change established organizational policies and cultures.

4.3.6.A Employ strategies and ways to share knowledge and skills

Description: Support the change with strategies, such as more planning how to organize the work, low learning curve, training, colocation and adding experience/coach to the team. Apply ways to share knowledge and skills via communities, demonstrations, templates etc.

Confirmation

	#1	#2	#3	#4	#5	#6
6A Employ strategies and ways to share knowledge and skills	+		+	+	+	+

Substantiation

Except for the External Consultant, the measure of using strategies and ways to share knowledge and skills is seen as important by all interviewees. Here again, the External Consultant indicates that he thinks this belongs more to Agile than specifically to CI/CD.

The IT Architect emphasizes that all measures for the Critical Success Factor 'Resistance to Change' are important and all three have their own timeline. Especially for the government organization, it is a matter of putting a dot on the horizon and then slowly working towards it. Here he also indicates that the resistance is not with the customer in their case. The Innovative Cloud Engineers also indicate that the resistance is not with the end users, but say that there is rather a not-invented-here-syndrome. Previously, they put more effort into conveying the knowledge about CI/CD but found that there was little response and thus did not have the intended effect. Functional Management states that it starts with sharing knowledge and skills but also indicates that they know of no resistance. The Product Manager also indicates that it starts with awareness by sharing the information. They do this in a bottem-up and top-down manner. The Product Owner indicates that they try to transfer knowledge in different ways such as workshops and road shows but notices that it is not always possible to do this properly.

Examples

- #6 PO 'What we try to do at least is what we do to get that to our users optimally though. So we have the delivery pipeline workshops, we have some other workshops on Enterprise Java development, we've done Roadshows in the past with teams and we try in as many ways as we can to do that.'
- '#1 ICE Our strategy was to engage with the reviewer, see what the positions are and ultimately let the future tell then.'
- '#1 ICE 'Well and during the department's monthly event, make nice presentation and prepare nicely so we have a nice story with a new topic and the latest technology.'
- '#1 ICE We also really did experience resistance when we explained what we were doing to people to really engage with them together. But someone was afraid and had the impression that their legs were being cut. And then he really goes on the defensive and you get really weird situations. I personally had a situation once where someone said to me 'I don't want you to interfere with my business', and I was really shocked. 'You shouldn't interfere, go away'. I was glad then that I wasn't alone with that person. People feel a bit threatened by it.'

- #5 PM 'So that's a little thing called, you start structuring and you provide that ability structure that knowledge and skills so presenting, doing together, doing by yourself. That's a mechanism that's used. And that does fit into this measure here, that's an example of that.'
- #5 PM 'Also a very important measure is that decision making has to do clear and open. So how does the decision-making process take place and what do we take a decision on, so to speak. What, for example, is the Works Council involved in and how are employees involved, so to speak, so that is also an intervention, so to speak, a measure that you can take, which is not yet in here where there is definitely attention. Well, we already had knowledge, so to speak, the measures for knowledge injection.'
- #5 PM 'There we have to make sure that you get to the right level of knowledge. So it actually starts with the injection of knowledge. And only when you have that can you take the next step and then you look at maturity levels and then you see that you have to reach a certain maturity level in terms of knowledge so to speak and only then can you start filling in Continuous Delivery. Because if you don't do that, then I'll give you a Ferrari, for example, even though you don't have a driver's license yet. You can say that you have a very nice Ferrari but you will probably crash it within an hour.'
- #6 PO 'So for that pipeline we have a template which makes so users easy to get started, so that lower learning curve.'
- #6 PO 'The adding of adding experience/coach to the team. Of that we actually say we're not into that. We get that question from time to time but as a Java development line we don't have the capacity for that. Teams have to solve that within their chain, solve it within the community or whatever.'

Quotes

- #3 ITA 'Yes all three important and all three have their own timeline and pace.'
- #4 FBI 'Sure, it starts with sharing knowledge and skills because unknown makes unloved.'

4.3.6.B Align strategies and policies, and establish a proper culture.

Description: Align rules, regulations, policies and strategies and establish a culture of open communication.

Confirmation

	#1	#2	#3	#4	#5	#6
6B Align strategies and policies, and establish a proper culture.			+	+	+	+

Substantiation

Four of the six interviewees indicated that the measure related to establishing the right culture is important within the government organization. And that right culture can be supported, according to the Product Manager, with well-designed information facilities. This is also indicated by the IT Architect, that a culture of open communication where it should also be possible to say when things have not gone well without being looked at. The IT Architect also indicates that resistance can come from unexpected quarters instead of the customer and that this is often because it is new and people tend to stick to what is known: "because we have always done it this way and we know how it will go". The External Consultant also indicates that the change in culture is the biggest problem and that therein lies also the norms and values. The Product Owner indicates that CD only works in a trinity of the organization, culture and technology. And that when an organization is divided into all separate pieces it becomes an almost impossible task to support this optimally.

The Innovative Cloud Engineers identify 'Align strategies and policies, and establish a proper culture.' as the most important measure for the Critical Success Factor 'Resistance to change'.

Examples

- #3 ITA 'Well maybe you know it from Fokke and Sukke. They are a couple of planners and they say: 'Well Thursday at a quarter past ten we are going to do the culture change. Well that's not going to work. You have to have a team that says it wants to do it. You have to invest in knowledge but also in learning on the job, just do it and let them make mistakes. You have to go for the result but you also have to accept that things will go wrong sometimes.'

- #5 PM 'But also, for example, leadership. So instead of a team manager who is very much in charge, he or she must actually convert, adopt a different kind of leadership, such as coaching. Let the idea come from the team and stimulate that too.'
- #4 FBI 'We have made agreements: "If you don't test, we can't release and then you are responsible or we release without you having tested, just say it. Well then you can address each other on those rules I think. And then I think you communicate at those moments when you get stuck somewhere and then you look for a solution together. But then you also take responsibility for things.'

Quotes

- #3 ITA 'We have a bit of a strange situation where the container hosting platform steering committee says they're not ready for it, there's not enough set up, and so we've said it's going to be deployed both on container hosting platform and on IPAS. That can probably be done without too many problems but so it indicates the coldness that is there for it. In itself we can maintain our CIU/CD so to speak, fortunately because otherwise we would be much further from home. So our pipeline we maintain, so our Continuous Integration. But with our Continuous Deployment, we had a set-back because even IPAS, our backup platform, was like, we're not in this world for that. So I think it's very funny that the resistance, in this particular case, is coming from our container hosting platform steering committee that would actually embrace this completely'

4.3.6.C Ensure appropriate norms, values and behaviour.

Description: Ensure top management support and leadership on continuous improvement, budgeting and tooling. Give the development team ownership and trust. Create awareness, the right culture and mindset.

Confirmation

	#1	#2	#3	#4	#5	#6
6C Ensure appropriate norms, values and behaviour.			+	-	+	+

Substantiation

For the measure 'ensure appropriate standards, values and behaviour', little additional information is actually given by the interviewees because it is seen as part of the culture adjustment. The Product Owner does indicate that it is important what you propagate that you also have to perform. The IT Architect also indicates that this is important. In an example, he indicates that the facilitating club in particular is not as approachable as you would like. The Functional Manager also indicates that many people are still not addressed about their behaviour.

Examples

- #3 ITA "But the resistance, strangely enough, we find with the container hosting platform. They say that, as a platform, they are not ready at all, while we want to go much faster and much further as a project. Very tellingly, if you have a question there, you can go to the website and you can fill out a form to ask that question and to request consultancy. And that's kind of like the container hosting platform. '
- #5 PM 'And one of the examples is of course the moment you get more Ops activities that the function changes. So standby walking then becomes something that is going to come within a job description. So you see that the norm changes there, the norm becomes that you become responsible for much more and actually have to be available 24/7 actually, in order to support an application or system. So you see that it's going to have an impact on the function but also the norms are changing and with that comes a certain kind of behavior. So that's an HR aspect you have to take into account and help people to understand what's going to happen, what it means to them. But you also have to make sure that things are arranged, such as balancing or otherwise. In any case, you have to be able to compensate for this in the end.'

Quotes

- #6 PO 'I do think 'practise what you preach', what we propagate we should do ourselves...'

4.3.7 Sales and intermediaries

Description: When user data is not accessible due to intermediaries.

There were no measures found in the literature related to the Critical Success Factor Sales and Intermediaries to test with the experts. The topic was also not addressed during the interviews because Sales and Intermediaries are not an issue for the government organization.

4.4. General conclusion interviews

In this chapter, interviews with practice experts of Continuous Integration and Continuous delivery were used to examine whether the theoretical results as presented in the Theoretical Framework (Appendix P) are actually applied in practice. The interview results were analysed by coding and analysed by measure.

It can be concluded that the measures found in the literature for the critical success factors of customer acceptance, communication, customer involvement, quality and resistance to change were confirmed by the majority of interviewees with some modifications. Examples of how each measure is applied were also provided. However, the results achieved through the application of the measures did not emerge in the interviews. For the Critical Success Factor "sales and intermediaries", no measures were found in the literature and therefore could not be validated during the interview.

Chapter 5 will further discuss the answering of the research questions for each Critical Success Factor reflecting the adaptations to the framework.

5. Discussion, conclusions and recommendations

This chapter presents a discussion of the research findings. In the process, the research question is answered and discussed. Section 5.1 presents the discussion and reflection of the research. In Section 5.2 the conclusions of the research are discussed. In section 5.3, some recommendations for practice are indicated and in section 5.4, recommendations for follow-up research are given.

5.1. Discussion – reflection

This section provides a reflection on the quality of the study and the value of the conclusions made based on the measures previously established in Section 3.4.

Internal validity

As indicated in 3.1.4, the risk of semi-structured interviews is that lack of standardization can lead to unreliability but this can be limited by avoiding three types of Bias, interviewer bias', 'interviewee/respondent bias' and 'participant bias'.

The interviews were partly conducted according to script by using standard questions after which there was room to elaborate on questions. A trial interview was held to see if the questions provided sufficient clarity. This showed that a good preparation gave more comfort during the interview itself. It also appeared that I found it difficult not to give my personal opinion and to stay with a neutral attitude. This was then taken into account during the execution of the follow-up interviews by indicating beforehand that I mainly wanted to hear the story of the interviewee and that I would therefore remain neutral in the questions to be asked.

By informing the interviewee in advance of the Measures that would be covered during the interview, the interviewees had a reasonable idea of what was expected of them. Sometimes it was necessary to pull out the examples to clarify the measure a little more. This depended a lot on how the interview went. During the interview the order of the framework was occasionally deviated from because the interviewee gave an answer that was a good bridge to another Critical Success Factor.

By only focusing on the measures for the customer, the information during the interview and the timeframe was more manageable. Most interviews were between sixty and ninety minutes total with an intro beforehand. After the interview, interviewees even stayed to chat for a while.

To increase replicability, the interviews were transcribed and coded as found in Appendix S. Because interviews were only conducted at one organization, there is a limitation on replicability. However, it was chosen to interview employees from different departments and different roles to still get a broad view of the organization. Because the case organization is a government organization, the results are more likely to be the same in a similar organization than when compared to a commercial organization.

Ethical aspects

With regard to the ethical aspects surrounding the conduct of interviews and the handling of sensitive data, the measures of 3.4.4 were followed. The privacy of the interviewees was taken into account, they were addressed with respect and their participation was voluntary. The interviews were developed into transcripts that had been submitted to the interviewees for review for agreement.

Limitations

There are a couple of limitations that were encountered during the interview process.

Maturity of CI/CD use case organization

When searching for employees for the interviews and during the interviews, it became apparent that the experiences with CI/CD are still limited and that the organization is not yet mature enough for a good and complete application of CI/CD. This is also reflected in the answers given where no results emerge from the application of the measures but also that answers are often given on the basis of a theory.

Transparency case organization

The organization where the interviews were conducted is a government organization that would like to be transparent but is not exactly that. This made it very difficult to get a clear idea of which actors within the organization were working with CI/CD and who could be approached for an interview. In the end a number of channels were set up within the own network so that the interview could be held with several employees. However, choices were made with regard to the selection criteria as explained in Section 4.2.2.

Number of interviews.

There were six interviews conducted with seven employees for this research. For the results, more interviews could have been held for better validation of the measures. Also, the limitation of one case organization in this case is too limited to state that the measures are actually applicable. A full theoretical saturation was not achieved with six interviews.

Information interviews

The interviewees were very enthusiastic and provided a lot of information. The focus of the interview was clearly communicated beforehand but still things turned out to be not quite clear during the interview. Sometimes it proved difficult to get more information when the interviewee stuck to a point of view. A lot of information was given that was certainly interesting but was less suitable as an answer to the questions asked.

5.2. Conclusions

Because there is currently no overview available to organizations of measures to successfully implement CI/CD processes in the literature, this research has focused on the question: 'What are measures of Critical Success Factors from a customer perspective of the CI/CD-processes?' This paragraph will answer this research question. The central research question is divided per Critical Success Factor, in the coming subsections an answer will be given per Critical Success Factor as to which measures contribute.

RQ2: Resistance to change

The measures found in the literature for the Critical Success Factor "Resistance to Change" were all confirmed by the interviewees. However, examining the resistance to CI/CD, one ends up not with the customer but with the organization itself. The culture change that must occur within the government organization is one that takes time. The IT Architect stated during the interview, "Yes, all three measures are important and all three have their own timeline and pace. The functional manager especially indicated, "Sure, it starts with sharing knowledge and expertise because unknown makes unloved.

The information provided during the interview about the measures of Resistance to Change did not provide additional information for the theoretical framework but it could be questioned whether this Critical Success Factor belongs among the Factors related to the customer.

RQ3: Complexity across customer organization boundary

Of the measures found in the literature related to the Critical Success Factor "complexity across customer organization boundaries", only the measure related to transparency of the status of development projects can be confirmed based on the interview. The measure "Increase Customer Involvement" might have been better understood with a better substantiation or with more examples from the literature. Now the interviewees mainly stuck to customer involvement in a very literal way. The measure 'Align Process Dependencies with Continuous Practices' was actually only confirmed by the Product Owner and negated by three interviewees with the justification that it was better to make sure there were less dependencies on other departments. This could confirm that the measure is not applicable, for at least this organisation.

RQ4: Acceptance by customer

The measures found in the literature related to "Acceptance by Customer" have all been confirmed during the interviews with a modification on the measure related to "Convince the Customer". When it comes to convincing the customer to adopt the use of CI/CD, both the application development side and the customer agree that the customer does not need to be convinced as long as well-functioning applications are delivered. Based on the substantiation and provided examples, the measure is confirmed as applied but the action lies more with the

developers to deliver good applications than with the customer to convince them. Thus, the application of the measure lies more on the development side and would be better stated as follows: 'Present the benefits of CI/CD to the customer'.

RQ5: Sales and intermediaries

No results were found in the academic literature regarding the sub-question related to the measures 'Sales and intermediaries'. During the interviews with the experts, this Critical Success Factor was not addressed either because there was no information to validate.

RQ6: Quality

All measures found in the literature related to the Critical Success Factor "Quality" were confirmed during the interview. With proper strategies, continuous self-use of the software, making the artifacts part of the development process, applying good agreements with regard to quality compromises, adopting social rules right from the start and developing an application that speaks for itself with any supporting documentation, high quality can be achieved when applying CI/CD according to the interviewees. This quality will improve even more when continuous testing is applied, but the organization still needs to become more mature in this process. The information given during the interview regarding the measures of Quality did not provide additional information to the theoretical framework.

RQ7: Customer involvement

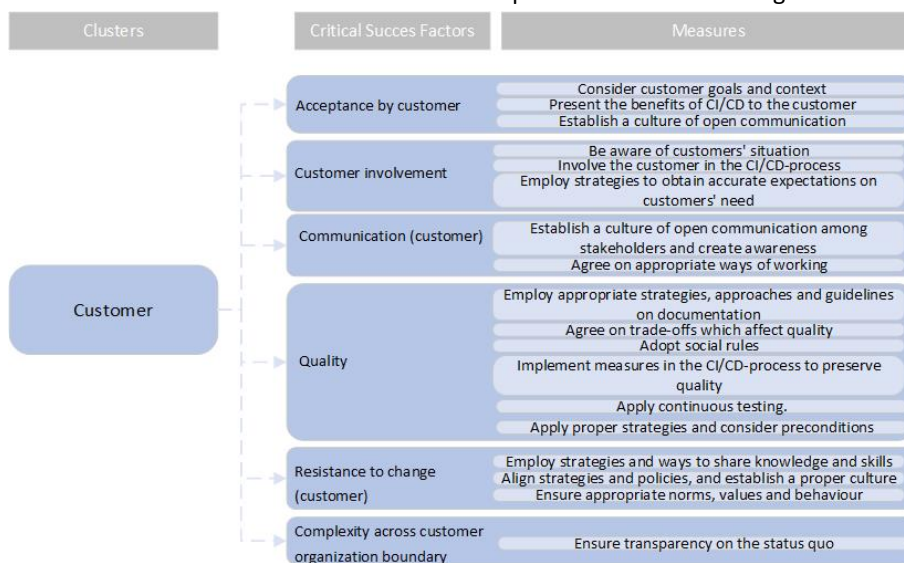
All three measures found in the literature related to "Customer Engagement" were confirmed in the interviews. By looking carefully at the customer's needs and the work that they will be using the applications for and involving the customer properly in the development process, the development process can run smoothly. The information given during the interview regarding the measures of Customer involvement did not provide additional information to the theoretical framework.

RQ8: Communication

The two measures regarding 'Communication' found in the literature were confirmed by the interviewees. With open communication and appropriate tooling in which a short feedback loop can be maintained and parties are allowed to communicate openly, knowledge can be exchanged quickly and a good product developed. The information given during the interview regarding the measures of Communication did not provide additional information to the theoretical framework.

Overall Conclusion

This research has mainly provided confirmation per measure but not so much additional data on the measures in the form of results of application. To answer the main question of this research 'What are measures of Critical Success Factors from a customer perspective of the CI/CD-processes?' the following practical framework is developed based on the literature research and the empirical validation at the governmental organization.



5.3. Recommendations for practice

With Rapid Developments in Information Technology and companies looking for ways to accelerate the software development cycle, this research was designed to develop a framework that organizations can use to evaluate themselves and properly prepare for the implementation of CI/CD processes. The result of this research, the practical framework as shown in the conclusion, consists of measures per Critical Success Factor that can support companies in this implementation. This framework can be used as a roadmap before, during and after the implementation of CI/CD processes to continue evaluating how the organization is doing with regard to the implementation of CI/CD and the relationship with the user. However, it should be taken into account that the framework in this study was only validated at a government organization and that perhaps follow-up research could result in additional measures or more substantiation of the measures.

5.4. Recommendations for further research

The focus of this research was the search and empirical validation of measures for Critical Success Factors directed at the customer for the implementation of CI/CD processes. Based on the research, a number of recommendations for follow-up research have been formulated.

Conducting interviews with the found measures from the other clusters

For this research, it was chosen to concentrate only on the measure for the critical success factors that focus on the customer. However, the overarching research was looking for measures that also apply to Process control/management/governance and CI/CD process. This report started by examining the measures in the literature related to Process control/management/governance but they were not tested with experts in this report.

Validate the measures found with other (smaller) organizations with a smaller distance to the customer

The interview for customer-related measures was only tested at a large government organization for this report. This government organization is a very large organization so there is sometimes a great distance from the customer. When there is a smaller distance between development and the customer this could give other results or on the contrary an extra validation with regard to the found measures that they are applicable in organizations independent of the distance to the customer.

Maturity of CI/CD use case organization

The organization in which the interviews were conducted is an organization that is still in the early stages of implementing CI/CD. If the measures are validated in an organization where the implementation of CI/CD has been in place for a longer period of time, this could provide a different result on the measures found.

Results of application measures

During the interviews for this study, confirmations of the application of the measures were particularly noted, rather than the results that the application of these measures actually provides for the organization. A follow-up study of the results of the applications of these measures could provide an extension of the framework.

Measures regarding to the CSF Resistance to change

The measures validated with regards to the customer brought the response that the resistance with regards to CI/CD was not really found with the customer but often in other places in the organization. The measures for this Critical Success Factor could be validated with other organizations to see if this resistance does occur there with the customer or if the resistance also occurs in other companies in other places in the organization.

Measures regarding to the CSF Complexity across customer organization boundary

The measures found regarding the 'Complexity across customer organization boundary' were not really applicable for the government organization. Therefore, these measures could be tested with other organizations to really exclude that these measures are not applicable for a successful adoption of CI/CD processes.

References

- Adams, B., Bellomo, S., Bird, C., Marshall-Keim, T., Khomh, F., & Moir, K. (2015). The Practice and Future of Release Engineering: A Roundtable with Three Release Engineers. *IEEE Software*, 32(2), 42-49. doi:10.1109/MS.2015.52
- Bass, L., Weber, I., & Zhu, L. (2015). *DevOps: A software architect's perspective*: Addison-Wesley Professional.
- Bolscher, R., & Daneva, M. (2019). *Designing Software Architecture to Support Continuous Delivery and DevOps: A Systematic Literature Review*. Paper presented at the ICSoft.
- Braun, V., & Clarke, V. (2006). Using thematic analysis in psychology. *Qualitative research in psychology*, 3(2), 77-101.
- Chen, L. (2015). Continuous Delivery: Huge Benefits, but Challenges Too. *IEEE Software*, 32(2), 50-54. doi:10.1109/MS.2015.27
- Chen, L. (2015). *Towards architecting for continuous delivery*. Paper presented at the 2015 12th Working IEEE/IFIP Conference on Software Architecture.
- Claps, G. G., Svensson, R. B., & Aurum, A. (2015). On the journey to continuous deployment: Technical and social challenges along the way. *Information and software technology*, 57, 21-31.
- Debbiche, A., Dienér, M., & Svensson, R. B. (2014). *Challenges when adopting continuous integration: A case study*. Paper presented at the International Conference on Product-Focused Software Process Improvement.
- Denyer, D., & Tranfield, D. (2009). Producing a systematic review. In *The Sage handbook of organizational research methods*. (pp. 671-689). Thousand Oaks, CA: Sage Publications Ltd.
- Eck, A., Uebernickel, F., & Brenner, W. (2014). Fit for continuous integration: How organizations assimilate an agile practice.
- Fitzgerald, B., & Stol, K.-J. (2017). Continuous software engineering: A roadmap and agenda. *Journal of Systems and Software*, 123, 176-189. doi:<https://doi.org/10.1016/j.jss.2015.06.063>
- Fowler, M., & Foemmel, M. (2006). Continuous integration. In.
- Hanna, P. (2012). Using internet technologies (such as Skype) as a research medium: A research note. *Qualitative research*, 12(2), 239-242.
- Humble, J., & Farley, D. (2010). *Continuous Delivery: Reliable Software Releases through Build, Test, and Deployment Automation (Adobe Reader)*: Pearson Education.
- Ilyas, M., & Khan, S. U. (2012). *Software integration model for global software development*. Paper presented at the 2012 15th International Multitopic Conference (INMIC).
- Khan, A. A., & Shameem, M. (2020). Multicriteria decision-making taxonomy for DevOps challenging factors using analytical hierarchy process. *Journal of Software: Evolution and Process*, e2263.
- Kitchenham, B., Pretorius, R., Budgen, D., Brereton, O. P., Turner, M., Niazi, M., & Linkman, S. (2010). Systematic literature reviews in software engineering—a tertiary study. *Information and software technology*, 52(8), 792-805.
- Kitchenham, B., Pretorius, R., Budgen, D., Pearl Brereton, O., Turner, M., Niazi, M., & Linkman, S. (2010). Systematic literature reviews in software engineering – A tertiary study. *Information and software technology*, 52(8), 792-805. doi:10.1016/j.infsof.2010.03.006
- Kotter, J. (2007). Leading strategy-why transformation efforts fail. *Harvard Business Review*.
- Laukkanen, E., Itkonen, J., & Lassenius, C. (2017). Problems, causes and solutions when adopting continuous delivery—A systematic literature review. *Information and software technology*, 82, 55-79.
- Laukkanen, E., Paasivaara, M., & Arvonen, T. (2015, 3-7 Aug. 2015). *Stakeholder Perceptions of the Adoption of Continuous Integration -- A Case Study*. Paper presented at the 2015 Agile Conference.
- Leppänen, M., Mäkinen, S., Pagels, M., Eloranta, V.-P., Itkonen, J., Mäntylä, M. V., & Männistö, T. (2015). The highways and country roads to continuous deployment. *IEEE Software*, 32(2), 64-72.
- Mårtensson, T., Ståhl, D., & Bosch, J. (2017). *Continuous integration impediments in large-scale industry projects*. Paper presented at the 2017 IEEE International Conference on Software Architecture (ICSA).
- Rodríguez, P., Haghighatkah, A., Lwakatere, L. E., Teppola, S., Suomalainen, T., Eskeli, J., . . . Oivo, M. (2017). Continuous deployment of software intensive products and services: A systematic mapping study. *Journal of Systems and Software*, 123, 263-291.
- Ross, D. T. (1977). Structured analysis (SA): A language for communicating ideas. *IEEE Transactions on software engineering*(1), 16-34.
- Saunders, M. N. K., Lewis, P., & Thornhill, A. (2019). *Research methods for business students*.

- Shahin, M., Babar, M. A., Zahedi, M., & Zhu, L. (2017). *Beyond continuous delivery: an empirical investigation of continuous deployment challenges*. Paper presented at the 2017 ACM/IEEE International Symposium on Empirical Software Engineering and Measurement (ESEM).
- Shahin, M., Babar, M. A., & Zhu, L. (2016). *The intersection of continuous deployment and architecting process: practitioners' perspectives*. Paper presented at the Proceedings of the 10th ACM/IEEE International Symposium on Empirical Software Engineering and Measurement.
- Shahin, M., Zahedi, M., Babar, M. A., & Zhu, L. (2019). An empirical study of architecting for continuous delivery and deployment. *Empirical Software Engineering*, 24(3), 1061-1108.
- Toh, M. Z., Sahibuddin, S., & Mahrin, M. N. r. (2019). *Adoption Issues in DevOps from the Perspective of Continuous Delivery Pipeline*. Paper presented at the Proceedings of the 2019 8th International Conference on Software and Computer Applications.
- van Belzen, M., Trienekens, J., & Kusters, R. (2019). Critical success factors of continuous practices in a DevOps context.
- Wettinger, J., Andrikopoulos, V., & Leymann, F. (2015). *Enabling DevOps collaboration and continuous delivery using diverse application environments*. Paper presented at the OTM Confederated International Conferences" On the Move to Meaningful Internet Systems".
- Yin, R. K. (2018). Case study research and applications. In: Sage.

Appendix A

Systematic Literature Review in steps

This appendix provides an overview of the steps that have been taken to obtain an overview of the literature that has been used as a baseline during this research. It first describes the literature review process as performed by van Belzen in his report "Critical Success Factors of Continuous Practices in a DevOps Context" (van Belzen et al., 2019). He made use of the Systematic Literature Review steps of Kitchenham (Kitchenham, Pretorius, Budgen, Brereton, et al.). Then, step by step, it was described how an additional study was carried out to add new literature related to Critical Success Factors to the literature overview.

Literature review by van Belzen (van Belzen et al.)

Six databases were used and one broad indexing service: IEEE computer society digital library, ACM digital library, SpringerLink, Web of Science, Citeseer and SCOPUS. The search period was between 2001 and June 2019. After a screening for unique articles, a first screening based on abstraction and then a first screening based on the full text, 19 articles remained. Further information regarding the search conduct that was carried out by van Belzen, M., Trienekens, J. & Kusters, R. (2019) can be found in section 3.1 'Research Methodology' of their article.

Additional Literature review

Because the research of van Belzen was carried out until June 2019, an additional literature review was carried out to verify whether articles were published after June 2019 regarding Critical Success Factors of the CI/CD process. This search was conducted with the same research process as van Belzen (2019).

DATABASES AND FIELDS USED

The following databases were used for the additional literature review:

Refnr	Database	Search field
1	IEEE computer society digital library	Abstract
2	ACM digital library	All fields (Full text)
3	SpringerLink	All fields
4	Web of Science	All fields
5	SCOPUS	Title, abstract and keywords
6	Citeseer	Abstract

SEARCH STRING

The following search strings have been used to search the databases:

Refnr	Database	Search string
1	IEEE computer society digital library	("Continuous Delivery" OR "Continuous Deployment" OR "Continuous Integration") AND DevOps
2	ACM digital library	
3	SpringerLink	
4	Web of Science	
5	SCOPUS	
6	Citeseer	Three separate search strings: "Continuous Delivery" AND DevOps, "Continuous Deployment" AND DevOps, "Continuous Integration" AND DevOps.

SEARCH PERIOD

During the search, the focus was on articles published in the next period of time:

Search period
Between June 2019 and August 2020.

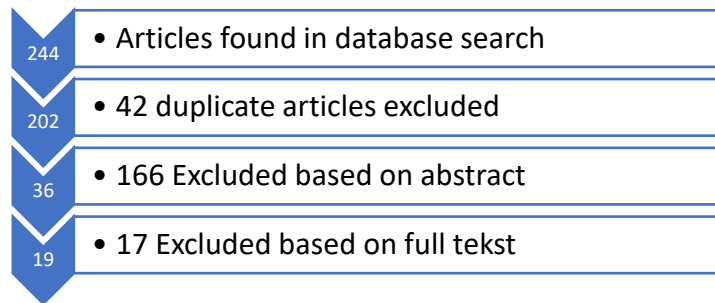
SEARCH RESULT

The search for additional literature resulted in the following number of articles:

Refnr	Database	Number of articles found
1	IEEE computer society digital library	23
2	ACM digital library	57
3	SpringerLink	75
4	Web of Science	17
5	SCOPUS	70
6	Citeseer	2
Total found		244

STUDY SELECTION

After the first search for literature in the database, 244 articles were found. The next step according to Kitchenham (2010) is to perform an initial screening of the found articles. In this case duplicates and relevance were examined. This process was carried out in the following steps with the results in the blue arrows.



After a screening for unique articles, a first screening based on abstraction and then a first screening based on the full text, 19 additional articles remained and are presented in the table on the next page.

QUALITY ASSESSMENT JOURNALS

In addition to the previous screening of the articles, the ranking of the journals in which the articles were published was also examined. A number of different rankings were examined; CORE, Scholar, SJR, ABDC, ERA and Qualis. This ranking can be found on the next page, but it has been decided to not exclude articles from the list anymore.

Title	Authors	Database	Journal ranking	Classification (1st estimate)
A Survey of DevOps Concepts and Challenges	Letta, L; Rocha, C; Kon, F; Milojicic, D; Meirelles, P	Webofscience	CORE: A*; SIR: h=143	Tooling; Coordination; Quality; Knowledge and training
Agile governance theory: operationalization	Alexandre J. H. de O. LunaMarcelo L. M. MarinhoHermano P. de Moura Fiorella ZampettiCarmine VassalloSebastiano PantchellaGerardo CarforaHarald GallMassimiliano Di Penta	Springerlink	Scholar: h5=13	Coordination
An empirical characterization of bad practices in continuous integration	Shahin, M; Zahedi, M; Babar, MA; Zhu, LM	Springerlink	CORE: A	Quality
An empirical study of architecting for continuous delivery and deployment	Hemion A., Lyonnnet B., Rowe F., Fitzgerald B.	Webofscience	CORE: A	Architecture; Coordination; Tooling
Conceptualizing the transition from agile to DevOps: A maturity model for a smarter is function	Shahin, M; Zahedi, M; Babar, MA; Zhu, LM	Scopus	?	Knowledge and training
Designing software architecture to support continuous delivery and DevOps: A systematic literature review	Bolscher R., Daneva M.	Scopus	CORE: B	Architecture
DevDocOps: Enabling continuous documentation in alignment with DevOps	Rong, GP; Jin, ZF; Zhang, H; Zhang, VW; Ye, WH; Shao, D	Webofscience	SIR: h=66	Quality
DevOps in practice: A multiple case study of five companies	Lwakatara, LE; Kilamo, T; Karvonen, T; Sauvola, T; Heikkila, V.; Itkonen, J; Kuvalja, P; Mikkonen, T; Olvo, M; Lassenius, C	Webofscience	CORE: A	Preconditions; Quality; Knowledge and training; Tooling; Tooling
DevOps Meets Dynamic Orchestration	Kiyana BahadortTullio Vardanega	Springerlink	?	Tooling
Documentation of Quality Requirements in Agile Software Development	Behutye, Woudschet; Sepp\ ajnen Alessandro CapranelleIsabella Di Nitto Damian Andrew	ACM	CORE: A; ERA: A; Qualis: B4	Quality
Fallacies and Pitfalls on the Road to DevOps: A Longitudinal Industrial Study	Behutye, Woudschet; Sepp\ ajnen Alessandro CapranelleIsabella Di Nitto Damian Andrew	Springerlink	?	Knowledge and training; Tooling
From Agile to DevOps: Smart Skills and Collaborations	Hemion, A.; Lyonnnet, B.; Rowe, F.; Fitzgerald, B	Webofscience	ABDC: A	Knowledge and training
Integrating Development and Operations in Cross-Functional Teams - Toward a DevOps Competency Model	Wiedemann, Anna; Wiesche, Manuel; Kromar, Helmut	ACM	ERA: B; CORE: B	Knowledge and training
Multicriteria decision-making taxonomy for DevOps challenging factors using analytical hierarchy process	Khan, AA; Shameem, M	Webofscience	CORE: B	Preconditions; Goals; Process design; Motivation; Resistance to change; Quality; Test complexity & source code control; Coordination; Communication; Knowledge and training; Tooling
Orchestrating automation and sharing in DevOps teams: a revelatory case of job satisfaction factors, risk and work conditions	Hemion-Hildgen, A; Rowe, F; Monnier-Senitcourt, L	Webofscience	ABDC: A*	Coordination; Communication; Knowledge and training; Tooling
Prioritization Based Taxonomy of DevOps Security Challenges Using PROMETHEE	Rafiq, S; Yu, W; Akbar, MA; Alsanad, A; Gunraj, A	Webofscience	SIR: h=86	Resistance to change; Coordination; Tooling
Releasing Fast and Slow: An Exploratory Case Study at ING	Kul'a, Elvan; Rastogi, Ayushi; Huijgens, Henne; Deursen, Arie van; Gousios, Georgios	ACM	Qualis: A2	Preconditions; Architecture; Process design; Quality; Test complexity & source code control; Communication; Knowledge and training; Tooling
Software Artifacts Consistency Management towards Continuous Integration: A Roadmap	Mvedeniya, DA; Rubasinsigle, ID; Perera, I	Webofscience	SIR: h=9	Tooling
What we know about software architecture styles in continuous delivery and devops?	Daneva M., Bolscher R.	Scopus	CORE: B; Qualis: B4	Architecture

Appendix B

Total literature overview Critical Success factors CI/CD

	Title	Reference
1	Implementing DevOps in Legacy Systems	Albuquerque & Cruz (2019)
2	Continuous Delivery: Huge Benefits, but Challenges Too	Chen (2015)(in IEEE software)
3	Continuous Delivery: Overcoming adoption challenges	Chen (2017)
4	On the journey to continuous deployment: Technical and social challenges along the way	Claps et al. (2015)
5	Software integration in global software development: Challenges for GSD vendors	Ilyas and Khan (2017)
6	Problems, causes and solutions when adopting continuous delivery—A systematic literature review	Laukkanen et al. (2017)
7	The highways and country roads to continuous deployment	Leppanen et al. (2015)
8	Continuous Integration, Delivery and Deployment: A Systematic Review on Approaches, Tools, Challenges and Practices	Shahin et al. (2017)(in IEEE Access)
9	Challenges When Adopting Continuous Integration: A Case Study	Debbiche et al. (2014)
10	Fit for Continuous Integration: How Organizations Assimilate an Agile Practice	Eck et al. (2014)
11	Towards Architecting for Continuous Delivery	Chen (2015)(in IFIP conf.)
12	An Empirical Investigation of the Software Integration Success Factors in GSD Environment	Ilyas and Khan (2017)(IEEE)
13	Stakeholder Perceptions of the Adoption of Continuous Integration -- A Case Study	Laukkanen et al. (2015)
14	Continuous Integration Impediments in Large-Scale Industry Projects	Martensson et al. (2017)
15	The Intersection of Continuous Deployment and Architecting Process: Practitioners' Perspectives	Shahin et al. (2016)
16	Beyond Continuous Delivery: An Empirical Investigation of Continuous Deployment Challenges	Shahin et al. (2017c)
17	Enhancing Lean Software Development by using DevOps Practices	Farid et al. (2017)
18	Inadequate Testing, Time Pressure, and (Over) Confidence: A Tale of Continuous Integration Users	Pinto et al. (2017)
19	Automated software integration flows in industry: a multiple-case study	Stahl and Bosch (2014)
20	DevOps for information management systems	Qumer Gill et al. (2018)
21	Climbing the Stairway to Heaven; -- A Multiple-Case Study Exploring Barriers in the Transition from Agile Development towards Continuous Deployment of Software	Olsson et al. (2012)
22	On continuous deployment maturity in customer projects	Virtanen et al., 2017
23	Continuous deployment of software intensive products and services: A systematic mapping study	Rodriguez et al. (2017)
24	Customer Involvement in Continuous Deployment: A Systematic Literature Review	Yaman et al. (2016)
25	GuideAutomator: Continuous Delivery of End User Documentation	Souza et al. (2017)
26	The Sources and Approaches to Management of Technical Debt: A Case Study of Two Product Lines in a Middle-Size Finnish Software Company	Yli-Huumo et al. (2014)
27	Trade-offs in continuous integration: assurance, security, and flexibility	Hilton et al. (2017)
28	Bottom-up Adoption of Continuous Delivery in a Stage-Gate Managed Software Organization	Laukkanen et al. (2016)
29	A Hundred Days of Continuous Integration	Miller (2008)
30	Stakeholder Perceptions of the Adoption of Continuous Integration - A Case Study	Laukkanen et al. (2015)
31	Building lean continuous integration and delivery pipelines by applying DevOps principles: a case study at Varidesk	Debroy et al. (2018)
32	Reuse (or Lack Thereof) in Travis CI Specifications: An Empirical Study of CI Phases and Commands	Sidhu et al. (2019)
33	Securing a Deployment Pipeline	Bass et al. (2015)
34	Continuous software engineering: A roadmap and agenda	Fitzgerald & Stol (2017)
35	A Survey of DevOps Concepts and Challenges	Leite et al., 2020
36	Agile governance theory: operationalization	Luna et al., 2020
37	An empirical characterization of bad practices in continuous integration	Zampetti et al., 2020
38	An empirical study of architecting for continuous delivery and deployment	Shahin et al., 2019
39	Conceptualizing the transition from agile to DevOps: A maturity model for a smarter is function	Hemon et al., 2019 (IFIP)
40	Designing software architecture to support continuous delivery and DevOps: A systematic literature review	Bolscher and Daneva, 2019
41	DevDocOps: Enabling continuous documentation in alignment with DevOps	Rong et al., 2020
42	DevOps in practice: A multiple case study of five companies	Lwakatare et al., 2019
43	DevOps Meets Dynamic Orchestration	Bahadori and Vardanega, 2019
44	Documentation of Quality Requirements in Agile Software Development	Behutiye et al., 2020
45	Fallacies and Pitfalls on the Road to DevOps: A Longitudinal Industrial Study	Caprarelli et al., 2020
46	From Agile to DevOps: Smart Skills and Collaborations	Hemon, Lyonnet, Rowe and Fitzgerald, March 2019
47	Integrating Development and Operations in Cross-Functional Teams - Toward a DevOps Competency Model	Wiedemann et al., 2019
48	Orchestrating automation and sharing in DevOps teams: a revelatory case of job satisfaction factors, risk and work conditions	Hemon-Hildgen et al., 2020
49	Prioritization Based Taxonomy of DevOps Security Challenges Using PROMETHEE	Rafi et al., 2020
50	Releasing Fast and Slow: An Exploratory Case Study at ING	Kula et al., 2019
51	Software Artefacts Consistency Management towards Continuous Integration: A Roadmap	Meedeniya et al., 2019
52	What we know about software architecture styles in continuous delivery and devops?	Daneva and Bolscher, 2020
53	Multicriteria decision-making taxonomy for DevOps challenging factors using analytical hierarchy process	Khan & Shameem (2020)

Appendix C

Overview Critical Success factors CI/CD process from a process management perspective

This appendix provides an overview that has been used as a starting point for the investigation of the various measures from a process management perspective mentioned in the literature for the implementation of Continuous Integration and Continuous delivery.

The table below provides a description for each Critical Success Factor, with examples found in the literature with their references.

PRECONDITIONS		
Description	Examples	References
Establishing the optimal provision of value (e.g. generating new capabilities, supporting routines and competencies, restructuring) for realization in use and context where standardization and routinization do not currently exist (Smyth, 2018). In other words, affairs which are not under direct control of the continuous integration and -delivery/deployment process.	<ul style="list-style-type: none"> • Applications that are not amenable to Continuous Deployment (Chen, 2015 (in IEEE software)) • vendor lock-in (Chen, 2015 (in IEEE software)) • dependencies with hardware and other (legacy) applications (Shahin et al., 2017) • legacy code considerations (Albuquerque & Cruz, 2019; Chen, 2017; Leppanen et al., 2015) • merging conflicts (Shahin et al., 2017; Laukkanen et al., 2017; Shahin et al., 2017 (in IEEE Access)) • partner plugins (Claps et al., 2015), third party integration (Laukkanen et al., 2017) • lack of compatibility (Ilyas and Khan, 2017) 	<ul style="list-style-type: none"> • Albuquerque & Cruz (2019) • Chen (2015)(in IEEE software) • Chen (2017) • Claps et al. (2015) • Ilyas and Khan (2017) • Laukkanen et al. (2017) • Leppanen et al. (2015) • Shahin et al. (2017)(in IEEE Access) • Shahin et al. (2017)
GOALS		
Description	Examples	References
Clear goals for the teams migrating towards continuous practices and assimilation metrics.	<ul style="list-style-type: none"> • Lack of setting up clear goals for the teams migrating towards CI is currently an impediment for the teams. (Debbiche et al., 2014) • tension exists between departments due to competing goals (Chen, 2015 (in IEEE software)) • examples of metrics: (outcome metric) "the number of integration errors towards the end of a release cycle" (a high number indicates that CI has not been adopted well) and (input metric) "CI assimilation can be expected to have progressed into a later stage if reflective activities such as continuous improvement processes have been institutionalized" (Eck et al., 2014) 	<ul style="list-style-type: none"> • Chen (2015)(in IEEE software) • Debbiche et al. (2014) • Eck et al. (2014)
STRATEGY AND APPROACH		
Description	Examples	References
Approaches to drive Continuous Integration and Continuous Delivery/-Deployment assimilation, and branching strategies.	<ul style="list-style-type: none"> • Bottom-up approach (e.g. low hanging fruits) or top-down approach (e.g. challenge-oriented) as assimilation paths (Eck et al., 2014) • branching strategies (e.g. feature- or test-motivated approaches) (Eck et al., 2014) • long-running branches and large commits (Laukkanen et al., 2017) 	<ul style="list-style-type: none"> • Eck et al. (2014); • Laukkanen et al. (2017)

ARCHITECT		
Description	Examples	References
Diverse aspects on architecture of the product and related infrastructure.	<ul style="list-style-type: none"> Application architecture (Chen, 2017; Chen, 2015 (in IFIP conf.); Shahin et al., 2016) dependencies in design and code (Shahin et al., 2017; Shahin et al., 2017 (in IEEE Access)) dependency at application level (Shahin et al., 2017c) developers must think about the complete system (Martensson et al., 2017) efficient incremental/continues integration (Ilyas and Khan, 2017) efficient specification for interface compatibility (Ilyas and Khan, 2017) lack of proper component interfaces (Ilyas and Khan, 2017) architecture mismatch (Ilyas and Khan, 2017) consistency in requirements and architecture design (Ilyas and Khan, 2017) modular and loosely coupled architecture (Martensson et al., 2017) product complexity (Debbiche et al., 2014) requirements breakdown (Debbiche et al., 2014) integration of big impact changes (Debbiche et al., 2014) architecture of the product (Laukkanen et al., 2015) difficult components (Laukkanen et al., 2015) system modularization (Laukkanen et al., 2017) unsuitable architecture (Laukkanen et al., 2017) internal dependencies (Laukkanen et al., 2017) inflexible build (Laukkanen et al., 2017) 	<ul style="list-style-type: none"> Chen (2015)(in IFIP conf.) Chen (2017); Debbiche et al. (2014) Ilyas and Khan (2017) Laukkanen et al. (2015) Laukkanen et al. (2017) Martensson et al. (2017) Shahin et al. (2016) Shahin et al. (2017) Shahin et al. (2017)(in IEEE Access)
PROCESS DESIGN		
Description	Examples	References
Institutionalizing the process of continuous integration and -delivery/deployment and aspects of the process e.g., design, effort to initially setting up the proces, management, planning, sufficient time/resources, waste in the process and accuracy of the process.	<ul style="list-style-type: none"> Three approaches to institutionalize CI: shared service, committee and community of practice (Eck et al., 2014) providing Continuous Integration with project start (Eck et al., 2014). lack of integration planning and lack of management (Ilyas and Khan, 2017) lack of time/resources (Laukkanen et al., 2015; Pinto et al., 2017) initially setting up continuous delivery requires effort (Laukkanen et al., 2017) difficulties managing the integration queu (Debbiche et al., 2014) activity sequencing/more synchronization between teams (Martensson et al., 2017) process suitability (meaning the possible integration frequency differs depending on what kind of work being carried out, the problem of using CI for all parts of the product and features, 	<ul style="list-style-type: none"> Chen (2015)(in IEEE software) Debbiche et al. (2014) Eck et al. (2014) Farid et al. (2017) Ilyas and Khan (2017) Laukkanen et al. (2017) Laukkanen et al. (2015) Martensson et al. (2017) Pinto et al. (2017) Stahl and Bosch (2014)

	<p>and the ability to break down software requirements)(Debbiche et al., 2014)</p> <ul style="list-style-type: none"> • construct the automated activities such that their scope affords a sufficient level of confidence in the artifacts processed by them (Stahl and Bosch, 2014) • wast in processes (e.g. a feature ready for release must go through a change advisory board) (Chen, 2015 (in IEEE software); Farid et al., 2017) 	
--	--	--

MOTIVATION

Description	Examples	References
<p>Motivation to adopt Continuous Integration, - Delivery/-Deployment and to get past early difficulties and effort, and discipline to commit often, test diligently, monitor the build status and fix problems as a team.</p>	<ul style="list-style-type: none"> • The use of Continuous Integration techniques require developers a certain level of discipline (Pinto et al., 2017) • scepticism and lack of trust on benefits of Continuous Integration, Continuous Delivery and Continuous Deployment (Debbiche et al., 2014; Shahin et al., 2017; Shahin et al., 2017 (in IEEE Access)) • company-wide effort to adopt CD (Claps et al., 2015) • lack of motivation and discipline (Laukkanen et al., 2017) 	<ul style="list-style-type: none"> • Claps et al. (2015) • Debbiche et al. (2014) • Laukkanen et al. (2017) • Pinto et al. (2017) • Shahin et al. (2017) • Shahin et al. (2017)(in IEEE Access)

RESISTANCE TO CHANGE

Description	Examples	References
<p>Difficulty to change established organizational policies and cultures.</p>	<ul style="list-style-type: none"> • own way of working (Chen, 2015 (in IEEE software)) • exposing work intention (Debbiche et al., 2014) • changing the old habits of team members (Debbiche et al., 2014; Shahin et al., 2017) • Not being facilitated in the change, new way of working not fitting the culture, unexpected social and technical implications (Laukkanen et al., 2017; Qumer Gill et al., 2018; Shahin et al., 2017 (in IEEE Access)) • lack of agile and suitable business model (Shahin et al., 2017 (in IEEE Access)) • changing long-lived feature branching to short-live one (Shahin et al., 2017 (in IEEE Access)) • time-consuming process to change team mindset (Shahin et al., 2017) 	<ul style="list-style-type: none"> • Debbiche et al. (2014) • Chen (2015)(in IEEE software) • Laukkanen et al. (2017) • Qumer Gill et al. (2018) • Shahin et al. (2017)(in IEEE Access) • Shahin et al. (2017)

Appendix D

Literature overview Critical Success factors CI/CD Process control/management/governance

Title	Reference	CSF
Implementing DevOps in Legacy Systems	Albuquerque & Cruz (2019)	Preconditions
Designing software architecture to support continuous delivery and DevOps: A systematic literature review	Bolscher and Daneva, 2019	Architecture
Continuous Delivery: Huge Benefits, but Challenges Too	Chen (2015)(in IEEE software)	Preconditions; Resistance to change; Goals; Process design
Towards Architecting for Continuous Delivery	Chen (2015)(in IFIP conf.)	Architecture
Continuous Delivery: Overcoming adoption challenges	Chen (2017)	Preconditions; Architecture
On the journey to continuous deployment: Technical and social challenges along the way	Claps et al. (2015)	Preconditions; Motivation
What we know about software architecture styles in continuous delivery and devops?	Daneva and Bolscher, 2020	Architecture
Challenges When Adopting Continuous Integration: A Case Study	Debbiche et al. (2014)	Preconditions; Resistance to change; Goals; Process design; Motivation
Fit for Continuous Integration: How Organizations Assimilate an Agile Practice	Eck et al. (2014)	Goals; Process design; Strategy and Approach
Enhancing Lean Software Development by using Devops Practices	Farid et al. (2017)	Process design
Software integration in global software development: Challenges for GSD vendors	Ilyas and Khan (2017)	Preconditions; Architecture; Process design
An Empirical Investigation of the Software Integration Success Factors in GSD Environment	Ilyas and Khan (2017)(IEEE)	Preconditions; Architecture; Process design
Multicriteria decision-making taxonomy for DevOps challenging factors using analytical hierarchy process	Khan & Shameem (2020)	Preconditions; Process design; Motivation; Resistance to change;
Releasing Fast and Slow: An Exploratory Case Study at ING	Kula et al., 2019	Preconditions; Architecture; Process design;
Stakeholder Perceptions of the Adoption of Continuous Integration -- A Case Study	Laukkanen et al. (2015)	Strategy and Approach; Architecture; Process design; Resistance to change; Preconditions
Problems, causes and solutions when adopting continuous delivery—A systematic literature review	Laukkanen et al. (2017)	Strategy and Approach; Architecture; Process design; Resistance to change; Preconditions
The highways and country roads to continuous deployment	Leppanen et al. (2015)	Preconditions
DevOps in practice: A multiple case study of five companies	Lwakatare et al., 2019	Preconditions
Continuous Integration Impediments in Large-Scale Industry Projects	Martensson et al. (2017)	Architecture; Process design
Inadequate Testing, Time Pressure, and (Over) Confidence: A Tale of Continuous Integration Users	Pinto et al. (2017)	Process design; Motivation
DevOps for information management systems	Qumer Gill et al. (2018)	Resistance to change;
Prioritization Based Taxonomy of DevOps Security Challenges Using PROMETHEE	Rafi et al., 2020	Resistance to change;
The Intersection of Continuous Deployment and Architecting Process: Practitioners' Perspectives	Shahin et al. (2016)	Architecture
Continuous Integration, Delivery and Deployment: A Systematic Review on Approaches, Tools, Challenges and Practices	Shahin et al. (2017)(in IEEE Access)	Preconditions; Resistance to change; Motivation; Architecture
Beyond Continuous Delivery: An Empirical Investigation of Continuous Deployment Challenges	Shahin et al. (2017c)	Architecture
An empirical study of architecting for continuous delivery and deployment	Shahin et al., 2019	Architecture
Automated software integration flows in industry: a multiple-case study	Stahl and Bosch (2014)	Process design

Appendix E

Measures definition

This appendix contains the terms that have been used by the authors to indicate 'measures' in their articles. It also shows the context in which it was found and the reference.

Term	Citation from article	Reference(s)
Prerequisite	Agile development methods are considered a prerequisite for CI [14], therefore findings by Turk et al. [18], might apply to the applicability of CI.	Debbiche et al (2014)
Prerequisite	A build system with the capacity to support frequent integration builds is often described as a prerequisite for continuous integration and related practices.	Mårtensson et al (2017)
Mitigation strategies	While a number of individual technical and social adoption challenges were uncovered by the case study in this research, most challenges were not faced in isolation. The severity of these challenges were reduced by a number of mitigation strategies employed by the case study organisation.	Claps et al. (2015)
Architecturally Significant Requirements / Requirements	The most salient implications of CD to architectures of software applications are about Architecturally Significant Requirements (ASRs). ASRs are those requirements that have a measurable impact on a software system's architecture [4]. It is this set of requirements that shape the architecture of the software applications.	Chen (2015) (IFIP)
Adoption actions	Adoption actions need to be performed by an organization to adopt CD.	Laukkanen (2017)
Guidelines	Based on this case study, three guidelines can be given to practitioners.	Laukkanen (2015)
Principles	We have identified a range of recurring architectural challenges (i.e., highly coupled monolithic architecture, team dependencies, and everchanging operational environments and tools.) and five main architectural principles (i.e., small and independent deployment units, not too much focus on reusability, aggregating logs, isolating changes, and testability inside the architecture).	Shahin et al (2016)
Operationalized constructs	Scholars who intend to quantitatively examine the extent of CI assimilation in organizations might find this framework to be a solid foundation for creating a set of operationalized constructs .	Eck et al (2014)
Solutions	We identified a total of 40 problems, 28 causal relationships and 29 solutions related to adoption of continuous delivery.	Laukkanen (2017)
Attributes	Following this there is a discussion on how to use a subset of the attributes and possible constraints (see Section 4.3).	Stahl & Bosch, (2014)
Required	Testing tools and the maturity of the infrastructure supporting the CI process is required in order to facilitate the daily tasks involved.	Debbiche (2014)

Appendix F

Research overview Critical success factors CI/CD

In this appendix the measures are presented per critical success factor that have been found in the literature. The table below looks at each critical success factor in the literature.

PRECONDITIONS			
Measure	Text from article	Measurability	References
Knowledge of the continuous deployment pipeline	Knowledge of the continuous deployment pipeline with its continuous integration, automated testing, and release deployment practices is a prerequisite for developers.		Leppanen (2015)
Mature infrastructure	Testing tools and the maturity of the infrastructure supporting the CI process is required in order to facilitate the daily tasks involved. Continuous integration advocates the use of automated tools to allow more frequent and efficient integrations.		Debbiche (2014)
Bottom up approach	However, the bottom up approach seems to have led some to believe that management could be more involved in the overall process of transitioning to CI. Another reason could be the use of a bottom-up approach where directions and guidance come from experience gained in pilot teams. This might have led to the confusion regarding a vision, since the most expected communication channel for organisational visions ought to be management.		Debbiche (2014)
Single and united organisational culture	The findings indicate that adopting the mindset aligned with CI is a challenge. Interviewees were sceptical about the benefits that they could gain from adopting CI. Similarly, Claps [2] found that teams adopting continuous deployment need to adopt the mindset needed for it. This means that there needs to be a shift towards a single and united organisational culture that adopts the principles of CI.		Debbiche (2014)
Successful cases of process change all share a common denominator, which is divided into 8 steps, liable for their success of change implementation.	Results from this study show that there is an ambiguity regarding the goals and the organisational vision for the implementation of CI. More than half (7 of 13) of the interviewed subjects had a hard time answering questions due to their lack of knowledge of what was expected from them personally. According to Kotter [10], successful cases of process change all share a common denominator, which is divided into 8 steps, liable for their success of change implementation. Kotter [10] highlights the importance of spending the necessary amount of time on each step, and failing to do so will lead to undesired results. For instance, lacking and under-communicating a vision might lead to confusion and incompatible results, which will take the organisation in the wrong direction. This might be a reason to the unclear goals and expectations regarding CI at the case company.		Debbiche (2014) (Kotter [10])
Agile development methods are considered a prerequisite for CI [14]	Additionally, research done by Turk et al. [18] on the suitability of agile development methods shows that assumptions (e.g. face-to-face communication, quality assurance, changing requirements) made by such methods are not appropriate for all organisations, products and projects. The authors [18] highlight important limitations of said methods, where two is of particular interest for this research, namely limited support for large complex software and large teams. Agile development methods are considered a prerequisite for CI [14], therefore		Debbiche (2014)

	findings by Turk et al. [18], might apply to the applicability of CI.		
A “lean” mind-set	The successful use of CD requires a “lean” mind-set. For organisations to become ‘lean’, Ries [37] suggests working in small batches		Claps et al. (2015)
more hardware resources for CI servers	Provide more hardware resources to a product’s CI servers to allow the software product to be continuously integrated as often as necessary, thus allowing the software product to be deployed at any time		Clap et al. (2015)
well prepared to handle technical and social adoption challenges with their existing expertise, processes and tools	From a managerial point of view, the findings indicate that organisations need to be well prepared to handle technical and social adoption challenges with their existing expertise, processes and tools before adopting the CD process.		Clap et al. (2015)
Adjust the architecture by changing technologies or components if needed.	First, understand that the product architecture has a significant effect on the adoption. However, do not let architectural problems keep you from implementing continuous integration. You may adjust the architecture by changing technologies or components if needed.		Laukkanen et al. (2015)

GOALS			
Measure	Text from article	Measurability	References
Set clear goals for teams	Lack of setting up clear goals for the teams migrating towards CI is currently an impediment for the teams.		Debbiche (2014)

STRATEGY AND APPROACH			
Measure	Text from article	Measurability	References
Well organized incident process	Employ a strategy of stopping the entire team (i.e. as many people as necessary) from doing what they are doing to fix issues to make sure the software is in an ever-ready state of being deployable		Clap et al. (2015)
Adopting the practice of small batches	Develop large features (dark features) in small batches that only appear visible to the customer when the entire feature is finally developed		Claps et al. (2015)
Availability of several branches per software product	Develop each feature for a software product on a separate branch and, when completed, merge that feature into the main branch of code. Having one single branch of code to maintain one working version of the software product in a CD environment		Claps et al. (2015)
Use parallel running systems for deployment	Use two parallel running systems (the old and the new system) to upgrade the user to the next version of software. Seamless upgrades – complications in implementation of seamless upgrades due to resource limitations, zero downtime deployment and customer data preservation		Claps et al. (2015)
Use document software products	Maintaining multiple documentation for each product which may have multiple offerings (e.g. a customer-hosted version and a version less Atlassian-hosted online version), where features may be released in one offering, but not in the other. Use Atlassian’s own product, Confluence (which is a wiki), to document software products		Claps et al. (2015)
Adopt ‘social rules’ which must be adhered to when deploying software.	A lack of understanding of the CD process by novice developers due to inconsistent documentation and a lack of industry standards		Clap et al. (2015)
Good overview on organization structure	The biggest challenge has been organizational. Release activities involve many divisions of the company. Each has its own interests, ways of working, and perceived territories of control. Tension existed between divisions due to competing goals. For example, we needed root access to the servers, and another team controlled this permission. Arriving at a solution involved		Chen (2015 IEEE)

	<p>much consultation and negotiation over six months. To address the organizational challenges, the leadership team restructured the organization to break down barriers among teams and promote a collaborative culture. The situation has improved since. Although literature on organizational change exists,4 little, if any, research focuses on introducing CD to an organization. Further research on this topic—for example, understanding the challenges in more depth and developing strategies and practices to tackle them more effectively—will significantly help an organization’s smooth adoption of CD.</p>		
invest in the communication	<p>Third, invest in the communication between different sites and avoid centralizing different competencies to certain sites. Understand the value of face-to-face communication and local support personnel when adopting CI.</p>		Laukkanen et al. (2015)
Management support	<p>Our analysis shows that compared to CDE, successfully adopting CD needs better management support. To achieve CD, organizations must break down barriers at production. This is mainly achievable by allowing developers to be part of deployment decision-making and placing more trust on them. By this, the manual approval process described in Section IV.C.5 will be significantly reduced.</p>		Shahin et al. (2017) (IEEE)
Devising an Assimilation Path	<p>Sources suggest various approaches to drive CI assimilation, which we characterize as low-hanging fruits, extend nucleus, and challenge-oriented. All approaches emphasize putting CI in a larger context of overall agile development assimilation. Stober and Hansmann (2010) propose a low-hanging fruits approach: those aspects should be introduced first that are easily implemented and require few behavioural changes. As the organizational members get accustomed to the change it can then explore more sophisticated practices, one of which is CI. Olsson et al. (2012) suggest an extend nucleus approach: starting with a core CI system that automates integration and testing, the ambition and reach of CI is extended continuously, until CI is used to leverage operational software for experimentation. Finally, Conboy et al. (2007) suggest a challenge-oriented approach: after prioritization of current challenges, the organization should implement those agile practices first that are likely to have the highest return. In conclusion, there is consensus that CI is not a means in itself and should be seen as component of agile development. Differences arise on who should have the initiative – ranging from a bottom-up attitude like low-hanging fruits up to a top-down stance as challenge-oriented.</p>	<p>Several authors discuss how to measure assimilation of an agile development practice such as CI. Stober and Hansmann (2010) propose an outcome metric, namely the number of integration errors towards the end of a release cycle. A high number indicates that CI has not been adopted well. As CI helps development teams to start integration activities early while continuing development until late in a project, there should be no observable spike in integration effort. An input metric is proposed by two sources (Conboy et al. 2007; Wang et al. 2012): CI assimilation – and to this end any agile development practice – can be expected to have progressed into a later stage if reflective activities such as continuous improvement processes have</p>	Eck et al. (2014)

		<p>been institutionalized. It is worth noticing that authors dismiss time since Fit for Continuous Integration Twentieth Americas Conference on Information Systems, Savannah, 2014 5 introductions as being suitable, because this metric has no qualitative component to it (Conboy et al. 2007; Wang et al. 2012). To summarize, we were able to identify one outcome metric and one input metric that complement each other.</p>	
Overcoming Initial Learning Phase	In conclusion, it is critical that organizations prepare for a dip in productivity, accept this initial negative impact as investment, and minimize the learning phase through targeted training.		Eck et al. (2014)
Clarifying Division of Labor	CI connects previously separated organizational functions through process automation, e.g. development and release management. A number of authors suggest that it is vital to clarify division of labor, such as establishing hand-over procedures along the CI chain (Maruping 2010; Thiyagarajan and Verma 2009) or introducing suitable policies, e.g. who is allowed to perform branch merging (Williams et al. 2011). However, organizations should resist re-establishing exactly those silos that CI is supposed to help overcome. A possible approach is to additionally assign source code ownership: across the entire CI chain, owners are made responsible to resolve integration errors affecting their code (Moore and Spens 2008; Stober and Hansmann 2010).		Eck et al. (2014)
CI and Distributed Development	CI has the potential to increase collaboration among spatially distributed development through replication, synchronization, and transparency. Modern CI systems support local replication of the codebase, which makes the CI chain resilient against connectivity disruptions (Sauer 2010; Poole 2004). The different development teams synchronize their work with every integration (Hillegersberg et al. 2011; Avritzer et al. 2010). There is a trade-off between synchronization frequency and communication needs, however. If changes are instantly synchronized against the full codebase, then communication between locations is likely to increase in case of encountered integration errors (Avritzer et al. 2010). Lastly, CI creates transparency about the project status, because every team member has access to the codebase (Bose 2008; Avritzer et al. 2010). In summary, the sources argue that the various elements of CI collectively foster collaboration in distributed development settings.		Eck et al. (2014)
Devising a Branching Strategy	A major feature of any CI system is branching, i.e. instantiating a copy of the common codebase, locally working with this copy, and feeding the		

	<p>changes back into the codebase after local testing. In its most basic use branching enables concurrent development. Practitioners leverage this feature to implement sophisticated strategies, which we characterize as either feature-motivated or test-motivated. A commonly observed feature-motivated strategy is to branch out every new feature, in order to not jeopardize integrity of the current codebase (Wilcox et al. 2007). Building on this idea, Stober and Hansmann (2010) propose cross-functional development teams when exploring risky technology changes not yet on the roadmap. These teams exist as long as the branch does, i.e. they are being dismantled once their goal is accomplished. To maintain consistency of the overall codebase, long-lived source code ownerships are assigned in parallel. Of the various test-motivated approaches, Dowling (2013) presents a variant in which a testing branch is created every two weeks. A dedicated team performs tests on this branch, while development continues uninterrupted. All encountered errors that could be fixed within this period are fed back into the codebase; a new test branch is then created and the cycle begins anew. Similarly, Ali et al. (2012) propose performing complex and time-consuming tests on a dedicated branch. Interestingly, none of these authors mention a hybrid strategy, combining feature-motivated and testing motivated approaches.</p>		
Work breakdown	<p>A way of working that supports work breakdown into small pieces that can be delivered to the software mainline. Directives on whether a commit to the mainline shall only include complete functions or not, and if a commit shall include tests and/or documentation (or if this could be delivered later – or earlier).</p>		Mårtensson et al (2017)
Resilience	<p>Design for failure is considered as the foundation of the architecture in a CD context, in which, instead of preventing failures (reliability), it is more important to learn how to deal with failures (resilience).</p>		Shahin et al (2019)

ARCHITECT			
Measure	Text from article	Measurability	References
Implement architecture principles	<p>Architecture principles should be put in place to guide the logging of the applications. The architecture principles can include rules on what to log, the log format, the logging mechanism, etc.</p>		Chen (2015 IFIP)
Modifiable architecture	<p>An application should be architected in a way that supports this style of software development. In general, the software application should exhibit a high degree of modifiability to allow constant incremental adding of small new features.</p>		Chen (2015 IFIP)
Monitorable software application	<p>Apart from putting in proper monitoring tools, the software application itself should be architected in a way that is amenable for monitoring. With CD, we sometimes need applications to expose additional monitoring interfaces to facilitate CD.</p>		Chen (2015 IFIP)
small and independently deployable units (e.g., service, component, and database) should be considered as a foundation for architecture design.	<p>According to Shahin there are three options for creating small and independently deployable units. Microservices: This architectural style tries to break complex systems into small, autonomous, independently deployable services. We found that some interviewees' organizations have successfully implemented this architectural style for enabling CD practice. Vertical layering: splitting out modules or components of the applications into vertical layers rather than employing horizontal</p>		Shahin et al. (2016)

	layered architecture and then letting each team to be responsible for their own module or component. Database as continuously deployable unit: it is vital to consider database as easily deployable unit like other software components. It is mainly because they are part of deployment process of software components. By incorporating the database into deployment process as the application, it is expected to potentially prevent unexpected issues that database updates may introduce at the point of deployment.		
Micro-services Architectures	There is a dominant architectural style present in these papers: micro-services. Micro-services are a set of small services that can be developed, tested, deployed, scaled, operated and upgraded independently, allowing organizations to gain agility, reduce complexity and scale their applications in the cloud in a more efficient way. Besides that, micro-services are very popular, they are being used and promoted by industry leaders such as Amazon, Netflix and LinkedIn (Villamizar, 2015). Shahin et al. describe micro-services as the first architectural style to be preferred for CD practice, by designing fine-grained applications as a set of small services (Shahin et al., 2016). Three papers (Stahl and Bosch, 2018; Pahl et al., 2018; Berger et al., 2017) state explicitly some specific benefits of employing the micro-services architecture concept. Micro-services are said to be helpful in increasing modularity and isolating changes and as a consequence increasing deployment frequency (Bass, 2017).		Bolscher et al (2019)
Focusing too much on reusability can be a huge bottleneck to continuously deploying software	Reusing at the component level as well as reusing packaged software within application may involve two side effects for CD: (i) it increases dependencies between teams, at which all teams depend on single and shared unit. In addition, there is a need to spend huge amount of time to maintain and obtain agreement from all teams before evolving a unit. So (re)-using packaged software may reduce ability to write the unit test and ability to put them on continuous deployment pipeline. Therefore, it is recommended that in the CD process everything should be decentralized.		Shahin et al. (2016)
Proper logging	It is absolutely critical to collect the operational data in order to both monitor the current state of application in the production and to fix issues and problems [3]. Therefore, log aggregation, analysis and monitoring are becoming increasingly important for deploying software on the continuous basis		Shahin et al. (2016)
Isolate changes and minimize the impact of changes	Some interviewees noted that they always considered that software units (e.g., component or service) had to be small enough to be modified, replaced and run independently in productions environment		Shahin et al. (2016)
Testability inside the architecture	Apart from choosing appropriate testing tools, the interviewees adopted following strategies in order to improve the testability of their system: Improving test quality: improve the quality of test (data). This includes having right number of test cases, improved quality of test data, and right time for testing. Making code more testable: the code should be written to be much more testable. Test automation: Interviewees argued that the architecture should support automating testing		Shahin et al. (2016)
Use a common service bus architecture.	In conclusion, it is very likely that CI infrastructure itself becomes heterogeneous and complex. Mordinyi et al. (2011) propose a common service		

	bus architecture around which the various elements are brought together.		
Micro-services architectural style	There is a dominant architectural style present in these papers: micro-services. All 8 papers either discuss micro-services or present them as the “go-to” architectural style for CD/DevOps practices. We observe that only 2 out of the 8 papers also mention other architectural styles or patterns. Micro-services are a set of small services that can be developed, tested, deployed, scaled, operated and upgraded independently, allowing organizations to gain agility, reduce complexity and scale their applications in the cloud in a more efficient way. Besides that, micro-services are very popular, they are being used and promoted by industry leaders such as Amazon, Netflix and LinkedIn (Villamizar, 2015). Shahin et al. describe micro-services as the first architectural style to be preferred for CD practice, by designing fine-grained applications as a set of small services (Shahin et al., 2016).		Bolscher et al (2019)
Modular and loosely coupled architecture	A modular architecture with small components, which makes it possible for many teams to work in parallel. Loosely coupled architecture with as few dependencies as possible between the components, which means that changes can be committed independently.		Mårtensson et al (2017)
small and independent deployment units	Compared with the work by Bass et al. (2015), Schermann et al. (2016), and Newman (2015), our study has independently identified the reasons why practicing CD within monoliths is difficult (Finding 1). We found that growing monoliths leads to increased complexity of internal and external dependencies, restricted automation (test and deployment), impeding teams’ ownership and having slow and inconsistent feedback (Laukkanen et al. 2017; Schermann et al. 2016; Arun 2015), which together can be roadblocks to frequent and automatic deployment. Similar to Newman (2015), and Lewis and Fowler (2010), we can conclude that the principle of “small and independent deployment units” is an alternative to monolithic systems for this purpose and serves as a foundation to design CD-driven architectures (Finding 4).		Shahin et al (2019)
delaying (architectural) design decisions	Compared with the less frequent releases, CD places greater emphasis on evolutionary changes. This requires delaying (architectural) design decisions to the last possible moment.		Shahin et al (2019)

PROCESS DESIGN			
Measure	Text from article	Measurability	References
Choose a good system design solution.	Four system design solutions were reported: system modularization, hidden changes, rollback and redundancy (Table 17). The design solutions considered what kind of properties the system should have to enable adopting CD.		Laukkanen et al (2017)
Institutionalizing CI	Three approaches have been discussed for how to institutionalize CI: shared service, committee, and communities of practice. Offering CI as shared service means establishing a dedicated team that maintains and improves CI, and which centrally provides training and other support. On the downside, all teams involved with CI depend on the central roadmap and cannot deviate easily (Heidrich et al. 2013). Alternatively, or as complement, a committee might be established. It consists of subject matter experts across the organization, who discuss the CI roadmap and		Eck et al. (2014)

	<p>issue authoritative guidelines for CI use (Williams et al. 2011). A further alternative are communities of practice: usually self-emergent, they are sustained by company-internal thought leaders who drive CI improvements forward and provide guidance on request. Through granting these thought leaders time for engaging in such activities, organizations choose to promote this model (Woodward et al. 2010). To summarize, different models exist to foster CI use and sophistication, ranging from centralized to decentralized.</p>		
Providing CI with Project Start	<p>It might be that CI is not available directly with project start: extensive manual preparation might be needed to set up the tool chain, or it needs to be amended with new elements that are required for the current project (Conboy et al. 2007; Poole 2004; Verweij, P. J. F. M. et al. 2010). Several authors argue that such delays may jeopardize CI altogether. If CI becomes available in the course of a project, developers are likely to abandon it (Kasoju et al. 2013; Avritzer et al. 2010). In addition, major architectural changes might become necessary to prepare the codebase for effective CI (Kane 2003).</p>		Eck et al. (2014)
Extending CI Beyond Source Code	<p>A number of sources propose capturing all artifacts of software development with CI, not just the source code (Bos and Vriens 2004; Kane 2003). Otherwise, changes in uncontrolled artifacts, e.g. configuration files, firmware, etc. might cause integration and testing errors that are hard to reproduce (Larsson et al. Eck et al. Contemporary Issues in Agile Development 8 Twentieth Americas Conference on Information Systems, Savannah, 2014 2009). However, organizations should carefully evaluate the benefits against its costs; it might make sense to purposefully leave gaps in the CI system (Talby et al. 2005).</p>		Eck et al. (2014)
Improved collaboration among teams and team members	<p>In many organizations, Development (Dev) and Operations (Ops) teams form silos that are possibly located in separate departments [31]. Whilst this structure was motivated by traditional methodologies (e.g., waterfall), it is not suitable for recent software development practices that simultaneously deal with agility, and maintaining software on different environments [22]. Iden et al. [23] highlight that effective cooperation between Dev and Ops teams has great impact on the quality of the final product. Any shortcomings in interaction of these members usually manifest in problems such as excluding IT operations from requirement specifications, poor communication and information flow, and lack of knowledge transfer [23]. Lwakatere et al. [29] indicate that collaboration can be enforced through practices such as broadening skill-set, information sharing and shifting responsibilities among these members. Nevertheless, implementing these practices demand changes in team structures, required responsibilities, the work culture of organization and mindset of team members [29; 31]. Some researchers have identified best practices to implement these changes (e.g., team structure). For example, Humble and Molskey [22] suggest re-architecting software product in form of strategic services, and assigning each service to a small cross-functional team who takes the full ownership of it during whole development lifecycle. Our study has revealed several organizational practices improving</p>		Shahin et al (2017)

	collaboration among teams and team members to effectively implement CD.		
Teams and responsibilities:	A project organization that supports both working with functional changes and at the same time takes responsibility for the architecture. The organization can be built on cross-functional teams or component teams (or a mix of both), but with explicit ownership for both a functional change and the design of the whole system.		Mårtensson et al (2017)
Activity sequencing:	Synchronization between the development teams in order to optimize the flow of activities when functions and systems are implemented. Scheduling of prototyping activities and prestudies related to architecture and system design.		Mårtensson et al (2017)
Developers must think about the complete system:	Developers understand the functions and design of the whole system, and not just their own sub-system. The developers have knowledge about how the functions utilize the different sub-systems and how different subsystems are connected to each other.		Mårtensson et al (2017)
Flexible organizational structure	Our experience in this study was that the process of changing to a CD organization requires more than providing tool support and automation. CD impacts on organizational structures (e.g., team structures), roles and responsibilities. From practitioners' perspectives, the inflexibility of organizational structures with the spirit of CD practices is the most critical challenge for implementing CD (Finding 3). A software organization can succeed in CD adoption once there is flexibility in optimizing organizational structures to be aligned with CD and certain skills and responsibilities need to be sought. Finding the best organizational structures that suit an organization depends on many factors, such as the flexibility of the current organizational structure, available skills and management procedures (Brown 2015; What Team Structure is Right for DevOps to Flourish 2017).		Shahin et al (2019)

MOTIVATION

Measure	Text from article	Measurability	References
Company-wide effort	Ensure that top-management implement a strategy to push the need to implement the CD process		Claps et al. (2015)

RESISTANCE TO CHANGE

Measure	Text from article	Measurability	References
Use a CI driver to implement CI	Similarly, a CI Driver involved in coaching and helping teams with transitioning to CI highlights the importance of questioning a change		Debbiche (2014)
Work with more experienced teams	Continuous integration emphasises early and frequent integrations. As a result, developers are compelled to expose their work earlier. Some interviewees (4 out of 13) found this to be challenge because they were used to big bang integrations at the end of a sprint. This gave them enough time to polish their code before integrating it. With the adoption of CI and the increase in integration frequencies, developers are worried about integrating low quality code that could be questioned by experts and managers, as an agile coach put it, "some teams they are not familiar or used to frequent delivery, because they feel safe if they can deliver once a month because they can make everything ready, if they have some changes, he can correct it on his own branch, don't have to deliver to the main		Debbiche (2014)

	branch and then everybody can see your faults right.” Developer confidence plays a role in this issue. Teams that are more experienced working with CI seem to be more comfortable about exposing their work earlier.		
increase pressure by increasing frequency of integrations in steps	The initiative to adopt CI has resulted in increased pressure on the teams according to some interviewees (4 out of 13). Despite the positive support and attitude towards the concept of CI, teams feel that management would like it to happen faster than currently possible which leads to increased pressure. Some developers feel that they lack the confidence and experience to reach desired integration frequencies. There seems to be a general consensus among developers that transitioning to CI carries risks, a period of chaos and increased pressure. Hence, the frequency of integrations and how to proceed should be done in steps in order to minimize the risk of increased pressure.		Debbiche (2014)
The mindset is an important factor in the success of implementing CI.	The mindset is an important factor in the success of implementing CI. Scepticism towards the introduction of a new process needs be considered in order to win over non-believers.		Debbiche (2014)
Team coordination – requires extra effort to coordinate multiple teams	Engage in an increased amount of planning in terms of planning around deployments with multiple systems that are interdependent		Claps et al. (2015)
Integrate the automated deployment of software using CD into the existing CI workflow of developers to ensure there is no, or a low learning curve	Team experience – having an experienced team is critical in the successful adoption of CD		Claps et al. (2015)
Improve communication among developers and managers.	Software developers may feel an increased amount of pressure to have code ready to be deployed immediately		Claps et al. (2015)
Promote a collaborative culture.	The biggest challenge has been organizational. Release activities involve many divisions of the company. Each has its own interests, ways of working, and perceived territories of control. Tension existed between divisions due to competing goals. For example, we needed root access to the servers, and another team controlled this permission. Arriving at a solution involved much consultation and negotiation over six months. To address the organizational challenges, the leadership team restructured the organization to break down barriers among teams and promote a collaborative culture. The situation has improved since. Although literature on organizational change exists,4 little, if any, research focuses on introducing CD to an organization. Further research on this topic—for example, understanding the challenges in more depth and developing strategies and practices to tackle them more effectively—will significantly help an organization’s smooth adoption of CD.		Chen (2015 IEEE)
Give the team time to adopt CI	Give enough time for your team members to overcome the initial learning phase. You have to lower the priority of new features during the adoption.		Laukkanen et al (2015)
Clarity, Visibility and awareness of a project status to the team	It is evident that sharing responsibilities of software delivery with all members [22] could promote coordination and collaboration in a team. We observed the practice of rotating roles [31] between developers and operations staff, i.e., involving developers in testing and QAs in development tasks. Our findings have highlighted the significant role of visibility and awareness of a project status for improving collaboration in a team and successfully adopting CD. Humble and Farley [21] recommend using big, ubiquitous		Shahin et al (2017)

	dashboards in each team room to visualize status of builds and sharing feedback with everybody. Our participants also indicated raising awareness in teams by involving operations staff in daily meetings [31] and interacting with Kanban board		
Sharing knowledge among team members	Broadening responsibilities of developers to a larger extent may negatively impact their productivity in core tasks. Shifting extensive amount of operations' responsibilities to developers could cause fear of losing jobs for Ops team and may negatively affect success of transition to CD. Organizations should extensively promote knowledge sharing among team members to complement areas of skill-set and collaboratively work towards a shared goal.		Shahin et al (2017)

Appendix G

Measurements against the inclusion and exclusion criteria

CSF	Definition measure (quote)	Reference	Inclusion Criteria	Measurability
Preconditions	Knowledge of the continuous deployment pipeline	Leppanen (2015)	needed/required	qualitatively
Preconditions	Mature infrastructure	Debbiche (2014)	needed/required	qualitatively
Preconditions	Bottom up approach	Debbiche (2014)	enable; support	qualitatively
Preconditions	Single and united organisational culture	Debbiche (2014)	needed/required; improve performance	qualitatively
Preconditions	Successful cases of process change all share a common denominator, which is divided into 8 steps, liable for their success of change implementation.	Debbiche (2014) (Kotter [10])	mitigate challenges	qualitatively
Preconditions	Agile development methods are considered a prerequisite for CI [14]	Debbiche (2014)	needed/required	qualitatively
Preconditions	A "lean" mind-set	Claps et al. (2015)	Action; enable	qualitatively
Preconditions	Hardware resources for CI servers	Claps et al. (2015)	improve performance	qualitatively
Preconditions	well prepared to handle technical and social adoption challenges with their existing expertise, processes and tools	Claps et al. (2015)	improve performance	qualitatively
Preconditions	Adjust the architecture by changing technologies or components if needed.	Laukkanen et al. (2015)	action; minimize risks	qualitatively
Goals	Set clear goals for teams	Debbiche (2014)	address concerns; improve performance	qualitatively
Strategy and Approach	Well organized incident process	Claps et al. (2015)	minimize risks	qualitatively
Strategy and Approach	Adopting the practice of small batches	Claps et al. (2015)	improve performance	qualitatively
Strategy and Approach	Availability of several branches per software product	Claps et al. (2015)	improve performance	qualitatively
Strategy and Approach	Use parallel running systems for deployment	Claps et al. (2015)	improve performance	qualitatively
Strategy and Approach	Use document software products	Claps et al. (2015)	minimize risks	qualitatively
Strategy and Approach	Adopt "social rules" which must be adhered to when deploying software.	Claps et al. (2015)	improve performance	qualitatively
Strategy and Approach	Good overview on organization structure	Chen (2015 IEEE)	mitigate challenges	qualitatively
Strategy and Approach	Invest in the communication	Laukkanen et al. (2015)	action; mitigate challenges	qualitatively
Strategy and Approach	Management support	Shahin et al. (2017) (IEEE)	support	qualitatively
Strategy and Approach	Devising an Assimilation Path	Eck et al. (2014)	minimize risks	qualitatively
Strategy and Approach	Overcoming Initial Learning Phase	Eck et al. (2014)	dealing with	qualitatively
Strategy and Approach	Clarifying Division of Labor	Eck et al. (2014)	improve performance; mitigate challenges	qualitatively
Strategy and Approach	CI and Distributed Development	Eck et al. (2014)	improve performance; mitigate challenges	qualitatively
Strategy and Approach	Devising a Branching Strategy	Eck et al. (2014)	enable; minimize risks	qualitatively
Strategy and Approach	Work breakdown	Mårtensson et al (2017)	support; minimize risks; improve performance	qualitatively
Strategy and Approach	Resilience	Shahin et al (2019)	improve performance	qualitatively
Architect	Implement architecture principles	Chen (2015 IFIP)	support	qualitatively
Architect	Modifiable architecture	Chen (2015 IFIP)	needed/required	qualitatively
Architect	Monitorable software application	Chen (2015 IFIP)	support	qualitatively
Architect	small and independently deployable units (e.g., service, component, and database) should be considered as a foundation for architecture design.	Shahin et al. (2016)	support; minimize risks; improve performance	qualitatively
Architect	Micro-services Architectures	Bolscher et al (2019)	support; minimize risks; improve performance	qualitatively
Architect	Focusing too much on reusability can be a huge bottleneck to	Shahin et al. (2016)	mitigate challenges	qualitatively
Architect	continuously deploying software	Shahin et al. (2016)	minimize risks; improve performance	qualitatively
Architect	Proper logging	Shahin et al. (2016)	minimize risks; improve performance	qualitatively
Architect	isolate changes and minimize the impact of changes	Shahin et al. (2016)	minimize risks; improve performance	qualitatively
Architect	Testability inside the architecture	Shahin et al. (2016)	support	qualitatively
Architect	Use a common service bus architecture.	Eck et al. (2014)	support; minimize risks	qualitatively
Architect	Micro-services architectural style	Bolscher et al (2019)	support; minimize risks; improve performance	qualitatively
Architect	Modular and loosely coupled architecture	Mårtensson et al (2017)	support; minimize risks	qualitatively
Architect	small and independent deployment units	Shahin et al (2019)	support; minimize risks; improve performance	qualitatively
Architect	delaying (architectural) design decisions	Shahin et al (2019)	support; improve performance	qualitatively
Process Design	Choose a good system design solution.	laukkanen et al (2017)	support; minimize risks; improve performance	qualitatively
Process Design	Institutionalizing CI	Eck et al. (2014)	mitigate challenges; minimize risks	qualitatively
Process Design	Providing CI with Project Start	Eck et al. (2014)	mitigate challenges; minimize risks; improve performance	qualitatively
Process Design	Extending CI Beyond Source Code	Eck et al. (2014)	improve performance; mitigate challenges	qualitatively
Process Design	Improved collaboration among teams and team members	Shahin et al (2017)	enable; address concerns	qualitatively
Process Design	Teams and responsibilities:	Mårtensson et al (2017)	mitigate challenges; minimize risks; improve performance	qualitatively
Process Design	Activity sequencing:	Mårtensson et al (2017)	minimize risks; improve performance	qualitatively
Process Design	Developers must think about the complete system:	Mårtensson et al (2017)	needed/required	qualitatively
Process Design	Flexible organizational structure	Shahin et al (2019)	support	qualitatively
Motivation	Company-wide effort	Claps et al. (2015)	mitigate challenges	qualitatively
Resistance to Change	Use a CI driver to implement CI	Debbiche (2014)	improve performance; mitigate challenges; support; address concerns	qualitatively
Resistance to Change	Work with more experienced teams	Debbiche (2014)	improve performance; minimize risks; enable	qualitatively
Resistance to Change	increase pressure by increasing frequency of integrations in steps	Debbiche (2014)	address concerns	qualitatively
Resistance to Change	Team coordination – requires extra effort to coordinate multiple teams	Claps et al. (2015)	dealing with; improve performance; address concerns	qualitatively
Resistance to Change	Improve communication among developers and managers.	Claps et al. (2015)	address concerns	qualitatively
Resistance to Change	Promote a collaborative culture.	Chen (2015 IEEE)	address concerns	qualitatively
Resistance to Change	Give the team time to adopt CI	Laukkanen et al. (2015)	address concerns	qualitatively
Resistance to Change	Clarity, Visibility and awareness of a project status to the team	Shahin et al (2017)	address concerns	qualitatively
Resistance to Change	Sharing knowledge among team members	Shahin et al (2017)	improve performance	qualitatively

Appendix H

Metaplan Session Notes

Process control_Resistance to change

Report date: 12-10-2020 15:46:13

Number of unsorted cards: 1

Number of sorted cards: 11

Total number of cards: 12

Unsorted cards:

a new name (a new description)

Topics:

Mindset and process ()

Owned cards:

Use a CI driver to implement CI (Similarly, a CI Driver involved in coaching and helping teams with transitioning to CI highlights the importance of questioning a change Debbiche (2014)

Improve communication among developers and managers. (Software developers may feel an increased amount of pressure to have code ready to be deployed immediately Claps et al. (2015)

Process ()

Owned cards:

- Work with more experienced teams (Continuous integration emphasises early and frequent integrations. As a result, developers are compelled to expose their work earlier. Some interviewees (4 out of 13) found this to be challenge because they were used to big bang integrations at the end of a sprint. This gave them enough time to polish their code before integrating it. With the adoption of CI and the increase in integration frequencies, developers are worried about integrating low quality code that could be questioned by experts and managers, as an agile coach put it, "some teams they are not familiar or used to frequent delivery, because they feel safe if they can deliver once a month because they can make everything ready, if they have some changes, he can correct it on his own branch, don't have to deliver to the main branch and then everybody can see your faults right." Developer confidence plays a role in this issue. Teams that are more experienced working with CI seem to be more comfortable about exposing their work earlier. Debbiche (2014)
- increase pressure by increasing frequency of integrations in steps (The initiative to adopt CI has resulted in increased pressure on the teams according to some interviewees (4 out of 13). Despite the positive support and attitude towards the concept of CI, teams feel that management would like it to happen faster than currently possible which leads to increased pressure. Some developers feel that they lack the confidence and experience to reach desired integration frequencies. There seems to be a general consensus among developers that transitioning to CI carries risks, a period of chaos and increased pressure. Hence, the frequency of integrations and how to proceed should be done in steps in order to minimize the risk of increased pressure. Debbiche (2014)
- Sharing knowledge among team members (Broadening responsibilities of developers to a larger extent may negatively impact their productivity in core tasks. Shifting extensive amount of operations' responsibilities to developers could cause fear of losing jobs for Ops team and may negatively affect success of transition to CD. Organizations should extensively promote knowledge sharing among team members to complement areas of skill-set and collaboratively work towards a shared goal. Shahin et al (2017)

Mindset ()

Owned cards:

Vervallen-geen maatregel ()

Owned cards:

- The mindset is an important factor in the success of implementing CI. (The mindset is an important factor in the success of implementing CI. Scepticism towards the introduction of a new process needs be considered in order to win over non-believers. Debbiche (2014)
- Integrate the automated deployment of software using CD into the existing CI workflow of developers to ensure there is no, or a low learning curve (Team experience – having an experienced team is critical in the successful adoption of CD Claps et al. (2015)

To process design

To topic planning: Team coordination - requires... + Give the team time...

- Team coordination -requires extra effort to coordinate multiple teams (Engage in an increased amount of planning in terms of planning around deployments with multiple systems that are interdependent Claps et al. (2015)
- Give the team time to adopt CI (Give enough time for your team members to overcome the initial learning phase. You have to lower the priority of new features during the adoption. Laukkanen et al. (2015)

To topic or organization...: Promote a collaborative ...)

Owned cards:

- Promote a collaborative culture. (The biggest challenge has been organizational. Release activities involve many divisions of the company. Each has its own interests, ways of working, and perceived territories of control. Tension existed between divisions due to competing goals. For example, we needed root access to the servers, and another team controlled this permission. Arriving at a solution involved much consultation and negotiation over six months. To address the organizational challenges, the leadership team restructured the organization to break down barriers among teams and promote a collaborative culture. The situation has improved since. Although literature on organizational change exists, little, if any, research focuses on introducing CD to an organization. Further research on this topic—for example, understanding the challenges in more depth and developing strategies and practices to tackle them more effectively—will significantly help an organization’s smooth adoption of CD. Chen (2015 IEEE)
- Clarity, Visibility and awareness of a project status to the team (It is evident that sharing responsibilities of software delivery with all members [22] could promote coordination and collaboration in a team. We observed the practice of rotating roles [31] between developers and operations staff, i.e., involving developers in testing and QAs in development tasks. Our findings have highlighted the significant role of visibility and awareness of a project status for improving collaboration in a team and successfully adopting CD. Humble and Farley [21] recommend using big, ubiquitous dashboards in each team room to visualize status of builds and sharing feedback with everybody. Our participants also indicated raising awareness in teams by involving operations staff in daily meetings [31] and interacting with Kanban board Shahin et al (2017)

Process control_Architecture

Number of unsorted cards: 1

Number of sorted cards: 14

Total number of cards: 15

Unsorted cards:

a new name (a new description)

Topics:

Standards (Principles, ...)

Owned cards:

- Focusing too much on reusability can be a huge bottleneck to continuously deploying software (Reusing at the component level as well as reusing packaged software within application may involve two side effects for CD: (i) it increases dependencies between teams, at which all teams depend on single and shared unit. In addition, there is a need to spend huge amount of time to maintain and obtain agreement from all teams before evolving a unit. So (re)-using packaged software may reduce ability to write the unit test and ability to put them on continuous deployment pipeline. Therefore, it is recommended that in the CD process everything should be decentralized. Shahin et al. (2016)
- Implement architecture principles (Architecture principles should be put in place to guide the logging of the applications. The architecture principles can include rules on what to log, the log format, the logging mechanism, etc. Chen (2015 IFIP)
- Isolate changes and minimize the impact of changes (Some interviewees noted that they always considered that software units (e.g., component or service) had to be small enough to be modified, replaced and run independently in productions environment Shahin et al. (2016)
- Modular and loosely coupled architecture (A modular architecture with small components, which makes it possible for many teams to work in parallel. Loosely coupled architecture with as few dependencies as possible between the components, which means that changes can be committed independently. Mårtensson et al (2017)
- small and independent deployment units (Compared with the work by Bass et al. (2015), Schermann et al. (2016), and Newman (2015), our study has independently identified the reasons why practicing CD within monoliths is difficult (Finding 1). We found that growing monoliths leads to increased complexity of internal and external dependencies, restricted automation (test and deployment), impeding teams’ ownership and having slow and inconsistent feedback (Laukkanen et al. 2017; Schermann et al. 2016; Arun 2015), which together can be roadblocks to frequent and automatic deployment. Similar to Newman (2015), and Lewis and Fowler (2010), we can conclude that the principle of “small and independent deployment units” is an alternative to monolithic systems for this purpose and serves as a foundation to design CD-driven architectures (Finding 4). Shahin et al (2019)
- delaying (architectural) design decisions (Compared with the less frequent releases, CD places greater emphasis on evolutionary changes. This requires delaying (architectural) design decisions to the last possible moment. Shahin et al (2019)

Product architecture (Different aspects which leverages agile principles e.g., monitorability, modifiability etc.)

Owned cards:

- Modifiable architecture (An application should be architected in a way that supports this style of software development. In general, the software application should exhibit a high degree of modifiability to allow constant incremental adding of small new features. Chen (2015 IFIP)
- Monitorable software application (Apart from putting in proper monitoring tools, the software application itself should be architected in a way that is amenable for monitoring. With CD, we sometimes need applications to expose additional monitoring interfaces to facilitate CD. Chen (2015 IFIP)
- small and independently deployable units (e.g., service, component, and database) should be considered as a foundation for architecture design. (According to Shahin there are three options for creating small and independently deployable units. Microservices: This architectural style tries to break complex systems into small, autonomous, independently deployable services. We found that some interviewees’ organizations have successfully implemented this architectural style for enabling CD practice. Vertical layering: splitting out modules or components of the applications into vertical layers rather than employing horizontal layered architecture and then letting each team to be responsible for their own module or component. Database as continuously deployable unit: it is vital to consider database as easily deployable unit like other software components. It is mainly because they are part of deployment process of software components. By incorporating the database into deployment

process as the application, it is expected to potentially prevent unexpected issues that database updates may introduce at the point of deployment. Shahin et al. (2016)

- Micro-services Architectures (There is a dominant architectural style present in these papers: micro-services. Micro-services are a set of small services that can be developed, tested, deployed, scaled, operated and upgraded independently, allowing organizations to gain agility, reduce complexity and scale their applications in the cloud in a more efficient way. Besides that, micro-services are very popular, they are being used and promoted by industry leaders such as Amazon, Netflix and LinkedIn (Villamizar, 2015). Shahin et al. describe micro-services as the first architectural style to be preferred for CD practice, by designing fine-grained applications as a set of small services (Shahin et al., 2016). Three papers (Stahl and Bosch, 2018; Pahl et al., 2018; Berger et al., 2017) state explicitly some specific benefits of employing the micro-services architecture concept. Micro-services are said to be helpful in increasing modularity and isolating changes and as a consequence increasing deployment frequency (Bass, 2017). Bolscher et al (2019)
- Proper logging (It is absolutely critical to collect the operational data in order to both monitor the current state of application in the production and to fix issues and problems [3]. Therefore, log aggregation, analysis and monitoring are becoming increasingly important for deploying software on the continuous basis Shahin et al. (2016)
- Testability inside the architecture (Apart from choosing appropriate testing tools, the interviewees adopted following strategies in order to improve the testability of their system: Improving test quality: improve the quality of test (data). This includes having right number of test cases, improved quality of test data, and right time for testing. Making code more testable: the code should be written to be much more testable. Test automation: Interviewees argued that the architecture should support automating testing Shahin et al. (2016)
- Micro-services architectural style (There is a dominant architectural style present in these papers: micro-services. All 8 papers either discuss micro-services or present them as the “go-to” architectural style for CD/DevOps practices. We observe that only 2 out of the 8 papers also mention other architectural styles or patterns. Micro-services are a set of small services that can be developed, tested, deployed, scaled, operated and upgraded independently, allowing organizations to gain agility, reduce complexity and scale their applications in the cloud in a more efficient way. Besides that, micro-services are very popular, they are being used and promoted by industry leaders such as Amazon, Netflix and LinkedIn (Villamizar, 2015). Shahin et al. describe micro-services as the first architectural style to be preferred for CD practice, by designing fine-grained applications as a set of small services (Shahin et al., 2016). Bolscher et al (2019)

Infrastructure supporting CI/CD process ()

Owned cards:

- Use a common service bus architecture. (In conclusion, it is very likely that CI infrastructure itself becomes heterogeneous and complex. Mordinyi et al. (2011) propose a common service bus architecture around which the various elements are brought together. Eck et al. (2014)

Process control_Preconditions

Report date: 12-10-2020 15:41:27

Number of unsorted cards: 1

Number of sorted cards: 10

Total number of cards: 11

Unsorted cards:

a new name (a new description)

Topics:

Knowledge management ()

Owned cards:

- Knowledge of the continuous deployment pipeline (Knowledge of the continuous deployment pipeline with its continuous integration, automated testing, and release deployment practices is a prerequisite for developers. Leppanen (2015)
- Single and united organisational culture (The findings indicate that adopting the mindset aligned with CI is a challenge. Interviewees were sceptical about the benefits that they could gain from adopting CI. Similarly, Claps [2] found that teams adopting continuous deployment need to adopt the mindset needed for it. This means that there needs to be a shift towards a single and united organisational culture that adopts the principles of CI. Debbiche (2014)
- A "lean" mind-set (The successful use of CD requires a “lean” mind-set. For organisations to become ‘lean’, Ries [37] suggests working in small batches Claps et al. (2015)
- well prepared to handle technical and social adoption challenges with their existing expertise, processes and tools (From a managerial point of view, the findings indicate that organisations need to be well prepared to handle technical and social adoption challenges with their existing expertise, processes and tools before adopting the CD process. Claps et al. (2015)

Infrastructuur (Tools, infrastructuur etc.)

Owned cards:

- Mature infrastructure (Testing tools and the maturity of the infrastructure supporting the CI process is required in order to facilitate the daily tasks involved. Continuous integration advocates the use of automated tools to allow more frequent and efficient integrations. Debbiche (2014)
- Agile development methods are considered a prerequisite for CI (Additionally, research done by Turk et al. [18] on the suitability of agile development methods shows that assumptions (e.g. face-to-face communication, quality assurance, changing requirements) made by such methods are not appropriate for all organisations, products and projects. The authors [18] highlight important limitations of said methods, where two is of particular interest for this research, namely limited support for large complex software and large teams. Agile development methods are considered a prerequisite for CI [14], therefore findings by Turk et al. [18], might apply to the applicability of CI. Debbiche (2014)

- Hardware resources for CI servers (Provide more hardware resources to a product's CI servers to allow the software product to be continuously integrated as often as necessary, thus allowing the software product to be deployed at any time Claps et al. (2015))
- Adjust the architecture by changing technologies or components if needed. (First, understand that the product architecture has a significant effect on the adoption. However, do not let architectural problems keep you from implementing continuous integration. You may adjust the architecture by changing technologies or components if needed. Laukkanen et al. (2015))

Strategy (Approach/strategy top-down, bottom-up etc. Developers.

Tevens organisatieveranderstrategie.)

Owned cards:

- Bottom up approach (However, the bottom up approach seems to have led some to believe that management could be more involved in the overall process of transitioning to CI. Another reason could be the use of a bottom-up approach where directions and guidance come from experience gained in pilot teams. This might have led to the confusion regarding a vision, since the most expected communication channel for organisational visions ought to be management. Debbiche (2014))
- Successful cases of process change all share a common denominator, which is divided into 8 steps, liable for their success of change implementation. (Results from this study show that there is an ambiguity regarding the goals and the organisational vision for the implementation of CI. More than half (7 of 13) of the interviewed subjects had a hard time answering questions due to their lack of knowledge of what was expected from them personally. According to Kotter [10], successful cases of process change all share a common denominator, which is divided into 8 steps, liable for their success of change implementation. Kotter [10] highlights the importance of spending the necessary amount of time on each step, and failing to do so will lead to undesired results. For instance, lacking and under-communicating a vision might lead to confusion and incompatible results, which will take the organisation in the wrong direction. This might be a reason to the unclear goals and expectations regarding CI at the case company. Debbiche (2014) (Kotter [10]))

Process control_ Process design

Report date: 12-10-2020 15:42:35

Number of unsorted cards: 1

Number of sorted cards: 9

Total number of cards: 10

Unsorted cards:

a new name (a new description)

Topics:

Naar CSF 'Architecture\product architecture' ()

Owned cards:

- Choose a good system design solution. (Four system design solutions were reported: system modularization, hidden changes, rollback and redundancy (Table 17). The design solutions considered what kind of properties the system should have to enable adopting CD. Laukkanen et al (2017))
- Developers must think about the complete system: (Developers understand the functions and design of the whole system, and not just their own sub-system. The developers have knowledge about how the functions utilize the different sub-systems and how different subsystems are connected to each other. Mårtensson et al (2017))

Organizing teams ()

Owned cards:

- Institutionalizing CI (Three approaches have been discussed for how to institutionalize CI: shared service, committee, and communities of practice. Offering CI as shared service means establishing a dedicated team that maintains and improves CI, and which centrally provides training and other support. On the downside, all teams involved with CI depend on the central roadmap and cannot deviate easily (Heidrich et al. 2013). Alternatively or as complement, a committee might be established. It consists of subject matter experts across the organization, who discuss the CI roadmap and issue authoritative guidelines for CI use (Williams et al. 2011). A further alternative are communities of practice: usually self-emergent, they are sustained by company-internal thought leaders who drive CI improvements forward and provide guidance on request. Through granting these thought leaders time for engaging in such activities, organizations choose to promote this model (Woodward et al. 2010). To summarize, different models exist to foster CI use and sophistication, ranging from centralized to decentralized. Eck et al. (2014))
- Improved collaboration among teams and team members (In many organizations, Development (Dev) and Operations (Ops) teams form silos that are possibly located in separate departments [31]. Whilst this structure was motivated by traditional methodologies (e.g., waterfall), it is not suitable for recent software development practices that simultaneously deal with agility, and maintaining software on different environments [22]. Iden et al. [23] highlight that effective cooperation between Dev and Ops teams has great impact on the quality of the final product. Any shortcomings in interaction of these members usually manifest in problems such as excluding IT operations from requirement specifications, poor communication and information flow, and lack of knowledge transfer [23]. Lwakatere et al. [29] indicate that collaboration can be enforced through practices such as broadening skill-set, information sharing and shifting responsibilities among these members. Nevertheless, implementing these practices demand changes in team structures, required responsibilities, the work culture of organization and mindset of team members [29; 31]. Some researchers have identified best practices to implement these changes (e.g., team structure). For example, Humble and Molskey [22] suggest re-architecting software product in form of strategic services, and assigning each service to a small cross-functional team who takes the full ownership of it during whole development lifecycle. Our study has revealed several organizational practices improving collaboration among teams and team members to effectively implement CD. Shahin et al (2017))

- Teams and responsibilities: (A project organization that supports both working with functional changes and at the same time takes responsibility for the architecture. The organization can be built on cross-functional teams or component teams (or a mix of both), but with explicit ownership for both a functional change and the design of the whole system. Mårtensson et al (2017)
- Activity sequencing: (Synchronization between the development teams in order to optimize the flow of activities when functions and systems are implemented. Scheduling of prototyping activities and prestudies related to architecture and system design. Mårtensson et al (2017)

Process conditions ()

Owned cards:

- Providing CI with Project Start (It might be that CI is not available directly with project start: extensive manual preparation might be needed to set up the tool chain, or it needs to be amended with new elements that are required for the current project (Conboy et al. 2007; Poole 2004; Verweij, P. J. F. M. et al. 2010). Several authors argue that such delays may jeopardize CI altogether. If CI becomes available in the course of a project, developers are likely to abandon it (Kasoju et al. 2013; Avritzer et al. 2010). In addition, major architectural changes might become necessary to prepare the codebase for effective CI (Kane 2003). Eck et al. (2014)
- Extending CI Beyond Source Code (A number of sources propose capturing all artifacts of software development with CI, not just the source code (Bos and Vriens 2004; Kane 2003). Otherwise, changes in uncontrolled artifacts, e.g. configuration files, firmware, etc. might cause integration and testing errors that are hard to reproduce (Larsson et al. Eck et al. Contemporary Issues in Agile Development 8 Twentieth Americas Conference on Information Systems, Savannah, 2014 2009). However, organizations should carefully evaluate the benefits against its costs; it might make sense to purposefully leave gaps in the CI system (Talby et al. 2005). Eck et al. (2014)
- Flexible organizational structure (Our experience in this study was that the process of changing to a CD organization requires more than providing tool support and automation. CD impacts on organizational structures (e.g., team structures), roles and responsibilities. From practitioners' perspectives, the inflexibility of organizational structures with the spirit of CD practices is the most critical challenge for implementing CD (Finding 3). A software organization can succeed in CD adoption once there is flexibility in optimizing organizational structures to be aligned with CD and certain skills and responsibilities need to be sought. Finding the best organizational structures that suit an organization depends on many factors, such as the flexibility of the current organizational structure, available skills and management procedures (Brown 2015; What Team Structure is Right for DevOps to Flourish 2017). Shahin et al (2019)

Process control_Strategy and approach

Report date: 12-10-2020 15:40:53

Number of unsorted cards: 1

Number of sorted cards: 16

Total number of cards: 17

Unsorted cards:

a new name (a new description)

Topics:

Process ()

Owned cards:

- Well organized incident process (Employ a strategy of stopping the entire team (i.e. as many people as necessary) from doing what they are doing to fix issues to make sure the software is in an ever-ready state of being deployable Claps et al. (2015)
- Adopting the practice of small batches (Develop large features (dark features) in small batches that only appear visible to the customer when the entire feature is finally developed Claps et al. (2015)
- Use parallel running systems for deployment (Use two parallel running systems (the old and the new system) to upgrade the user to the next version of software. Seamless upgrades – complications in implementation of seamless upgrades due to resource limitations, zero downtime deployment and customer data preservation Claps et al. (2015)
- Work breakdown (A way of working that supports work breakdown into small pieces that can be delivered to the software mainline. Directives on whether a commit to the mainline shall only include complete functions or not, and if a commit shall include tests and/or documentation (or if this could be delivered later – or earlier). Mårtensson et al (2017)
- Resilience (Design for failure is considered as the foundation of the architecture in a CD context, in which, instead of preventing failures (reliability), it is more important to learn how to deal with failures (resilience). Shahin et al (2019)
- Use document software products (Maintaining multiple documentation for each product which may have multiple offerings (e.g. a customer-hosted version and a version less Atlassian-hosted online version), where features may be released in one offering, but not in the other. Use Atlassian's own product, Confluence (which is a wiki), to document software products Claps et al. (2015)

Architecture ()

Owned cards:

- Availability of several branches per software product (Develop each feature for a software product on a separate branch and, when completed, merge that feature into the main branch of code. Having one single branch of code to maintain one working version of the software product in a CD environment Claps et al. (2015)
- Devising a Branching Strategy (A major feature of any CI system is branching, i.e. instantiating a copy of the common codebase, locally working with this copy, and feeding the changes back into the codebase after local testing. In its most basic use branching enables concurrent development. Practitioners leverage this feature to implement sophisticated strategies, which we characterize as either feature-motivated or test-motivated. A commonly observed feature-motivated strategy is to branch out every new feature, in order to not jeopardize integrity of the current codebase (Wilcox et al. 2007). Building on this idea, Stober and Hansmann (2010) propose cross-functional development teams when exploring risky technology changes not yet on the roadmap. These teams exist as long as the branch does, i.e. they are being dismantled once their goal is accomplished. To

maintain consistency of the overall codebase, long-lived source code ownerships are assigned in parallel. Of the various test-motivated approaches, Dowling (2013) presents a variant in which a testing branch is created every two weeks. A dedicated team performs tests on this branch, while development continues uninterrupted. All encountered errors that could be fixed within this period are fed back into the codebase; a new test branch is then created and the cycle begins anew. Similarly, Ali et al. (2012) propose performing complex and time-consuming tests on a dedicated branch. Interestingly, none of these authors mention a hybrid strategy, combining feature-motivated and testing motivated approaches. Eck et al. (2014)

Knowledge management ()

Owned cards:

- Adopt 'social rules' which must be adhered to when deploying software. (A lack of understanding of the CD process by novice developers due to inconsistent documentation and a lack of industry standards Claps et al. (2015))

Management (Organization structure, collaboration etc.)

Owned cards:

- Good overview on organization structure (The biggest challenge has been organizational. Release activities involve many divisions of the company. Each has its own interests, ways of working, and perceived territories of control. Tension existed between divisions due to competing goals. For example, we needed root access to the servers, and another team controlled this permission. Arriving at a solution involved much consultation and negotiation over six months. To address the organizational challenges, the leadership team restructured the organization to break down barriers among teams and promote a collaborative culture. The situation has improved since. Although literature on organizational change exists,4 little, if any, research focuses on introducing CD to an organization. Further research on this topic—for example, understanding the challenges in more depth and developing strategies and practices to tackle them more effectively—will significantly help an organization's smooth adoption of CD. Chen (2015 IEEE)
- Management support (Our analysis shows that compared to CDE, successfully adopting CD needs better management support. To achieve CD, organizations must break down barriers at production. This is mainly achievable by allowing developers to be part of deployment decision-making and placing more trust on them. By this, the manual approval process described in Section IV.C.5 will be significantly reduced. Shahin et al. (2017) (IEEE)
- invest in the communication (Third, invest in the communication between different sites and avoid centralizing different competencies to certain sites. Understand the value of face-to-face communication and local support personnel when adopting CI. Laukkanen et al. (2015)
- Clarifying Division of Labor (CI connects previously separated organizational functions through process automation, e.g. development and release management. A number of authors suggest that it is vital to clarify division of labor, such as establishing hand-over procedures along the CI chain (Maruping 2010; Thiyagarajan and Verma 2009) or introducing suitable policies, e.g. who is allowed to perform branch merging (Williams et al. 2011). However, organizations should resist re-establishing exactly those silos that CI is supposed to help overcome. A possible approach is to additionally assign source code ownership: across the entire CI chain, owners are made responsible to resolve integration errors affecting their code (Moore and Spens 2008; Stober and Hansmann 2010). Eck et al. (2014)
- Devising an Assimilation Path (Sources suggest various approaches to drive CI assimilation, which we characterize as low-hanging fruits, extend nucleus, and challenge-oriented. All approaches emphasize putting CI in a larger context of overall agile development assimilation. Stober and Hansmann (2010) propose a low-hanging fruits approach: those aspects should be introduced first that are easily implemented and require few behavioural changes. As the organizational members get accustomed to the change it can then explore more sophisticated practices, one of which is CI. Olsson et al. (2012) suggest an extend nucleus approach: starting with a core CI system that automates integration and testing, the ambition and reach of CI is extended continuously, until CI is used to leverage operational software for experimentation. Finally, Conboy et al. (2007) suggest a challenge-oriented approach: after prioritization of current challenges, the organization should implement those agile practices first that are likely to have the highest return. In conclusion, there is consensus that CI is not a means in itself and should be seen as component of agile development. Differences arise on who should have the initiative – ranging from a bottom-up attitude like low-hanging fruits up to a top-down stance as challenge-oriented. Eck et al. (2014)
- CI and Distributed Development (CI has the potential to increase collaboration among spatially distributed development through replication, synchronization, and transparency. Modern CI systems support local replication of the codebase, which makes the CI chain resilient against connectivity disruptions (Sauer 2010; Poole 2004). The different development teams synchronize their work with every integration (Hillegersberg et al. 2011; Avritzer et al. 2010). There is a trade-off between synchronization frequency and communication needs, however. If changes are instantly synchronized against the full codebase, then communication between locations is likely to increase in case of encountered integration errors (Avritzer et al. 2010). Lastly, CI creates transparency about the project status, because every team member has access to the codebase (Bose 2008; Avritzer et al. 2010). In summary, the sources argue that the various elements of CI collectively foster collaboration in distributed development settings. Eck et al. (2014)

<verplaatsen naar Preconditions> (Overcoming initial learning phase -> Topic 'Knowledge Management'.)

Owned cards:

- Overcoming Initial Learning Phase (In conclusion, it is critical that organizations prepare for a dip in productivity, accept this initial negative impact as investment, and minimize the learning phase through targeted training. Eck et al. (2014)

Transferred to my CSF:

Customer_Complexity

Naar CSF 'Architurre' ()

Owned cards:

- Have an architecture that supports continuous practices (flexible and modular architecture). In example loosely coupled architecture (Flexible and Modular Architecture As discussed in Section IV.D.3.a, technical dependency between codes or components can act as an obstacle to adopt CDE and CD. The reviewed studies reported that delivering software in days instead of months requires architectures that support CDE adoption [S7, S12, S28, S30, S45, S51, S57]. The software architecture should be designed in a way that software features can be developed and deployed independently. Loosely coupled architecture minimizes the impact of changes as well. For example, Laukkanen et al. [S45] observed that the studied organization had to rearchitect their product (e.g., removing components caused trouble) to better adopt CI and CDE. Laukkanen et al. (2017) Shahin et al. (2017)
- Implement rollback and redundancy properties. (Rollback and redundancy. Rollback and redundancy are properties of the system and are important when releasing the system. Rollback means that the system is built so that it can be downgraded automatically and safely if a new version causes unexpected problems [C5]. Thus, rollback mechanism reduces the risk of deploying more bugs. Redundancy means that the production system contains multiple copies of the software running simultaneously. This allows seamless updates, preserving customer data [C5] and reducing deployment downtime [C5, C25c]. Laukkanen et al. (2017)

Appendix I

Grouped measures after Metaplan session

SQ2: Preconditions

The measures of Preconditions are:

Topics	Measure	Measure context
Knowledge management	Knowledge of the continuous deployment pipeline	Developers must have sufficient proficiency and knowledge of typical continuous-deployment practices. Also, developers' reputations are on the line: deploying a broken build to customers could strain the relationship between parties and create an unwanted user experience. Any lack of confidence in an application's quality is amplified by the knowledge that any and all changes are immediately deployed. Knowledge of the continuous-deployment pipeline with its continuous integration, automated testing, and release deployment practices is a prerequisite for developers. Leppanen (2015)
	Single and united organisational culture	The findings indicate that adopting the mindset aligned with CI is a challenge. Interviewees were skeptical about the benefits that they could gain from adopting CI. Similarly, Claps [2] found that teams adopting continuous deployment need to adopt the mindset needed for it. This means that there needs to be a shift towards a single and united organisational culture that adopts the principles of CI. Debbiche (2014)
	A "lean" mind-set	The successful use of CD requires a "lean" mind-set. For organisations to become 'lean', Ries [37] suggests working in small batches Claps et al. (2015)
	well prepared to handle technical and social adoption challenges with their existing expertise, processes and tools	From a managerial point of view, the findings indicate that organisations need to be well prepared to handle technical and social adoption challenges with their existing expertise, processes and tools before adopting the CD process. Claps et al. (2015)
Infrastructuur (Tools, infrastructuur etc.)	Mature infrastructure	Testing tools and the maturity of the infrastructure supporting the CI process is required in order to facilitate the daily tasks involved. Continuous integration advocates the use of automated tools to allow more frequent and efficient integrations. Debbiche (2014)
	Agile development methods are considered a prerequisite for CI	Additionally, research done by Turk et al. [18] on the suitability of agile development methods shows that assumptions (e.g. face-to-face communication, quality assurance, changing requirements) made by such methods are not appropriate for all organisations, products and projects. The authors [18] highlight important limitations of said methods, where two is of particular interest for this research, namely limited support for large complex software and large teams. Agile development methods are considered a prerequisite for CI [14], therefore findings by Turk et al. [18], might apply to the applicability of CI. Debbiche (2014)
	Hardware resources for CI servers	Provide more hardware resources to a product's CI servers to allow the software product to be continuously integrated as often as necessary, thus allowing the software product to be deployed at any time Claps et al. (2015)
	Adjust the architecture by changing technologies or components if needed.	First, understand that the product architecture has a significant effect on the adoption. However, do not let architectural problems keep you from implementing continuous integration. You may adjust the architecture by changing technologies or components if needed. Laukkanen et al. (2015)
Strategy (Approach/strategy top-down, bottom-up etc. Developers.Tevens organisatieveranderingstrategie.)	Bottom up approach	However, the bottom up approach seems to have led some to believe that management could be more involved in the overall process of transitioning to CI. Another reason could be the use of a bottom-up approach where directions and guidance come from experience gained in pilot teams. This might have led to the confusion regarding a vision, since the most expected communication channel for organisational visions ought to be management. Debbiche (2014)
	Step-by-step implementation	Results from this study show that there is an ambiguity regarding the goals and the organisational vision for the implementation of CI. More than half (7 of 13) of the interviewed subjects had a hard time answering questions due to their lack of knowledge of what was expected from them personally. According to Kotter [10], successful cases of process change all share a common denominator, which is divided into 8 steps, liable for their success of change implementation. Kotter [10] highlights the importance of spending the necessary amount of time on each step, and failing to do so will lead to undesired results. For instance, lacking and under-communicating a vision might lead to confusion and incompatible results, which will take the organisation in the wrong direction. This might be a reason to the unclear goals and expectations regarding CI at the case company. Debbiche (2014) (Kotter [10])

SQ3: Goals

The found measures of Preconditions is:

Goals	Set clear goals for teams	Lack of setting up clear goals for the teams migrating towards CI is currently an impediment for the teams. Debbiche (2014)
-------	---------------------------	---

SQ4: Strategy and approach

The measures of Strategy and approach are:

Knowledge management	Overcoming Initial Learning Phase	In conclusion, it is critical that organizations prepare for a dip in productivity, accept this initial negative impact as investment, and minimize the learning phase through targeted training. Eck et al. (2014)
	Adopt 'social rules' which must be adhered to when deploying software.	A lack of understanding of the CD process by novice developers due to inconsistent documentation and a lack of industry standards Claps et al. (2015))

Process	Well organized incident process	Employ a strategy of stopping the entire team (i.e. as many people as necessary) from doing what they are doing to fix issues to make sure the software is in an ever-ready state of being deployable Claps et al. (2015)
	Adopting the practice of small batches	Develop large features (dark features) in small batches that only appear visible to the customer when the entire feature is finally developed Claps et al. (2015) A way of working that supports work breakdown into small pieces that can be delivered to the software mainline. Directives on whether a commit to the mainline shall only include complete functions or not, and if a commit shall include tests and/or documentation (or if this could be delivered later – or earlier). Mårtensson et al (2017)
	Use parallel running systems for deployment	Use two parallel running systems (the old and the new system) to upgrade the user to the next version of software. Seamless upgrades – complications in implementation of seamless upgrades due to resource limitations, zero downtime deployment and customer data preservation Claps et al. (2015)
	Resilience	Design for failure is considered as the foundation of the architecture in a CD context, in which, instead of preventing failures (reliability), it is more important to learn how to deal with failures (resilience). Shahin et al (2019)
	Use document software products	Maintaining multiple documentation for each product which may have multiple offerings (e.g. a customer-hosted version and a versionless Atlassian-hosted online version), where features may be released in one offering, but not in the other. Use Atlassian's own product, Confluence (which is a wiki), to document software products Claps et al. (2015)
Architecture	Devising a Branching Strategy	A major feature of any CI system is branching, i.e. instantiating a copy of the common codebase, locally working with this copy, and feeding the changes back into the codebase after local testing. In its most basic use branching enables concurrent development. Practitioners leverage this feature to implement sophisticated strategies, which we characterize as either feature-motivated or test-motivated. A commonly observed feature-motivated strategy is to branch out every new feature, in order to not jeopardize integrity of the current codebase (Wilcox et al. 2007). Building on this idea, Stober and Hansmann (2010) propose cross-functional development teams when exploring risky technology changes not yet on the roadmap. These teams exist as long as the branch does, i.e. they are being dismantled once their goal is accomplished. To maintain consistency of the overall codebase, long-lived source code ownerships are assigned in parallel. Of the various test-motivated approaches, Dowling (2013) presents a variant in which a testing branch is created every two weeks. A dedicated team performs tests on this branch, while development continues uninterrupted. All encountered errors that could be fixed within this period are fed back into the codebase; a new test branch is then created and the cycle begins anew. Similarly, Ali et al. (2012) propose performing complex and time consuming tests on a dedicated branch. Interestingly, none of these authors mention a hybrid strategy, combining feature-motivated and testingmotivated approaches. Eck et al. (2014)
	Availability of several branches per software product	Develop each feature for a software product on a separate branch and, when completed, merge that feature into the main branch of code. Having one single branch of code to maintain one working version of the software product in a CD environment Claps et al. (2015)
Management (Organization structure, collaboration etc.)	Management support	Our analysis shows that compared to CDE, successfully adopting CD needs better management support. To achieve CD, organizations must break down barriers at production. This is mainly achievable by allowing developers to be part of deployment decision-making and placing more trust on them. By this, the manual approval process described in Section IV.C.5 will be significantly reduced. Shahin et al. (2017) (IEEE)
	Good overview on organization structure	The biggest challenge has been organizational. Release activities involve many divisions of the company. Each has its own interests, ways of working, and perceived territories of control. Tension existed between divisions due to competing goals. For example, we needed root access to the servers, and another team controlled this permission. Arriving at a solution involved much consultation and negotiation over six
	invest in the communication	Third, invest in the communication between different sites and avoid centralizing different competencies to certain sites. Understand the value of face-to-face communication and local support personnel when adopting CI. Laukkanen et al. (2015)
	Clarifying Division of Labor	CI connects previously separated organizational functions through process automation, e.g. development and release management. A number of authors suggest that it is vital to clarify division of labor, such as establishing hand-over procedures along the CI chain (Maruping 2010; Thiagarajan and Verma 2009) or introducing suitable policies, e.g. who is allowed to perform branch merging (Williams et al. 2011). However, organizations should resist re-establishing exactly those silos that CI is supposed to help overcome. A possible approach is to additionally assign source code ownership: across the entire CI chain, owners are made responsible to resolve integration errors affecting their code (Moore and Spens 2008;
	Devising an Assimilation Path	Sources suggest various approaches to drive CI assimilation, which we characterize as low-hanging fruits, extend nucleus, and challenge-oriented. All approaches emphasize putting CI in a larger context of overall agile development assimilation. Stober and Hansmann (2010) propose a low-hanging fruits approach: those aspects should be introduced first that are easily implemented and require few behavioral changes. As the organizational members get accustomed to the change it can then explore more sophisticated practices, one of which is CI. Olsson et al. (2012) suggest an extend nucleus approach: starting with a core CI system that automates integration and testing, the ambition and reach of CI is extended continuously, until CI is used to leverage operational software for experimentation. Finally Conboy et al. (2007) suggest a challenge-oriented approach: after prioritization of current challenges, the organization should implement those agile practices first that are likely to have the highest return. In conclusion, there is consensus that CI is not a means in itself and should be seen as component of agile development. Differences arise on who should have the initiative – ranging from a bottom-up attitude like low-hanging fruits up to a top-down
	CI and Distributed Development	CI has the potential to increase collaboration among spatially distributed development through replication, synchronization, and transparency. Modern CI systems support local replication of the codebase, which makes the CI chain resilient against connectivity disruptions (Sauer 2010; Poole 2004). The different development teams synchronize their work with every integration (Hillegersberg et al. 2011; Avritzer et al. 2010). There is a trade-off between synchronization frequency and communication needs, however. If changes are instantly synchronized against the full codebase, then communication between locations is likely to increase in case of encountered integration errors (Avritzer et al. 2010). Lastly, CI creates transparency about the project status, because every team member has access to the codebase (Bose 2008; Avritzer et al. 2010). In summary, the sources argue that the various elements of CI collectively

SQ6: Process Design

The measures of Process Design are:

Organizing teams	Improved collaboration among teams and team members	In many organizations, Development (Dev) and Operations (Ops) teams form silos that are possibly located in separate departments [31]. Whilst this structure was motivated by traditional methodologies (e.g., waterfall), it is not suitable for recent software development practices that simultaneously deal with agility, and maintaining software on different environments [22]. Iden et al. [23] highlight that effective cooperation between Dev and Ops teams has great impact on the quality of the final product. Any shortcomings in interaction of these members usually manifest in problems such as excluding IT operations from requirement specifications, poor communication and information flow, and lack of knowledge transfer [23]. Lwakatere et al. [29] indicate that collaboration can be enforced through practices such as broadening skill-set, information sharing and shifting responsibilities among these members. Nevertheless, implementing these practices demand changes in team structures, required responsibilities, the work culture of organization and mindset of team members [29; 31]. Some researchers have identified best practices to implement these changes (e.g., team structure). For example, Humble and Molskey [22] suggest re-architecting software product in form of strategic services, and assigning each service to a small cross-functional team who takes the full ownership of it during whole development lifecycle. Our study has revealed several organizational practices improving collaboration among teams and team members to effectively implement CD. Shahin et al (2017)
	Institutionalizing CI	Three approaches have been discussed for how to institutionalize CI: shared service, committee, and communities of practice. Offering CI as shared service means establishing a dedicated team that maintains and improves CI, and which centrally provides training and other support. On the downside, all teams involved with CI depend on the central roadmap and cannot deviate easily (Heidrich et al. 2013). Alternatively or as complement, a committee might be established. It consists of subject matter experts across the organization, who discuss the CI roadmap and issue authoritative guidelines for CI use (Williams et al. 2011). A further alternative are communities of practice: usually self-emergent, they are sustained by company-internal thought leaders who drive CI improvements forward and provide guidance on request. Through granting these thought leaders time for engaging in such activities, organizations choose to promote this model (Woodward et al. 2010). To summarize, different models exist to foster CI use and sophistication, ranging from centralized to decentralized. Eck et al. (2014)
	Teams and responsibilities:	A project organization that supports both working with functional changes and at the same time takes responsibility for the architecture. The organization can be built on cross-functional teams or component teams (or a mix of both), but with explicit ownership for both a functional change and the design of the whole system. Mårtensson et al (2017)
	Activity sequencing	Synchronization between the development teams in order to optimize the flow of activities when functions and systems are implemented. Scheduling of prototyping activities and prestudies related to architecture and system design. Mårtensson et al (2017)
	Clarity, Visibility and awareness of a project status to the team	It is evident that sharing responsibilities of software delivery with all members [22] could promote coordination and collaboration in a team. We observed the practice of rotating roles [31] between developers and operations staff, i.e., involving developers in testing and QAs in development tasks. Our findings have highlighted the significant role of visibility and awareness of a project status for improving collaboration in a team and successfully adopting CD. Humble and Farley [21] recommend using big, ubiquitous dashboards in each team room to visualize status of builds and sharing feedback with everybody. Our participants also indicated raising awareness in teams by involving operations staff in daily meetings [31] and interacting with Kanban board. Shahin et al (2017)
	Promote a collaborative culture.	The biggest challenge has been organizational. Release activities involve many divisions of the company. Each has its own interests, ways of working, and perceived territories of control. Tension existed between divisions due to competing goals. For example, we needed root access to the servers, and another team controlled this permission. Arriving at a solution involved much consultation and negotiation over six months. To address the organizational challenges, the leadership team restructured the organization to break down barriers among teams and promote a collaborative culture. The situation has improved since. Although literature on organizational change exists,4 little, if any, research focuses on introducing CD to an organization. Further research on this topic—for example, understanding the challenges in more depth and developing strategies and practices to tackle them more effectively—will significantly help an organization's smooth adoption of CD. Chen (2015 IEEE)
	Strong and proper communication and coordination between multiple teams	Team structures and interactions among multiple teams working on a same codebase system play an important role in successfully implementing CDE and CD practices. Several of the reviewed studies [S6, S31, S45, S50, S56, S57] reported that high cross-team dependencies prohibited development teams to develop, evolve and deploy applications or components and services into production independently of each other. This issue also has major impact on practicing CI as a small build break or test failure may have ripple effects on different teams [S50]. The author of [S56] argued that feature and module (hardware) teams developing embedded domain systems were highly dependent, in which each feature was compiled, tested and built by a combination of both teams. This required a strong and proper communication and coordination among them. Two studies [S50, S57] in this group also discussed that nonexistence of a suitable architecture can increase the cross-team dependency. Shahin et al. (2017)
Planning	Interdependent deployment planning.	Engage in an increased amount of planning in terms of planning around deployments with multiple systems that are interdependent. Claps et al. (2015)
	Team coordination	requires extra effort to coordinate multiple teams (Engage in an increased amount of planning in terms of planning around deployments with multiple systems that are interdependent Claps et al. (2015)
	Give the team time to adopt CI	Give enough time for your team members to overcome the initial learning phase. You have to lower the priority of new features during the adoption. Laukkanen et al. (2015)

Process conditions	Providing CI with Project Start	It might be that CI is not available directly with project start: extensive manual preparation might be needed to set up the tool chain, or it needs to be amended with new elements that are required for the current project (Conboy et al. 2007; Poole 2004; Verweij, P. J. F. M. et al. 2010). Several authors argue that such delays may jeopardize CI altogether. If CI becomes available in the course of a project, developers are likely to abandon it (Kasoju et al. 2013; Avritzer et al. 2010). In addition, major architectural changes might become necessary to prepare the codebase for effective CI (Kane 2003). Eck et al. (2014)
	Extending CI Beyond Source Code	(A number of sources propose capturing all artifacts of software development with CI, not just the source code (Bos and Vriens 2004; Kane 2003). Otherwise, changes in uncontrolled artifacts, e.g. configuration files, firmware, etc. might cause integration and testing errors that are hard to reproduce (Larsson et al. Eck et al. Contemporary Issues in Agile Development 8 Twentieth Americas Conference on Information Systems, Savannah, 2014 2009). However, organizations should carefully evaluate the benefits against its costs; it might make sense to purposefully leave gaps in the CI system (Talby et al. 2005). Eck et al. (2014)
	Flexible organizational structure	Our experience in this study was that the process of changing to a CD organization requires more than providing tool support and automation. CD impacts on organizational structures (e.g., team structures), roles and responsibilities. From practitioners' perspectives, the inflexibility of organizational structures with the spirit of CD practices is the most critical challenge for implementing CD (Finding 3). A software organization can succeed in CD adoption once there is flexibility in optimizing organizational structures to be aligned with CD and certain skills and responsibilities need to be sought. Finding the best organizational structures that suit an organization depends on many factors, such as the flexibility of the current organizational structure, available skills and management procedures (Brown 2015; What Team Structure is Right for DevOps to Flourish 2017). Shahin et al (2019)

SQ7: Motivation

The found measures of Motivation is:

Motivation	Company-wide effort	Ensure that top-management implement a strategy to push the need to implement the CD process. Claps et al. (2015)
------------	---------------------	---

SQ8: Resistance to Change

The measures of Resistance to Change are:

Mindset and process	Use a CI driver to implement CI	Similarly, a CI Driver involved in coaching and helping teams with transitioning to CI highlights the importance of questioning a change Debbiche (2014)
	Improve communication among developers and managers.	Software developers may feel an increased amount of pressure to have code ready to be deployed immediately Claps et al. (2015)
Process	Work with more experienced teams	Continuous integration emphasises early and frequent integrations. As a result, developers are compelled to expose their work earlier. Some interviewees (4 out of 13) found this to be challenge because they were used to big bang integrations at the end of a sprint. This gave them enough time to polish their code before integrating it. With the adoption of CI and the increase in integration frequencies, developers are worried about integrating low quality code that could be questioned by experts and managers, as an agile coach put it, "some teams they are not familiar or used to frequent delivery, because they feel safe if they can deliver once a month because they can make everything ready, if they have some changes, he can correct it on his own branch, don't have to deliver to the main branch and then everybody can see your faults right." Developer confidence plays a role in this issue. Teams that are more experienced working with CI seem to be more comfortable about exposing their work earlier. Debbiche (2014)
	Step by step increase integration frequency	The initiative to adopt CI has resulted in increased pressure on the teams according to some interviewees (4 out of 13). Despite the positive support and attitude towards the concept of CI, teams feel that management would like it to happen faster than currently possible which leads to increased pressure. Some developers feel that they lack the confidence and experience to reach desired integration frequencies. There seems to be a general consensus among developers that transitioning to CI carries risks, a period of chaos and increased pressure. Hence, the frequency of integrations and how to proceed should be done in steps in order to minimize the risk of increased pressure. Debbiche (2014)
	Sharing knowledge among team members	Broadening responsibilities of developers to a larger extent may negatively impact their productivity in core tasks. Shifting extensive amount of operations' responsibilities to developers could cause fear of losing jobs for Ops team and may negatively affect success of transition to CD. Organizations should extensively promote knowledge sharing among team members to complement areas of skill-set and collaboratively work towards a shared goal. Shahin et al (2017)

Appendix J

The composite framework of the clusters Process control/management/governance and Customer

CSF	Measure	Description	Examples
Process control/management/governance Measures			
Preconditions	Implement Knowledge management strategy	A good shared knowledge in advance of CI and Lean	<ul style="list-style-type: none"> - Knowledge of the continuous deployment pipeline (Leppanen (2015)) - Single and united organisational culture (Debbiche (2014)) - A "lean" mind-set (Claps et al. (2015)) - well prepared to handle technical and social adoption challenges with their existing expertise, processes and tools (Claps et al. (2015))
	Select a good infrastructure and resource strategy	A mature infrastructure with good resources available	<ul style="list-style-type: none"> - Mature infrastructure (Debbiche (2014)) - Agile development methods are considered a prerequisite for CI (Debbiche (2014)) - Hardware resources for CI servers (Claps et al. (2015)) - Adjust the architecture by changing technologies or components if needed. (Laukkanen et al. (2015))
	Select an appropriate management strategy	Selecting a suitable strategy and implementing this step by step	<ul style="list-style-type: none"> - Bottom up approach (Debbiche (2014)) - Step-by-step implementation (Debbiche (2014) (Kotter [10]))
Goals	Establishing proper and well-defined goals	Making goals as clear as possible so that everyone understands what the goal is	<ul style="list-style-type: none"> - Set clear goals for teams (Debbiche (2014))
Strategy and approach	Choose a well-organized process strategy	A well-organized process with space for resilience (use of small batches, parallel systems, well-organized incident process)	<ul style="list-style-type: none"> - Well organized incident process (Claps et al. (2015)) - Adopting the practice of small batches (Claps et al. (2015) Mårtensson et al (2017)) - Use parallel running systems for deployment (Claps et al. (2015)) - Resilience (Shahin et al (2019)) - Use document software products (Claps et al. (2015))
	Select the right branch strategy	Choosing the industry strategy with the availability of multiple branches can be beneficial.	<ul style="list-style-type: none"> - Devising a Branching Strategy (Eck et al. (2014)) - Availability of several branches per software product (Claps et al. (2015))
	Maintain a knowledge management strategy	Giving developers the time during the implementation/learning phase	<ul style="list-style-type: none"> - Overcoming Initial Learning Phase (Eck et al. (2014)) - Adopt 'social rules' which must be adhered to when deploying software. (Claps et al. (2015))
	Implement an appropriate management strategy	Applying a suitable management strategy	<ul style="list-style-type: none"> - Management support (Shahin et al. (2017) (IEEE)) - Good overview on organization structure (Chen (2015 IEEE)) - invest in the communication (Laukkanen et al. (2015)) - Clarifying Division of Labor (Eck et al. (2014)) - Devising an Assimilation Path (Eck et al. (2014)) - CI and Distributed Development (Eck et al. (2014))

Architecture	Establishing well-defined Architecture guidelines	Provide well-described guidelines where it is made clear what the architecture must comply with	<ul style="list-style-type: none"> - Implement architecture principles (Chen (2015 IFIP)) - Focusing too much on reusability can be a huge bottleneck to continuously deploying software (Shahin et al. (2016)) - Isolate changes and minimize the impact of changes (Shahin et al. (2016)) - Have an architecture that supports continuous practices (flexible and modular architecture). In example loosely coupled architecture. (Laukkanen et al. (2017) Shahin et al. (2017)) - small and independent deployment units (Shahin et al (2019)) - delaying (architectural) design decisions (Shahin et al (2019))
	Establish clear guidelines for product architecture	Provide clear guidelines for the architecture (various aspects using agile principles, such as controllability, modifiability, logability)	<ul style="list-style-type: none"> - Monitorable software application (Chen (2015 IFIP)) - Modifiable architecture (Chen (2015 IFIP)) - Micro-services Architectures: small and independently deployable units (Shahin et al. (2016)) - Proper logging (Shahin et al. (2016)) - Testability inside the architecture (Shahin et al. (2016)) - Choose a good system design solution. (Laukkanen et al (2017)) - Developers must think about the complete system. (Mårtensson et al (2017)) - Implement rollback and redundancy properties. (Laukkanen et al (2017))
	Use a good defined architecture strategy	Select the appropriate infrastructure to support the CI/CD process	<ul style="list-style-type: none"> - Use a common service bus architecture. (Eck et al. (2014))
Process Design	Establish a collaboration strategy	Improve the (internal) collaboration possibilities with transparent communication	<ul style="list-style-type: none"> - Improved collaboration among teams and team members (Shahin et al (2017)) - Institutionalizing CI (Eck et al. (2014)) - Teams and responsibilities (Mårtensson et al (2017)) - Activity sequencing (Mårtensson et al (2017)) - Clarity, Visibility and awareness of a project status to the team (Shahin et al (2017)) - Promote a collaborative culture. (Chen (2015 IEEE)) - Strong and proper communication and coordination between multiple teams (Shahin et al. (2017))
	Establish the right process conditions from the start	When initiating the adoption of the CI/CD process, it is important to be well equipped and prepared.	<ul style="list-style-type: none"> - Providing CI with Project Start (Eck et al. (2014)) - Extending CI Beyond Source Code (Eck et al. (2014)) - Flexible organizational structure (Shahin et al (2019))
	Create a proper adoption planning	Take the start-up phase into account during the planning phase	<ul style="list-style-type: none"> - Interdependent deployment planning. (Claps et al. (2015)) - Team coordination (Claps et al. (2015)) - Give the team time to adopt CI (Laukkanen et al. (2015))
Motivation	Adopt the CI/CD process with the entire company	Adopting must be a company-wide approach	<ul style="list-style-type: none"> - Company-wide effort (Claps et al. (2015))
Resistance to change	Guide organisational aspects of change	Provide guidance in the organisational aspects of change	<ul style="list-style-type: none"> - Use a CI driver to implement CI (Debbiche (2014)) - Improve communication among developers and managers. (Claps et al. (2015))
	Guide process aspects of change	Provide guidance in the process aspects of change	<ul style="list-style-type: none"> - Work with more experienced teams (Debbiche (2014)) - Step by step increase integration frequency (Debbiche (2014)) - Sharing knowledge among team members (Shahin et al (2017))

Appendix K

Interview Protocol

This appendix provides an overview of the steps that will be followed to establish the data collection through the use of interviews with experts in the field.

Step 1 Selecting the Internet-mediated interview tool

The web conferencing program used in the case organization is Webex. Webex also provides the ability to record the conversation and transcribe the audio directly. The employees of the case organization are familiar with the application and have easy access to it.

The interview will be conducted with Dutch speaking employees and therefore the interview will also be conducted in the Dutch language. The transcription will also be made in Dutch. During the analysis phase there will be a translation to English.

Step 2 Drawing up a Interview Guide

For conducting the interview, an Interview Guide is going to be prepared. This guide can be used as a guideline during the interviews with the different interviewees. It will contain the information of the interview such as the name and the function the interviewee but also the time and date at which the interview is carried out.

In this the intro of the interview is developed based on the example given by Saunders (2018) in Box 10.9, 'Opening a semi-structured interview'. Then the different topics are written out with additional information and the questions that can be asked during the interview to maintain a good structured interview.

The interview guideline with the topics and interview questions is included in Appendix N.

Step 3 Selecting candidates to be interviewed

My focus here is on an Government Organization. This is an organization that is constantly evolving and trying to embrace new work processes. A number of employees within the organization are looking at the possibilities of working with CI/CD. However, this is still on a try-out basis which means that selecting the right people can be a challenge. In chapter 3.2 are the selection criteria

The following candidates were selected for the interviews:

nr	Function	Identification	Interview date:	
1	Innovation Cloud Engineer	ICE1 ICE 2	05-02-2021	14:00 – 15:00
2	External Consultant	EC	19-02-2021	11:15 – 12:15
3	IT Architect	ITA	19-02-2021	15:00 – 16:00
4	Functional Manager	FBI	24-02-2021	10:00 – 11:00
5	Program Manager	PM	05-03-2021	14:00 – 15:00
6	Product Owner	PO	08-03-2021	11:00 – 12:00

Step 4 Send invitation to interviewee

To invite the selected candidates, an initial contact will be made by phone or internal chat application. If interested in participating, an email will be sent. This email will briefly explain the purpose of the study and the relevance of the interview.

After sending the email there will be a telephonic contact after 3 working days if no response has been given in order to provide some extra information by telephone if necessary.

Step 6 preparing the interview

- checking operation of Webex and recording function
- Send to interviewee:

- Information sheet (Annex M)
- confidentiality statement (Annex L)
- check signature of confidentiality statement before interview

Step 5 conducting a pilot interview

A pilot interview will be conducted to test the interview guide with one of the selected interviewees. This will then indicate whether the questions and measures are clear and the examples can be used properly. Also, the technique can be checked for proper functioning. After this interview a short evaluation will take place.

Step 7 Interviewing the interviewee

During the interview the interview will be conducted using the Interview Guide.

First the interviewee will be thanked for participating and a short introduction of the interviewer will be given. Then reference will be made to the information sheet and asked if there are any questions about this.

It is again indicated that confidentiality and anonymity are guaranteed and that the interviewee may always come back to this. The rights of the interviewee are also discussed again, as stated on the information sheet, such as the right not to answer questions and what will happen with the data. It will be indicated that a summary of the research findings will also be shared when available. Permission to record the interview will be asked again. After this, a small summary will be given indicating that consent was given to conduct the interview, what the topic of the interview is, how long it will take, and whether the consent form was read and signed. (saunders, 2018). These steps are included with the research questions in the interview guide which is included in Appendix N.

Step 8 Data processing after the interview

After the interview, the audio interview will be transcribed and the notes will be included in a document. The transcribed interview will be provided to the interviewees with a request to agree to the interview conducted and to ask if they have any additions to the interview held. For this purpose, a response period of approximately 3 working days will be applied to after which the interviewee will be contacted to respond as needed.

Appendix L

Privacy Statement Interview

In this appendix is the privacy Statement signed by the interviewees and the interviewer for ensuring the privacy of the interviewee.

Toestemmingsformulier

Studie: Business Process management & IT
Onderzoek: Measures of Critical Success Factors of CI/CD processes.
Onderzoeker: Nancy de Vries

Rechten van de geïnterviewde:

- Deelname aan dit onderzoeksproject is vrijwillig en zonder betaalde beloning.
- Tijdens het interview staat het u vrij om niet te antwoorden op vragen en bij de oplevering van de notities kunt u ook antwoorden terug trekken
- U heeft controle over de audio-opnames die tijdens het interview worden gemaakt.
- U hebt het recht om u op elk moment uit het onderzoeksproject terug te trekken.
- Persoonlijke informatie van de geïnterviewde wordt geanonimiseerd door te identificeren op functie.

Verzamelde data

De data die wordt verzameld tijdens het interview wordt na het interview getranscribeerd en worden verstrekt aan u. De geanalyseerde informatie en de geanonimiseerde transcripties zullen worden gedeeld met de studiebegeleiders en op worden genomen in de scriptie. Het eindresultaat zal uiteindelijk worden gedeeld op de onderzoek database van de Open Universiteit en kan daarna wellicht worden gebruikt voor verder onderzoek.

Overgenomen met betrekking tot het afnemen van interviews met Rijks-medewerkers:

- A. de geïnterviewde Rijks-medewerker de gelegenheid geeft commentaar te leveren op het verslag van het interview;
- B. geheimhouding zal betrachten t.a.v. alle aan een interview ontleende gegevens welke niet voor publicatie worden vrijgegeven;
- C. de Staat vrijwaart van eventuele aansprakelijkheid, voortvloeiende uit een interview;
- D. de geïnterviewde Rijks-medewerker desgewenst anoniem opvoert;
- E. geen tot individuele natuurlijke personen en rechtspersonen herleidbare gegevens in de openbaarheid zal brengen, dan met nadrukkelijke toestemming van de geïnterviewde;
- F. geen contact opneemt met de personen, hun familieleden of nabestaanden die mogelijk in een interview worden genoemd, zonder vooraf verkregen toestemming van de geïnterviewde;
- G. het concept van een voorgenomen publicatie aan de betreffende partij/ het betreffende departement zal voorleggen ter verificatie of hij/zij heeft voldaan aan de onder E en F genoemde vereisten;
- H. erkent dat de betreffende partij/ het betreffende departement de bevoegdheid heeft publicatie van bepaalde uit een interview afkomstige gegevens te verbieden;
- I. de betreffende partij/ het betreffende departement de gelegenheid geeft commentaar te leveren op het concept van het eindrapport;
- J. van de publicatie een exemplaar aan de directeur van de betreffende partij/ het betreffende departement ter beschikking zal stellen;
- K. de gedragscode voor Onderzoek en Statistiek naleeft van de Vereniging voor Beleidsonderzoek (VBO), de Vereniging voor Statistiek en Onderzoek (VSO) en de MarktOnderzoekAssociatie.nl (MOA).

Bij ondertekening gaat u akkoord met de audio-opname tijdens het interview. Tevens gaat u akkoord dat de (geanonimiseerde) verzamelde informatie zal worden gebruikt ten behoeve van het onderzoek van de interviewer en voor eventueel toekomstig onderzoek.

Geïnterviewde

Interviewer

Naam:

Naam:

Datum:

Datum:

Handtekening:

Handtekening:

Appendix M

Participant Information Sheet

Deze information sheet is opgesteld ten behoeve van uw deelname aan het interview dat uitgevoerd zal worden voor het onderzoek naar 'Measures of Critical Success Factors of CI/CD processes' (Maatregelen van kritieke succesfactoren van CI/CD-processen). Op deze sheet kunt u informatie vinden met betrekking tot het onderzoek, vindt u de informatie over hoe het interview uit zal worden gevoerd en informatie over de topics die tijdens het interview besproken zullen worden.

Inhoud van het onderzoek

Onderzoeksproject:	Measures of Critical Success Factors of CI/CD processes
Opleidingsinstituut	Open Universiteit, faculteit Management, Science & Technology
Opleiding:	Masteropleiding Business Process Management & IT
Onderzoeker:	Nancy de Vries
Supervisors:	Prof.dr.ir. Jos Trienekens Michiel van Belzen MSc.

Doel van het onderzoek

Korte omschrijving:	Het doel van het onderzoek waarvoor dit interview wordt uitgevoerd is het ontwikkelen van een Framework met maatregelen voor Kritieke Succesfactoren dat organisaties een richtlijn biedt voor de adoptie van CI/CD processen. Er is voor het vervolg van het onderzoek gekozen om de focus te leggen op de relatie met de 'klant'. De topics die besproken zullen worden zijn dan ook gerelateerd aan de klant.
---------------------	--

Topics van het interview

Klantacceptatie
betrokkenheid van de klant
Communicatie (Klant)
Kwaliteit
Weerstand tegen verandering (klant)
Complexiteit over de grens van de klantorganisatie heen

Vereisten voor deelname

Methode van dataverzameling:	Interview via Webex. Dit gezien de huidige Covid-19 restricties
Wijze van vastlegging data:	Webexopname, notities op papier
Tijdsduur:	+/- 1 uur
Oplevering transcripties:	Na het interview zal de transcriptie beschikbaar worden gesteld ter inzage zodat de geïnterviewde eventueel hierop nog kan reageren.

Contactinformatie

Mochten er nog vragen zijn naar aanleiding van deze sheet, het onderzoek of het interview dan mag u altijd contact opnemen.

Interviewer:	Nancy de Vries
Functie:	Functioneel beheerder Inning
Afdeling:	
Telefoonnummer:	
Email:	
Adres:	

Appendix N

Interview Guide

Informatie Geïnterviewde

Naam geïnterviewde (or#):	
Functie van geïnterviewde:	
Datum/tijdstip interview:	
Hoe lang werkzaam met CI / CD:	

Intro
<ul style="list-style-type: none">• “Hoi (naam), bedankt dat je mee wil doen met dit interview.”
<ul style="list-style-type: none">• “Mijn naam is Nancy de Vries en ik ben werkzaam als functioneel beheerder bij CAP inning. Ik houd mij hier vooral bezig met de applicatie IED en de Mini One Stop Shop. Ik ben bezig met de opleiding Business Information Management & IT aan de Open Universiteit en dit interview is in kader van mijn afstudeeropdracht.”
<ul style="list-style-type: none">• Je hebt als het goed is een informatie sheet ontvangen met daarop informatie over dit interview, heb je daar nog vragen over?
<ul style="list-style-type: none">• “Vertrouwelijkheid en anonimiteit zijn gegarandeerd tijdens dit interview”
<ul style="list-style-type: none">• “Uw deelname aan dit onderzoeksproject is vrijwillig.• Tijdens het interview staat het u vrij om niet te antwoorden op vragen en bij de oplevering van de notities kunt u ook antwoorden terug trekken• U hebt het recht om u op elk moment uit het onderzoeksproject terug te trekken.”
<ul style="list-style-type: none">• “De data dit wordt verzameld zal vertrouwelijk worden behandeld en dus zonder informatie waardoor dit herleidbaar zou zijn naar u als deelnemer. Afspraken zijn voor beide onderzoeker overeengekomen. Het eindresultaat zal uiteindelijk worden gedeeld op de onderzoek database van de Open Universiteit en kan daarna wellicht worden gebruikt voor verder onderzoek.”
<ul style="list-style-type: none">• “Na afloop van het interview ontvangt u het uitgewerkte interview per mail zodat u hierop toestemming kunt geven, mochten er toch informatie zitten waar u zich niet prettig bij voelt dan mag u dit aangeven en houden wij hier rekening mee”
<ul style="list-style-type: none">• “Wanneer beschikbaar zal het onderzoeksrapport worden gedeeld.”
<ul style="list-style-type: none">• “Heeft u verder nog vragen?”
<ul style="list-style-type: none">• “Dan wil ik u nogmaals vragen om toestemming voor het opnemen van dit interview.”
Start audio-opname
<ul style="list-style-type: none">• “Dan is de audio-opname nu gestart, ik heb uw toestemming gekregen voor het afnemen van dit interview. We zullen tijdens dit interview de maatregelen valideren die behoren tot de succes factoren van het CI/CD proces en de maatregelen die wij graag willen toetsen bij experts in de praktijk. Dit zal ongeveer een uur in beslag nemen. U heeft het toestemmingsformulier gelezen en ondertekend? Dan kunnen we beginnen met het eerste topic.

Interviewvragen herhalend bij elk topic:

- Welke geïdentificeerde maatregelen worden toegepast?
- Hoe wordt maatregel 1 (2, 3, etc.) toegepast? Kun je voorbeelden noemen?
- Wat zijn de resultaten van maatregel 1 (2, 3, etc.) in de context van de bijbehorende CSF(s)? En waaruit (bijv. meetgegevens) blijken die resultaten?
- Waarom zijn dat de resultaten van maatregel 1 (2, 3, etc.)?
- Welke maatregelen hebben jullie toegepast die niet in het overzicht stonden?
- *Als je de maatregelen in volgorde van belangrijkheid zou mogen zetten, hoe zou jij dit weergeven?*

Appendix O

Conceptual Model Customer translated

1	Acceptatie door klant	
	Beschrijving: Aanvaarding van de praktijk van voortdurende releases. Klantperceptie van hun betrokkenheid bij ontwikkeling en klantgedrag. Domeinbeperkingen. Ontdekking van kenmerken.	
	Maatregel	Beschrijving
1A	Houd rekening met de doelstellingen en de context van de klant	Houd rekening met de doelstellingen en de context van de klant bij het maken van afwegingen inzake snelheid, releasefrequentie, veiligheid, leercurve van eindgebruikers, enz.
1B	Overtuig de klant.	Overtuig de klant van de voordelen van continue praktijken door effectieve communicatie (bv. blogs, materialen, workshops, demonstraties), releases van hoge kwaliteit, juist gedrag, enz.
1C	Een cultuur van open communicatie tot stand brengen.	Een cultuur van open communicatie tot stand brengen, bijvoorbeeld toestemming vragen om informatie te verzamelen.

2	Communicatie	
	Beschrijving: Intra- en interteamcommunicatie, de juiste communicatiemiddelen, bewustwording en transparantie.	
	Maatregel	Beschrijving
2A	Een cultuur van open communicatie tussen belanghebbenden tot stand brengen en zorgen voor bewustmaking.	Informatie en kennis delen, activiteiten transparant maken, frequent communiceren, belanghebbenden erbij betrekken en bewustwording creëren.
2B	Overeenstemming over passende werkwijzen.	Maak afspraken over de juiste manier van werken tussen de teamleden en met de klant, neem de juiste beslissingen en trade-offs tijdens het ontwerp van het CI/CD-proces, en wees transparant.

3	Complexiteit over de grenzen van de klantorganisatie heen	
	Beschrijving: Geen toegang tot of controle over een productieomgeving of diversiteit en complexiteit van klantensites, wat het moeilijker maakt om het implementatieproces volledig te automatiseren.	
	Maatregel	Beschrijving
3A	Procesafhankelijkheden afstemmen op continue praktijken.	Overtuig alle actoren in het proces om continue praktijken toe te passen en de actualiseringsmechanismen van de betrokken systemen en apparaten aan te passen.
3B	Vergroot de betrokkenheid van de klant.	Betrek de organisatorische eenheden die het raakvlak met de klanten vormen.
3C	Zorgen voor transparantie over de status quo.	Transparant zijn over de stand van ontwikkelingsprojecten.

4	Betrokkenheid van de klant	
	Omschrijving: Voorbereiden en ontvangen van klanteninput, opzetten van een klantensteekproefgroep, en leveren van functiegroei.	
	Maatregel	Beschrijving
4A	Wees je bewust van de situatie van de klanten.	Gebruik feedbackmechanismen, wees u bewust van mogelijke belemmeringen en neem zelfs de rol van een klant op u.
4B	Betrek de klant bij het CI/CD-proces.	Betrek de klant als actor in het CI/CD-proces, neem maatregelen om feedback te krijgen, houd rekening met de behoeften van de klant en bereid het ontvangende einde voor.
4C	Gebruik strategieën om nauwkeurige verwachtingen over de behoefte van klanten te verkrijgen.	Verschillende maatregelen toepassen om snel feedback te krijgen, een passend model voor klantenbetrokkenheid te ontwikkelen en andere continue praktijken mogelijk te maken (bv. continue verbetering, continue planning).

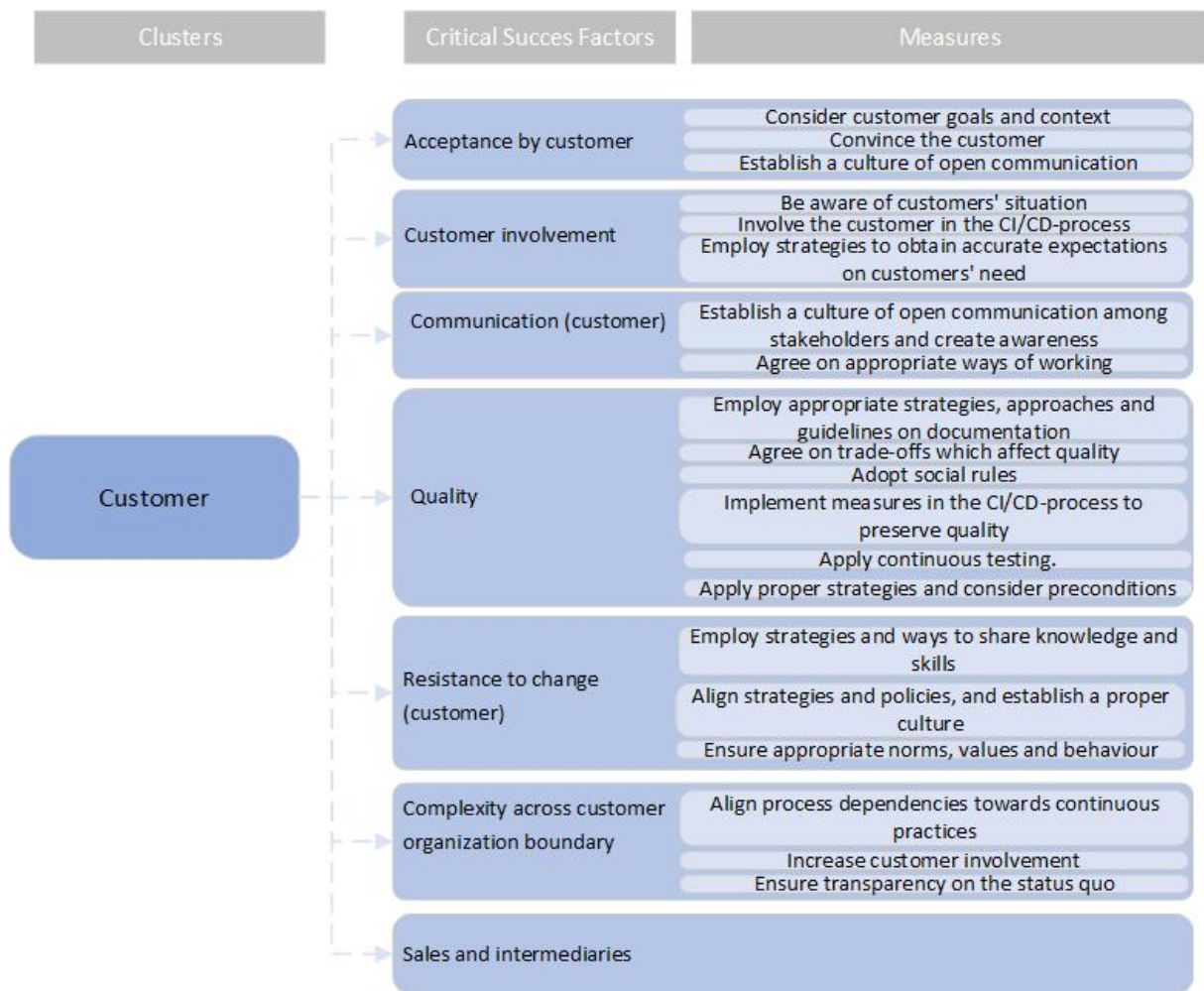
5		Kwaliteit
Beschrijving: Behoud van kwaliteit en adequate documentatie.		
Maatregel		Beschrijving
5A	Gebruik passende strategieën, benaderingen en richtsnoeren inzake documentatie te hanteren.	Gebruik feedbackmechanismen, wees u bewust van mogelijke belemmeringen en kruip zelfs in de huid van een klant.
5B	Overeenstemming bereiken over compromissen die de kwaliteit beïnvloeden.	Overweeg afwegingen inzake integratiefrequentie, veiligheid, zekerheid, omvang van de incrementen, kwaliteitspoorten en betrokkenheid van belanghebbenden.
5C	Neem sociale regels aan.	Samenwerken om aan de verwachtingen van de klant te voldoen, zoals feedback van klanten toepassen, reageren op waarschuwingen, kapotte builds en bugs, zorgen voor compatibiliteit en indien nodig terugdraaien. Denk na over het volledige systeem.
5D	Implementeer maatregelen in het CI/CD-proces om de kwaliteit te behouden.	Zorg voor snelle feedback en code reviews, beheer artefacten en systeemconfiguratie, integreer kwaliteitscontroles en fool proofing mechanismen.
5E	Pas continu testen toe.	Test automatisch onmiddellijk na een code commit, test nieuwe functies in het echte gebruik, betrek de klant bij het testen en beoordeel veranderingen tijdens het testen.
5F	Pas de juiste strategieën toe en houd rekening met de randvoorwaarden.	Pas continue strategieën toe op refactoring, verbetering, monitoring, meting, compliance, beveiliging, gebruik, innovatie, enz. Overweeg randvoorwaarden, zoals strategieën om technische schuld te verminderen, modularisering van ontwikkeling, betrouwbare testomgevingen. En pas op met het top-down opleggen van een metrisch gebaseerde evaluatie.

6		Weerstand tegen verandering
Beschrijving: Moeilijkheid om gevestigde organisatorische beleidslijnen en culturen te veranderen.		
Maatregel		Beschrijving
6A	Gebruik strategieën en manieren om kennis en vaardigheden te delen.	Ondersteun de verandering met strategieën, zoals meer planning hoe het werk te organiseren, lage leercurve, training, colocatie en het toevoegen van ervaring/coach aan het team. Pas manieren toe om kennis en vaardigheden te delen via communities, demonstraties, sjablonen enz.
6B	Strategieën en beleid op elkaar afstemmen en een juiste cultuur tot stand brengen.	Stem regels, voorschriften, beleidsmaatregelen en strategieën op elkaar af en creëer een cultuur van open communicatie.
6C	Zorgen voor passende normen, waarden en gedrag.	Zorg voor steun en leiderschap van het topmanagement op het gebied van voortdurende verbetering, budgettering en tooling. Geef het ontwikkelingsteam eigenaarschap en vertrouwen. Creëer bewustzijn, de juiste cultuur en mindset.

7		Verkoop en tussenpersonen
Beschrijving: Wanneer gebruikersgegevens niet toegankelijk zijn door tussenpersonen.		
Maatregel		Beschrijving

Appendix P

Conceptual Model Customer Edited



Appendix Q

Email request participation interview

Beste Collega,

Mijn naam is Nancy de Vries en ik ben binnen de ██████████ werkzaam als functioneel beheerder ██████ bij ██████████. Momenteel ben ik bezig met de opleiding Business Proces Management and IT aan de Open Universiteit en op dit moment ben ik bezig met mijn afstudeeropdracht.

Voor mijn afstudeeropdracht zijn wij bezig om op basis van bestaande theorie een framework te ontwikkelen om de maatregelen in kaart te brengen voor een succesvolle adoptie van Continuous Integration en Continuous Delivery. Hierbij kijken wij vooral naar de maatregelen die een bijdrage kunnen leveren aan de Kritische Succesfactoren in de relatie tot de 'klant'.

De volgende stap is nu om interviews af te nemen met mensen die in de praktijk al werken met Continuous Integration en Continuous Delivery om de maatregelen te valideren. Dit is voor jou meteen een mooi moment om terug te kijken op het implementatie/gebruiksproces en om aan te geven welke maatregelen effect hadden op het succes van het CI/CD-proces. Als je meewerkt aan het interview krijg dan stuur ik je het uiteindelijke rapport op wat je kan gebruiken om meer mensen te motiveren om de overstap naar Continuous Integration en Continuous Delivery toe te maken.

Volgens mijn informatie heb jij ervaring met Continuous Integration en Continuous Delivery en mijn vraag is dat ook of jij mee zou willen werken aan een interview om in gesprek te gaan over de maatregelen die wij hebben gevonden of deze overeenkomen met de praktijk. Het gesprek zal uitgevoerd worden via Webex, het zal ongeveer 60 minuten duren en de audio zal worden opgenomen. Hierbij zal je privacy wel worden gewaarborgd.

Mijn vraag is of jij dit zou willen? Zou jij met mij een moment willen inplannen zodat ik je kan interviewen aan de hand van het opgestelde Framework?

Met vriendelijke groet,

Nancy de Vries
Functioneel beheerder Inning

Appendix R

Confirmation of application of measure

Legend table

+	Confirmation	-	Denied	A	Agile
---	--------------	---	--------	---	-------

Confirmation of application of measure	#1 ICE	#2 EC	#3 ITA	#4 FBI	#5 PM	#6 PO
1 Acceptance by customer						
1A Consider customer goals and context	+	-	+	+	+	+
1B Convince the customer.	-	+	+/-	-	-	+
1C Establish a culture of open communication.	+	-	+/-	+		+
2 Communication						
2A Establish a culture of open communication among stakeholders and create awareness	+	+/-	+	+	+	+
2B Agree on appropriate ways of working.	+	-	+	+	+/-	
3 Complexity across customer organization boundary						
3A Align process dependencies towards continuous practices.			-	-	-	+
3B Increase customer involvement.			+		+	
3C Ensure transparency on the status quo	+	+/-A	+		+	+
4 Customer involvement						
4A Be aware of customers' situation.	+	+/-A	+	+	+	+
4B Involve the customer in the CI/CD-process.	+	+/-A	+	+	+	+
4C Employ strategies to obtain accurate expectations on customers' need.	-	+/-A	+		+	-
5 Quality						
5A Employ appropriate strategies, approaches and guidelines on documentation.	-	+		+	+	+
5B Agree on trade-offs which affect quality.	-	+	+	+	+	+
5C Adopt social rules.	+		+	+	+	
5D Implement measures in the CI/CD-process to preserve quality.	+		+	-	-	+
5E Apply continuous testing.	-	+	+	+/-	+	+
5F Apply proper strategies and consider preconditions.	+		+		+	
6 Resistance to change						
6A Employ strategies and ways to share knowledge and skills	+		+	+	+	+
6B Align strategies and policies, and establish a proper culture.			+	+	+	+
6C Ensure appropriate norms, values and behaviour.			+	-	+	+
7 Sales and intermediaries						

Appendix S

complete data including substantiation, examples and quotes

1 Acceptance by customer

M1A Consider customer goals and context

MA Confirmation

ID	Document	Quotation Content	Codes	Reference
8:9	#1 ICE	Ik denk wel dat ze daar verwachtingen over hebben	M1A Consider customer goals and context MA Confirmation	59 - 59
8:34	#1 ICE	Nou dat zagen ze zitten en dat gaan we samen uitwerken	M1A Consider customer goals and context MA Confirmation	116 - 116
10:4	#3 ITA	Wij houden ook rekening met de doelstellingen en de context van de klant	M1A Consider customer goals and context MA Confirmation	17 - 17
10:15	#3 ITA	En we verpakken daarin wel juist die doelstellingen en de context van de klant.	M1A Consider customer goals and context MA Confirmation	29 - 29
11:12	#4 FBI	Ja	M1A Consider customer goals and context MA Confirmation	46 - 46
12:5	#5 PM	Jazeker	M1A Consider customer goals and context MA Confirmation	11 - 11
13:5	#6 PO	Dus dat is een hele belangrijke doelstelling en manier waarop wij onze voorzieningen leveren.	M1A Consider customer goals and context MA Confirmation	19 - 19

MA Substantiation

ID	Document	Quotation Content	Codes	Reference
8:10	#1 ICE	Ze verwachten dat je dat beheerst en verantwoordelijk doet.	M1A Consider customer goals and context MA Substantiation	59 - 59
8:11	#1 ICE	Ja maar ook de manier waarop je de software maakt dat het gewoon een fatsoenlijk proces is. Hoe het precies in elkaar steekt dat wordt pas relevant voor hen als dat niet loopt en dat zij er last van hebben.	M1A Consider customer goals and context MA Substantiation	63 - 63
8:33	#1 ICE	En we hebben vooral geluisterd naar hun omstandigheden en werkprocessen, waarna we samen hebben gekeken: als we nou eens met dat stuk aan de slag gaan. Nou dat zagen ze zitten en dat gaan we samen uitwerken.	M1A Consider customer goals and context MA Substantiation	116 - 116
10:5	#3 ITA	wij hebben dit ook meegegeven als, 'wat jij wil zeg maar dan ben je ook heel erg geholpen met CI/CD' of met containerhosting. Dus daar houden we zeker rekening mee en dat konden wij ook doen omdat wij heel erg in gesprek waren met de klant.	M1A Consider customer goals and context MA Substantiation	17 - 17
12:6	#5 PM	Dus de vraag is of je rekening houdt met de doelstellingen en de context van de klant. Jazeker want dat is de eerste vraag namelijk die wij stellen, wat is voor jou de goede reden om Continuous Delivery in te gaan voeren, waarom wil je dat?	M1A Consider customer goals and context MA Substantiation	11 - 11

12:7	#5 PM	Dus je moet een hele goede reden hebben dat je dit wilt doen want je krijgt een heel hoop verantwoordelijkheid naar je toe geschoven en dat betekent dus ook dat je kennis en competenties daarvan moet ontwikkelen.	M1A Consider customer goals and context MA Substantiation	11 - 11
13:3	#6 PO	Dus dat is een hele belangrijke doelstelling en manier waarop wij onze voorzieningen leveren. De grote uitdaging daarin is de verplichtheid van de aspecten, vanuit compliance. Maar in principe is dat verder de teams, de gebruikers, onze klanten die zijn verantwoordelijk voor hun eigen pipeline, voor hun eigen voortbreng. Wij helpen ze daarbij met voorzieningen en met consultancy.	M1A Consider customer goals and context MA Substantiation	19 - 19

MA Example

ID	Document	Quotation Content	Codes	Reference
9:6	#2 EC	Dus er zit zeker een heel groot gat in kennis tussen, zowel vanuit de techneuten naar 'wat is nou eigenlijk de behoeften van de klant' als vanuit de klant naar 'ja maar beschrijf nou je behoefte en dan gaat de techneut wel zorgen dat het goed komt en ga je nou niet zorgen maken over hun stukje. Of doe je best om dat een beetje beter te begrijpen.	M1A Consider customer goals and context MA Example	31 - 31
11:2	#4 FBI	De applicatie wordt ontwikkeld in Openshift en dat is ook eigenlijk een heel nieuw platform wat voor beheer eigenlijk nog wennen is. Hebben wij wel heel korte lijntjes met iedereen 'goh we zien nu iets, wat is dat'. En soms is het dan iets en vaak is het niet erg maar weet je dat nog niet omdat je het allemaal nog niet kent.	M1A Consider customer goals and context MA Example	22 - 22
11:9	#4 FBI	Dat gaat straks vast gebeuren, we hebben nu alleen nog maar een pilot gehad en dan viel dat continue releasen wel tegen want dan loop je tegen de structuur van de overheidsorganisatie aan. Waarbij de releases toch gewoon aangevraagd moeten worden, er moeten wachtwoorden zijn en weet ik veel wat allemaal. Maar er wordt altijd met ons overlegd wanneer een release is, wanneer er gepland wordt of dit liever op een zaterdag, weekend of doordeweeks kan.	M1A Consider customer goals and context MA Example	38 - 38
11:10	#4 FBI	Ja, er worden ook kennissessies gepland voor die nieuwe technieken. Wij zijn zelfs opgegeven voor een opleiding in Openshift, dat vind ik wel heel belangrijk. Ik kom niet uit de technische wereld, ik heb er soms wat meer moeite mee en dan vind ik het wel prettig om een beetje bagage te hebben.	M1A Consider customer goals and context MA Example	42 - 42
13:4	#6 PO	Wat wij doen als Java-ontwikkelstraat is dat wij generieke modules maken waarmee de teams zelf hun eigen voortbrenging kunnen inrichten. Team a heeft stap A nodig, en een ander team heeft stap B en C nodig. En dat is ook wat wij bieden. Je kan zeggen wij bieden ondersteuning voor A, ondersteuning voor B en ondersteuning voor C, je kan dat in principe zelf in elkaar klikken omdat toe te passen	M1A Consider customer goals and context MA Example	19 - 19

NM Most Important Measure

ID	Document	Quotation Content	Codes	Reference
11:7	#4 FBI	Ja dan denk ik toch wel 1A	M1A Consider customer goals and context NM Most Important Measure	34 - 34

11:8	#4 FBI	Uiteindelijk bouwen ze voor ons natuurlijk en niet voor zichzelf. Zij kunnen wel iets heel leuks bedenken en heel open over communiceren en mij proberen te overtuigen, maar als dat totaal voorbij gaat aan de doelstellingen die wij als business hebben dan heb je er niets aan.	M1A Consider customer goals and context MA Substantiation NM Most Important Measure	34 - 34
------	--------	---	---	---------

M1B Convince the customer

MA Confirmation


ID	Document	Quotation Content	Codes	Reference
9:1	#2 EC	Ja dat gevoel heb ik wel dat de klant overtuigd wil worden	M1B Convince the customer MA Confirmation	27 - 27
10:1	#3 ITA	Daar hebben we de klant overtuigd	M1B Convince the customer MA Confirmation	17 - 17
13:1	#6 PO	Overtuigen, ja	M1B Convince the customer MA Confirmation	19 - 19

MA Substantiation

ID	Document	Quotation Content	Codes	Reference
8:1	#1 ICE	Ik denk dat die klant het helemaal niks interesseert op het moment dat wij kwalitatief hoogwaardige software op de afgesproken tijd leveren.	M1B Convince the customer MA Substantiation	37 - 37
8:2	#1 ICE	Mee eens. Dat interesseert de klant niet.	M1B Convince the customer MA Substantiation	39 - 39
8:3	#1 ICE	Dat is gewoon technische voortbrenging. De klant gaat er van uit dat het goed geregeld is.	M1B Convince the customer MA Substantiation	41 - 41
8:4	#1 ICE	N Oké, dus je hebt niet het gevoel dat je de klant moet overtuigen dat dit de betere manier is om te werken? ICE1 Ik denk het niet. Als je gewoon laat dat je op tijd en volgens afspraak de kwaliteit levert.	M1B Convince the customer MA Substantiation	43 - 45
8:5	#1 ICE	Ja, ons eindproduct is waar het de klant om gaat.	M1B Convince the customer MA Substantiation	49 - 49
8:6	#1 ICE	Ons dictaat is om het maandag op tijd te leveren, dat is de klant tevreden. Maar dat komt je misschien bekend voor. Onze klant wil werken met software en als de developers dat voortbrengen door overtypen dan maakt de klant dat niet uit.	M1B Convince the customer MA Substantiation	51 - 51
8:7	#1 ICE	Ik denk dat de klant niet echt een mening heeft, als je het over de eindgebruiker hebt, over hoe wij voortbrengen. Als het maar werkt. Het maakt hen ook niet uit hoe whatsapp wordt gemaakt, als het maar werkt.	M1B Convince the customer MA Substantiation	55 - 55
10:2	#3 ITA	Daar hebben we de klant overtuigd want die had echt wel behoefte aan verandervermogen en we hebben aan kunnen tonen, zelfs door middel van een business case, in de oude applicatie had je gewoon geen verandervermogen	M1B Convince the customer MA Substantiation	17 - 17
10:3	#3 ITA	En de klant CAP is zelf degene die zegt richting containerhostingstuurgroep van joh wij willen die risico's wel nemen, dus die zit echt ons straatje qua acceptatie enz.	M1B Convince the customer MA Substantiation	17 - 17
10:10	#3 ITA	Jaja, zeker. Maar dus op een conceptuele manier	M1B Convince the customer MA Substantiation	21 - 21

10:11	#3 ITA	Het niet zo zeer accepteren van CI/CD maar wel het accepteren van de software.	M1B Convince the customer MA Substantiation	21 - 21
10:13	#3 ITA	geef ook gewoon aan wat de voordelen zijn	M1B Convince the customer MA Substantiation	29 - 29
13:2	#6 PO	in principe komen wij, nou ja langzamerhand is dat niet zo nodig meer dat ze dat nodig hebben. Ze beginnen langzamerhand, maar nog niet iedereen hoor. Er zijn er nog steeds een heleboel die het allemaal wel goed vinden, dus er is nog steeds, en dat moet ook echt verschillende niveaus. Dus wij proberen met zowel de teams zelf (de ontwikkelaars) maar ook met productowner en ook met de ketenmanagers en ook met de architecten om tafel te zitten van dit is wat jij moet willen. Maar de essentie is, ik ben van oudsher heel erg overtuigen maar dat is meestal niet de beste stijl omdat te doen. Het is inzicht geven in en doordat andere teams dit doen en doordat bij andere teams hun voorbrenging beter wordt gaan anderen daar ook op mee	M1B Convince the customer MA Substantiation	19 - 19
13:6	#6 PO	Nou dat is eigenlijk wat intern een beetje op andere manier, teams hebben contact met andere teams en daar horen ze dat dingen beter lopen en dus willen ze het op een gegeven moment ook. Dan mag dat wat mij betreft binnen de overheidsorganisatie een heel stuk beter, de kennisuitwisseling, ervaringsuitwisseling, maar dat is weer een ander topic.	M1B Convince the customer MA Substantiation	23 - 23

MA Not applied

ID	Document	Quotation Content	Codes	Reference
10:8	#3 ITA	Maar we communiceren echt met de klant niet over CI/CD	M1B Convince the customer MA Not applied	17 - 17
11:6	#4 FBI	Dus ja daar hoef je me dan verder niet zo van te overtuigen, dat snap ik wel dat dat veel handiger is. 	M1B Convince the customer MA Not applied	26 - 26
12:1	#5 PM	Nou nee	M1B Convince the customer MA Not applied	15 - 15

MA Substantiation Not applied

ID	Document	Quotation Content	Codes	Reference
10:9	#3 ITA	Maar we communiceren echt met de klant niet over CI/CD, dat is denk ik nog wel steeds een stuk waar we wel inzicht geven maar dat is het een éénwega-communicatie naar hun toe om hun inzicht te geven maar niet zozeer dat zij zeggen 'oh maar doen moet je dit of dat doen. Dat niet want zij zijn heel erg gefocusseerd op hun eigen doelen.	M1B Convince the customer MA Not applied MA Substantiation	17 - 17
12:2	#5 PM	Nou nee ik ga een klant niet overtuigen als een klant niet wil. Kijk ik laat een aantal voor de hand liggende baten zien, die hebben wij natuurlijk al van klanten die daar mee bezig zijn opgehaald en die zijn ook getoetst natuurlijk met de buitenwereld, andere organisaties die ervaringen hebben met Continuous Delivery. Zie je eigenlijk, als je het wat oprolt naar boven, een zevental baten die je kunt realiseren met Continuous Delivery. Nou ik geef dat wat voorbeelden bij, dus ik geef daar wat	M1B Convince the customer MA Not applied MA Substantiation	15 - 15

		context bij en dan zie je dus dat eventuele applicatieteams of zelfs Business daarop aanslaan.		
--	--	--	--	--

MA Example

ID	Document	Quotation Content	Codes	Reference
9:2	#2 EC	Ja het beste voorbeeld hierbij vind ik bij mijn eigen werkgever, ik denk ook dat ik daar het meeste over ga vertellen, dat is het meest concreet. Binnen beide overheidsorganisaties is het gewoon dat een technische oplevering tegelijkertijd ook business functionaliteit gaat opleveren. En omdat dat aan elkaar gekoppeld is heb je het gesprek met de klant, met de acceptant van het opleveren. En wat ik gedaan heb ik discussies hier omtrent, is de vraag stellen 'wil je dat wel?'. Want dat is eigenlijk helemaal niet fijn want als er dan technisch iets mis gaat heb jij je functionaliteit niet. Dus ik heb gezocht naar wat nou de zorg is, wat de angst is, hoe ik deze kan wegnemen en welke behoeftes heb je en hoe kan ik die invullen terwijl ik technisch gezien releases blijf uitrollen naar productie zoveel als ik wil. Maar dat is echt een gesprek	M1B Convince the customer MA Example	23 - 23
9:3	#2 EC	Precies omdat ik die twee uit elkaar wil trekken. Ik zie gewoon niet dat een klant het fijn vindt dat zijn functionele releases gekoppeld zijn aan mijn technische releases. Hij heeft wel een behoefte, hij heeft wel een angst en ook 'Ja maar zo doen we het al tien jaar, dus waarom zouden we het nu anders doen'. Dus ook dat soort niet rationele overwegingen zijn er en ik wil graag niet de klant dwingen, dat is niet mijn stijl. Maar je moet wel degelijk je klant meenemen en overtuigen, 'nee maar het komt echt goed, dit is beter en de behoeftes die je hebt kunnen we nog steeds aan voldoen'.	M1B Convince the customer MA Example	27 - 27
11:5	#4 FBI	Nou ik snap wel waarom het belangrijk is waarom zij nu gaan ontwikkelen met web services en open shift en dat heb ik gedeeltelijk uit mijn opleiding gehaald en ook wel uit de uitleg die zij daar over geven.	M1B Convince the customer MA Example	26 - 26

NM Most Important Measure

ID	Document	Quotation Content	Codes	Reference
10:12	#3 ITA	Nou ja overtuig klant is de belangrijkste	M1B Convince the customer NM Most Important Measure	29 - 29

M1C Establish a culture of open communication

MA Confirmation

ID	Document	Quotation Content	Codes	Reference
8:12	#1 ICE	Het is een deel van het proces, denk ik	M1C Establish a culture of open communication MA Confirmation	67 - 67

8:14	#1 ICE	Daar worden ze echt heel gelukkig van, die open communicatie.	M1C Establish a culture of open communication MA Confirmation	67 - 67
10:14	#3 ITA	En ook dat je Agile werkt en dat je dus ook open bent in dat je dingen gaat uit zoeken	M1C Establish a culture of open communication MA Confirmation	29 - 29
11:3	#4 FBI	Dus ik vind wel dat er een hele lopen communicatie is.	M1C Establish a culture of open communication MA Confirmation	22 - 22
13:7	#6 PO	Ja	M1C Establish a culture of open communication MA Confirmation	23 - 23

MA Substantiation

ID	Document	Quotation Content	Codes	Reference
10:7	#3 ITA	Zijn wij specifiek aan het communiceren over CI/CD? Dat niet wij zijn vooral aan het praten over welk doel wij zouden willen bereiken. Dus we hebben wel een toolset en we kunnen ook uitleggen waarom we veel sneller kunnen ontwikkelen en waarom de kwaliteit verbeterd enz.	M1C Establish a culture of open communication MA Substantiation	17 - 17
10:16	#3 ITA	En die communicatie, die gaat nog komen. Maar dat gaat pas komen op het moment dus als je daadwerkelijk heel vaak dingen in productie gaat brengen, en zover zijn we nog niet. Maar dat gaat echt het komende half jaar gaat dat steeds verder tot uiting komen.	M1C Establish a culture of open communication MA Substantiation	29 - 29
13:8	#6 PO	De oplossingen die wij beiden moeten uiteindelijk tot standaard voorzieningen worden, waarvan iedereen weet hoe dat werkt en waarvan iedereen weet van daar kan ik bij. Ik vind dat alle testrapporten en beeldjobs in principe publiekelijk beschikbaar moeten zijn. Uiteindelijk vind ik ook dat alles wat wij hier doen dat dat eigenlijk een soort opensource cultuur zou moeten hebben. We zijn een overheidsbedrijf, we doen het voor het publiek, ik vind eigenlijk dat we al onze code publiek moeten maken, maar dat is weer een ander verhaal	M1C Establish a culture of open communication MA Substantiation	23 - 23

MA Not applied

ID	Document	Quotation Content	Codes	Reference
9:4	#2 EC	Moeilijk, het is vaak een technisch domein,	M1C Establish a culture of open communication MA Not applied	31 - 31
10:6	#3 ITA	Dat niet	M1C Establish a culture of open communication MA Not applied	17 - 17

MA Substantiation Not applied

-

MA Example

ID	Document	Quotation Content	Codes	Reference
8:13	#1 ICE	Als jij zeg maar continue met je klant kan overleggen over dingen die misschien nodig zijn of als jij als producent van software iets online zet ziet dat het kapot gaat ergens in een klein hoekje, als je daarover open met elkaar communiceert met die klant, je zegt we gaan het fixen en er komt een update aan zodra we het gefixt hebben, dat kan over 5 minuten zijn maar kan ook 2	M1C Establish a culture of open communication MA Example	67 - 67

		dagen zijn, en dan laat je dat weer horen als je dit gedaan hebt, dan denk ik dat je het in toren 1c al hebt afgetikt.		
9:5	#2 EC	het is vaak een technisch domein, een technisch verhaal tegen iemand die alleen maar functionaliteit wil en geen idee heeft wat daar eigenlijk voor nodig is of wat daar gebeurt.	M1C Establish a culture of open communication MA Example	31 - 31
11:1	#4 FBI	Wij zitten in het DevOps-team, daar zit bij de architect, de ontwikkelaars, de product-owner. Daarmee bespreken we al een heleboel dingen	M1C Establish a culture of open communication MA Example	22 - 22
11:4	#4 FBI	Je praat met de architecten en als je uitleg wil worden die sessies gepland, met het voorbereidingsteam, noem maar op	M1C Establish a culture of open communication MA Example	22 - 22

NM Most Important Measure

-

NM Agile

ID	Document	Quotation Content	Codes	Reference
12:3	#5 PM	Nou open communicatie zit meer bij het Agile werken	M1C Establish a culture of open communication MA Agile	19 - 19

ID	Document	Quotation Content	Codes	Reference
12:4	#5 PM	Ja het heeft er mee te maken maar open communicatie zit hem in de manier hoe je samenwerkt. Agile heb je nodig voor Continuous Delivery en andersom. Dus je kan zeggen dat dit een no-brainer is, je werkt multidisciplinair met multidisciplinaire teams. Als je DevOps dichter op elkaar brengt dan moet je een open cultuur en open communicatie hebben	M1C Establish a culture of open communication MA Agile MA Substantiation	19 - 19

M1 New Measure

ID	Document	Quotation Content	Codes	Reference
9:7	#2 EC	mogelijkheden bieden.	M1 New Measure EC MA New measure	35 - 35
9:8	#2 EC	Ik heb in gesprekken gehoord maar wij willen graag een campagne doen en de gebruikers iets vertellen over de nieuwe functionaliteit voor dat die functionaliteit beschikbaar komt.	M1 New Measure EC MA Substantiation	35 - 35
9:9	#2 EC	'Ik heb een wetswijzigingsdatum' Exact. En het wordt dan mooi als je kan zeggen, 'wil jij jouw wetswijzigingsdatum of wil jij jouw lanceringsdatum afhankelijk maken van technische implementatie, technische installatie. Nou dat klinkt niet heel goed. Want dan zegt hij 'nee liever niet'. 'Dat kunnen we voor je regelen'.	M1 New Measure EC MA Example	47 - 47

2 Communication

M2A Establish a culture of open communication among stakeholders and create awareness

MA Confirmation

ID	Document	Quotation Content	Codes	Reference
----	----------	-------------------	-------	-----------

9:23	#2 EC	Ja, maar dat is Agile. Dat is niet Continuous Delivery.	M2A Establish a culture of open communication among stakeholders and create awareness MA Agile MA Confirmation	126 - 126
------	-------	---	--	-----------

ID	Document	Quotation Content	Codes	Reference
8:15	#1 ICE	Zeker.	M2A Establish a culture of open communication among stakeholders and create awareness MA Confirmation	71 - 71
8:27	#1 ICE	Ja, 100%.	M2A Establish a culture of open communication among stakeholders and create awareness MA Confirmation	95 - 95
10:17	#3 ITA	Heel open zijn in je communicatie	M2A Establish a culture of open communication among stakeholders and create awareness MA Confirmation	35 - 35
11:14	#4 FBI	Ja kennissessies	M2A Establish a culture of open communication among stakeholders and create awareness MA Confirmation	50 - 50
11:19	#4 FBI	Ja	M2A Establish a culture of open communication among stakeholders and create awareness MA Confirmation	58 - 58
12:8	#5 PM	ja	M2A Establish a culture of open communication among stakeholders and create awareness MA Confirmation	27 - 27
13:9	#6 PO	we proberen open te communiceren.	M2A Establish a culture of open communication among stakeholders and create awareness MA Confirmation	27 - 27

MA Substantiation

ID	Document	Quotation Content	Codes	Reference
10:19	#3 ITA	Hoe doen we dat? Nou deze twee maatregelen zijn wel heel erg belangrijk daarin. Heel open zijn in je communicatie en het implementeren van DevOps, om zeg maar 2B overeenstemming van passende werkwijzen goed te krijgen.	M2A Establish a culture of open communication among stakeholders and create awareness M2B Agree on appropriate ways of working MA Substantiation	35 - 35

ID	Document	Quotation Content	Codes	Reference
9:24	#2 EC	Wat je wil doen is een korte feedbackloop hebben op dat wat je ontwikkeld. Of dat ok naar productie gaat of niet, dat is een andere vraag. Wat je probeert aan te	M2A Establish a culture of open communication among stakeholders	126 - 126

		tonen en te onderzoeken, is of de betrokkenheid van de klant bijdraagt aan het succes. Ik denk dat je daarvan moet zeggen dat dit indirect is. Want de snelle feedbackloop die je in een Agile manier van ontwikkelen hebt die trek je door tot en met dat je het aan de klant kan leveren. Maar dat hoeft dus niet dat leveren. Je maakt dus wat je levert aan de klant en de terugkoppeling daarop, die ring maak je nog groter want tot en met productie kan iemand zeggen dat hij het niet mooi vindt. 'Nou zeg het maar, over één uur heb je een andere kleur'. Dus dat helpt wel, maar het is meer het doortrekken wat je in Agile al doet, wat je nu doortrekt tot in je productieomgeving.	and create awareness MA Agile MA Substantiation	
--	--	--	---	--

ID	Document	Quotation Content	Codes	Reference
12:10	#5 PM	Dus het zit hem in name in van wat betekent het nou als je met CI/CD aan de slag gaat.	M2A Establish a culture of open communication among stakeholders and create awareness MA Confirmation MA Substantiation	27 - 27

ID	Document	Quotation Content	Codes	Reference
8:17	#1 ICE	En tegelijkertijd wil je mensen ook kunnen aankijken op het moment dat je een gesprek hebt.	M2A Establish a culture of open communication among stakeholders and create awareness MA Substantiation	71 - 71
8:19	#1 ICE	Wij laten die klant aan het woord over wat ze nodig hebben. En dan krijg je vervolgens wel de afweging van wat past er op korte termijn en wat moet op langere termijn en waar moet beter over nagedacht worden. Maar we proberen daar de lijntjes wel zo kort mogelijk te houden.	M2A Establish a culture of open communication among stakeholders and create awareness MA Substantiation	76 - 76
8:28	#1 ICE	Als je open communicatie hebt en je maakt van je hart geen moordkuil dan kom je er altijd uit.	M2A Establish a culture of open communication among stakeholders and create awareness MA Substantiation	95 - 95
8:29	#1 ICE	Maar wij zijn niet echt een standaard team hoor, dus het hoeft niet helemaal representatief te zijn. Wij zijn met zijn tweeën, als je een iets groter team hebt en je hebt 1 of 2 mensen van de business erbij en een paar anderen dan wordt de club alweer te groot en dan krijg je daar weer de complexiteit, dan wordt het iets ingewikkelder. Maar in onze omstandigheden werkt het als een tierelier	M2A Establish a culture of open communication among stakeholders and create awareness MA Substantiation	95 - 95
11:20	#4 FBI	Ja, want wij hebben dan Confluence en Jira daar staat dat precies in beschreven, helemaal uitgewerkt. Dus dat is allemaal gewoon toegankelijk en als je uitleg wil hebben dan kun je dat krijgen.	M2A Establish a culture of open communication among stakeholders and create awareness MA Substantiation	66 - 66
13:10	#6 PO	Ik probeer met onze activiteiten om dat zoveel mogelijk open neer te zetten, dat lukt lang niet altijd. Dat is ook moeilijk soms.	M2A Establish a culture of open communication among stakeholders and create awareness MA Substantiation	27 - 27

MA Not applied

-

MA Substantiation Not applied

ID	Document	Quotation Content	Codes	Reference
9:10	#2 EC	Continuous delivery werkt het prettigst in een DevOps omgeving	M2A Establish a culture of open communication among stakeholders and create awareness M2B Agree on appropriate ways of working MA DevOps MA Not applied	67 - 67

ID	Document	Quotation Content	Codes	Reference
9:11	#2 EC	Ik denk dat die communicatie altijd moeilijk is en hoe beter dat loopt hoe beter de hele voortbrenging is, niet per se Continuous Delivery. Misschien wel iets meer kans op Continuous Delivery. Ik vind hem niet zo heel sterk eigenlijk als voorwaarde.	M2A Establish a culture of open communication among stakeholders and create awareness M2B Agree on appropriate ways of working MA Not applied MA Substantiation	67 - 67

MA Example

ID	Document	Quotation Content	Codes	Reference
8:16	#1 ICE	Kijk in ons geval is als je software maakt en je hebt een eindgebruiker dan ga je tussentijds laten zien waar je staat. En dat moet gewoon een fatsoenlijk verhaal zijn met beeld daarbij. En in deze situatie van thuiswerken allemaal, dus je moet je scherm kunnen delen. En regie op je scherm kunnen delen dus zodat iemand anders iets kan laten zien	M2A Establish a culture of open communication among stakeholders and create awareness MA Example	71 - 71
8:18	#1 ICE	Dus bij ons in de praktijk hebben we Webex aan voor het beeld zodat we elkaar kunnen zien en we delen onze virtuele desktop via de Horizon Client. Maar daar kunnen we maar maximaal met 5 mensen zitten dus dat is beperkt en dan zijn er nog wel mogelijkheden met jabber, maar dat is niet heel praktisch allemaal maar dat zijn wel de dingen die je wel wil als je met elkaar in gesprek bent over het eindproduct voor die klant.	M2A Establish a culture of open communication among stakeholders and create awareness MA Example	71 - 71
8:20	#1 ICE	Dus als zij roepen ja dit willen we graag anders want dit is heel belangrijk dat we dat dan zeg maar dat het er bij wijze van spreken een uur later het er in zou zitten. De hele laag ertussen die zit er bij ons niet in, zeg maar. Het is de techneut praat met de klant.	M2A Establish a culture of open communication among stakeholders and create awareness MA Example	76 - 76
8:23	#1 ICE	In dat gesprek ja, het is niet zenden zeg maar. Het is wel ook luisteren, we vertellen wat we plan zijn of wat we gedaan hebben en dat kost 5 of 10 minuten en daarna onderbreken zij ons ook continue. Het is niet zeg maar dat we een sprintdemo geven en dat iedereen gezapig achterover leunt en ik denk ik ga weer naar de koffie. Het is wel heel interactief	M2A Establish a culture of open communication among stakeholders and create awareness MA Example	86 - 86
8:24	#1 ICE	En dat komt ook omdat we een vrij compact groepje hebben en met onze klant zijn het meestal 4 mensen en 1 van die 4 is de teamlead daar en drie ervaren mensen uit het team en wij met zijn tweeën. Weet je, er zit ook vrijwel niks bij en als ik zelf kijk naar webexmeetings, als de groep groter is dan een man of 5/6 dan haak ik al vaak af. Dan trek ik mijn mond niet zo snel meer open omdat er dan ook anderen zijn. Als het klein is dan kan	M2A Establish a culture of open communication among stakeholders and create awareness MA Example	88 - 88

		je iedereen meer tot zijn recht laten komen denk ik. En wordt dus dan ook daar het product beter van.		
8:25	#1 ICE	Plus zo'n teamlead die trekt ook zeg maar afhankelijk van de situatie en hoe het aan hun kant is omdat we dat niet allemaal kunnen zien, ja selectief eens iemand anders erbij, uit een andere hoek van hun afdeling. Dus als we het alleen over OB hebben dan zitten we met die mensen terwijl het op dat moment misschien ook interessant is voor het IH stukje dan halen ze iemand uit de uitvoering van de IH erbij. Maar die krijgt dat het uitgebreide verhaal van ons en dan ja die haakt dan niet bij die front aan maar pas als dat van belang wordt. En dan zijn ze wel op de hoogte. En wij ook over hun wensen en dat kunnen we dan in ons achterhoofd meenemen.	M2A Establish a culture of open communication among stakeholders and create awareness MA Example	90 - 90
11:13	#4 FBI	Ja kennissessies, maar ook voorlichting sessies dat wij een bepaalde service doornemen van wat doet die, wat zijn de foutmeldingen. Dus ja daar wordt echt wel veel aandacht aan besteed. En ook bijna continue zit er een architect bij, dus dat is wel heel prettig.	M2A Establish a culture of open communication among stakeholders and create awareness MA Example	50 - 50
11:18	#4 FBI	Ja. Ik heb van de week gevraagd voor welke foutmeldingen ik de architect 's nachts voor uit bed mocht bellen. Dat is maar een grapje maar uiteindelijke ga je daar wel naar toe natuurlijk. Dat je bij bepaalde fouten, dat je in de weekenden stand-by moet staan.	M2A Establish a culture of open communication among stakeholders and create awareness MA Example	58 - 58
13:12	#6 PO	We zitten met name met het ontwikkelteam. Het team van de Java-ontwikkelstraat die bijvoorbeeld de workshops doet, dus die zitten eigenlijk voornamelijk met het ontwikkelteam. Ik als Productowner, onze architect en wat andere mensen, soms het team ook die zitten met name met de andere mensen er om heen.	M2A Establish a culture of open communication among stakeholders and create awareness MA Example	31 - 31

NM Most Important Measure

ID	Document	Quotation Content	Codes	Reference
10:21	#3 ITA	Ja als je meer met de business gaat praten zal 2A belangrijker	M2A Establish a culture of open communication among stakeholders and create awareness MA Substantiation NM Most Important Measure	43 - 43

ID	Document	Quotation Content	Codes	Reference
8:26	#1 ICE	Die open communicatie vind ik.	M2A Establish a culture of open communication among stakeholders and create awareness NM Most Important Measure	94 - 94
8:45	#1 ICE	Ja, 100%.	M2A Establish a culture of open communication among stakeholders and create awareness	95 - 95

			NM Most Important Measure	
--	--	--	---------------------------	--

M2B B Agree on appropriate ways of working

MA Confirmation

ID	Document	Quotation Content	Codes	Reference
8:21	#1 ICE	Dat is tweewekelijks	M2B Agree on appropriate ways of working MA Confirmation	80 - 80
10:18	#3 ITA	en het implementeren van DevOps	M2B Agree on appropriate ways of working MA Confirmation	35 - 35
11:16	#4 FBI	Ja	M2B Agree on appropriate ways of working MA Confirmation	54 - 54
12:12	#5 PM	Kijk bij Deployment wel	M2B Agree on appropriate ways of working MA Confirmation	31 - 31

MA Substantiation

ID	Document	Quotation Content	Codes	Reference
10:19	#3 ITA	Hoe doen we dat? Nou deze twee maatregelen zijn wel heel erg belangrijk daarin. Heel open zijn in je communicatie en het implementeren van DevOps, om zeg maar 2B overeenstemming van passende werkwijzen goed te krijgen.	M2A Establish a culture of open communication among stakeholders and create awareness M2B Agree on appropriate ways of working MA Substantiation	35 - 35

ID	Document	Quotation Content	Codes	Reference
12:14	#5 PM	Dus de impact zeg maar voor de klant is eigenlijk bij Continuous Delivery nul, want die afspraak zal niet veranderen. Wat wel gaat veranderen is op het moment dat Continuous Delivery invoert dan zie je dus dat Ops activiteiten naar het applicatieteam gaan. Nou is het zo dat bijvoorbeeld functioneel beheer een rol heeft in het incidentenproces, of in een probleemproces. Nou daar bij die Ops kant zie je dus zodra je na Continuous Delivery dingen in productie staan dan gaan wel dingen veranderen. Maar dat is DevOps en niet Continuous Delivery. Dus an sich staat de klant daar los van maar het gevolg van Continuous Delivery is wel dat (doordat de Ops geactiveerd wordt) daar wel een verandering zit.	M2B Agree on appropriate ways of working MA DevOps MA Substantiation	31 - 31

ID	Document	Quotation Content	Codes	Reference
8:30	#1 ICE	De klant die is niet het formulier met de vakjes die moeten worden ingevuld. Maar de klant is die gebruiker die goed werkende software wil. En wat er veel mis gaat, denk ik, in communicatie naar de klant is dat je belooft iets maar vervolgens hou je je aan de	M2B Agree on appropriate ways of working MA Substantiation	96 - 96

		afspraken van een afvinklijstje in plaats van wat je tegen die klant hebt gezegd.		
10:20	#3 ITA	Kijk in de traditionele indeling binnen de organisatie is zit Functioneel beheer bij de business en waar wij zelf mee bezig zijn is om FB veel meer in ons team te betrekken. En ik vind dat wel een, en daar heb je het dan vooral over 2B, overeenstemmen passen de werkwijzen, want dat moet één goed geoliede machine worden. Want als wij sneller release willen doen dan moeten zij eigenlijk al weten waar het over gaat. En niet pas als wij aankomen met 'hey kun je dit accepteren'. Dus we zijn heel erg bezig om hun steeds meer in dat proces te betrekken.	M2B Agree on appropriate ways of working MA Substantiation	35 - 35
12:13	#5 PM	daar moet je vooraf dingen ingeregeld hebben want Continuous Deployment wil zeggen dat je zonder tussenkomst van goedkeuring door een klant bijvoorbeeld naar productie gaat. Continuous Delivery is er altijd een goedkeuring nodig van de belanghebbende of in ieder geval degene die gemandateerd is om de goedkeuring te geven voor in productie gaan. En dat in dit geval de klant zijn. En dat processtukje zal niet zomaar veranderen. Als dat een procesafpraak is blijft dat altijd bij Continuous Delivery gehouden.	M2B Agree on appropriate ways of working MA Substantiation	31 - 31
13:11	#6 PO	Wat wij doen, wij geven workshops, dat is wat wij de delivery pipeline versie twee workshops noemen, wat dus impliceert dat er ook een versie één was en dat er straks een versie drie en vier wellicht gaan komen. Initieel gaan we met team om de tafel zitten om samen met hun, nou eigenlijk ondersteunen wij een team zodat zij een eigen pipeline inrichten en eigen CI/CD proces programmeren. Het team is eigenaar van hun eigen voortbrenging, dat is de essentie daarin. Dat betekent dat er wat kennis en vaardigheden bij moet komen en dat door een initiële workshop ze aan kunnen sluiten op de enabler van pipelineversie twee geven we ze het eerste stapje mee om dat te doen en daarna moeten ze het in principe zelf doen en dan kunnen ze altijd consultancy bij ons komen ophalen	M2B Agree on appropriate ways of working MA Substantiation	27 - 27
13:15	#6 PO	Ja en ik vind de essentie, hoe kunnen we onze ontwikkelaars maximaal enablen. Ik vind eigenlijk dat in de missie/visie statement van de ICT-organisatie wel duizend keer de ontwikkelaar langs komen. Maar ga je dat er op na slaan dan hebben we het over allerlei dingen maar niet over de ontwikkelaar. En ik denk dat daar eigenlijk in essentie de oorzaak ligt van waarom we zijn waar we nu zitten. We moeten echt de ontwikkelaar enablen en de ontwikkelaar moet centraal staan en die moet niet gehinderd worden en dat soort zaken	M2B Agree on appropriate ways of working MA Substantiation	44 - 44

MA Not applied

ID	Document	Quotation Content	Codes	Reference
9:10	#2 EC	Continuous delivery werkt het prettigst in een DevOps omgeving	M2A Establish a culture of open communication among stakeholders and create awareness M2B Agree on appropriate ways of working	67 - 67

			MA DevOps MA Not applied	
--	--	--	-----------------------------	--

ID	Document	Quotation Content	Codes	Reference
12:11	#5 PM	Voor Continuous Delivery is de klant an sich niet nodig.	M2B Agree on appropriate ways of working MA Not applied	31 - 31

MA Substantiation Not applied

ID	Document	Quotation Content	Codes	Reference
9:11	#2 EC	Ik denk dat die communicatie altijd moeilijk is en hoe beter dat loopt hoe beter de hele voortbrenging is, niet per se Continuous Delivery. Misschien wel iets meer kans op Continuous Delivery. Ik vind hem niet zo heel sterk eigenlijk als voorwaarde.	M2A Establish a culture of open communication among stakeholders and create awareness M2B Agree on appropriate ways of working MA Not applied MA Substantiation	67 - 67

MA Example

ID	Document	Quotation Content	Codes	Reference
8:22	#1 ICE	Dat is tweewekelijks zeg maar dat we gewoon standaard staan en dat is een half uurtje en dat kan dan soms 10 minuten zijn of 5 minuten en soms is dat een half uur of langer, net wat er besproken wordt.	M2B Agree on appropriate ways of working MA Example	80 - 80
11:15	#4 FBI	Ja, dat zie ik meer onder 2B vallen en daar hebben we ook echt afspraken in van wat komt er in het DevOps-team aan de orde en wat voor afspraken maken we voor dingen buiten het DevOps-team	M2B Agree on appropriate ways of working MA Example	54 - 54
13:13	#6 PO	We hebben ook geen overkoepelend bedrijfsvisie hierop, geen afdeling die zegt die zegt dat ze vinden dat er in het algemeen onze teams wel redelijk autonoom moeten zijn en van alles zelf kunnen beslissen en daarin gaan we ze maximaal enablen en daarom vinden we dat ze het voortbrengingsproces op een bepaalde manier vorm moet krijgen. Daar wordt wel aan gewerkt maar dat duurt heel lang.	M2B Agree on appropriate ways of working MA Example	35 - 35
13:14	#6 PO	Dus aan de ene kant wil je een generieke bedrijfsbrede oplossing hebben maar ja daar hebben we organisatorisch helemaal niets voor. Dat is nog steeds een enorme omissie wat mij betreft, het is me nog steeds niet gelukt om dat in de afgelopen drie jaar te veranderen. Er veranderen wel dingen dus dat zijn hele lange ademdingen dus daar zitten ik en de Architect van de Java-ontwikkelstraat wel met verschillende mensen om de tafel, het nieuwe management of de reorganisatie zou hierin iets kunnen betekenen. Maar voordat die allemaal gereorganiseerd zijn ben je ook weer verder. Dus we zouden daar dolgraag tegen aan willen bemoeien maar die mogelijkheid hebben we lang niet altijd.	M2B Agree on appropriate ways of working MA Example	40 - 40

NM Most Important Measure

ID	Document	Quotation Content	Codes	Reference
----	----------	-------------------	-------	-----------

10:22	#3 ITA	en met functioneel beheer heb je meer gemeenschappelijke processen, daar heb je ook het releaseproces wat daar speelt. Dus dan is 2B weer misschien belangrijker.	M2B Agree on appropriate ways of working MA Substantiation NM Most Important Measure	43 - 43
-------	--------	---	--	---------

ID	Document	Quotation Content	Codes	Reference
13:63	#6 PO	Nou die gemeenschappelijke benadering, dat is voor mij echt essentieel. Die eilandjes moeten samengevoegd worden op één of andere manier. Dat hoeft niet fysiek te zijn maar. Ik vind het al lastig om te alignen met alle verschillende partijen. Dat is echt een fundamentele	M2B Agree on appropriate ways of working NM Most Important Measure	92 - 92

NM New Measure

ID	Document	Quotation Content	Codes	Reference
9:12	#2 EC	Want met DevOps ga je operatie en development in één team stoppen. En daarmee wordt communicatie echt anders en teamdynamiek en teamsturing wordt dan ook echt anders.	M2 New Measure EC DevOps MA DevOps MA New measure	73 - 73

3 Complexity across customer organization boundary

M3A Increase customer involvement

MA Confirmation

ID	Document	Quotation Content	Codes	Reference
13:18	#6 PO	Maar wat daar staat is wel een duidelijke en met name die procesafhankelijkheden afstemmen op continue praktijken. Je moet niet automatiseren, je moet niet alleen het huidige proces as-ist nemen, dat is vanuit het verleden vanuit het niet geautomatiseerde en een bepaalde technologische inrichting gekomen en wellicht heel logisch voor die situatie maar je moet altijd ook ver de hele keten heen kijken. Dat is mijn benadering en dat lijkt heel lastig te zijn. De mensen van Deployment kijken naar Deployment, de mensen van GH kijken naar incidentmanagement, de mensen van ontwikkeling kijken naar het ontwikkelproces.	M3A Align process dependencies towards continuous practices MA Confirmation	48 - 48

MA Context

ID	Document	Quotation Content	Codes	Reference
10:27	#3 ITA	Ik word een beetje getriggerd door 'wat het moeilijk maakt om een implementatieproces volledig te automatiseren'. En dat is wel een mooie want waar wij tegen aan lopen is dat wij binnen de organisatie een releaseproces hebben van eens per week en eventueel een noodrelease waarbij je nog 26 releases kan doen maar je moet allerlei mensen op de hoogte brengen dat jij gaat releasen. Dus jij kan heel makkelijk je software tot acceptatie geheel geautomatiseerd voortbrengen en dan kom je dus bij een hobbel en moet je het hele releaseproces in waarbij je bij alle partijen een vinkje moet halen. En dan moet je daar weer op wachten. Daar heb je een hobbel te nemen voor het een grote succesfactor gaat worden in je CI/CD. Dus Continuous Integration gaat heel makkelijk, Continuous Deployment is iets waar je dus allerlei	M3A Align process dependencies towards continuous practices MA Context	62 - 62

		afhankelijkheden moet zien te managen, zowel in het proces, en daar zijn we ook mee bezig en dat is ook best wel belangrijke maategel om daar verder in te gaan. Maar ook die afhankelijkheden die je hebt met andere partijen en je er snel genoeg kan deployen zonder dat je op iets anders hoeft te wachten.		
--	--	---	--	--

MA Substantiation

ID	Document	Quotation Content	Codes	Reference
10:23	#3 ITA	Je hebt ook altijd met afhankelijkheden te maken met andere teams, en of ze nu CI/CD toepassen dat maakt niet zoveel uit. Als jij heel snel kan wil dat nog niet zeggen dat je heel snel kunt, soms moet je gewoon echt wachten tot een andere partij, maar die worden hier niet genoemd.	M3A Align process dependencies towards continuous practices MA Substantiation	58 - 58
10:26	#3 ITA	Die afhankelijkheden moet je minimaliseren en dan heb je ook veel meer vrijheid om CI/CD voor jezelf in te richten	M3A Align process dependencies towards continuous practices MA Substantiation	62 - 62

MA Not applied

ID	Document	Quotation Content	Codes	Reference
10:24	#3 ITA	Nee het is meer dat je de afhankelijkheden die je hebt minimaliseert	M3A Align process dependencies towards continuous practices MA Not applied	62 - 62
11:21	#4 FBI	Daar zijn wij niet mee bezig, dat wordt vast gedaan maar op een ander niveau,	M3A Align process dependencies towards continuous practices MA Not applied	74 - 74
12:15	#5 PM	Nee	M3A Align process dependencies towards continuous practices MA Not applied	39 - 39
13:17	#6 PO	Je moet dus echt met teams gaan zitten om eigenlijk te zeggen 'nou wat zijn al jouw stapjes die er nodig zijn om van code naar productie te gaan'. Niet alleen die Deployment stap, niet allen die teststap, allemaal en dat ik wat wij proberen en wat we te weinig doen omdat we te weinig tijd hebben. Maar hopelijk als we dat een aantal keren gedaan hebben dat we dat makkelijker uit kunnen dragen.	M3A Align process dependencies towards continuous practices MA Not applied	48 - 48

MA Substantiation Not applied

ID	Document	Quotation Content	Codes	Reference
9:13	#2 EC	Ik denk dat Continuous Delivery over het leveren gaat van digitale diensten. En het maakt veel minder uit wat voor diensten dat zijn als dat je het stukje voortbrenging en een stukje datacenter hebt. Dus als je het hebt over over grenzen van organisaties heen dan is dat vooral die Development en Operations, daaroverheen ligt Continuous Delivery en Continuous Integration. De klant daarboven heeft daar veel minder mee te maken denk ik.	M3A Align process dependencies towards continuous practices MA Not applied MA Substantiation	82 - 82
10:25	#3 ITA	Nee het is meer dat je de afhankelijkheden die je hebt minimaliseert, maar dat is niet zo zeer op procesniveau maar meer dat je de producten die je oplevert weinig tot geen afhankelijkheden hebben met	M3A Align process dependencies towards continuous practices	62 - 62

			MA Not applied MA Substantiation	
12:16	#5 PM	Nee, bereik je helemaal niets mee. Je kan elk deel proces binnen een applicatie verbeteren. Dus je kan Continuous Delivery in een bepaalde mate zeker verbeteren. Maar de grootste stap zit hem erin dat je over die applicaties heen te kunnen gaan sturen. Dat is nu lastig, het systeem van applicaties werkt als een monoliet, je kunt niet één deeltje er uitpakken en daar iets aan gaan sleutelen. Om de integriteit te kunnen checken van het deelsysteem, de optelsom van die applicaties als één monoliet te samenwerken zul je altijd een integratietest moeten gaan doen om er zeker van te zijn dat de data niet beschadigd is. De manier van hoe het nu ingeregeld is, dat de uitkomst nog hetzelfde is. Dus het kan alleen maar als je de applicatiearchitectuur veranderd en de onderdelen ver-serviced en dat die Epi's hebben met contractafspraken van die Epi's (wat voor soort data, een soort stempel van hoe de data over het lijntje heen gaat). En dat is niet zo. A op de koppelvlakken is geen sprake van contracten, dus het koppelvlak ook zelf een versienummer heeft met een contract, daar zijn afspraken van. Laat staan dat de applicatie zelf ver-serviced is en dat je onafhankelijk, zonder dat het uitmaakt of de ander bestaat, als die het even niet doet dan kan die andere service mooi nog doordraaien, dat is niet de situatie. Als één applicatie niet werkt, dan heeft dat meteen gevolgen voor de andere applicaties waarmee zij gekoppeld zijn. Dus met het antwoord kan je kort zijn, je kan processen optimaliseren tot je een ons weegt maar dat is niet de marche.	M3A Align process dependencies towards continuous practices MA Not applied MA Substantiation	47 - 47

MA Example

ID	Document	Quotation Content	Codes	Reference
10:30	#3 ITA	Impliciet kun je teams gaan vormen, wat wij nu doen, maar uiteindelijk zul je ook je organisatie moeten aanpassen zodat het beter hierop aansluit, maar op dat moment moet dan iedereen hetzelfde gaan werken. Of gaat iedereen hetzelfde werken.	M3A Align process dependencies towards continuous practices MA Example	62 - 62
11:22	#4 FBI	En ik neem aan dat dat ook een stukje bij de architecten ligt, die afhankelijkheden afstemmen.	M3A Align process dependencies towards continuous practices MA Example	74 - 74
12:19	#5 PM	Dus stel dat een applicatie of programma veel storingen heeft en we kijken er naar hoe we dat in de toekomst kunnen voorkomen, dat is een gemeenschappelijk iets. Of als je ziet dat er in Europese samenwerkingen projecten draaien die tijdsafhankelijk zijn, dus is het zo dat nieuwe technologie en werkwijzen het mogelijk maken om sneller projectafspraken in te vullen. Dan zou je kunnen zeggen dan is het gewoon handig om bewust meteen te kiezen voor nieuwe infrastructuur, nieuwe methodologieën waar Continuous Delivery ook een aspect in is. En draagt het ook wel wezenlijk bij wat mij betreft.	M3A Align process dependencies towards continuous practices MA Example	55 - 55

NM Most Important Measure

ID	Document	Quotation Content	Codes	Reference
10:28	#3 ITA	Dus die 3A vind ik een hele belangrijke om daar dus in het proces in af te stemmen en dan heb je het voornamelijk over die releases	M3A Align process dependencies towards continuous practices NM Most Important Measure	62 - 62

NM New Measure

-

M3B Increase customer involvement

MA Confirmation

ID	Document	Quotation Content	Codes	Reference
10:29	#3 ITA	3B heeft meer betrekking op bijvoorbeeld functioneel beheer dat veel dichter tegen het team aanzit en waar wij eigenlijk op mikken is een DevOp-team, in de pilot die we draaien noem ik het een virtueel DevOp-team omdat het harkje dat er nu is niet zomaar omgeturnd is	M3B Increase customer involvement MA Confirmation	62 - 62
12:17	#5 PM	Nou ja zeker.	M3B Increase customer involvement MA Confirmation	55 - 55
12:27	#5 PM	Maar daarom moet je de klant wel erbij betrekken als je hiermee start om aan te geven dat daar geen consequenties zijn.	M3B Increase customer involvement MA Confirmation	67 - 67

MA Substantiation

ID	Document	Quotation Content	Codes	Reference
12:18	#5 PM	Uiteindelijk moet het doel hetzelfde zijn. Uiteindelijk vorm je samen een drijvend belang van de overheidsorganisatie in het grotere geheel. Kijk het begint natuurlijk aan de voorkant. Uiteindelijk creëer jij waarde voor de burgers/instanties die gebruik maken van de producten van de overheidsorganisatie	M3B Increase customer involvement MA Confirmation MA Substantiation	55 - 55
12:20	#5 PM	Dus ja het begint bij de waarde creatie die we willen bereiken met elkaar.	M3B Increase customer involvement MA Confirmation MA Substantiation	55 - 55
12:28	#5 PM	En dan moet je ze ook continue meenemen, je krijgt het vertrouwen niet zomaar	M3B Increase customer involvement MA Confirmation MA Substantiation	67 - 67

ID	Document	Quotation Content	Codes	Reference
8:32	#1 ICE	Nou ik vind die betrokkenheid van de klant vergroten, dat vind ik raar. Want als je iets maakt waar de klant op zit te wachten, dan is die betrokkenheid er sowieso al. Dus ik vraag me dan af, als je daar moeite voor moet doen maak je dan wel het goeie.	M3B Increase customer involvement MA Substantiation	116 - 116
9:15	#2 EC	Maar daar hoeft die klant niet echt wat van te vinden anders dan ik wil het sneller.	M3B Increase customer involvement MA Substantiation	90 - 90
11:29	#4 FBI	Ik denk dat dat verschil kleiner zou moeten worden in een DevOps-team en dat moet nog groeien.	M3B Increase customer involvement MA Substantiation	82 - 82

12:21	#5 PM	Dus ja helemaal terecht wat je zegt de klant is zeker betrokken, alle partijen moeten daarbij betrokken zijn en er achter gaan staan. Alleen je moet je kunnen koppelen aan het hogere doel. De applicatie-exploitanten die nu het beheer doen zeggen ook dat het harstikke ingewikkeld en dat ze niet weten of je dat zou moeten willen. Ja natuurlijk, als je dat aan hun vraag dan is dat belang ook helemaal anders. Het gaat om hun werk, waar ze trots op zijn. Dan ga ik vertellen, nou beste mensen over een jaar hebben jullie geen werk meer, nou dat slaat natuurlijk in als een bom.	M3B Increase customer involvement MA Substantiation	55 - 55
12:31	#5 PM	Dus je moet ze wel betrekken daarin alleen zit dat niet zo zeer in het 'actor' zijn daarin maar eerder een toeschouwer.	M3B Increase customer involvement MA Substantiation	67 - 67
13:19	#6 PO	Maar daar ligt de essentie en neem automatisering als uitgangspunt en niet als afgeleid product van 'we gaan het eerst handmatig doen'. Nee! Maar dat is met name een cultuur, een gewoonteding. En daar hebben we vaak veel te weinig invloed want ontwikkelteams, wij bepalen niet wat ze doen dat bepalen ze zelf met de productowner. Dus als zij hun processen niet willen automatiseren, we kunnen alleen maar laten zien hoe het wel zou kunnen werken.	M3B Increase customer involvement MA Substantiation	48 - 48
13:22	#6 PO	Ja. Ik denk dat dat ook onze eilandjescultuur.	M3B Increase customer involvement MA Substantiation	56 - 56

MA Agile

ID	Document	Quotation Content	Codes	Reference
9:29	#2 EC	Ja dat is ook zo, maar de vraag is dus zijn betrokkenheid van de klant en kwaliteit onderwerpen van Continuous Delivery en kun je die daar noemen of kun je beter zeggen dat een goed Agile proces een succesfactor is voor Continuous Delivery	M3B Increase customer involvement M4B Involve the customer in the CI/CD-process M5F Apply proper strategies and consider preconditions MA Agile	146 - 146

MA Substantiation Not applied

-

MA Example

ID	Document	Quotation Content	Codes	Reference
9:14	#2 EC	Kijk als ik naar de overheidsorganisatie kijk dan is daar bij Interactie een applicatie die voor de helpdeskmedewerkers is. Die medewerkers die deze software gebruiken hebben vrij weinig te maken met of dat Continuous Delivery is of niet. Die willen gewoon functionaliteit, dus als je dat definieert als de klant dan heeft die er vrij weinig mee te maken.	M3B Increase customer involvement MA Example	86 - 86
13:26	#6 PO	Het is belangrijk dat ontwikkelaars dat weten en dat zij geholpen worden dus je moet hun in hun ontwikkel en bouwproces rapporten daarover leveren, maar je moet ze ook zorgen dat dus management, productowner of wat dan ook, systemteams dat ze het overall plaatje een beetje zien. Niet om de teams er mee te slaan, nee de teams moeten zelf vinden dat ze daar van zijn. Als je alleen maar informatie aan management levert dan heeft het team zoiets van 'komen ze weer aan, ze weten geen donder waar het over gaat en er heeft in enen iemand een klepel	M3B Increase customer involvement MA Example	56 - 56

		gehoord over de onveilige libraries', maar die gebruiken we helemaal niet.		
--	--	--	--	--

NM Most Important Measure

-

NM New Measure

-

M3C Ensure transparency on the status quo

MA Confirmation

ID	Document	Quotation Content	Codes	Reference
8:35	#1 ICE	Je moet altijd transparant zijn omtrent de stand van een ontwikkelingstraject.	M3C Ensure transparency on the status quo MA Confirmation	124 - 124
10:32	#3 ITA	Ja	M3C Ensure transparency on the status quo MA Confirmation	62 - 62
12:30	#5 PM	Je moet wel transparant en open zijn over wat je doet zeg maar,	M3C Ensure transparency on the status quo MA Confirmation	67 - 67
13:23	#6 PO	Ja. Ik denk dat dat ook onze eilandjescultuur. Het lijkt mij al nauwelijks om alle partijen te vinden waar het over gaat. En dan word je van het kastje naar de muur gestuurd en moet je ergens iets escaleren maar hoeveel gewicht breng je in de schaal en dat soort zaken.	M3C Ensure transparency on the status quo MA Confirmation	56 - 56

MA Substantiation

ID	Document	Quotation Content	Codes	Reference
9:21	#2 EC	Het automatiseren van testen is wel heel belangrijk en een voorwaarde voor Continuous Delivery En daar zit zeker betrokkenheid van klanten en een stukje transparantie in. Laat je zien wat je test, laat je zien wat je niet test en dan kom je eigenlijk op risicomanagement want testen is eigenlijk voor risicomanagement. Wat is de kans dat er iets omvalt of stukgaat, en dat ga je dan testen. Zo gaan alle testen altijd over risicomanagement. Risicomanagement heeft weer te maken met een stukje vertrouwen. Hoe hard vertrouw ik dat wat er aan veranderingen zijn dat dat ten goede komt aan de uiteindelijke business die ik wil doen. Dat vertrouwen bouwen en die risicoafweging, daar moet je de klant wel bij betrekken.	M3C Ensure transparency on the status quo M5E Apply continuous testing MA Confirmation MA Substantiation	114 - 114

ID	Document	Quotation Content	Codes	Reference
12:29	#5 PM	Je moet wel transparant en open zijn over wat je doet zeg maar, dat het herkenbaar is dat er inderdaad niets verandert.	M3C Ensure transparency on the status quo MA Confirmation MA Substantiation	67 - 67

ID	Document	Quotation Content	Codes	Reference
----	----------	-------------------	-------	-----------

8:36	#1 ICE	Dat is een verplichting die je hebt, je werkt voor hun.	M3C Ensure transparency on the status quo MA Substantiation	124 - 124
8:37	#1 ICE	Ben je dat niet dan ben je gelijk de betrokkenheid heel snel kwijt.	M3C Ensure transparency on the status quo MA Substantiation	125 - 125
8:38	#1 ICE	Inderdaad en het vertrouwen.	M3C Ensure transparency on the status quo MA Substantiation	126 - 126
10:31	#3 ITA	Ja maar dat is iets dat we nu al wel doen, in de vorm van Agile werken en met Safe proberen we dat dus wel inzicht in te krijgen en daar helder in te zijn. Het heeft misschien niet geheel wat te maken met CI/CD, maar is wel iets dat nodig is.	M3C Ensure transparency on the status quo MA Substantiation	62 - 62
13:20	#6 PO	Ja dat staat al heel lang hoog op mijn lijstje om te zien met dashboard of wat dan ook inzicht te laten zien en waar men mee bezig is. Daar zijn wij te langzaam in om dat goed te faciliteren, dat hebben we te weinig. Maar ik denk dat dat essentieel is om dat goed te kunnen doen.	M3C Ensure transparency on the status quo MA Substantiation	48 - 48
13:21	#6 PO	Nou uiteindelijk zou dat het moeten zijn als jij een dashboard kan laten zien van wat is de inceptie tijd oftewel van code tot aan productie of waar dat gewoon in staat en als je dat ziet dat er z'n blok in staat voor je Deployment ten opzichte van drie kleine blokjes voor test en andere dingen dan kan je dat gewoon laten zien en dan kun jouw productowner kan daarmee aan de gang. Maar nu ligt het daartussen in en is het niet zichtbaar, niet transparant en dus blijft het allemaal maar een beetje van 'ja het duurt lang'. Ja hoelang dan, ja dat weet ik eigenlijk niet precies.	M3C Ensure transparency on the status quo MA Substantiation	52 - 52
13:25	#6 PO	En die informatie daarover transparant beschikbaar stellen. Je moet als team ook trots zijn op het feit, daarom vind ik ook dat wij als Overheidsorganisatie de code publiek inzichtelijk moet zijn. Moet je eens opletten hoeveel beter onze code dan wordt als jij als ontwikkelaar weet dat heel Nederland jouw code kan zien. Als we die transparantie hebben, ik denk dat enorm bijdraagt.	M3C Ensure transparency on the status quo MA Substantiation	56 - 56
13:37	#6 PO	Maar door dus transparant te zijn en zorgen dat de informatie die relevant is openbaar beschikbaar te stellen kan je dat wel optimaal faciliteren.	M3C Ensure transparency on the status quo MA Substantiation	68 - 68

MA Agile

ID	Document	Quotation Content	Codes	Reference
9:17	#2 EC	Wat ik mij dus afvraag is of je transparantie moet noemen of dat je het beter Agile en DevOps kan noemen, als Succes Factoren. Want die zijn veel groter, en ja een stukje daarvan is transparantie. Maar er zit ook een stukje in vertrouwen, er zit ook een stukje in fast feedback.	M3C Ensure transparency on the status quo MA Agile MA DevOps MA Substantiation	98 - 98

MA Substantiation Not applied

ID	Document	Quotation Content	Codes	Reference
----	----------	-------------------	-------	-----------

9:16	#2 EC	Continuous Delivery werkt het makkelijkst als het DevOps is en DevOps bestaat eigenlijk niet echt zonder Agile. En Agile bouwt voort op de mensen die er zijn. En mensen gaan goed op trust/vertrouwen en autonomie. Er zijn een soort basiswaardes waar je op verder bouwt. Kun je Continuous Delivery hebben zonder al deze dingen, ja dat kan wel. Maar de waarde, de transparantie waar we het nu over hebben die zit veel meer in het Agile bouwblokje. Je kan wel Continuous Delivery doen zonder, het helpt wel. Dus is het een Succesfactor? Ja dat denk ik wel, maar die hoort meer bij 'doe je Agile' en daarna doe je dan DevOps en dat helpt allemaal wel met Continuous Delivery, maar die waardes die daar allemaal bij horen, en één van de uitkomsten is dan dat Continuous Delivery beter gaat. Maar om hem direct te koppelen aan alleen Continuous Delivery vind ik dan lastig.	M3C Ensure transparency on the status quo MA Agile MA DevOps MA Not applied MA Substantiation	94 - 94
------	-------	--	---	---------

MA Example

ID	Document	Quotation Content	Codes	Reference
8:39	#1 ICE	Maar het is niet erg om te zeggen dat iets niet gelukt is of dat het heel lastig is, dat je het niet weet. Wat er aan de hand is zeg het ze gewoon, leg het ze uit en laat merken dat je je stinkende best aan het doen bent	M3C Ensure transparency on the status quo MA Example	126 - 126
8:40	#1 ICE	We hadden drie demo's achter elkaar eigenlijk waarbij we eigenlijk niks functioneels op hebben geleverd voor de klant omdat we er gewoon veel te veel shit om heen hadden, nou dat leg je dan gewoon uit wat voor gedoe er omheen is, wat we daarmee gedaan hebben en dat is dan misschien een beetje technisch maar ik vind dat we niet bang moeten zijn om dat aan die klant te vertellen. Plus, het zet hun ook aan het denken dat er meer is dan alleen dat scherm wat ze krijgen. En we leggen dan ook uit wat het buiten dat het voor hen ook voor de hele organisatie dat we dan met andere dingen even bezig zijn. Omdat we er zelf last van hebben maar dan pakken we het groter aan dan alleen ons eigen stukje	M3C Ensure transparency on the status quo MA Example	127 - 127
8:41	#1 ICE	We hadden drie demo's achter elkaar eigenlijk waarbij we eigenlijk niks functioneels op hebben geleverd voor de klant omdat we er gewoon veel te veel shit om heen hadden, nou dat leg je dan gewoon uit wat voor gedoe er omheen is, wat we daarmee gedaan hebben en dat is dan misschien een beetje technisch maar ik vind dat we niet bang moeten zijn om dat aan die klant te vertellen. Plus, het zet hun ook aan het denken dat er meer is dan alleen dat scherm wat ze krijgen. En we leggen dan ook uit wat het buiten dat het voor hen ook voor de hele organisatie dat we dan met andere dingen even bezig zijn. Omdat we er zelf last van hebben maar dan pakken we het groter aan dan alleen ons eigen stukje.	M3C Ensure transparency on the status quo MA Example	127 - 127
8:42	#1 ICE	En wat ook nog een positief effect is van het echt vertellen wat je doet, dus als je heel transparant bent, zij hebben recht om te weten wat er onder de motorkap gebeurt. Dat het ze dat soms te veel van hun bed staat of dat het heel technisch is dat komt dan vanzelf naar voren. Het is waarschijnlijk een technisch verhaal maar ik vind belangrijk dat je weet dat dit en dit aan de hand is en dat geeft dan ook echt vertrouwen. Dat schakelt allerlei drogredenen uit en rookgordijnen die wij ook wel eens teams zien opwerpen.	M3C Ensure transparency on the status quo MA Example	128 - 128

9:18	#2 EC	Ik neem aan dat je de filmpjes van de Spotify Cultuur ook al wel bekeken hebt, daar blijft ik telkens op terug komen omdat daar hele goede essentiële dingen in zitten. Want als een release moeilijk is dan ben je geneigd dat minder vaak te doen. Als de release makkelijker is dan doe je dat gewoon nog een keer. Dus als je komt in het gebruik, de headit van releasen is makkelijk, dan ben je geneigd dat vaker te doen, daarmee wordt releasen nog makkelijker en dan kom je gewoon in het gebruik van heel veel. Maar dat werkt wel binnen een agile context waarin je gebaseerd bent op vertrouwen.	M3C Ensure transparency on the status quo MA Example	98 - 98
13:24	#6 PO	Een ontwikkelaar wil ontwikkelen, en die moet je dus helpen en je moet dus een dashboard neer zetten waarin dus de informatie die hij nodig heeft.	M3C Ensure transparency on the status quo MA Example	56 - 56

NM Most Important Measure

-

NM New Measure

ID	Document	Quotation Content	Codes	Reference
8:43	#1 ICE	Ja soms heb je situaties waarvan je zegt, eigenlijk kan dat niet en als je die situatie hebt dan kan het ook niet en dan moet je een punt maken, dit kan niet. En dan moet je het ook gewoon niet doen. Want dan lijdt je pijn, dan heb je daar last van en je moet trouw blijven aan je eigen waarden daarin	M3D New Measure MA New measure MA New Measure ICE	132 - 132

ID	Document	Quotation Content	Codes	Reference
10:33	#3 ITA	En wat ik hieraan toe zou voegen is die afhankelijkheden minimaliseren met afnemende partijen. Meer op de producten die je oplevert. Als je van die afhankelijkheden te veel hebt zeg maar dan hoeft er maar één kink in de kabel en dat staat de hele boel stil. Dus er zit nog een afhankelijkheid in.	M3E New Measure ITA	62 - 62

4 Customer involvement

M1A Consider customer goals and context

MA Confirmation

ID	Document	Quotation Content	Codes	Reference
9:19	#2 EC	Ja dat hoort meer bij Agile denk ik. Ja ik vind dat niet specifiek voor Continuous Delivery.	M4A Be aware of customers' situation M4B Involve the customer in the CI/CD-process M4C Employ strategies to obtain accurate expectations on customers' need MA Agile MA Confirmation	102 - 102

ID	Document	Quotation Content	Codes	Reference
10:37	#3 ITA	Ja bij functioneel beheer wel	M4A Be aware of customers' situation M4B Involve the customer in the CI/CD-process	66 - 66

			M4C Employ strategies to obtain accurate expectations on customers' need MA Confirmation	
--	--	--	---	--

ID	Document	Quotation Content	Codes	Reference
12:22	#5 PM	Nou ja 4A en 4C die herken ik	M4A Be aware of customers' situation M4C Employ strategies to obtain accurate expectations on customers' need MA Confirmation	59 - 59

ID	Document	Quotation Content	Codes	Reference
8:51	#1 ICE	Ja	M4A Be aware of customers' situation MA Confirmation	151 - 151
8:52	#1 ICE	Ja dat denk ik ook ja	M4A Be aware of customers' situation MA Confirmation	152 - 152
10:43	#3 ITA	Dan moet 4 A sowieso, maakt niet uit wat, maar heeft niets te maken met CI/CD.	M4A Be aware of customers' situation MA Confirmation	77 - 77
10:47	#3 ITA	4A is altijd	M4A Be aware of customers' situation MA Confirmation	85 - 85
11:33	#4 FBI	Ze denken daar wel in mee ja.	M4A Be aware of customers' situation MA Confirmation	98 - 98
13:52	#6 PO	Even kijken, wees je bewust van de situatie van de klanten. Ja hetzelfde verhaal volgens mij. Dat is iets waar we momenteel heel hard mee bezig zijn.	M4A Be aware of customers' situation MA Confirmation	84 - 84

MA Substantiation

ID	Document	Quotation Content	Codes	Reference
8:53	#1 ICE	Wij zijn wel echt over de vloer geweest om te kijken hoe zij hun werk doen en wat voor drama, wat er goed of wat er niet goed gaat en wat er beter kan. Om naar het proces te kijken. En dat doe je dan met meerdere brillen op. En met name ook met een vernieuwende bril omdat wij in de vernieuwing zitten. En dan kijk je dus verder dan het proces dat geautomatiseerd moet worden, dan ga je ook kijken naar het werk dat zij gedaan willen krijgen en dan kijk je naar het proces dat daarbij zit en dan stel je vragen over hoe iets is ingericht en dan kan je kiezen om ook het proces samen met die klant aan te pakken.	M4A Be aware of customers' situation MA Substantiation	153 - 153

MA Not applied

-

MA Substantiation Not applied

-

MA Example

ID	Document	Quotation Content	Codes	Reference
----	----------	-------------------	-------	-----------

8:63	#1 ICE	Eigenlijk zou je hun werk moeten doen. En als je dat wil doen en je bent nog steeds gelukkig of je bent zelfs gelukkig geworden, dan ben je een goede softwareontwikkelaar. Is dat niet zo dan moet je terug naar de tekentafel en dan moet je het eerst aan die klant geven.	M4A Be aware of customers' situation MA Example	170 - 170
10:42	#3 ITA	Wij zien het vooral in de vorm van foutmeldingen. Kijk wij worden straks aangeroepen door een portaal bijvoorbeeld. En dan is de klant bij de burger die zijn informatie zelf aanlevert zonder dat daar een kantoormedewerker aan te pas komt. Een aantal van onze processen hebben nog wel een kantoorelement in zich, een soort handmatige controle voordat we een afspraak mogen opvoeren. Maar dat staat weer los van de automaat en de service die we aanbieden. Dus dan heb je het wel weer over afhankelijkheden die je hebt. De vervangende applicatie is hier geen goed voorbeeld in maar als je kijkt naar een klant die zich direct kan bemoeien met het proces/ met de schermen van een applicatie dan heb je hier wel heel erg mee te maken.	M4A Be aware of customers' situation MA Example	77 - 77
12:32	#5 PM	Nou kijk, je begint dus met een hele goede reden dat je dit zou willen, dat is eigenlijk gewoon een vraag, letterlijk. Dus je betreft ze bij de eerste sessie, de intakes ook dat er aan de klant gevraagd wordt welke voordelen er voor hen inzitten, of die er zijn	M4A Be aware of customers' situation MA Example	71 - 71
13:47	#6 PO	Aan de ene kant willen we heel faciliteren maar we willen ze ook een beetje opvoeden.	M4A Be aware of customers' situation MA Example	84 - 84

NM Most Important Measure

ID	Document	Quotation Content	Codes	Reference
11:37	#4 FBI	Het is leuk als ze zich bewust zijn maar als ze er niets mee doen dan heb ik er niets aan. En dan mogen zij mij betrekken maar dan heb ik er nog niets aan. Uiteindelijk moet ik ermee kunnen werken.	M4A Be aware of customers' situation MA Substantiation NM Most Important Measure	111 - 111

NM New Measure

-

M4B Involve the customer in the CI/CD-process

MA Confirmation

ID	Document	Quotation Content	Codes	Reference
9:19	#2 EC	Ja dat hoort meer bij Agile denk ik. Ja ik vind dat niet specifiek voor Continuous Delivery.	M4A Be aware of customers' situation M4B Involve the customer in the CI/CD-process M4C Employ strategies to obtain accurate expectations on customers' need MA Agile MA Confirmation	102 - 102

ID	Document	Quotation Content	Codes	Reference
----	----------	-------------------	-------	-----------

10:37	#3 ITA	Ja bij functioneel beheer wel	M4A Be aware of customers' situation M4B Involve the customer in the CI/CD-process M4C Employ strategies to obtain accurate expectations on customers' need MA Confirmation	66 - 66
-------	--------	-------------------------------	--	---------

ID	Document	Quotation Content	Codes	Reference
8:54	#1 ICE	Ja, die horen daar gewoon bij.	M4B Involve the customer in the CI/CD-process MA Confirmation	157 - 157
10:38	#3 ITA	Ik vind het wel lastig wanneer je opschrijft dat we een DevOps team hebben dan hebben we functioneel beheer niet als klant, dan zijn ze meer een onderdeel van het team. Zolang je dus een functioneel beheerachtige rol bij je business hebt dan heb je deze maatregelen nodig en ook zolang je BizDevOps nog niet hebt ingericht moet je tot op zekere hoogte moeite in stoppen. Zoals ik net als zei, betrek je je klant bij het CI/CD proces, Ja voor functioneel beheer. Ik denk dat als jij het CI/CD proces succesvol kan gebruiken om stabiele succesvolle opleveringen te doen dan heb je alleen maar een blijde klant. Dan hoeft je het niet bij het proces te betrekken maar dan zeg je dat het er over twee dagen staat, dan is hij ook al blij.	M4B Involve the customer in the CI/CD-process MA Confirmation	66 - 66
10:44	#3 ITA	Bij 4B kan je dat heel erg gebruiken in je testtraject of voorbrengingstraject door al heel snel te zeggen of iets wel of niet werkt	M4B Involve the customer in the CI/CD-process MA Confirmation	77 - 77
10:48	#3 ITA	4B kan als de klant redelijk dichtbij zit.	M4B Involve the customer in the CI/CD-process MA Confirmation	85 - 85
11:31	#4 FBI	Dat wordt allemaal met ons doorgenomen nu.	M4B Involve the customer in the CI/CD-process MA Confirmation	86 - 86
12:23	#5 PM	4B dat heb ik net uitgelegd.	M4B Involve the customer in the CI/CD-process MA Confirmation	59 - 59
12:26	#5 PM	Jaja, zeker. Daar verandert niets aan.	M4B Involve the customer in the CI/CD-process MA Confirmation	63 - 63
13:44	#6 PO	Nou ja dat 4B betrekken is dat wij feitelijk zeggen 'jullie proces is leidend, jullie moeten nadenken over je voortbrengingsproces', tegelijkertijd hebben we natuurlijk een standaard implementatie.	M4B Involve the customer in the CI/CD-process MA Confirmation	84 - 84

MA Substantiation

ID	Document	Quotation Content	Codes	Reference
8:46	#1 ICE	Nou ik denk dat het heel belangrijk is dat je met de juiste mensen om tafel zit. Als je met de verkeerde mensen om tafel zit dan krijg je ook niet het resultaat wat voor de partij van belang is.	M4B Involve the customer in the CI/CD-process MA Substantiation	141 - 141
8:47	#1 ICE	Ja dat is ook een veelgehoorde klacht van echte uitvoerende medewerkers, er wordt even een keer een meeting belegd maar daarna wordt er alleen maar met managers of teamleiders gepraat en niet	M4B Involve the customer in the CI/CD-process MA Substantiation	142 - 142

		met de echte personen die het echte werk moeten doen.		
12:24	#5 PM	Het betrekken van een klant is zeg maar indirect.	M4B Involve the customer in the CI/CD-process MA Substantiation	59 - 59
13:43	#6 PO	Als we zon workshop met teams doen hebben wij gewoon een standaard template waarvan we zeggen, eigenlijk doe je altijd dit en ziet een proces er zo uit, moeten we ook niet ingewikkelder maken dan het is. Veel dingen zijn gewoon redelijk recht toe, recht aan. Waar het lastig wordt is het feit dat we als ontwikkelaars afwijken van de standaarden en dan denken dat we een afwijkend proces hebben en dat gefaciliteerd willen hebben, Nou dan kan je je afvragen, moeten we dan het proces veranderen om dat te faciliteren of moeten we onze eigen organisatie veranderen. Nou in principe een beetje van beide	M4B Involve the customer in the CI/CD-process MA Substantiation	84 - 84
13:50	#6 PO	En ik zou veel meer feedback en input willen hebben van onze klantenkring, de gebruikerskring.	M4B Involve the customer in the CI/CD-process MA Substantiation	84 - 84
13:51	#6 PO	Maar daar wil ik dus ook onze dashboards voor hebben, ik wil nog wat dashboard hebben met de adoptie van onze producten en dat soort zaken zodat iedereen dat kan zien en dat we daarop kunnen sturen.	M4B Involve the customer in the CI/CD-process MA Substantiation	84 - 84

MA Not applied

-

MA Substantiation Not applied

-

MA Example

ID	Document	Quotation Content	Codes	Reference
10:39	#3 ITA	Ja maar dan heb je nog steeds je V-vorm, van hoog naar laag en weer terug. Je hebt ergens het idee, de impuls enz. en die wordt dan uitgewerkt, steeds gedetailleerd en gebouwd. Die gedetailleerde uitwerking ga je testen en uiteindelijk zeg je: 'Is jouw impuls nou geïmplementeerd?' Alleen dan wat kleiner denk ik dan. Dus als je het hier hebt over 'wees je bewust van de situatie van de klant' dan wil je met CI/CD eigenlijk inspelen op die klantwensen. Maar dat geldt niet alleen voor CI/CD.	M4B Involve the customer in the CI/CD-process MA Example	70 - 70
10:41	#3 ITA	Dus je moet een proces inregelen om impulsen binnen te krijgen alleen zullen dat steeds kleinere dingen zijn, je zal directer op de klant zitten. Ik vind het ook wel moeilijk om te vertalen naar de vervangende applicatie, dan heb je daar heel erg te maken met dat we vooral een basisadministratie zijn. Dus als je CI/CD op een andere applicatie los zou laten dan zou je veel meer met de klant gaan samenzitten op het moment dat jij gaat lopen sparren over hoe een scherm eruit moet zien. 'Kijk eens we hebben nu dit en is dit goed?'. Dan staat dit dezelfde middag nog in productie. En op dat moment kun je ook de klant gebruiken in je hele testproces door te zeggen van ik bouw gewoon wat, je laat het zien, er komt commentaar op, je past het aan. Met Continuous Integration en Continuous Delivery hoef je niet alles in productie te testen, zeker	M4B Involve the customer in the CI/CD-process MA Example	73 - 73

		dingen als schermen of processen, dan kun je ze gewoon meenemen tijdens het testtraject om de veranderingen te laten zien. En dan heb je inderdaad een grote betrokkenheid van de klant tijdens het CI/CD proces, dat maakt het wel gemakkelijker		
10:64	#3 ITA	Ook de klant moet heel erg goed doorhebben dat een DevOps team niet aan applicatieontwikkeling doet zoals we dat van vroeger kennen dat als de applicatie klaar is dan is het gedaan. Nee een DevOps-team moet vooral ook als geheel die ondersteuning blijvend bieden. Je moet dus een stabiel voortbrengingsteam voor het behoud van de applicatie behouden. Je hebt dat zelf ook gezien waarbij eerst met één ontwikkelteam is begonnen, toen weer uitgebreid is naar twee teams, daarna iedereen de deur uit gedaan omdat het geld op was, maar de applicatie niet klaar was. Toen toch maar weer eens een keer gaan opschalen waardoor niemand meer wist waar het over ging behalve misschien één bouwer, één ontwerper en één functioneel beheerder. Dat is ook wel iets als je CI/CD doet is dat ook wel een aspect waar je de klant van bewust moet maken is dat als je dit wil laten slagen dan is dit wel een randvoorwaarde. Dus misschien dat je dat nog ergens bij de acceptatie door de klant kan meenemen.	M4B Involve the customer in the CI/CD-process MA Example	117 - 117
11:30	#4 FBI	Ik heb bijvoorbeeld allemaal beheer features die heb ik ingediend, dat wordt door de architect opgepakt, die plant daar sessies voor met ons om door te spreken over wat we precies willen hebben, of we dat in een schermje willen of willen we query's draaien, hoe wil je je foutmeldingen zien, welke foutmeldingen, foutmeldingen waar je direct actie op moet nemen wil je die bij wijze van spreken in het rood hebben, ik noem maar iets. Dat wordt allemaal met ons doorgenomen nu.	M4B Involve the customer in the CI/CD-process MA Example	86 - 86
11:32	#4 FBI	Wij zijn in gesprek met ontwerp, architect, productowner en ook soms met de ontwikkelaars erbij, zeker met de refinementsessies zullen ze erbij zijn.	M4B Involve the customer in the CI/CD-process MA Example	94 - 94
12:25	#5 PM	Het is meer de ontzorging dat er in die zin niet meteen iets verandert met Continuous Delivery an sich zelf. Dus in dat proces van inrichten Continuous Delivery houden we eigenlijk de goedkeuringsstap, de vrijgave, die houden we in stand. Dus daar verandert an sich niets aan. Maar het gevolg van Continuous Delivery wat ik net zei, daar heeft de klant natuurlijk wel mee te maken.	M4B Involve the customer in the CI/CD-process MA Example	59 - 59
13:45	#6 PO	Je hebt teams die in het verleden de keuzen hebben gemaakt om al hun code in één code-repository te stoppen, als daar vijf componenten bij elkaar zitten met een verschillend release-cycle dan is dat een erg ingewikkeld om daar één proces op te bouwen. Nou dan kun je zeggen dat we dat toch gewoon doen of je gaat met elkaar zeggen om eens die code-repository uit elkaar te gaan trekken. Ja en dat heeft consequenties voor de ontwikkelaar.	M4B Involve the customer in the CI/CD-process MA Example	84 - 84

13:46	#6 PO	Dus dan zou je met elkaar dat gesprek aan moeten gaan.	M4B Involve the customer in the CI/CD-process MA Example	84 - 84
-------	-------	--	---	---------

NM Most Important Measure

-

NM New Measure

-

NM Agile

ID	Document	Quotation Content	Codes	Reference
9:29	#2 EC	Ja dat is ook zo, maar de vraag is dus zijn betrokkenheid van de klant en kwaliteit onderwerpen van Continuous Delivery en kun je die daar noemen of kun je beter zeggen dat een goed Agile proces een succesfactor is voor Continuous Delivery	M3B Increase customer involvement M4B Involve the customer in the CI/CD-process M5F Apply proper strategies and consider preconditions MA Agile	146 - 146

M4C Employ strategies to obtain accurate expectations on customers' need

MA Confirmation

ID	Document	Quotation Content	Codes	Reference
9:19	#2 EC	Ja dat hoort meer bij Agile denk ik. Ja ik vind dat niet specifiek voor Continuous Delivery.	M4A Be aware of customers' situation M4B Involve the customer in the CI/CD-process M4C Employ strategies to obtain accurate expectations on customers' need MA Agile MA Confirmation	102 - 102

ID	Document	Quotation Content	Codes	Reference
10:37	#3 ITA	Ja bij functioneel beheer wel	M4A Be aware of customers' situation M4B Involve the customer in the CI/CD-process M4C Employ strategies to obtain accurate expectations on customers' need MA Confirmation	66 - 66

ID	Document	Quotation Content	Codes	Reference
12:22	#5 PM	Nou ja 4A en 4C die herken ik	M4A Be aware of customers' situation M4C Employ strategies to obtain accurate expectations on customers' need MA Confirmation	59 - 59

ID	Document	Quotation Content	Codes	Reference
----	----------	-------------------	-------	-----------

12:34	#5 PM	Je bedoelt bij het realiseren van Continuous Delivery? Dat zit in de aanpak. Dus de voice of the customers zeg maar is echt het startpunt voor dat je überhaupt begint.	M4C Employ strategies to obtain accurate expectations on customers' need MA Confirmation MA Substantiation	75 - 75
-------	-------	---	--	---------

MA Substantiation

ID	Document	Quotation Content	Codes	Reference
12:34	#5 PM	Je bedoelt bij het realiseren van Continuous Delivery? Dat zit in de aanpak. Dus de voice of the customers zeg maar is echt het startpunt voor dat je überhaupt begint.	M4C Employ strategies to obtain accurate expectations on customers' need MA Confirmation MA Substantiation	75 - 75

ID	Document	Quotation Content	Codes	Reference
8:56	#1 ICE	dat komt nu uit dat overdracht moment, die momenten dat we met ze praten. Maar het staat hun ook vrij en dat herhalen we ook iedere keer, neem contact op als er iets is via mail of chat SEP: ICE1 En dat hebben ze ook een paar keer gedaan SEP: ICE2 Ja, klopt. Nou en dan kijken we gelijk even.	M4C Employ strategies to obtain accurate expectations on customers' need MA Substantiation	161 - 163
8:57	#1 ICE	En mocht het team groter worden zeg maar dan moet je dat wel meer gaan formaliseren maar dan wil je denk ik gewoon in de applicatie de mogelijkheid hebben om te zeggen dat iets niet goed gaat. In plaats van dat dit via de helpdesk op een stapel ergens terecht komt om drie weken later te horen dat iets niet opgelost wordt. Je wilt direct, want anders is dat die betrokkenheid, vertrouwen en transparantie die je kwijt raakt	M4C Employ strategies to obtain accurate expectations on customers' need MA Substantiation	163 - 163
8:59	#1 ICE	Dat hele stuk hoort ook bij CI/CD, het nazorgtraject. Maar dat moet super efficiënt en effectief zijn, als een klant, en vooral als ze zelf veel kunnen en zelf kunnen zien, ze gaan het eerst zelf proberen of vragen een collega doordat ze betrokken zijn bij de ontwikkeling. En dan is het ook meer van hunzelf. In plaats van dat jij zelf iets koopt in de winkel en het dan maar terugbrengt omdat het niet bevalt.	M4C Employ strategies to obtain accurate expectations on customers' need MA Substantiation	163 - 163
10:40	#3 ITA	Het is iets dat je sowieso moet implementeren want anders gaat geen enkel project goed.	M4C Employ strategies to obtain accurate expectations on customers' need MA Substantiation	73 - 73
10:45	#3 ITA	Ja op die manier, een beetje het 'vond u dit moeilijk-knopje'. Ik kan me er iets bij voorstellen, is heel situatieafhankelijk in hoeverre dit bruikbaar is. Eigenlijk, hoe verder de klant van je afstaat, en dan zie ik bijvoorbeeld de klant als burger die zijn rekeningnummer moet aanpassen, hoe belangrijker dit wordt.	M4C Employ strategies to obtain accurate expectations on customers' need MA Substantiation	81 - 81
10:46	#3 ITA	Dus 4C vind ik wel situatieafhankelijk.	M4C Employ strategies to obtain accurate expectations on customers' need MA Substantiation	85 - 85

13:49	#6 PO	We hadden in het verleden ook wat structuren om ook in gesprek te zijn met de ontwikkelaar, dat is een beetje minder geworden. We zijn nu heel druk bezig om onze stakeholder Management meer te verbeteren om op die manier dus ook voor ons, Continuous improvement, om dat ook echt toe te passen en ook andersom, hoe krijgen we onze gebruikers, onze klanten, onze stakeholders optimaal betrokken. Dus daar zijn wij heel druk mee bezig en ik had het liever gister dan vandaag af maar de praktijk is wat dat betreft wat weerbarstiger	M4C Employ strategies to obtain accurate expectations on customers' need MA Substantiation	84 - 84
-------	-------	--	---	---------

MA Not applied

ID	Document	Quotation Content	Codes	Reference
8:55	#1 ICE	Op dit moment is dat nog niet geformaliseerd zeg maar	M4C Employ strategies to obtain accurate expectations on customers' need MA Not applied	161 - 161
13:48	#6 PO	Nou en het gebruik de strategieën. Ja eigenlijk hebben we dat op dit moment niet goed genoeg meer.	M4C Employ strategies to obtain accurate expectations on customers' need MA Not applied	84 - 84

MA Substantiation Not applied

MA Example

ID	Document	Quotation Content	Codes	Reference
11:34	#4 FBI	Er zijn structureel wekelijks bepaalde overleggen ingepland. En wij hebben dan Mattermost, dat is eigenlijk een soort chat online waarin je in een bepaalde groep wordt ingedeeld, wij hebben dat voor de pilot waar we heel snel met elkaar kunnen communiceren en informatie kunnen uitwisselen.	M4C Employ strategies to obtain accurate expectations on customers' need MA Example	102 - 102
11:35	#4 FBI	Dat is wel de bedoeling om dat te bouwen, om dashboards voor ons te bouwen.	M4C Employ strategies to obtain accurate expectations on customers' need MA Example	107 - 107
12:33	#5 PM	En 4C gaat over de strategieën om nauwkeurige verwachtingen over de behoefte van klanten te verkrijgen. Die hebben we, we hebben een aanpak en een strategie hoe je dit moet gaan doen.	M4C Employ strategies to obtain accurate expectations on customers' need MA Example	71 - 71

NM Most Important Measure

ID	Document	Quotation Content	Codes	Reference
11:36	#4 FBI	Nou ja 4C denk ik.	M4C Employ strategies to obtain accurate expectations on customers' need NM Most Important Measure	111 - 111

M4D New Measure

ID	Document	Quotation Content	Codes	Reference
8:49	#1 ICE	Nou nee dat je met de juiste om tafel zit. Dat je echt met de juiste mensen praat. Ik denk dat dat een onderschatte factor is.	M4D New Measure MA New measure	146 - 146
8:50	#1 ICE	En dan kan je soms wel beter met alleen de echte gebruikers praten waar het teamlead niet bij is dan dat het teamlead er wel bij is. Gelukkig hebben wij	M4D New Measure MA Substantiation	147 - 147

		een team dat supergezellig is en dat elkaar ook vertrouwd maar er zijn ook teams waar de gebruiker de mond niet open doet zodra een teamlead erbij is. Want dat is levensgevaarlijk.		
8:60	#1 ICE	Nou ik denk dat als je niet met de juiste mensen om tafel zit dat het dan sowieso mis gaat.	M4D New Measure MD4 New Measure ICE NM Most Important Measure	167 - 167
8:61	#1 ICE	Ja idd, de eerste is die mensen.	M4D New Measure NM Most Important Measure	168 - 168
8:62	#1 ICE	Als het dan mis gaat dan heb je het grootste risico dat het überhaupt flopt. En dan moet je altijd helemaal verplaatsen in de klanten, op hun stoel mee gaan kijken om te beleven waarmee zij werken en wat zij nodig hebben.	M4D New Measure MA Substantiation	169 - 169

5 Quality

M5A Employ appropriate strategies, approaches and guidelines on documentation

MA Confirmation

ID	Document	Quotation Content	Codes	Reference
11:23	#4 FBI	A	M5A Employ appropriate strategies, approaches and guidelines on documentation MA Confirmation	115 - 115
11:38	#4 FBI	Ja dat gaan ze wel doen.	M5A Employ appropriate strategies, approaches and guidelines on documentation MA Confirmation	119 - 119
11:41	#4 FBI	Ja	M5A Employ appropriate strategies, approaches and guidelines on documentation MA Confirmation	125 - 125
12:35	#5 PM	Gebruik passende strategieën, benaderingen en richtlijnen inzake documentatie te hanteren.	M5A Employ appropriate strategies, approaches and guidelines on documentation MA Confirmation	81 - 81
13:31	#6 PO	Ja dat proberen wij ook te ondersteunen	M5A Employ appropriate strategies, approaches and guidelines on documentation MA Confirmation	64 - 64

ID	Document	Quotation Content	Codes	Reference
9:25	#2 EC	Niet anders dan bij traditioneel.	M5A Employ appropriate strategies, approaches and guidelines on documentation MA Confirmation MA Substantiation	130 - 130

MA Substantiation

ID	Document	Quotation Content	Codes	Reference
9:25	#2 EC	Niet anders dan bij traditioneel.	M5A Employ appropriate strategies, approaches and guidelines on documentation MA Confirmation MA Substantiation	130 - 130

ID	Document	Quotation Content	Codes	Reference
8:65	#1 ICE	We hebben nog wel gekeken voor als we eventueel gaan opschalen is een soort tour dat je in je software kan maken, een soort digitale rondleiding door je applicatie. Maar we weten nog niet of we dat gaan gebruiken, ik heb liever dat het zo intuïtief is dat mensen gewoon snappen wat de bedoeling ermee is.	M5A Employ appropriate strategies, approaches and guidelines on documentation MA Substantiation	174 - 174
8:66	#1 ICE	Ja idd, en voor de rest kan ik me wel voorstellen dat er wel documentatie komt voor higher level als het breder toegepast gaat worden, dat je meer het concept uitlegt. Maar ik stel mezelf altijd de vraag waar ik de handleiding van Whatsapp of Facebook kan vinden, iedereen gebruikt het maar niemand heeft de handleiding ooit gezien, maar ja dan is het goede software. Het scheelt ook heel veel werk tijdens je CI/CD proces anders moet je voor je kan releasen je documentatie in orde hebben. Je kan een hele kudde mensen aan het werk zetten, maar je kan ook de software goed maken.	M5A Employ appropriate strategies, approaches and guidelines on documentation MA Substantiation	175 - 175
8:86	#1 ICE	dat die betrokkenheid/ die kwaliteit bereik je ook door zelf te gebruiken. Het probleem is natuurlijk, vaak is het een team dat software voortbrengt dat niet de software ook gebruikt in onze organisatie, maar dan moet je jezelf wel de vraag stellen zou ik de software willen gebruiken gedurende twee maanden. Wil ik op die stoel zitten of loop ik gillend weg. En ik vind dat dat een goede maatstaf zou zijn als je daar eerlijk naar kijkt naar de kwaliteit van je software.	M5A Employ appropriate strategies, approaches and guidelines on documentation MA Substantiation	233 - 233
9:26	#2 EC	De documentatie daarbij is niet meer of minder belangrijk denk ik.	M5A Employ appropriate strategies, approaches and guidelines on documentation MA Substantiation	134 - 134
9:27	#2 EC	Wat de verschillen zijn tussen traditionele voortbrenging en Continuous Delivery is automatisering. En dat is zowel automatisering van de testen als de automatisering van de voortbrenging. Hoe ga ik van de code naar een uitgerold en geïnstalleerd systeem dat door de gebruiker gebruikt kan worden. Wat de succesfactor van Continuous Delivery van die twee dingen automatiseren is dat testen veel strikter herhaalbaar zijn en dat die voortbrenging ook veel strikter herhaalbaar is. Dat levert een veel stabielere kwaliteit op dan wanneer je iets hebt gedaan dat daarna naar een tester gaat die met zijn Excel sheet en een mindere dag heeft vandaag dan gister, dan test die gewoon anders. Dat stukje menselijke dynamiek/variatie die haal je uit het hele proces. En uit het test proces en uit het voortbrengingsproces. Dus daarmee verhoog je de kwaliteit en dat is denk ik zeker een succesfactor.	M5A Employ appropriate strategies, approaches and guidelines on documentation MA Substantiation	134 - 134
10:61	#3 ITA	Wat je hier wel een beetje mee zit, documentatie is wel belangrijk inderdaad. Niet dat je een heel beheerhandboek steeds opnieuw moet uitbrengen. Maar dat dat inderdaad volgt uit die documentatie uit de code, maar dat vergt van klanten ook wel weer een aanpassing waar ze het moeten vinden en hoe ze het moeten lezen. Dus dat is best een uitdaging. Wat hier ook al een uitdaging in is en misschien een extra maatregel is	M5A Employ appropriate strategies, approaches and guidelines on documentation MA Substantiation	101 - 101

		dat als jij aanpassingen doet in je proces of in je schermen dan is het binnen onze organisatie al snel dat er een heel opleidingstraject op moeten worden gestart zodat alle medewerkers ermee overweg kunnen. En dat is wel iets dat CI/CD in de weg staat		
11:39	#4 FBI	En sommige dingen komen ook met voortschrijdend inzicht als je nu een andere applicatie monitort op Openshift dan zie je wel wat je ook nog moet hebben 'omdat we het nu onvoldoende kunnen zien'.	M5A Employ appropriate strategies, approaches and guidelines on documentation MA Substantiation	119 - 119
11:40	#4 FBI	Ja, die wordt heel goed, heel uitvoerig bijgehouden.	M5A Employ appropriate strategies, approaches and guidelines on documentation MA Substantiation	125 - 125
11:42	#4 FBI	Nee zijn geen handleidingen maar wat ze doen. In Confluence en Jira allemaal, daar staan alle usecases beschreven, de services staan beschreven.	M5A Employ appropriate strategies, approaches and guidelines on documentation MA Substantiation	129 - 129
12:36	#5 PM	Wij gebruiken een boek van Continuous Delivery van Jez Humble en David Farley, dat zijn twee bekende Amerikanen die dat hebben geschreven die heel veel er over nagedacht hebben en de factor standaards hebben bepaald voor Continuous Delivery. Dat boek gebruiken wij als naslagwerk en het geven wij ook aan team want dat is namelijk de taal die wij spreken als we het hebben over Continuous Delivery, daar staan begrippen in en voorbeelden in van Continuous Delivery die we hanteren. En daarbij komen we eigenlijk op het niveau dat we dezelfde taal spreken.	M5A Employ appropriate strategies, approaches and guidelines on documentation MA Substantiation	81 - 81
13:32	#6 PO	De aardigheid is datgene wat wij aan klanten leveren moeten wij ook zelf doorleven en dat zijn we niet altijd omdat we ook vanuit een jarenlange praktijk, niet iedereen is daar even volwassen mee opgegroeid dus dat is altijd heel interessant. Nee maar belangrijk, hoe eerder we klanten een informatie kunnen geven over de kwaliteit van de applicatie hoe eerder ze ook kunnen reageren	M5A Employ appropriate strategies, approaches and guidelines on documentation MA Substantiation	64 - 64

MA Not applied

ID	Document	Quotation Content	Codes	Reference
8:64	#1 ICE	Oh maar wij documenteren voor de klant eigenlijk niks, het moet vanzelf spreken	M5A Employ appropriate strategies, approaches and guidelines on documentation MA Not applied	174 - 174

MA Substantiation Not applied

-

MA Example

ID	Document	Quotation Content	Codes	Reference
10:62	#3 ITA	Dus je gaat iets aanpassen in je proces, een wat grotere aanpassing in je scherm hebt en misschien zelf het veranderen van de naam op een knopje van 'Submit' naar 'Opsturen'. Das zeggen sommige afdelingen dat zij eerst de documentatie aan moeten passen voor er geleased kan worden. Dat is wel iets dat botst met IC/CD, en dat is inderdaad een uitdaging om dat goed te krijgen. En daar heb ik geen antwoord op	M5A Employ appropriate strategies, approaches and guidelines on documentation MA Example	101 - 101
10:63	#3 ITA	Behalve dan dat je heel bewust, zeker als je een kantoorachtige applicatie hebt of een portaal dat je documentatie in je scherm meeneemt. Van die i-tjes er neer zetten en waar ze op kunnen klikken en extra informatie kunnen krijgen en dat je daar ook die feedback op wil hebben zodat je wanneer iets niet duidelijk is dat snel aan kan passen. Die mechanismen zijn er wel maar dat is wel een hele lastige. Wat je misschien mee kan nemen, je hebt van die UX Designers (user Experience Designer), die sowieso al met je meegaan tijdens het CI/CD traject om te kijken of het allemaal wel zo werkbaar is zoals jij misschien bedacht hebt. Dus dat is ook wel een strategie, een feedbackmechanisme, die je daarin kan toepassen. Maar het is wel een heel hardnekkig ding en daar moet je dus idd, een beetje afhankelijk van je situatie, moet je dus denken aan die UX Designers of gerichte documentatie in je schermen om te voorkomen dat dit je gaat ophouden.	M5A Employ appropriate strategies, approaches and guidelines on documentation MA Example	105 - 105
12:37	#5 PM	Dus als we het hebben over passende strategie is dat boek een handig uitgangspunt, als kennis, voor gemeenschappelijke taal, voor gemeenschappelijke begrippenkader. Dat gebruiken wij uiteraard.	M5A Employ appropriate strategies, approaches and guidelines on documentation MA Example	81 - 81
12:38	#5 PM	We hebben ook een solution architectuur die eigenlijk een soort richting aangeeft, die zeg maar echt de kadering aangeeft als we het hebben over invoering van Continuous Delivery, dat we allemaal hetzelfde doen als we hiermee aan de slag gaan. Dus dat zijn eigenlijk de strategieën die we daarin gebruiken	M5A Employ appropriate strategies, approaches and guidelines on documentation MA Example	81 - 81
13:33	#6 PO	Dat is ook een verschil tussen de v1 pipeline en de v2 pipeline, v2 zit in principe op de hele code, op alle branches waar de teams mee bezig zijn zodat ze al vroegtijdig op de hoogte gebracht worden van wat ze aan het doen zijn. Betekend wel weer dat je verschillende aspecten of verschillende momenten wil kunnen adresseren. Een feature branche ga je niet helemaal tot aan acceptatie brengen, een feature branche wil je waarschijnlijk ook niet statische codeanalyse op doen. Want dan krijgt de gebruiker 'hey hij is waarschijnlijk in die feature in die code wat aan het reorganiseren' en dan wordt die continue om de oren geslagen met allerlei code issues. Dus daarin proberen we onze eigen ervaring te stoppen samen met wat klanten willen hebben en waar ze tegen aanlopen. Dus wat dat betreft, ja.	M5A Employ appropriate strategies, approaches and guidelines on documentation MA Example	64 - 64

NM Most Important Measure

-

NM New Measure

-

M5B Agree on trade-offs which affect quality

MA Confirmation

ID	Document	Quotation Content	Codes	Reference
10:60	#3 ITA	Dus 5F en 5B ga je doen bij een bepaald volwassenheidsniveau van je CI/CD.	M5B Agree on trade-offs which affect quality M5F Apply proper strategies and consider preconditions MA Confirmation	97 - 97

ID	Document	Quotation Content	Codes	Reference
11:24	#4 FBI	ik denk B ook wel	M5B Agree on trade-offs which affect quality MA Confirmation	115 - 115
12:39	#5 PM	Overeenstemming bereiken over compromissen die de kwaliteit beïnvloeden.	M5B Agree on trade-offs which affect quality MA Confirmation	81 - 81
13:35	#6 PO	uiteindelijk is het team er verantwoordelijk voor. Die is verantwoordelijk voor hun kwaliteit, voor hun security en al dat soort zaken.	M5B Agree on trade-offs which affect quality MA Confirmation	68 - 68

ID	Document	Quotation Content	Codes	Reference
9:28	#2 EC	Er zit daar wel een interactie met de klant, wat is voldoende kwaliteit en hoe tonen we die voldoende kwaliteit aan. ^[P] _[SEP]	M5B Agree on trade-offs which affect quality MA Confirmation MA Substantiation	138 - 138
10:56	#3 ITA	Hierin kun je wel keuzes maken	M5B Agree on trade-offs which affect quality MA Confirmation MA Substantiation	93 - 93

MA Substantiation

ID	Document	Quotation Content	Codes	Reference
9:28	#2 EC	Er zit daar wel een interactie met de klant, wat is voldoende kwaliteit en hoe tonen we die voldoende kwaliteit aan. ^[P] _[SEP]	M5B Agree on trade-offs which affect quality MA Confirmation MA Substantiation	138 - 138
10:56	#3 ITA	Hierin kun je wel keuzes maken	M5B Agree on trade-offs which affect quality MA Confirmation MA Substantiation	93 - 93

ID	Document	Quotation Content	Codes	Reference
10:58	#3 ITA	En hoe beter jij CI/CD hebt ingericht en hoe meer vertrouwen de klant daar in heeft hoe eerder je dit kan doen. Je kan altijd kiezen om terug te draaien. Dus dat is niet het probleem. Maar liever heb je het gewoon gelijk opgelost en wel gewoon je nieuwe functionaliteit. Maar dan moet je dus heel snel kunnen schakelen als ware.	M5B Agree on trade-offs which affect quality MA Substantiation	93 - 93

12:40	#5 PM	Ja weet je, wij kennen de maturity model. Dus er zijn een aantal stadia waar een team inzit en het is aan het team zelf om daar te bepalen hoe snel zij qua maturity vooruit willen dan wel dat ze ergens blijven hangen. Dat is voor ons geen discussie omdat je zelf je snelheid bepaalt. Wat wij doen is dat we de structuur uitleggen en zorgen dat ze op kennisniveau zitten we maken een lijst met acties die in hun backlog terecht moet komen en dan is het aan het team zelf. Want dat is namelijk het mooie van Continuous Delivery en DevOps, je bent zelfsturend. Je bent autonoom. Ik ga niet zeggen waar jij volgend jaar moet staan, dat bepaal jij zelf want je had ook een doelstelling weet je nog die je met de klant hebt afgesproken notabene. Dus ja, ik ga niet zeggen je moet de komende drie sprints moet je allemaal Continuous Delivery artifacts gaan produceren en realiseren	M5B Agree on trade-offs which affect quality MA Substantiation	81 - 81
12:50	#5 PM	Een van de voorwaarden is juist flexibel te kunnen zijn. Ja dan wordt er wel samen met het team gekeken hoe kun je flexibel zijn en waarom ben je nu niet flexibel. Waar ligt dat dan aan. Ja en dat zou kunnen zijn dat je nog te grote brokken hebt die je niet in één sprint naar een artifact kan brengen, dus naar code kan brengen of naar een configuratiefile of weet ik veel wat. Dus als dat zo is dan kunnen we kijken of je dat niet in kleinere stukjes kan hakken. Maar nogmaals dat voldoe je eigenlijk niet aan de randvoorwaarde van Continuous Delivery of eigenlijk van Agile werken zou je kunnen zeggen.	M5B Agree on trade-offs which affect quality MA Substantiation	89 - 89
13:34	#6 PO	En we hebben een hele goede relatie met security maar in het verleden, dat is tegen ons ook, zeggen ze 'ja jullie zijn van je security dus vertel ons maar waarmee wij jullie kunnen helpen'. Dus ik zeg 'Nee, draai het om, jullie weten wat security is, vertel ons waar wij aan moeten voldoen. En dan kunnen wij daar rekening mee houden.	M5B Agree on trade-offs which affect quality MA Substantiation	68 - 68
13:41	#6 PO	Maar je moet daar met zijn allen een bewuste keuze in maken. Datgene wat voor de ene een hoge kwaliteit is dat voor de ander niet	M5B Agree on trade-offs which affect quality MA Substantiation	80 - 80

MA Not applied

ID	Document	Quotation Content	Codes	Reference
8:68	#1 ICE	Ja weet je, wij zijn ook in deze niet een standaard team want wij hebben niet een lange termijnplanning en target wanneer iets beloofd is dat iets in de lucht moet zijn of iets anders dat onbeschikbaar maakt. Dat helpt in deze situatie wel	M5B Agree on trade-offs which affect quality MA Not applied	179 - 179

MA Substantiation Not applied

ID	Document	Quotation Content	Codes	Reference
13:36	#6 PO	Tegelijkertijd is dat ook voor grote bedrijven en eilandjes een excuus om er dan maar niet van te hoeven zijn.	M5B Agree on trade-offs which affect quality	68 - 68

			MA Not applied MA Substantiation	
--	--	--	-------------------------------------	--

MA Example

ID	Document	Quotation Content	Codes	Reference
10:57	#3 ITA	en één van die keuzes die je kan maken, wat je met de klant kan afspreken is dat je iets toepast dat wij noemen FailForward. Als wij een fout tegenkomen bij een release dan gaan wij die fout oplossen en opnieuw releasen. Dat is een andere keuze dan terugdraaien. Maar dat is het een keuze die je samen met de klant moet maken.	M5B Agree on trade-offs which affect quality MA Example	93 - 93
13:42	#6 PO	Daarom is een tool als Sonarqube zo lastig, het is aan de ene kant heel gaaf maar het zijn wel generieke regels. En wat nou als jouw applicatie, als jij met zijn allen zegt die regel vinden wij niet belangrijk, vinden wij als bedrijf zijnde dat wel? Vinden we dat we daar allemaal aan moeten voldoen? Waarom vinden we dat dan? Er staan 400 regels in. Ik vind dat ook een hele lastige want zoals ik al zei, die 400 regels, wat vind ik ervan als bedrijf zijnde, maar als afdeling gaan we daar ook helemaal niet over. We hebben daar wel een mening over maar we gaan daar niet over. Wij leveren de faciliteiten zodat teams en management, architecten kunnen vaststellen/bepalen wat zij willen doen en nastreven. Maar daarin moet je dus iedereen enables en trainen en opvoeden en dat soort zaken. Dan moet je het vakmanschap goed op orde hebben. En dat is ook best een lastige. In een bedrijf waar je senior kan worden door anciënniteit in plaats van door vakmanschap.	M5B Agree on trade-offs which affect quality MA Example	80 - 80

NM Most Important Measure

-

NM New Measure

-

MA Context

ID	Document	Quotation Content	Codes	Reference
8:67	#1 ICE	als de druk hoger is dan is behoefte aan afspraken en het halen van die afspraken groter denk ik	M5B Agree on trade-offs which affect quality MA Context	179 - 179

M5C Adopt social rules

MA Confirmation

ID	Document	Quotation Content	Codes	Reference
10:52	#3 ITA	5C is een hele goede regel om direct op te starten als je deze werkwijze gaat toepassen.	M5C Adopt social rules MA Confirmation	89 - 89
11:25	#4 FBI	C ook	M5C Adopt social rules MA Confirmation	115 - 115
12:41	#5 PM	Nou neem sociale regels aan, we zijn daar heel simpel in.	M5C Adopt social rules MA Confirmation	81 - 81

ID	Document	Quotation Content	Codes	Reference
8:71	#1 ICE	Er zijn wel sociale regels maar die zijn impliciet. Niet expliciet. Als je een demo geeft moet je altijd iedereen de kans geven om te reageren vind ik, je moet ruimte in bouwen waar mensen vragen kunnen stellen. Respectvol met elkaar om te gaan staat boven alles natuurlijk. En ze uitnodigen om ernaar te kijken, laat het maar weten, ze kunnen altijd bellen, als het nodig is mag het zelfs 's avonds of in het weekend. Je moet werken aan het vertrouwen.	M5C Adopt social rules MA Confirmation MA Substantiation	187 - 187

MA Substantiation

ID	Document	Quotation Content	Codes	Reference
8:71	#1 ICE	Er zijn wel sociale regels maar die zijn impliciet. Niet expliciet. Als je een demo geeft moet je altijd iedereen de kans geven om te reageren vind ik, je moet ruimte in bouwen waar mensen vragen kunnen stellen. Respectvol met elkaar om te gaan staat boven alles natuurlijk. En ze uitnodigen om ernaar te kijken, laat het maar weten, ze kunnen altijd bellen, als het nodig is mag het zelfs 's avonds of in het weekend. Je moet werken aan het vertrouwen.	M5C Adopt social rules MA Confirmation MA Substantiation	187 - 187

ID	Document	Quotation Content	Codes	Reference
10:53	#3 ITA	Want wat je daarmee doet is verantwoordelijkheid nemen voor je eigen toko. Dat kan in de vorm van feedback van klanten, noem dat incidentmanagement. Maar ik heb het liever zelfs nog proactief reageren op waarschuwingen.	M5C Adopt social rules MA Substantiation	89 - 89
10:55	#3 ITA	Maar dan ben je qua kwaliteit al veel verder omdat je een feedback krijgt dat het niet werkt, omdat je het al hebt opgelost voor dat hij belt. Dus die zou ik eigenlijk in tweeën splitsen, enerzijds proactief en anderzijds het verwerken van feedback.	M5C Adopt social rules MA Substantiation	89 - 89
12:42	#5 PM	De enige Code Of Conduct die we hebben is; 'Je moet het willen, je moet bereid zijn om hier energie in te steken (m.a.w. je moet enthousiast zijn), en je bent zelf verantwoordelijk. Dus dat is een hele simpele. En sociale regels in de zin van 'we zijn allemaal mensen, fouten maken mag, en we zijn direct en open maar wel fair naar elkaar toe'. En dat is eigenlijk min of meer de Code Of Conduct. Maar dat laatste spreken we niet eens uit, dat eerste spreken we altijd wel uit van 'we willen werken met enthousiaste mensen. Je moet het willen en een hele goede reden hebben om hieraan te gaan werken en je moet van ons altijd een doel definiëren. En als je dat niet wil, ook goede vrienden maar dan gaan wij er geen energie in steken. En daar zijn we ook duidelijk in, wel een beetje zakelijk.	M5C Adopt social rules MA Substantiation	81 - 81

MA Not applied

-

MA Substantiation Not applied

ID	Document	Quotation Content	Codes	Reference
13:38	#6 PO	Naja dat gaat dus ook over die sociale regels en dat soort dingen. Uiteindelijk zijn het de teams die daar van zijn maar we moeten ze daar in faciliteren en daar zit een ander punt, dan moeten ze opgevoed worden en wie gaat die teams nou opvoeden. Wie gaat die teams ter verantwoording roepen? Jarenlang hebben wij gevonden dat wij daar voor een deel voor waren, voor een deel misschien ook wel maar tegelijkertijd zeg ik dat de Java-ontwikkelstraat de voorziening levert zodat teams maximaal kunnen ontwikkelen maar wij zijn niet van het vakmanschap van teams en van ontwikkelaars. Wij zijn ook niet van of een team de juiste verantwoordelijkheden neemt. Daar zij anderen binnen dit bedrijf van. Dus als wij verantwoording geven aan een team, als organisatie, op welke manier wordt dan die verantwoording getoetst/getest, tegen wie moeten ze verantwoordelijkheid afleggen	M5C Adopt social rules MA Not applied MA Substantiation	68 - 68

MA Example

ID	Document	Quotation Content	Codes	Reference
8:72	#1 ICE	Precies, en dan terugkoppeling geven dat dwing je niet af door echt harde afspraken te maken dat is hoe je in elkaar steekt en wat je belangrijk vindt	M5C Adopt social rules MA Example	191 - 191
8:73	#1 ICE	En dan mag je ook zeggen wat je verkeerd hebt gedaan, en dat vinden ze dan niet meer erg. Zij kunnen ook fouten maken, wij kunnen ook fouten maken. Dat maakt dan niet meer uit. Als iedereen maar weet dat die fout gemaakt is en dat er aan wordt gewerkt om het op te lossen.	M5C Adopt social rules MA Example	193 - 193
8:74	#1 ICE	En om het te voorkomen	M5C Adopt social rules MA Example	194 - 194
10:54	#3 ITA	Als jij in je ligging waarschuwingen of fouten ziet zeg maar dan moet je daar proactief op inspelen. Als je dat doet dan komt de klant niet eens met slechte feedback maar hooguit met 'joh, ik moet drie keer op een knop drukken, kan dat niet één keer? Maar dat is meer een functionele feedback, die ook belangrijk is.	M5C Adopt social rules MA Example	89 - 89

NM Most Important Measure

ID	Document	Quotation Content	Codes	Reference
8:84	#1 ICE	Ja sociale regels zijn omgangsvormen hoe je met elkaar omgaat, dat vind ik het allerbelangrijkste.	M5C Adopt social rules NM Most Important Measure	227 - 227

NM New Measure

M5D Implement measures in the CI/CD-process to preserve quality

MA Confirmation

ID	Document	Quotation Content	Codes	Reference
----	----------	-------------------	-------	-----------

8:77	#1 ICE	Maar hoe wij dat nu doorgeven is door zelf die software continue te gebruiken.	M5D Implement measures in the CI/CD-process to preserve quality MA Confirmation	202 - 202
10:50	#3 ITA	En ook 5D zit in je pipeline als het ware.	M5D Implement measures in the CI/CD-process to preserve quality MA Confirmation	89 - 89
13:39	#6 PO	Ja	M5D Implement measures in the CI/CD-process to preserve quality MA Confirmation	72 - 72

MA Substantiation

ID	Document	Quotation Content	Codes	Reference
8:75	#1 ICE	Het liefst heb je, en wij hebben dus niet, een geautomatiseerd testverhaal waar rapporten uit komen. Dat zouden wij heel graag willen. Maar hoe wij dat nu doorgeven is door zelf die software continue te gebruiken. Voor onze eigen processen.	M5D Implement measures in the CI/CD-process to preserve quality MA Substantiation	202 - 202

MA Not applied

ID	Document	Quotation Content	Codes	Reference
11:26	#4 FBI	D nog niet omdat we het eigenlijk nog niet doen	M5D Implement measures in the CI/CD-process to preserve quality MA Not applied	115 - 115
12:44	#5 PM	Nee, want dat is aan het team zelf.	M5D Implement measures in the CI/CD-process to preserve quality MA Not applied	85 - 85

MA Substantiation Not applied

ID	Document	Quotation Content	Codes	Reference
12:43	#5 PM	De laatste is 'Implementeer maatregelen in het CI/CD-proces om de kwaliteit te behouden'. Dit is wat Continuous Delivery eigenlijk doet. En niet zozeer een eis aan het implementeren van Continuous Delivery.	M5D Implement measures in the CI/CD-process to preserve quality MA Not applied MA Substantiation	81 - 81

MA Example

ID	Document	Quotation Content	Codes	Reference
13:40	#6 PO	we hebben gewoon standaard modules voor Sonarqube analyse, om de boel naar Sig te sturen en er staan er zo'n stapel op de backlog, tollgates, dat zijn wel de belangrijkste, artifact-management.	M5D Implement measures in the CI/CD-process to preserve quality MA Example	72 - 72

ID	Document	Quotation Content	Codes	Reference
8:76	#1 ICE	En continue kijken naar elkaars codewerk	M5D Implement measures in the CI/CD-process to preserve quality MA Example MA Substantiation	204 - 204

10:51	#3 ITA	Het begint bij ons al, dat is tegenwoordig vrij standaard, dat je die codereviews hebt. Mooie van container hosten is dat zowel artefacten als systeemconfiguratie onder versiebeheer staan. En we hebben die kwaliteitscontroles, sig-metingen, sonarqube, de vulnerabilities, de securitycheck, die wil je ook allemaal geautomatiseerd hebben. Maar dat is meer op een hoger niveau dat je dus op dat soort dingen niet op je bek gaat. Dus dat passen we toe en dat moet ook.	M5D Implement measures in the CI/CD-process to preserve quality MA Example MA Substantiation	89 - 89
-------	--------	---	--	---------

NM Most Important Measure

-

NM New Measure

-

NM Context

ID	Document	Quotation Content	Codes	Reference
8:70	#1 ICE	Je wilt niet dat je software onderuitzakt. Dus het moet wel gewoon kwalitatief van dusdanig niveau zijn dat je niet uit je bed wordt gebeld als het niet werkt of van je vakantie wordt teruggehaald als het niet werkt. En dat kan je met een CI/CD straat opvangen door gewoon iedere commit te testen.	M5D Implement measures in the CI/CD-process to preserve quality MA Context	183 - 183

M5E Apply continuous testing

MA Confirmation

ID	Document	Quotation Content	Codes	Reference
10:49	#3 ITA	Nou ja 5E is sowieso een uitgangspunt bij CI/CD, moet geïmplementeerd worden het geautomatiseerd testen.	M5E Apply continuous testing MA Confirmation	89 - 89
11:44	#4 FBI	Ja	M5E Apply continuous testing MA Confirmation	137 - 137
12:45	#5 PM	Ja, dat is er wel eentje.	M5E Apply continuous testing MA Confirmation	85 - 85
13:27	#6 PO	Ja,	M5E Apply continuous testing MA Confirmation	60 - 60
13:28	#6 PO	pas Continue testen toe daar faciliteren wij in	M5E Apply continuous testing MA Confirmation	60 - 60

ID	Document	Quotation Content	Codes	Reference
9:21	#2 EC	Het automatiseren van testen is wel heel belangrijk en een voorwaarde voor Continuous Delivery En daar zit zeker betrokkenheid van klanten en een stukje transparantie in. Laat je zien wat je test, laat je zien wat je niet test en dan kom je eigenlijk op risicomanagement want testen is eigenlijk voor risicomanagement. Wat is de kans dat er iets omvalt of stukgaat, en dat ga je dan testen. Zo gaan alle testen altijd over	M3C Ensure transparency on the status quo M5E Apply continuous testing MA Confirmation MA Substantiation	114 - 114

		<p>risicomanagement. Risicomanagement heeft weer te maken met een stukje vertrouwen. Hoe hard vertrouw ik dat wat er aan veranderingen zijn dat dat ten goede komt aan de uiteindelijke business die ik wil doen. Dat vertrouwen bouwen en die risicoafweging, daar moet je de klant wel bij betrekken.</p>		
--	--	---	--	--

MA Substantiation

ID	Document	Quotation Content	Codes	Reference
9:21	#2 EC	<p>Het automatiseren van testen is wel heel belangrijk en een voorwaarde voor Continuous Delivery En daar zit zeker betrokkenheid van klanten en een stukje transparantie in. Laat je zien wat je test, laat je zien wat je niet test en dan kom je eigenlijk op risicomanagement want testen is eigenlijk voor risicomanagement. Wat is de kans dat er iets omvalt of stukgaat, en dat ga je dan testen. Zo gaan alle testen altijd over risicomanagement. Risicomanagement heeft weer te maken met een stukje vertrouwen. Hoe hard vertrouw ik dat wat er aan veranderingen zijn dat dat ten goede komt aan de uiteindelijke business die ik wil doen. Dat vertrouwen bouwen en die risicoafweging, daar moet je de klant wel bij betrekken.</p>	<p>M3C Ensure transparency on the status quo M5E Apply continuous testing MA Confirmation MA Substantiation</p>	114 - 114

ID	Document	Quotation Content	Codes	Reference
8:79	#1 ICE	<p>Iedere commit testen... Doe me er twee mensen bij in ons team, dan gaan we dat regelen. Maar dat staat heel hoog op het lijstje. Er zijn drie dingen die heel hoog op het lijstje staan en dat is er 1 van.</p>	<p>M5E Apply continuous testing MA Substantiation</p>	208 - 208
9:22	#2 EC	<p>Ik pleit er altijd voor om vergaand te automatiseren, dat betekent dat de testen aantonen dat de applicatie werkt en als de testen allemaal goed zijn dat doet ie het en kan die door naar productie. Als je dat niet vertrouwd en het risico te groot vindt dan ontbreken er testen, dan heb je te weinig testen</p>	<p>M5E Apply continuous testing MA Substantiation</p>	118 - 118
11:43	#4 FBI	<p>Ja omdat we ook zo een VDI krijgen, dat is ook hoofdzakelijk voor het testen zodat wij Postmen kunnen gebruiken en dan kunnen we ook geautomatiseerd testen. Dus dan kunnen we a la minute dingen inschieten.</p>	<p>M5E Apply continuous testing MA Substantiation</p>	137 - 137
12:46	#5 PM	<p>Dus dat heeft te maken met het feit dat je dus testen automatisch na code commit, dat klopt, dat is een uitgangspunt. En die hoort overigens bij Continuous Integration.</p>	<p>M5E Apply continuous testing MA Substantiation</p>	85 - 85
13:29	#6 PO	<p>En daarin werken we ook samen met de andere teams die daar over gaan, je hebt GTO die gaat over testen en dat moet je dan vanuit die pipeline kunnen ondersteunen. Integratietesten, performancetesten, securitytesten al dat soort zaken</p>	<p>M5E Apply continuous testing MA Substantiation</p>	60 - 60

13:30	#6 PO	Nou heel belangrijk want als je niet continue kan testen kan je ook geen CD doen. Zo simpel is dat. En dat is met name richting de Deco's de andere afdelingen waar nog allemaal handmatige controlestappen zitten. Op het moment dat je dus alles automatiseert inclusief testen, controles, kwaliteit en dat soort zaken dan moet je hun weer meenemen, dat zijn dus niet de ontwikkelteams maar degene die erover gaan, de DBA'ers.	M5E Apply continuous testing MA Substantiation	60 - 60
-------	-------	--	---	---------

MA Not applied

ID	Document	Quotation Content	Codes	Reference
8:69	#1 ICE	Wij hebben dat nog niet ingericht met ons minuscule team maar in principe wil je gewoon geautomatiseerd testen en dan direct door naar productie	M5E Apply continuous testing MA Not applied	183 - 183
8:78	#1 ICE	Dat is nu handmatig, maar dat hoort in de CI/CD straat.	M5E Apply continuous testing MA Not applied	208 - 208
11:27	#4 FBI	E ook nog niet	M5E Apply continuous testing MA Not applied	115 - 115

MA Substantiation Not applied

-

MA Example

ID	Document	Quotation Content	Codes	Reference
11:28	#4 FBI	En nog een simpel voorbeeld tussen het testen. Nu heb je gewoon je scherpjes in een applicatie en straks heb je web services en heb je geen scherpjes en om dat te kunnen testen heb je een VDI nodig. We hebben werkelijk vier maand lang hemel en aarde moeten bewegen maar het schijnt dat we nu VDI krijgen. En dan krijg je als antwoord dat het niet voor de business is maar alleen voor de IV. En dan denk ik, Overheidsorganisatie, jullie willen nieuwe technieken gebruiken maar ga dan ook eens naar dit soort dingen kijken. Daar hebben we de leidinggevende voor ingezet, de Gedelegeerd Business Owner en uiteindelijk heeft de IT Architect het voor elkaar gekregen voor ons. Dat soort dingen.	M5E Apply continuous testing MA Example	78 - 78

NM Most Important Measure

-

NM New Measure

-

MA Context

ID	Document	Quotation Content	Codes	Reference
8:83	#1 ICE	We hebben geen monitoring erop, dat zou je wel willen naast dat je dat geautomatiseerd testen wil hebben wil je daar ook monitoring op hebben welke geautomatiseerd in je ontwikkelprocessen staan welke componenten in mijn hele stuk software hebben nieuwe versies. En dat is tegenwoordig zelfs te automatiseren naast je inrichting en wat onderhoud. Om de componenten één voor één te upgraden naar de laatste versie en te kijken of het nog door de beeld en test komt als je dat allemaal geautomatiseerd in je teststraat hebt staan krijg je automatisch de upgrades van je componenten. En als dat geen problemen oplevert in je voorbrenging uit die test dan kan dat systeem zelf een pull request maken om die versie te upgraden.	M5E Apply continuous testing MA Context	213 - 213

M5F Apply proper strategies and consider preconditions

MA Confirmation

ID	Document	Quotation Content	Codes	Reference
10:60	#3 ITA	Dus 5F en 5B ga je doen bij een bepaald volwassenheidsniveau van je CI/CD.	M5B Agree on trade-offs which affect quality M5F Apply proper strategies and consider preconditions MA Confirmation	97 - 97

ID	Document	Quotation Content	Codes	Reference
8:80	#1 ICE	Dat passen we gewoon toe hoor	M5F Apply proper strategies and consider preconditions MA Confirmation	212 - 212
12:49	#5 PM	Dus dat wordt zeker meegenomen.	M5F Apply proper strategies and consider preconditions MA Confirmation	85 - 85

MA Substantiation

ID	Document	Quotation Content	Codes	Reference
8:82	#1 ICE	nee maar dat voorkomen wij wel door zeg maar regelmatig, ondanks dat het werk is, alle onderliggende componenten te upgraden.	M5F Apply proper strategies and consider preconditions MA Substantiation	213 - 213
10:59	#3 ITA	En dan kijk ik ook meteen naar 5F want als jij die kwaliteit allemaal goed hebt ingeregeld dan kun je ook met veel meer vertrouwen gaan refactoren bijvoorbeeld. Want refactoring is vaak 'we doen het technisch	M5F Apply proper strategies and consider preconditions MA Substantiation	97 - 97

		beter' qua onderliggend verandert de functionaliteit niet. Dat volgt dan vanuit je testen. Je kunt op basis van alleen al je functionele testen gaan refactoren of je technische schuld gaan verminderen of dingen uit elkaar plukken en zo verder.		
12:48	#5 PM	Dus wat we doen is eigenlijk. In de maturity kijken we naar wat voor soort testsoorten gebruik je, wat voor soort testen gebruik je allemaal. Waar we naar op zoek zijn is altijd een soort bal van Demming, ken je het? Plan-do-check-act. Dat zit hier ook een beetje in, 'ben jij in staat om eigenlijk in die Continuous Delivery '. Stel dat je iets, een issue aantreft bijvoorbeeld bij een performance of functionele test blijkt dat dat er ergens een issue zit, hoe snel is dat issue dan terug bij het development team waar het thuis hoort. Dat is natuurlijk wel iets, daar zit die bal van Demming. Hoe snel is dat geborgd en is dat geborgd dan ook dat het kwaliteitsverbetering in één keer wordt meegenomen. Of is het software-onmogelijk dat toch een release door loopt zonder dat iedereen het weet. Zit alles in een monitor dan en wordt ergens het beeld gemonitord. Dus dat wordt zeker meegenomen.	M5F Apply proper strategies and consider preconditions MA Substantiation	85 - 85

MA Not applied

-

MA Substantiation Not applied

-

MA Example

ID	Document	Quotation Content	Codes	Reference
8:81	#1 ICE	Compliance kijken we ook naar, beveiliging, gebruik innovatie. We hebben geen specifieke meting van technical debt.	M5F Apply proper strategies and consider preconditions MA Example	212 - 212

NM Most Important Measure

ID	Document	Quotation Content	Codes	Reference
8:85	#1 ICE	En daarna, heel dicht daarbij, is de strategieën toepassen, want dat is gewoon je vakmanschap als softwareontwikkelaar. Dat je de refactoring doet, de beveiliging, al die thema's waar je aandacht aan moet geven.	M5F Apply proper strategies and consider preconditions NM Most Important Measure	227 - 227

NM New Measure

-

MA Agile

ID	Document	Quotation Content	Codes	Reference
9:29	#2 EC	Ja dat is ook zo, maar de vraag is dus zijn betrokkenheid van de klant en kwaliteit onderwerpen van Continuous Delivery en kun je die daar noemen of kun je beter zeggen dat een goed Agile proces een succesfactor is voor Continuous Delivery	M3B Increase customer involvement M4B Involve the customer in the CI/CD-process M5F Apply proper strategies and consider	146 - 146

			preconditions MA Agile	
--	--	--	---------------------------	--

6 Resistance to change

M6A Employ strategies and ways to share knowledge and skills

MA Confirmation

ID	Document	Quotation Content	Codes	Reference
10:65	#3 ITA	Ja alle drie belangrijk en alle drie hebben ze een eigen tijdslijn en een eigen tempo	M6A Employ strategies and ways to share knowledge and skills M6B Align strategies and policies, and establish a proper culture M6C Ensure appropriate norms, values and behaviour MA Confirmation	129 - 129

ID	Document	Quotation Content	Codes	Reference
8:89	#1 ICE	Ja maar we hebben dat wel enigszins los moeten laten omdat we zagen dat dat niet het effect had wat we beoogden.	M6A Employ strategies and ways to share knowledge and skills MA Confirmation	257 - 257
11:46	#4 FBI	Tuurlijk, het begint met kennis en vaardigheden delen, want onbekend maakt onbemind	M6A Employ strategies and ways to share knowledge and skills MA Confirmation	149 - 149
12:51	#5 PM	Het antwoord is Ja	M6A Employ strategies and ways to share knowledge and skills MA Confirmation	93 - 93

ID	Document	Quotation Content	Codes	Reference
13:54	#6 PO	Wat wij in ieder geval proberen is datgene wat wij doen om dat wel optimaal bij onze gebruikers te krijgen. Zo hebben we dus de delivery pipeline workshops, we hebben wat andere workshops op het gebied van Enterprise Java-ontwikkeling, we hebben in het verleden Roadshows gegeven met teams en we proberen op zoveel mogelijk manier om dat te doen.	M6A Employ strategies and ways to share knowledge and skills MA Confirmation MA Example	88 - 88

MA Substantiation

ID	Document	Quotation Content	Codes	Reference
10:68	#3 ITA	Ze zijn dus eigenlijk alle drie belangrijk, alleen het heeft heel erg tijd nodig en je moet het langzaam ontwikkelen. Belangrijk is de stip op de horizon te zetten en inderdaad er langzaam naar toe werken. Zeker als je het hebt over iets als de overheidsorganisatie. ^{SEP}	M6A Employ strategies and ways to share knowledge and skills M6B Align strategies and policies, and establish a proper culture M6C Ensure appropriate norms, values and behaviour MA Substantiation	129 - 129

ID	Document	Quotation Content	Codes	Reference
12:62	#5 PM	kijk op het moment dat wij beginnen dan beginnen wij eerst met bewustwording zoals de informatie delen en de bewustwording van je wil een stap gaan zetten richting de doelen doe je wilt bereiken, zoals bijvoorbeeld sneller kunnen voortbrengen, je wilt onafhankelijk kunnen zijn,	M6A Employ strategies and ways to share knowledge and skills MA Confirmation MA Substantiation	23 - 23

ID	Document	Quotation Content	Codes	Reference
8:90	#1 ICE	Het not-invented-here-syndroom. Als het niet zelf bedacht is, is het niet goed. Nou ja we zagen al op een eerder project als we een demo gaven van veel transacties verwerken, sommeren naar groepen/buckets, iedereen handen op elkaar die architecten. Maar ja het hield op bij die handen op elkaar en er werd geen vraag of iets gesteld. En je kan heel erg gaan roepen en zenden etc., maar in the end je krijgt pas echt een aanhang als iedereen naar je kerk komt in plaats van dat jij maar continue dat boek uitdeelt.	M6A Employ strategies and ways to share knowledge and skills MA Substantiation	261 - 261
8:93	#1 ICE	Die weerstand zit niet bij de eindgebruiker, die wil dat heel graag. Het liefst wil die iedere dag nieuwe functionaliteit die al klaar is.	M6A Employ strategies and ways to share knowledge and skills MA Substantiation	267 - 267
9:31	#2 EC	Het is een stukje kennis, het is een stukje transparantie en ten opzichte van traditioneel voelt het heel anders. Weerstand bij mensen komt vanuit angst, komt vanuit 'ja maar ik wil graag controle hebben en als ik krijg wat ik had dan snap ik het tenminste. En nu wordt het allemaal fluïde en snel en dat voelt alsof ik geen controle meer heb'. En waar het gesprek dan over moet gaan denk ik is welke controle wil je precies, waar zit je angst. Maar dat komt toch ook weer meer op Agile neer, wat willen we nou belangrijker vinden. Willen we de procedures belangrijker vinden of vinden we de mensen belangrijker.	M6A Employ strategies and ways to share knowledge and skills MA Substantiation	158 - 158
9:32	#2 EC	Nou nee maar ik denk dat het stukje kennis het makkelijkste is in te vullen. Dus ja er is wel een kennisgat maar er is wel zoveel content over Agile en Continuous Delivery dus dat is niet echt het probleem.	M6A Employ strategies and ways to share knowledge and skills MA Substantiation	162 - 162
10:69	#3 ITA	Mijn ervaring met de klant van mijn project is zoals ik net al zei, zij zien waarom we het doen en ondersteunen ons daar ook in. Dus de weerstand zit niet bij de klant bij ons. Dat wil niet zeggen dat dat altijd zo is natuurlijk.	M6A Employ strategies and ways to share knowledge and skills MA Substantiation	133 - 133
11:45	#4 FBI	Ik denk dat dat van de persoon afhankelijk is. Ik heb dat niet nodig, mijn collega ook niet. Maar iemand die continue in de weerstand zit, ik denk dat je dan dit soort dingen wel nodig hebt.	M6A Employ strategies and ways to share knowledge and skills MA Substantiation	145 - 145
12:52	#5 PM	Maar let op we vragen het niet alleen aan developers, we vragen het ook aan de directeur van IBS wat of hij wil bereiken. Dus het is zowel Bottom-up als Top-down approach. Kijk en stel dus dat wij weggaan dan loop je dus zeg maar. De situatie is van stel je daar dan weg maar dan	M6A Employ strategies and ways to share knowledge and skills MA Substantiation	93 - 93

		is de vraag of de maatregel om kennis en vaardigheden te delen.		
13:53	#6 PO	Dit vind ik een hele lastige want wat ik eerder zei, we zouden dolgraag onze ontwikkelaars willen trainen en dat soort zaken en het vakmanschap bij brengen. Maar om dat echt goed voor elkaar te krijgen, dat kunnen wij als Java-ontwikkelstraat niet, op wat voor manier moet je dat inzetten?	M6A Employ strategies and ways to share knowledge and skills MA Substantiation	88 - 88

MA Not applied

-

MA Substantiation Not applied

-

MA Example

ID	Document	Quotation Content	Codes	Reference
13:54	#6 PO	Wat wij in ieder geval proberen is datgene wat wij doen om dat wel optimaal bij onze gebruikers te krijgen. Zo hebben we dus de delivery pipeline workshops, we hebben wat andere workshops op het gebied van Enterprise Java-ontwikkeling, we hebben in het verleden Roadshows gegeven met teams en we proberen op zoveel mogelijk manier om dat te doen.	M6A Employ strategies and ways to share knowledge and skills MA Confirmation MA Example	88 - 88

ID	Document	Quotation Content	Codes	Reference
8:88	#1 ICE	Onze strategie was in gesprek gaan met de beoordelaar, bekijken wat de posities zijn en uiteindelijk laat de toekomst het maar uitwijzen dan.	M6A Employ strategies and ways to share knowledge and skills MA Example	245 - 245
8:91	#1 ICE	Nou ja en tijdens het maandelijkse event van de afdeling, mooie presentatie maken en netjes voorbereidt zodat we een mooi verhaal met een nieuw onderwerp en de nieuwste technologie hebben, dan wordt met aandacht geluisterd maar in the end, je krijgt eens een keer een vraag en dat is het.	M6A Employ strategies and ways to share knowledge and skills MA Example	261 - 261
8:92	#1 ICE	We hebben ook echt wel weerstand ervaren als we uitlegden wat we aan het doen waren bij mensen om echt met ze samen in gesprek te gaan. Maar dat iemand bang was en de indruk had dat er aan zijn stoelpoten gezaagd werd. En dan gaat die echt in de verdediging en dan krijg je echt rare situaties. Ik heb persoonlijk een keer een situatie gehad waarbij iemand tegen mij zei 'ik wil niet dat jij met mijn dingen bemoeit', daar was ik echt van in shock. 'Je moet je er niet mee bemoeien, ga weg'. Ik was toen blij dat ik niet alleen met die persoon zat. Mensen voelen zich een beetje bedreigd erdoor.	M6A Employ strategies and ways to share knowledge and skills MA Example	265 - 265
12:53	#5 PM	Dus dat is een dingetje zeg maar, je gaat structureren en je zorgt voor die ability structuur dat kennis en vaardigheden dus voordoen, samen doen, zelf doen. Dat is een mechanisme dat gebruikt wordt. En dat past hier wel in deze maatregel, dat is een voorbeeld daarvan.	M6A Employ strategies and ways to share knowledge and skills MA Example	93 - 93

12:59	#5 PM	Ook een hele belangrijke maatregel is dat besluitvorming helder en open moet doen. Dus hoe vindt het besluitvormingsproces plaats en waarover nemen we dan een beslissing zeg maar. Waarover wordt bijvoorbeeld de OR betrokken en hoe worden medewerkers daarbij betrokken zeg maar, dus dat is ook een interventie zeg maar een maatregel die je kunt nemen, die staat hier nog niet in waar wel degelijk aandacht voor is. Nou ja kennis hebben we al gehad zeg maar, de maatregelen voor kennisinjectie zeg maar.	M6A Employ strategies and ways to share knowledge and skills MA Example	97 - 97
12:63	#5 PM	Daar moeten we zorgen dat je op het juiste kennisniveau komt. Dus het begint eigenlijk al bij het injecteren van kennis. En als je dat hebt dan pas kun je de volgende stap zetten en dan kijk je naar maturitylevels en dan zie je dus dat je voor een bepaald maturitylevel moet komen qua kennis zeg maar en dan pas kun je Continuous Delivery gaan invullen. Want als je dat niet doet, dan geef ik jou bijvoorbeeld een Ferrari terwijl je nog geen rijbewijs hebt. Dan kun je wel zeggen dat je een hele mooie Ferrari hebt maar waarschijnlijk rijd je hem binnen een uur in de prak.	M6A Employ strategies and ways to share knowledge and skills MA Example	23 - 23
13:55	#6 PO	Voor die pipeline hebben we dus een template wat dus gebruikers makkelijk maakt om op te starten, dus die lagere leercurve.	M6A Employ strategies and ways to share knowledge and skills MA Example	88 - 88
13:56	#6 PO	De toevoegen van toevoegen van ervaring/coach aan het team. Daarvan zeggen wij eigenlijk dat we daar niet van zijn. We krijgen af en toe die vraag maar als Java-ontwikkelstraat hebben wij daar de capaciteit niet voor. Dat moeten teams binnen hun keten oplossen, binnen de community oplossen of wat dan ook.	M6A Employ strategies and ways to share knowledge and skills MA Example	88 - 88

NM Most Important Measure

-

NM New Measure

-

MA Agile

ID	Document	Quotation Content	Codes	Reference
9:30	#2 EC	Ja grappig, ik denk echt dat het meer vanuit Agile is dan Continuous Delivery.	M6A Employ strategies and ways to share knowledge and skills MA Agile	158 - 158

M6B Align strategies and policies, and establish a proper culture


MA Confirmation

ID	Document	Quotation Content	Codes	Reference
10:65	#3 ITA	Ja alle drie belangrijk en alle drie hebben ze een eigen tijdslijn en een eigen tempo	M6A Employ strategies and ways to share knowledge and skills M6B Align strategies and policies, and establish a proper culture M6C Ensure appropriate norms, values and	129 - 129

			behaviour MA Confirmation	
--	--	--	------------------------------	--

ID	Document	Quotation Content	Codes	Reference
11:47	#4 FBI	En vervolgens inderdaad afstemmen voor ene juiste cultuur.	M6B Align strategies and policies, and establish a proper culture MA Confirmation	149 - 149
12:55	#5 PM	Dus die wordt wel toegepast zeg maar	M6B Align strategies and policies, and establish a proper culture MA Confirmation	93 - 93
13:57	#6 PO	Ja	M6B Align strategies and policies, and establish a proper culture MA Confirmation	88 - 88

MA Substantiation

ID	Document	Quotation Content	Codes	Reference
10:68	#3 ITA	Ze zijn dus eigenlijk alle drie belangrijk, alleen het heeft heel erg tijd nodig en je moet het langzaam ontwikkelen. Belangrijk is de stip op de horizon te zetten en inderdaad er langzaam naar toe werken. Zeker als je het hebt over iets als de overheidsorganisatie. 	M6A Employ strategies and ways to share knowledge and skills M6B Align strategies and policies, and establish a proper culture M6C Ensure appropriate norms, values and behaviour MA Substantiation	129 - 129

ID	Document	Quotation Content	Codes	Reference
9:33	#2 EC	Het probleem is veel meer de cultuur en de verandering van cultuur. Dat is ook waarom Spotify filmpjes heeft wat volgens mij een groot pro-Agile en pro-Continuous Delivery verhaal is maar de titel van het filmpje is 'Cultuur'. En dat zit hem in cultuur en normen en waarden denk ik	M6B Align strategies and policies, and establish a proper culture MA Substantiation	162 - 162
10:67	#3 ITA	Ja en dan heb je het ook over 6B he, creëer een cultuur van open communicatie en op en communicatie is ook van het kunnen zeggen dat het niet goed is gegaan of als er fouten zijn gemaakt zonder dat je daarop wordt afgefakkeld en daarmee het project gestopt wordt. Dus je zal echt wel het agile principe 'als het niet werkt dan gooi je het overboord en ga je het opnieuw doen', maar weet iig dat het niet werkt.	M6B Align strategies and policies, and establish a proper culture MA Substantiation	129 - 129
10:71	#3 ITA	We hebben een beetje de vreemde situatie waarbij containerhostingplatformstuurgroep zegt dat ze er niet klaar voor zijn, er is te weinig ingeregeld, en dus hebben we gezegd dat het zowel op containerhostingplatform wordt gedeployed als op IPAS. Dat kan waarschijnlijk zonder al te veel problemen maar geeft dus wel de koudwatervrees aan die er voor is. Op zich kunnen we onze CIU/CD zeg	M6B Align strategies and policies, and establish a proper culture MA Substantiation	133 - 133

		<p>maar in stand houden, gelukkig want anders waren we nog veel verder van huis. Dus onze pipeline houden we in stand, dus onze Continuous Integration. Maar bij onze Continuous Deployment hebben we een set-back gehad omdat zelfs IPAS, onze back-up platform, zo iets had van daarvoor zijn we niet in deze wereld. Dus ik vind het heel grappig dat de weerstand, n dit specifieke geval, vanuit onze containerhostingplatform-stuurgroep komt die dit eigenlijk helemaal zou embracen</p>		
10:72	#3 ITA	<p>Dus ja er is weerstand tegen verandering, ja je zal je organisatie daarvoor moeten klaar stomen en je zal dat geleidelijk moeten doorgroeien. Maar de weerstand kan ook uit een onverwachte hoek komen. Dat hoeft echt niet alleen maar van uit de klant te zijn. Want ik denk dat als je met functioneel beheer gaat praten dat ze er echt wel voor zullen zijn, maar die zien ook genoeg problemen maar niet omdat ze onoverkomelijk zijn of omdat het in CI/CD zou liggen. Maar vooral omdat het nieuw is en we moeten daar gewoon een vorm in zien te vinden. En ja dan moet je relatief veel tijd in steken en dat is vaak de weerstand die je hebt 'omdat we het altijd zo hebben gedaan en we weten hoe dat gaat lopen'. Zo kom je ook niet verder. Ja maatregelen ja, maar mijn kanttekening is dat het heel lastig is.</p>	<p>M6B Align strategies and policies, and establish a proper culture MA Substantiation</p>	137 - 137
12:54	#5 PM	<p>Dus dat je gaat kijken van welke cultuur aspecten zijn er nu anderszijds als je naar een ingenieurscultuur wil bewegen wat heb je dan nodig zeg maar. Ook kennis zou een onderdeel van kunnen zijn. Maar ook bijvoorbeeld leiderschap. Dus in plaats van een teammanager heel erg dirigerend is moet die eigenlijk zichzelf omzetten, een ander leiderschap gaan hanteren naar coachen bijvoorbeeld. Het idee laten komen uit het team en dat ook gaan stimuleren. Dus dat is er wel eentje die daarin zit.</p>	<p>M6B Align strategies and policies, and establish a proper culture MA Substantiation</p>	93 - 93
13:58	#6 PO	<p>zowel ik als ook onze architect, en dat zijn de belangrijkste twee denk ik, zijn heel erg bezig binnen de organisatie met verschillende veranderonderdelen met hopelijk binnenkort de toekomstige reorganisatie bij een afdeling om een betere invulling te kunnen geven, Wat ik al zei, CD kan alleen in een drie-eenheid van de organisatie, cultuur en technologie. En als je de organisaties splitst in allemaal afzonderlijke onderdeeljes en dat is het bijna onmogelijk om een gecombineerde visie optimaal te ondersteunen.</p>	<p>M6B Align strategies and policies, and establish a proper culture MA Substantiation</p>	88 - 88
13:59	#6 PO	<p>Maar we hebben dat talloze keren aangegeven bij management en dat is ons nog niet gelukt tot nu toe en ook niet om het zelf te doen. Ik denk dat daar wel als we echt optimaal adresseren, willen we echt optimaal DevOps gaan doen, en optimaal CD te gaan doen dan moet dat echt veranderen.</p>	<p>M6B Align strategies and policies, and establish a proper culture MA Substantiation</p>	88 - 88
13:60	#6 PO	<p>Dus dan is die slagvaardigheid volgens mij ook weer zoek.</p>	<p>M6B Align strategies and policies, and establish a</p>	88 - 88

			proper culture MA Substantiation	
--	--	--	-------------------------------------	--

MA Not applied

-

MA Substantiation Not applied

-

MA Example

ID	Document	Quotation Content	Codes	Reference
10:66	#3 ITA	Nou ja misschien ken je hem wel van Fokke en Sukke. Dan zijn ze een paar planners en dan zeggen ze: 'Nou donderdag om kwart over tien dan gaan we de cultuurverandering doen. Nou dat gaat dus niet lukken. Je moet wel een team hebben dat uitspreekt dat het wil doen. Je moet investeren in kennis maar ook in learning on the job, gewoon doen en laat maar fouten maken. Je moet wel voor het resultaat gaan maar je moet ook wel een keertje accepteren dat het ook wel eens fout gaat.	M6B Align strategies and policies, and establish a proper culture MA Example	129 - 129
11:49	#4 FBI	We hebben afspraken gemaakt, 'van als jij niet test dan kunnen we niet releasen en dan ben jij veroorzaker of we releasen zonder dat jij getest hebt, zeg het maar'. Nou dan kan je elkaar ook aanspreken op die regels vind ik. En dan denk ik communiceer je op die momenten wanneer jij ergens vastloopt en dan zoek je naar een oplossing samen. Maar dan pak je ook verantwoordelijkheid voor dingen.	M6B Align strategies and policies, and establish a proper culture MA Example	149 - 149
12:58	#5 PM	Kijk één maatregel wat ik hier bij weerstand mis is communicatie, dus de informatievoorziening naar mensen toe dat het goed is ingericht.	M6B Align strategies and policies, and establish a proper culture MA Example	97 - 97

NM Most Important Measure

ID	Document	Quotation Content	Codes	Reference
8:94	#1 ICE	maar ik denk dat juist de cultuur het belangrijkste is, alleen dat tot stand brengen, dat zie ik niet	M6B Align strategies and policies, and establish a proper culture NM Most Important Measure	289 - 289
8:95	#1 ICE	Ik denk ook niet dat je dat veranderd krijgt. Het schip is te groot. Onbekend maakt vaak onbemind.	M6B Align strategies and policies, and establish a proper culture NM Most Important Measure	291 - 291

NM New Measure

-

M6C Ensure appropriate norms, values and behaviour

MA Confirmation

ID	Document	Quotation Content	Codes	Reference
10:65	#3 ITA	Ja alle drie belangrijk en alle drie hebben ze een eigen tijdslijn en een eigen tempo	M6A Employ strategies and ways to share knowledge and skills M6B Align strategies and policies, and establish a proper culture M6C Ensure appropriate	129 - 129

			norms, values and behaviour MA Confirmation	
--	--	--	--	--

ID	Document	Quotation Content	Codes	Reference
12:56	#5 PM	Ik zei het net eigenlijk al dat je een faciliterend leiderschap moet gaan faciliteren dus die zit daar zeker ook in.	M6C Ensure appropriate norms, values and behaviour MA Confirmation	93 - 93
13:61	#6 PO	Ja sowieso een voorbeeldfunctie	M6C Ensure appropriate norms, values and behaviour MA Confirmation	88 - 88

MA Substantiation

ID	Document	Quotation Content	Codes	Reference
10:68	#3 ITA	Ze zijn dus eigenlijk alle drie belangrijk, alleen het heeft heel erg tijd nodig en je moet het langzaam ontwikkelen. Belangrijk is de stip op de horizon te zetten en inderdaad er langzaam naar toe werken. Zeker als je het hebt over iets als de overheidsorganisatie. ¹ _{SEP}	M6A Employ strategies and ways to share knowledge and skills M6B Align strategies and policies, and establish a proper culture M6C Ensure appropriate norms, values and behaviour MA Substantiation	129 - 129

ID	Document	Quotation Content	Codes	Reference
13:62	#6 PO	Ik vind wel 'practise what you preach', wat we uitdragen moeten we zelf ook doen. En volgens mij doen we dat ook. En inderdaad geef het ontwikkelteam eigenaarschap en vertrouwen. Daar gaan wij minder over als de Productowners en andere mensen, maar we zijn wel met de mensen in gesprek die dat zouden kunnen doen. En direct met de teams zodat ze dat krijgen maar ook dat ze dat moeten krijgen van hun productowner en hun ketenmanagers en dergelijke. En ik zou willen dat het meer was, maar dat lukt niet altijd.	M6C Ensure appropriate norms, values and behaviour MA Substantiation	88 - 88

MA Not applied

ID	Document	Quotation Content	Codes	Reference
11:48	#4 FBI	Normen, waarden en gedrag, ik zie nog steeds dat veel mensen niet aangesproken worden op gedrag.	M6C Ensure appropriate norms, values and behaviour MA Not applied	149 - 149

MA Substantiation Not applied

-

MA Example

ID	Document	Quotation Content	Codes	Reference
10:70	#3 ITA	Maar de weerstand, gek genoeg, vinden wij bij het containerhostingplatform. Die zegt dat zij er als platform nog helemaal niet klaar voor zijn terwijl wij al veel sneller en veel verder willen als project zijnde. Heel tekenend, als jij daar een vraag hebt, dan kun je naar de website gaan en kun je een formuliertje invullen om die vraag te stellen en om consultancy aan te vragen. En dat is zeg maar het containerhostingplatform.	M6C Ensure appropriate norms, values and behaviour MA Example	133 - 133
12:57	#5 PM	En één van de voorbeelden is natuurlijk het moment dat je meer Ops-activiteiten krijgt dat de functie verandert. Dus stand-by lopen wordt dan iets dat binnen een functiebeschrijving gaat komen. Dus je ziet dus dat de norm daar wijzigt, de norm wordt dus dat je voor veel meer verantwoordelijk wordt en eigenlijk 24/7 misschien wel beschikbaar moet zijn eigenlijk, om zo een applicatie of systeem te ondersteunen. Dus je ziet dus dat het impact gaat hebben op de functie maar ook de normen veranderen en daar hoort een bepaald soort gedrag bij. Dat is dus een HR kant waar je rekening mee moet gaan houden en mensen moet gaan helpen om dat a. te snappen wat er gaat gebeuren, wat betekend dat dan voor hun. Maar ook zorgen dat er dingen geregeld worden bijvoorbeeld is saldering of anderzijds zeg maar. Je moet dat in ieder geval kunnen compenseren uiteindelijk	M6C Ensure appropriate norms, values and behaviour MA Example	93 - 93

NM Most Important Measure

-

NM New Measure

-

MA Agile

-