

MASTER'S THESIS

The Role of Procrastination in Agile IT Projects

Diepstraten, M.

Award date:
2022

[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: 02. Jul. 2022

Open Universiteit
www.ou.nl



The Role of Procrastination in Agile IT Projects

Degree program: Open University of the Netherlands, Faculty Science
Master of Science Business Process Management & IT

Course: IM9806 Business Process Management and IT Graduation Assignment

Student: Maud Diepstraten

Date: January 29, 2022

Thesis supervisor: Dr. Nick Benschop

Second reader: Dr. Montserrat Prats López

Status: Final

Abstract

Procrastination is a bias that can cause project failure in Information Technology (IT). Research on procrastination is mainly studied in traditional waterfall projects, despite the increased use of the relatively new and considerably different Agile method. This might lead to a blind spot in project management and could be a risk in recognizing the bias and preventing or reducing the impact of procrastination. The main research question is: What role does procrastination play in Agile IT projects? This research is qualitative and explorative, based on literature research and eighteen semi-structured interviews. The research results show that procrastination is a widely spread problem in Agile IT projects and harms the timing, costs, quality, and functionality, as with waterfall projects. Postponing resolving and preventing technical debt is more common in Agile IT projects because of the high business pressure to deliver functionality. Capacity allocation could mitigate this. Understanding the causes of procrastination (task averseness, personal characteristics, and business influences) is relevant for recognizing and acting upon procrastination. Agile feedback mechanisms, open feedback culture, and an engaged and mature Agile team enable detecting procrastination early and limiting the impact. This research suggests that working with Agile can counteract procrastination in IT projects.

Keywords: Information Technology, project escalation, bias, procrastination, Agile, technical debt

Preface

You are reading the thesis ‘the role of procrastination in Agile IT projects’. This research consists of interviews conducted among IT departments in three organizations in The Netherlands. I wrote this thesis to fulfill the graduation requirements of the master degree Business Process Management and IT at the Open University in The Netherlands. From February 2021 to January 2022, I was engaged in researching and writing this thesis.

The research question for this thesis was formulated together with my supervisor, dr. Nick Benschop. The research was complex but educational and fun. After an extensive literature review and broad qualitative research, I was able to answer the research question. I would like to thank my thesis supervisor, dr. Nick Benschop for his extensive feedback that allowed me to continue my research with new insights.

I would like to thank my fellow graduates for the exciting discussions during meetings and of course, for conducting part of the interviews. I would also like to thank all participants who were willing to share their honest experiences. Without their cooperation, I would not have been able to do this analysis.

This thesis will also conclude a period of two years in which I studied for my master’s degree. I would like to thank my study group, with whom I have carried out various courses and learned a lot. In addition, I would like to thank my colleagues for their flexibility in recent years. Above all, I would like to thank my beloved for their understanding and patience in the past period. Special thanks go to my parents, who are a true inspiration. Michael, thanks for your support. You kept me motivated and always encouraged me to go the extra mile.

I hope you enjoy your reading.

Maud Diepstraten

Eindhoven, January 29, 2022

Summary

Complications in Information Technology (IT) projects have been occurring for decades. To make projects more successful, an increasing amount of IT projects are managed by Agile project management methods instead of traditional waterfall methods. Nevertheless, IT projects still face recurring problems. Human decision errors, also called biases, can cause project failing. This paper focuses on the bias procrastination. Procrastination is a complex bias where people tend to voluntarily delay a task or action despite being worse off because of the delay. Research on procrastination is mainly studied in traditional waterfall projects rather than in the relatively new Agile method. Waterfall and Agile project management methods are considerably different, suggesting that biases probably play a different role in Agile IT projects than in waterfall IT projects.

Not much is known about the role of procrastination in Agile IT projects, which might lead to a blind spot in project management and could be a risk in recognizing the bias and preventing or reducing the impact of procrastination. The main research question is: What role does procrastination play in Agile IT projects? With understanding the role of procrastination, this research aims to clarify the manifestation, causes, consequences, and possible countermeasures of procrastination in Agile IT projects.

This research is qualitative and explorative to get a more abundant, coherent, and clear understanding of procrastination in Agile IT projects. Limited information about the role of procrastination is known, so a combination of literature research and semi-structured interviews is executed to collect information from literature and practice. Three different interviewers conducted eighteen interviews. Participants varied in their Agile role, the number of years of experience working with Agile, and the sector they work. Thematic analysis is used to analyze the qualitative data.

The research results show that procrastination is a problem in Agile IT projects and manifests itself across different Agile roles and various types of work. Procrastination negatively impacts the timing, costs, quality, and functionality of Agile IT projects, just as with waterfall projects. The postponement of code refactoring, documentation, or other tasks to prevent or solve technical debt can result in the accumulation of technical debt. Procrastination and technical debt have similar causes. The pressure to deliver functionality to the business and the fluid and fast-changing environment make the accumulation of technical debt more likely to emerge in Agile IT projects.

The literature and interviews show that task averseness and personal characteristics, like motivation failure, focusing too much on details, and insecurity, are causes of procrastination that arise in both waterfall and Agile IT projects. Team composition and team maturity are other causes of procrastination in Agile IT projects that emerged in the interviews. Understanding these causes of procrastination is relevant for practice since it enables people to recognize the bias and act upon it. For IT projects that lack the novelty of discovery-oriented projects, it is essential to pay attention to task averseness since this occurs more often in this kind of projects.

A transparent backlog with clear priorities will mitigate the risk of technical debt. Especially capacity allocation, a method to allocate resources among each type of work, is precious in Agile IT projects for preventing and solving technical debt, personal development, and refinements. Having an open feedback culture, a mature and competent Agile team, and a clear backlog and transparent priorities can prevent causes of procrastination and reduce the risk of procrastination occurring. It is a myth that procrastination could completely be prevented, so it is essential to detect procrastination as soon as possible to act upon it. An addition to the body of knowledge is the insight that feedback mechanisms, an open feedback culture, and an engaged and mature Agile team will aim to detect procrastination in an early stage so that measures can be taken to reduce the impact of it. Feedback mechanisms include ceremonies, short iterative cycles, and monitoring. An open feedback culture is about transparent communication and speaking up. All countermeasures mentioned are typically for the Agile way of working and, therefore a solid argument for adopting the Agile way of working. Personal and business-related countermeasures, namely meaningfulness, improving stakeholder relationships, increasing personal motivation, and improving skills and knowledge are also countermeasures against procrastination in Agile and waterfall projects.

This research suggests that working with Agile project management methods can reduce the risk and the impact of procrastination in Agile IT projects. For teams that already work Agile, it would be good practice to look at the pressure from the business, personal development, team composition, backlog and priorities, and feedback loops in the team. For IT teams struggling with procrastination, it would be worthwhile to work Agile and pay particular attention to feedback mechanisms, clear backlog and transparent priorities, feedback culture, and a competent and engaged Agile team.

Content

1. Introduction	1
1.1. Background	1
1.2. Exploration of Project Management Methods	1
1.3. Problem Statement	1
1.4. Research Objective and Questions	2
1.5. Relevance	2
1.6. Main Lines of Approach	3
2. Theoretical Framework	3
2.1. Research Approach and Implementation	3
2.2. IT Project Escalation and Biases	3
2.3. Procrastination in (Waterfall) Projects	4
2.4. Procrastination in Agile	5
2.5. Results and Conclusions	6
2.6. The Objective of the Follow-up Research	6
3. Methodology	7
3.1. Conceptual Design	7
3.2. Technical Design	7
3.3. Data Analysis	7
3.4. Reflection on Validity, Reliability, and Ethical Aspects	8
4. Results	8
4.1. Interviewees and Data Analysis	8
4.2. Manifestation	9
4.3. Causes	10
4.4. Effects	11
4.5. Countermeasures	12
4.6. Theoretical Saturation	14
4.7. Summary and Propositions	14
5. Discussion, Conclusions, and Recommendations	16
5.1. Discussion and Reflection	16
5.2. Conclusions	18
5.3. Recommendations for Practice	19
5.4. Recommendations for Further Research	19
5.5. Main Conclusions	20
References	21
Appendix 1: Interview Protocol	23
Appendix 2: Interview Protocol (Dutch)	26
Appendix 3: Interviewees Characteristics	29
Appendix 4: Coding Scheme	30
Appendix 5: Overview Thematic Analyses	37
Appendix 6: Overview of Codes per Interview	38
Appendix 7: General Procrastination Scale	40

1. Introduction

1.1. Background

Complications in Information Technology (IT) projects have been occurring for decades (Keil & Mann, 1997; Project Management Institute, 2018; The Standish Group International, 1995, 2015, 2020). Often waterfall methods were used to manage IT projects. To make projects more successful, an increasing amount of IT projects are managed by Agile project management methods (Cervone, 2011). In research by Holz (2019), nearly half of the organizations use Agile for all or most of their developments. Nevertheless, IT projects still face recurring problems. A report from Johnson and Mulder (2020) shows that in the period between 2015 and 2020, 65% of the IT projects were not successful in terms of budget, time, and/or customer satisfaction. Human decision errors, also called biases, can cause project failing (Keil, 1995). Agile is a relatively novel approach for managing projects, and little research is conducted about this way of working and biases that comes with it.

1.2. Exploration of Project Management Methods

This section will focus on the waterfall and Agile methods and explain the main differences. Waterfall projects, like PRINCE2, are typically well planned, heavily documented, and stage-oriented (Great Britain. Office of Government Commerce, 2002; Pawar & Mahajan, 2017). Agile project management allows project teams to manage projects more effectively by decreasing the amount of overhead (Cervone, 2011). Scrum is the most famous and most used Agile method (Cervone, 2011; Holz, 2019). The Scaled Agile Framework (SAFe) is an Agile framework that integrates Scrum and combines the Agile and lean principles for large enterprises (Turetken, Stojanov, & Trienekens, 2017). The most fundamental difference between the waterfall and Agile project management methods is that a Scrum team is better able to adapt quickly to rapidly changing environments by empowering an iterative process of continuous review (Cervone, 2011). After every iteration, there is a feedback moment with the stakeholders, so the risk of developing something worthless is small in Agile methods (Cervone, 2011). This in contrast to poor change and risk management in waterfall projects, where changes are not possible in the middle of the project (Pawar & Mahajan, 2017).

The rapid development, a brisk environment, and fast development cycles in Agile projects also have a downside, as technical debt can mount more quickly (Kruchten, Nord, & Ozkaya, 2012). Technical debt describes the phenomenon whereby problems arise if tasks are not performed adequately during (software) development (Rios, de Mendonça Neto, & Spínola, 2018). Examples of technical debt are pending code refactoring, outdated documentation, and non-execution of tests. Waterfall and Agile project management methods are considerably different, suggesting that biases probably play a different role in Agile IT projects than in waterfall IT projects.

1.3. Problem Statement

Keil and Mann (1997, p. 139) describe project escalation as “*a continued commitment to a failing course of action*”. Several human decision errors (biases) are known that can lead to project escalation (Cuellar, Keil, & Johnson, 2006; Jani, 2008; Janis, 2008; Nuijten, Keil, & Commandeur, 2016; Park, Im, & Keil, 2008; Wu, Ramachandran, & Krishnan, 2014). This paper focuses on the bias procrastination. Procrastination is defined as “*voluntarily delay an intended course of action despite expecting to be worse off for the delay*” (Steel, 2007, p. 8). More and more projects use modern Agile project management methods (Holz, 2019). However, previous research about procrastination is mainly based on traditional waterfall project management or studied in academic and student contexts (Klein et al., 2017; Rozental et al., 2018). The literature shows significant differences in the mentioned project management methods (Pawar & Mahajan, 2017). This suggests that biases probably play a different role in Agile and waterfall methods. Ambiguity and ignorance about procrastination in Agile is a potential risk to day-to-day project management. Given these points, the problem statement of this study is: not much is known about the role of procrastination in Agile IT projects, which might lead to a blind spot in project management and could be a risk in recognizing the bias and preventing or reducing the impact of procrastination.

1.4. Research Objective and Questions

The objective of this study is to get a better understanding of the role of procrastination in Agile IT projects. Only limited information about the role of procrastination in Agile IT projects is known. The contribution of this study is to obtain a more abundant and coherent insight into the bias procrastination in Agile IT projects. The main research question is:

What role does procrastination play in Agile IT projects?

With understanding the role of procrastination, this research aims to clarify the manifestation, causes, consequences, and possible countermeasures of procrastination in Agile IT projects. Five sub-questions regarding this objective will be answered to answer the research question.

1. How does procrastination manifest itself in Agile IT projects?
2. What are the causes of procrastination in Agile IT projects?
3. What effect can procrastination have on time, costs, functionality, and quality in Agile IT projects?
4. What are possible countermeasures against procrastination in Agile IT projects?
5. What is the difference between the role of procrastination in Agile IT projects compared to traditional waterfall IT projects?

This research is qualitative and explorative, and all sub-questions will be answered by a combination of literature research and semi-structured interviews. This combination is chosen to compare literature with information from practice about the role of procrastination in Agile IT projects. The interviews will be focused on understanding interviewees' experiences on the manifestation, causes, effects, and countermeasures of the bias. The first sub-question aims to clarify who procrastinates, what is procrastinated, when people procrastinate, and how big the problem is in Agile IT projects. Sub-question two tries to reveal the origin of procrastination in Agile IT projects and so shed light on recognizing the bias. The third sub-question is related to the consequences of procrastination in Agile IT projects. Consequences can relate to the effects of time, costs, functionality, and quality. The fourth sub-question will attempt to describe the countermeasures of procrastination. This will be relevant for the practical recommendations of this study. The Agile experts who will be interviewed do not necessarily have experience with the traditional waterfall way of working and might be unable to answer questions about waterfall projects. The insights into the role of procrastination in Agile IT projects derived from the interviews will be combined with the information about waterfall projects from the literature to answer sub-question five. Answering the sub-questions will lead to the answer to the main question.

1.5. Relevance

Previous research has shown that multiple factors influence project failure (Keil & Mann, 1997). Although more and more projects use Agile methods, the role of procrastination in Agile IT projects is underexposed in the literature. As mentioned in Section 1.2, the role of procrastination may differ for Agile methods compared to waterfall methods, as the two methods show significant differences. Conducting this research could show new insights into the role of procrastination in Agile IT projects and give a relevant contribution to the existing literature. This research may suggest that procrastination manifests itself differently in Agile than in waterfall projects and will contribute to the body of knowledge about the advantages and disadvantages of the Agile method. If the research shows that procrastination manifests itself in Agile IT projects, information about found countermeasures is relevant to the literature.

Because little is known about the role of procrastination in Agile IT projects, this could lead to a blind spot in project management. This study could contribute to practical adjustments in managing IT projects. By understanding the causes of procrastination in Agile IT projects, people can recognize and act against procrastination. If this research shows that specific Agile characteristics act as countermeasures towards procrastination, this could accelerate the adoption of the Agile methodology in organizations. On the other hand, if this research shows that Agile methods cannot handle procrastination well (likely resulting in more technical debt), this could lead to adjustments in the Agile method or choosing different project management methods in the future.

1.6. Main Lines of Approach

This research is based on literature research and semi-structured interviews. The next chapter describes the theoretical framework based on existing literature on procrastination. In Chapter 3, this study's research method and design are illustrated. Chapter 4 will encounter the analysis and results of the semi-structured interviews. The discussion, conclusions, and recommendations of this research are described in Chapter 5.

2. Theoretical Framework

This chapter aims to partly answer the sub-questions formulated in Section 1.3 by formulating a theoretical framework based on previous research. In Section 2.1, the research approach will be discussed. In Section 2.2 until 2.4, a theoretical framework is formulated. Section 2.5 enclose the results and conclusion. Finally, the objective for the follow-up research is described in Section 2.6.

2.1. Research Approach and Implementation

A research approach is created to conduct a critical literature review. Both primary and secondary literature sources are used to find relevant information. In this research, the user-friendly online search engine Google Scholar is used to search for scientific papers. With this search engine, numerous databases worldwide can be searched from one point. A combination of search parameters, search terms, search connectors, and filters are used to scope the search. The most important search terms are: 'project escalation', 'procrastination', 'it project', 'waterfall', 'agile', and 'technical debt'. These terms were searched for in the full text. The language of publication searched for is English. Peer-reviewed recent (2017-2022) articles from refereed academic journals are preferred. Some exceptions are made for original articles because they may be older. Table 1 shows an overview of the search strategy and the number of hits.

Table 1: Overview of the Search Strategy

Search term	# Hits
"project escalation" and (it or is)	1870
procrastination and ("it project" or "is project")	638
procrastination and project	61300
procrastination and project and – student and -academic	16700
procrastination and project and (waterfall or prince2)	1640
procrastination and project and agile	3230
"technical debt" and project and agile	4260
deadlines and agile and "project escalation"	86

Another search method used is finding interesting new papers based on references mentioned or referring to already known articles. This is called backward and forward snowballing. With this approach, twenty papers were found. In total, more than two hundred articles were scanned briefly, 95 articles were intensively reviewed, of which 42 articles were selected. Articles were selected based on relevance (relatedness to the research questions), value (preferably refereed peer-reviewed academic journals without bias), and sufficiency (are authors, ideas, and conclusions recognized by other papers).

2.2. IT Project Escalation and Biases

The Standish Group International (2020) reported that from 2015 to 2020, only 35% of the software projects were successful. Project escalation is a major issue in failing IT projects. Keil and Mann (1997, p. 139) describe project escalation as *"a continued commitment to a failing course of action"*. A typology that is often used to explain project escalation includes four influencing factors: project factors, social factors, organizational factors, and psychological factors, see Figure 1 (Keil & Mann, 1997; Sabherwal, Sein, & Marakas, 2003). Project factors include costs, benefits, expected difficulty, and duration of a project (Keil & Mann, 1997). Social factors can arise in the various groups that are involved, like competitive rivalry (Sabherwal et al., 2003). Organizational factors can promote escalation by involving the structural and political environment, such as slack resources and loose management controls (Keil & Mann, 1997). Lastly, psychological factors include the individual participants in the

process. Those factors include cognitive biases that can affect how information concerning the project is perceived or processed (Keil, 1995). Cognitive biases are subconscious and systematic human errors (Keil, Depledge, & Rai, 2007). Due to the typical uncertainty in decision-making, humans tend to use heuristics in making decisions, which results in biases (Keil, 1995). An example of this is related to the prior history of project success. Earlier project success can cause decision-makers to ignore or downplay negative information, promoting escalation (Keil, 1995). IT projects are typically dynamic, not tangible, and requirements constantly change, resulting in a highly complex environment with many decision moments (Sleesman, Lennard, McNamara, & Conlon, 2018). This complex environment makes it difficult to make rational decisions, making IT projects more prone to biases. Korzaan and Morris (2009, p. 1321) also mention that *“most problems associated with software development are behavioral in nature”*. Various cognitive biases are mentioned in the literature that causes project failure (Sabherwal et al., 2003). This research is part of a larger study into the role of biases in Agile IT projects. This research will focus exclusively on the bias procrastination.

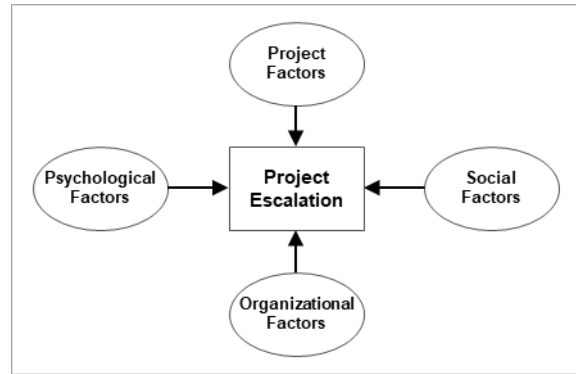


Figure 1: Typology of factors that influence escalation behavior (Keil & Mann, 1997)

This complex environment makes it difficult to make rational decisions, making IT projects more prone to biases. Korzaan and Morris (2009, p. 1321) also mention that *“most problems associated with software development are behavioral in nature”*. Various cognitive biases are mentioned in the literature that causes project failure (Sabherwal et al., 2003). This research is part of a larger study into the role of biases in Agile IT projects. This research will focus exclusively on the bias procrastination.

2.3. Procrastination in (Waterfall) Projects

Procrastination is a complex bias with cognitive, affective, and behavioral components (Steel, 2007; Watson, 2001). Steel (2007, p. 8) defines procrastination as *“voluntarily delay an intended course of action despite expecting to be worse off for the delay”*. In business, this means that teams consistently postpone most of their work until deadlines are approaching (Steel, 2007). The following paragraphs will elaborate on the effects, causes, and countermeasures of procrastination in general and waterfall projects.

2.3.1. Effects

Procrastinating can have several adverse effects on a project. For example, people who delay work closer to the deadline will have a more considerable risk of not meeting the project deadline and enlarging the project duration (Chen, Lan, Zhao, & Shang, 2019). Postponing deadlines could result in the inability to add functionality in the given project duration. As a result of procrastination, increasing the project duration will most likely result in budget overruns. Procrastination also comes with a quality risk. Procrastinators try to meet the deadline by cramming in later stages of the project, potentially resulting in mistakes (Wu et al., 2014). Finally, Steel (2007) mentioned underperformance and feeling miserable in the long term as two individual effects of procrastination. In summary, procrastination harms time, costs, functionality, and quality in projects.

2.3.2. Causes

Many different causes are linked to procrastination. In the first place, the psychological tendency to overvalue current utilities is a fundamental cause of delaying work (Chen et al., 2019). In other words, procrastinators tend to attach a greater salience to immediate-term efforts when allocating efforts over time (Chen et al., 2019; Steel, 2007; Wu et al., 2014). So procrastinators are more likely to tackle issues that are important at the moment, like helping a colleague or doing unrelated tasks, than working on long-term project goals. This cause is linked to the construal-level theory, where events far in time tend to be more abstractly represented than events close in time (McCrea, Liberman, Trope, & Sherman, 2008). Due to the lengthy phases in waterfall projects and deadlines further in time (Cervone, 2011), waterfall projects may deal with a high level of construal. Therefore, is assumed that procrastination often occurs in waterfall projects. Secondly, task aversiveness is why people procrastinate (Steel, 2007). Task aversiveness can occur in IT projects that lack the novelty of discovery-oriented projects (Wu et al., 2014). Thirdly, McCrea et al. (2008) argued that procrastinators focus too much on task details and feel overwhelmed. Since waterfall projects are often heavily weighted and spend much time on details, procrastination is likely to be common in waterfall projects. Van Oorschot, Sengupta, and Van Wassenhove (2018) mentioned the negative effect of schedule pressure in IT projects that can cause team members to use time inefficiently. According to a report by Johnson and Mulder (2020), a typical waterfall project will incur about

80% overhead. The overhead in waterfall projects can be seen as a cause of procrastination, which often results in delayed developments. Other causes of procrastination mentioned in the literature are low self-efficacy, impulsiveness, conscientiousness, motivation failure, planning fallacy, and an overall increase of procrastination in society (Klein et al., 2017; Rozental & Carlbring, 2014; Steel, 2007).

2.3.3. Countermeasures

The literature describes various countermeasures to prevent or reduce the effects of procrastination. Project teams could reduce the postponement of work by setting specific and concrete subgoals (McCrea et al., 2008). Furthermore, intermediate deadlines can improve the performance of team members (Ariely & Wertenbroch, 2002). Although generally seen as a negative effect, schedule pressure can help project teams use time more efficiently, resulting in less delay (Van Oorschot et al., 2018). The behavior of employees is not only influenced by their degree of procrastination but also by that of their colleagues. Wu et al. (2014) indicate that a diverse project team with bigger and smaller procrastinators performs better than a team with an average procrastination level. Finally, competition between team members could reduce the risk of delaying work (Wu et al., 2014).

2.4. Procrastination in Agile

The enormous amount of overhead and frequent changes in requirements is why the Agile project management mindset is developed (Cervone, 2011). Since Agile project management is a very different method than waterfall project management, see 1.2 for most essential differences, procrastination probably plays a different role in the methods. The following paragraphs will outline the (expected) effects, causes, and countermeasures of procrastination in Agile projects.

2.4.1. Effects

Procrastination in Agile projects is expected to negatively affect project success, just like waterfall projects. Procrastination can also result in technical debt. Technical debt is the concept whereby problems arise if tasks are performed inadequately or are postponed (Rios et al., 2018). Examples are outdated documentation or pending code refactoring. Kruchten et al. (2012) indicate that technical debt can mount more quickly in Agile projects than in traditional waterfall projects. Technical debt can result in expanding project duration, large cost overruns, severe quality issues, and the inability to add new features (Rios et al., 2018). An example of this is whenever a customer raises quality issues or security features, developers of an Agile team could always say that those items are put on the backlog for a later moment (Sneed, 2014). This will harm the quality of the product. Additionally, a product owner who defers items with significant long-term benefits may push complex functionality to the future, preventing new features from being created. In summary, procrastination in Agile projects will likely negatively influence the timeliness, costs, quality, and functionality of an IT project.

2.4.2. Causes

Since procrastination and technical debt both originate as a result of overvaluing short-term benefits (Rios et al., 2018; Wu et al., 2014), this may imply they have similar causes. Kruchten et al. (2012) clarify that developing and delivering very rapidly are the reasons why technical debt is more common in Agile projects. The causes of technical debt relate to the fact that most Agile projects deliver products and services frequently (Turetken et al., 2017) because they deal with fluid and rapidly changing environments (Cervone, 2011). Since procrastination and technical debt have similarities, it is expected that the fast and complex environment is a cause of procrastination in Agile IT projects as well.

2.4.3. Countermeasures

The following section outlines several Agile characteristics that could act as a countermeasure to procrastination. Firstly, working in short iterative cycles is one of the most common characteristics of Agile methods (Cervone, 2011; Van Oorschot et al., 2018) and relates to intermediate deadlines, which is a countermeasure (Ariely & Wertenbroch, 2002). It is assumed that short iterative cycles will lower the construal level and, as a result, lower the risk of procrastination. Next to intermediate deadlines, Scrum comes with specific goals per iterative cycle: sprint goals. McCrea et al. (2008) recognized that concrete subgoals could reduce delaying work. Thirdly, communication with stakeholders and team members is an essential characteristic of Agile (Cervone, 2011). This daily Scrum is an extra moment of communication and could give

team members commitment not to delay and meet the deadline. Other major components in Scrum are Scrum artifacts, including the sprint backlog and burndown charts. Since procrastinators delay beginning or completing work tasks, a burndown chart can be an extra motivation to burn down items gradually and not delay to the end of the sprint. Besides that, a burndown chart could be a catalyst for positive competition between team members, mitigating the tendency to procrastinate (Wu et al., 2014). A sprint backlog, another Scrum artifact, is a list of all items to work on during an iteration and gives a clear overview of the goal of the iteration. Items need to be very clear before pulling into an iteration since project requirements cannot be changed during an iteration (Cervone, 2011). A concrete sprint backlog can help employees avoid focusing too much on details, which is a cause of procrastination (McCrea et al., 2008), reducing the risk of procrastination occurring.

2.5. Results and Conclusions

This chapter is the first attempt to answer the research sub-questions. Procrastination is a complex bias where people voluntarily delay actions or tasks despite expecting to be worse off. The expectation is that the role of procrastination is different in waterfall projects than in Agile projects. Waterfall projects are heavily weighted, focus on details, and have lengthy phases, resulting in a risk for procrastination and negatively influencing the timeliness, costs, quality, and functionality of a project. Conversely, Agile projects have short iterative cycles, concrete subgoals, use Scrum artifacts and Scrum activities, which could reduce the causes of procrastination and thus reduce the risk of procrastination occurring. On the other hand, technical debt occurs more often in Agile projects than in waterfall projects, which is more likely to result in delays, cost overruns, quality issues, and less functionality. In Table 2, the (possible) effects, causes, and countermeasures are summarized for waterfall and Agile projects.

Table 2: Overview of effects, causes, and countermeasures of (waterfall) projects and Agile projects

	(Waterfall) Projects	Agile Projects
Effects	<ul style="list-style-type: none"> • Time: not meeting deadlines or enlarging project duration • Costs: budget overruns • Functionality: inability to add functionality • Quality: potential mistakes • Individual effects: underperformance and feeling miserable 	<ul style="list-style-type: none"> • Time: expanding project duration. • Costs: cost overruns • Functionality: pushing complex functionality forward • Quality: postpone quality issues
Causes	<ul style="list-style-type: none"> • Overvaluing short-term utilities and a higher level of construal • Task averseness • Focus too much on details • Schedule pressure • Waterfall overhead • General: Low self-efficacy, impulsiveness, conscientiousness, motivation failure, planning fallacy, and an overall increase in society 	<ul style="list-style-type: none"> • Fast development and frequent delivery • Fluid and rapidly changing environments
Counter-measures	<ul style="list-style-type: none"> • Setting specific subgoals • Intermediate deadlines • Schedule pressure • Diverse project team • Competition between team members 	<ul style="list-style-type: none"> • Short iterative cycles • Specific sprint goals • Communication and commitment • Sprint backlog • Burndown chart

2.6. The Objective of the Follow-up Research

The objective of the follow-up research is to explore the statements made with caution in this chapter in practice and to uncover possible new elements about the role of procrastination in Agile IT projects. The follow-up research will consist of semi-structured interviews with experts in Agile IT projects.

3. Methodology

This chapter elaborates on the research method and design. Section 3.1. will describe the conceptual design and explain the choices of the chosen research method. The technical design will describe the research method in more detail in Section 3.2. In Section 3.3. will outline the data analysis approach. The final section of this chapter will focus on reflecting the quality risks and ethical aspects of this research.

3.1. Conceptual Design

This section will clarify the conceptual design and the selected research methods. The objective of this research is to get a better understanding of the role of procrastination in Agile IT projects. Little is known about the role of procrastination in Agile IT projects, and therefore little literature is available to get a better understanding of this subject. Exploratory studies are particularly useful to clarify understandings of a phenomenon (Saunders, Lewis, & Thornhill, 2019) which is the reason to conduct explorative research. A common quantitative research method to measure procrastination is the experiment using questionnaires (Ariely & Wertenbroch, 2002; Battoa, 2019; Benschop et al., 2020; Cadena, Schoar, Cristea, & Delgado-Medrano, 2011; McCrea et al., 2008; Van Oorschot et al., 2018). Questionnaires do usually not fit exploratory research because of the following disadvantages: needs to be standardized and pre-defined, do not allow the respondent to come up with subjects, and asking additional questions is not possible (Saunders et al., 2019). Although most research about cognitive biases is quantitative, Nuijten, Benschop, Rijsenbilt, and Wilmlink (2020) used interviews in their exploratory research about cognitive biases because they wanted *“to obtain a more rich and coherent insight on cognitive biases”* (Nuijten et al., 2020, p. 10). A semi-structured interview is structured to allow comparisons of important themes between participants, yet flexible because the order does not matter, and additional questions can be asked during the interview (Saunders et al., 2019). This makes semi-structured interviews best suited for this exploratory research over other methods. With the use of literature research and semi-structured interviews, the sub-questions will be answered to answer the main question.

3.2. Technical Design

This section will discuss the technical design of the interviews. To get a complete picture of the role of procrastination in Agile IT projects, eighteen semi-structured interviews were conducted with experts with a varied set of roles (product owners, scrum masters, developers, and agile coaches) and sectors (high tech, financial services, and business services). To ensure that participants can best represent the reality of the role of procrastination in Agile, participants need at least two years of experience with Agile working in IT projects. The interviews were conducted via video call on a one-to-one basis, were audio-recorded, and the spoken language was Dutch. Therefore, all quotes in this paper are translated into English. This research is part of a broader research, and questions regarding other biases were also included in the interviews. Three different researchers each interviewed six experts about three biases. This makes it possible to interview multiple participants from different organizations in a short time frame. Every participant was questioned for thirty minutes about three different biases, resulting in an interview of approximately ninety minutes. The interviews are semi-structured, which means the questions are prepared in advance per theme. Questions are derived from the sub-questions mentioned in Section 1.4 or based on the literature research. A pilot of one interview was conducted to gain experience in conducting an interview and testing whether the interview protocol worked as planned. The proposed order of questions was slightly modified based on the pilot results, and the definition of technical debt was added to the interview protocol. The pilot was an effective method to improve the interview protocol. In Appendix 1 (Appendix 2 for Dutch) the interview protocol including the interview questions is given.

3.3. Data Analysis

After the interviews, a data analysis took place. This approach is explained in this section. The interview audio recordings were manually transcribed to textual data and anonymized. Thematic analysis is used to search for themes or patterns related to the sub-questions of this research. Thematic analysis offers a systematic yet flexible approach to analyze qualitative data (Saunders et al., 2019). Atlas.ti, a qualitative data analysis tool, is used to code the data. The themes are based on the sub-questions of this research: manifestation, causes, effects on time, effects on costs, effects on quality, effects on functionality, and measures, see Figure 2. Specific

sub-themes and codes per theme are based on the literature research or emerge from the interviews. Based on the thematic analysis, a coding scheme is set up. The coding scheme provides a clear description of the content of the themes, sub-themes, and codes and describes how these are related. This includes establishing explicit coding rules to avoid subjectivity and avoid interpretation issues.

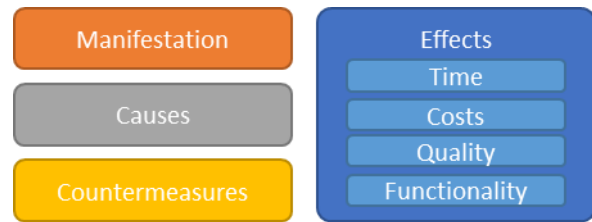


Figure 2: Themes derived from the research sub-questions

3.4. Reflection on Validity, Reliability, and Ethical Aspects

This section will focus on the quality risks and ethical aspects of this research. The generalizability of this study (external validity) is increased by working together with three other researchers. This made it possible to increase the number of participants with various roles and corporate sectors. One of the quality risks lies in the variety of interviewers. In all cases, the interviewers worked at the same organization as the interviewees, resulting in possible researcher errors and bias. Researcher errors and bias refer to the factor which alters the researcher's interpretation and the researcher's recording of response. A few interventions are taken to mitigate the risk of distortion of the researcher. Firstly, an interview schedule is designed with an overview of questions. In line with a semi-structured interview for exploratory research, this interview schedule allows the comparison of themes, and yet there is plenty of room to ask additional questions. In addition, all interviews are transcribed to recognize possible incorrect interpretations. Finally, to avoid interpretation issues and validate the results, research data is sent back to the participants to confirm its accuracy.

For qualitative research, internal validity (or credibility) refers to the extent findings of the research match with what the participants intended (Saunders et al., 2019). The risks to internal validity are mitigated by the interventions related to researcher bias, as described in the previous paragraph. In addition, all interviewers and interviewees have access to information and definitions about the bias so that everyone understands the bias as best as possible, and the risk of misunderstanding is reduced.

The reliability of research refers to replication and consistency (Saunders et al., 2019). The reliability of this research is increased by clearly describing the research design and being transparent about how the data is obtained. This includes using an interview schedule and conducting the interviews under the same conditions (90 minutes, one-to-one online interview, during working hours). In addition, a coding scheme is set up to increase transparency, reduce the risk of subjectivity, and increase the reproducibility of this research.

All participants were informed of this study by email and consented to participate. Interviewees were asked for their voluntary participation in this research, and it is explicitly made clear that they always have the right to withdraw from this study. All information of the interviewees has been anonymized to comply with ethical standards. To guarantee privacy and to prevent socially desirable answers, it has been made clear to the interviewees that no judgment is made on the projects or products the participants work on.

4. Results

This chapter presents the findings of this research. The first section will describe the characteristics of the interviewees, the data analysis, and the coding scheme. Sections 4.2 until 4.5 will outline the results of the interviews per theme and sub-theme. In Section 4.6, the theoretical saturation will be discussed. The closing of this chapter will concern the formulation of propositions based on the results.

4.1. Interviewees and Data Analysis

As described in the research design in Chapter 3, the research consists of eighteen semi-structured interviews. The interviewees vary in their current role, corporate sector, Agile experience, and current Agile method. On average, the number of years of experience in Agile IT projects is 5.5 years. The interviewees' experience variation in Agile and waterfall methods is given in Figure 3. Figure 4 gives an overview of the diversity of roles of the interviewees. Seven of the interviewees fulfill a combination of two roles, for example, scrum master and

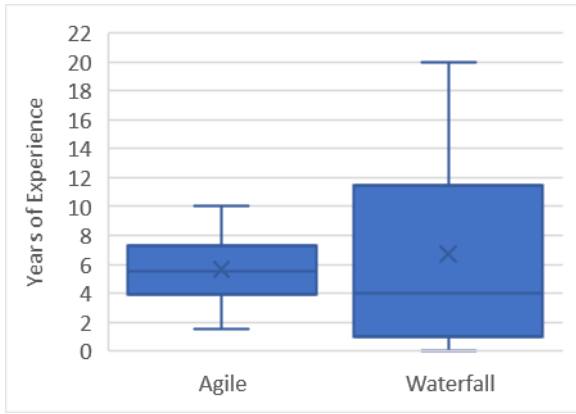


Figure 3: Boxplot of years of experience in Agile and waterfall projects among interviewees

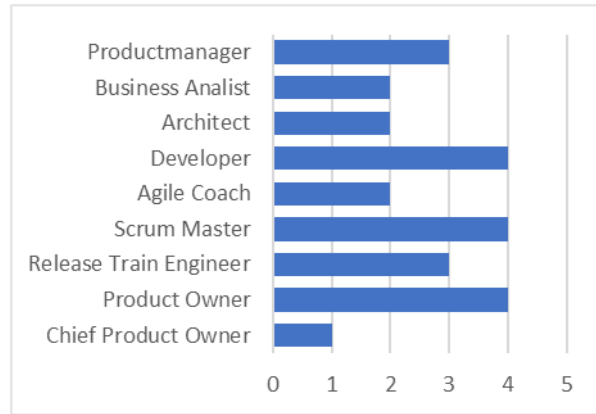


Figure 4: Number of roles among interviewees (some interviewee have a double role)

developer. In Appendix 3, a complete overview of the characteristics of the interviewees is given. A thematic analysis followed the interviews. Based on the thematic analysis, a coding scheme is set up. See Appendix 4 for this coding scheme. Next to the themes and sub-themes that are based on the sub-questions of this research, new sub-themes arrived from the data analysis. An overview of all themes, sub-themes, and codes is given in Appendix 5. The outer ring of the sun burst visualizations used in this chapter represents the size of a code, the specific frequency per code can be found in Appendix 4. The results of the data analysis will be described in the following sections per theme and sub-theme.

4.2. Manifestation

The theme manifestation is focused on clarifying who, what, and when people procrastinate and on the size of the problem in Agile IT projects. The following paragraphs will give more insights into these subjects, see Figure 5 for an overview.

4.2.1. Who, What, and When

Although many interviewees gave examples of individuals procrastinating, it also occurs in groups, teams, or pairs. Many interviewees indicated that developers, scrum masters, product owners, and other team members show procrastination. However, some interviewees could not confirm this for their team. The item most put off in Agile IT projects, according to the interviewees, is resolving or preventing technical debt. This included postponing documentation, refactoring of code, and improving the performance. A few interviewees noted that developers “are not afraid to start something, but have difficulties finishing it” – interviewee 2. According to the interviewees, other items that are postponed are strategical and architectural choices, migrations, stakeholder verification, preparation of work (refinements), operational work, development work, personal development, and communication. The interviewees point out that people postpone work to the end of a deadline, for example, the end of a sprint or the end of a Product Increment (PI).

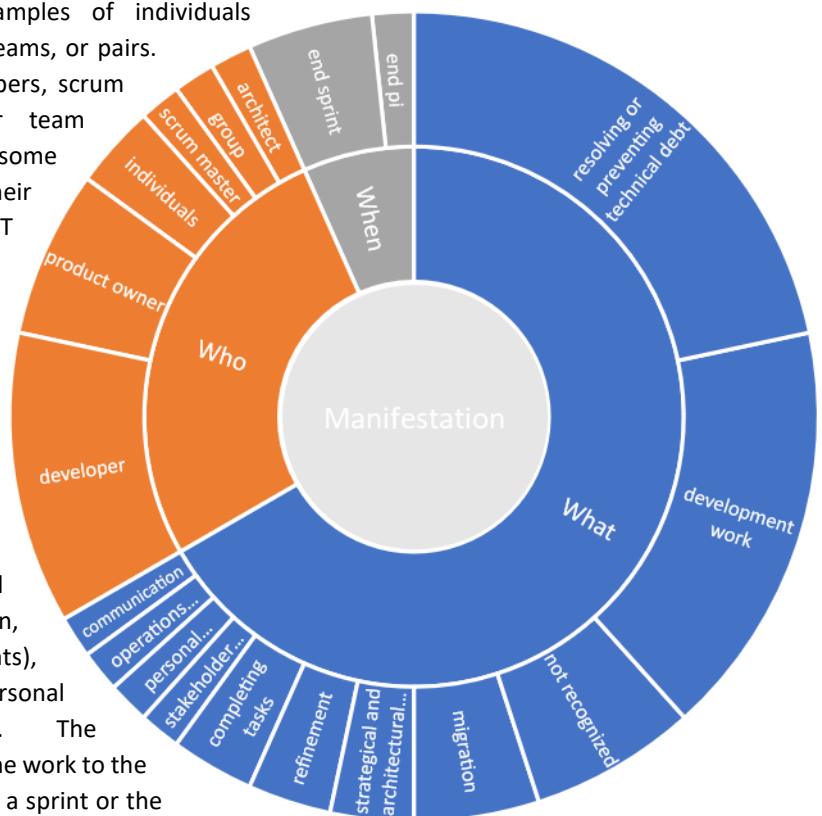


Figure 5: Overview of theme manifestation

4.2.2. Size of the Problem

Interviewees were asked what percentage procrastination is a problem in Agile IT projects. Although most interviewees indicated that this was difficult to estimate, fourteen of the eighteen interviewees gave an estimation. The average was taken if interviewees gave an estimation within a range (so “I think 60% or 70%” – interviewee 18, became 65%). The answer on how often procrastination is a problem in Agile IT projects varies widely, as shown in Figure 6. However, almost all interviewees indicate that the problem exists.

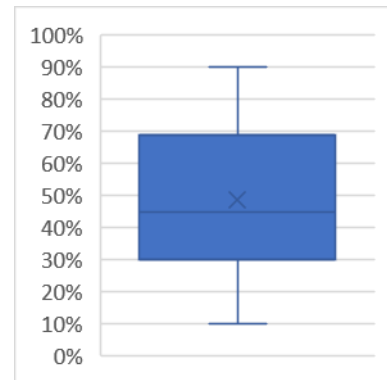


Figure 6: Boxplot of how often procrastination is a problem in Agile IT projects

4.3. Causes

The theme causes describe the identified origins of procrastination in Agile IT projects. Figure 7 provides an overview of the identified causes. In the following paragraphs, each sub-theme of causes will be described.

4.3.1. Tasks

Steel (2007) recognized task averseness as one of the reasons for people to procrastinate. Eleven interviewees identified that the originator of procrastination in Agile IT projects also relates to some form of task aversion. The interviewees of this research identified that most developers pick up easy, small, and short stories first and postpone hard, large, and long stories for later. In addition, people postpone boring and unchallenging tasks. Furthermore, tasks with lots of dependencies and unclear tasks are often delayed by people in Agile IT projects. Task aversiveness is not typical Agile, as it occurs in other methods as well.

4.3.2. Person

Interviewee 17 describes: “it is human to procrastinate” and interviewee 3 responded: “We are all human beings of course (...) everyone suffers from this in their way”. Interviewees identified that the cause of procrastination could partly be explained by the human nature to procrastinate, without mentioning a specific cause. Another personality aspect is the level of introversion of people. Interviewee 1 wondered “if a typical IT person is a bit introverted by nature”. The code introvert also includes how

comfortable people are to speak up, give push-back, and ask for help. If a person is introvert he or she may not feel comfortable asking stakeholders for verification tests and may postpone these tasks. The interviewees indicated that people could also procrastinate if the person is not competent to execute the task or is insecure about whether he/she can perform the task. They pick up stories with which they are familiar first. As mentioned in Chapter 2, motivation failure and focusing too much on details are causes of procrastination in projects (McCrea et al., 2008; Steel, 2007). These causes are also observed by the interviewees working in Agile IT projects. This sub-theme is focused on the person, and it is questionable whether this cause is Agile specific as Steel (2007) and McCrea et al. (2008) also recognize multiple personal characteristics in other fields.

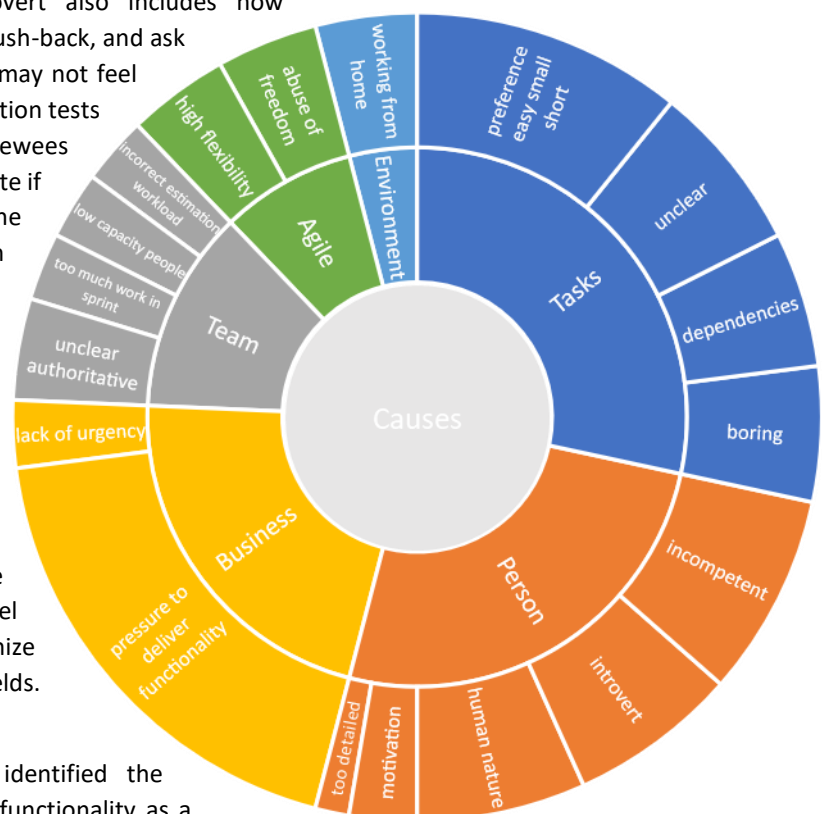


Figure 7: Overview of theme causes

4.3.3. Business

Thirteen of the eighteen interviewees identified the pressure of the business to deliver (new) functionality as a cause of procrastination. This business pressure can cause an

Agile team to postpone technical debt. *“Matters that generate value for the business are given a higher priority because the prevention of technical debt is often not immediately reflected in the end product”* – interviewee 8. The characteristic of an Agile project is that *“somehow we always have more work than can fit in those two weeks”*, which means that stories without direct business value (like resolving or preventing technical debt) are postponed because *“it does not really fit right now”* - interviewee 6. In contrast with the high pressure of the business, a lack of urgency can also be a cause of procrastination. This can manifest itself if stakeholders do not come to review sessions to give feedback and explain the importance of a feature so that the developers also do not feel the urgency of the feature and are less inclined to deliver and thus will delay.

4.3.4. Team

The sub-theme team is related to team composition and team maturity. Interviewee 4 explained that Agile teams *“Do not think about the risks that can occur in a sprint and all things from the Definition of Done, which results in an incorrect estimation of work”*. This incorrect estimation of workload can result in too much work in a sprint and can cause one to take more time on one task and delay other tasks as a result. Additionally, the unclear authority could cause procrastination in Agile IT projects. An example is given by interviewee 18 *“there was no one who dared to make a decision, and then you put it off again”*. A low capacity of people could cause procrastination as well because if there is no capacity, not everything can be picked up, and things have to be postponed.

4.3.5. Agile

An interesting sub-theme that could cause procrastination in Agile IT projects originates from the Agile way of working. Other causes mentioned in this chapter can relate to Agile IT projects, but may not directly result from the Agile way of working. Interviewee 3 expresses his opinion about the flexibility in Agile as follows: *“The real go/no go moments that you have in waterfall projects, you do not have in Agile projects, and I think that makes it easier to procrastinate in Agile projects”*. In addition, Agile teams often have a high degree of autonomy. Young teams could abuse this freedom and delay the start or completion of work.

4.3.6. Environment

Lastly, interviewees point out that environmental aspects that are not explicitly related to the Agile way of working could cause procrastination. This sub-theme covers working from home, where there is less control over people, and *“there is no real contact with your colleagues”* – interviewee 18, making people more prone to procrastinate.

4.4. Effects

Consequences of procrastination that are based on literature are time, costs, functionality, and quality. Other effects of procrastination are a weaker reputation of a team, unsatisfied stakeholders, and personal health issues, like burnout or a lack of personal development. Steel (2007) also mentions these personal issues as a general result of procrastination. See Figure 8 for an overview of all effects of procrastination. In Section 4.4.4. the effects of procrastination are ranked.

4.4.1. Time and Costs

Late deliveries and not meeting deadlines are commonly mentioned as a result of procrastination. Completing the tasks have a longer lead time which can result in more working hours, which in turn can result in overwork.

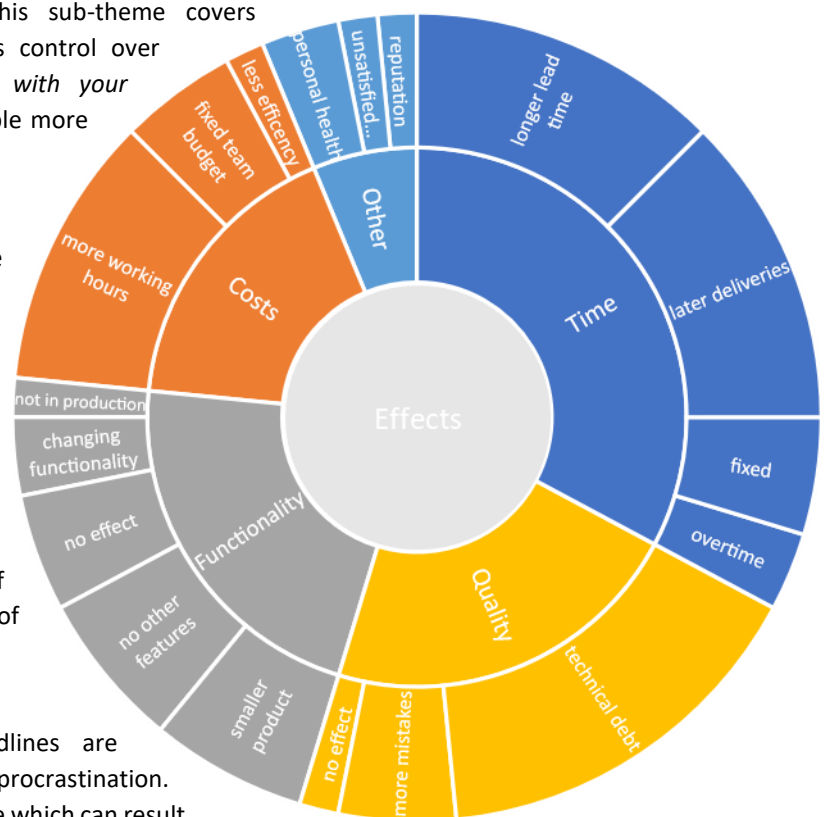


Figure 8: Overview of theme effects

The multitude of interviewees indicates that time and costs are related. If people spend more working hours on a product, this directly affects the costs. Procrastination could also lead to less efficiency, which could result in higher costs. Some interviewees argued that procrastination does not affect time and costs at all in Agile IT projects since the Agile team's budgets and timing are fixed.

4.4.2. Quality

More than half of the interviewees recognized technical debt as a quality effect of procrastination. *“If you take shortcuts to get something into production as quickly as possible and postpone a structural solution, you will eventually run into technical debt”* – interviewee 5. Some interviewees recognized that procrastination could result in rushing just before a deadline, resulting in more errors and less quality. Wu et al. (2014) also recognized this effect.

4.4.3. Functionality

Due to procrastination, there is *“less capacity to pick up other things”* – interviewee 6. Interviewees indicate that a smaller product with fewer features will be delivered if people procrastinate. Procrastination can extend the lead time of a project, which can pose a risk for changing functionality over time. If a team keeps putting off certain tasks, it may no longer be of value because the environment has changed. Changing environments is typical for Agile IT projects (Cervone, 2011). Postponing tasks until the end of a deadline can also lead to products that are not deployed in the production environment, which does not bring value for the stakeholders, even though this is considered important in Scrum.

4.4.4. Ranked

To prevent the interviewees from identifying the effects equally important, the interviewees ranked the effects in terms of time, costs, functionality, and quality. Sixteen of the eighteen interviewees gave a ranking. An average value was assigned if the interviewee gave a shared place to certain effects. There is a considerable variation in the ranking, as shown in Figure 9. According to the interviewees, procrastination has the most impact on the timing of a project, followed by costs. On average, interviewees rate the functionality and quality of a project as having less impact than the timing and costs.

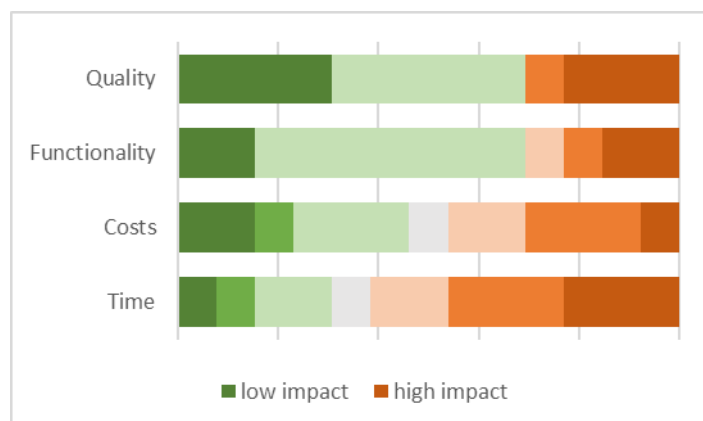


Figure 9: Impact of procrastination on quality, functionality, costs, and time

4.5. Countermeasures

This theme describes two different types of countermeasures of procrastination. Countermeasures that allow early detection of procrastination enable teams to take action to reduce procrastination and thus limit the impact of procrastination. Countermeasures could also aim to prevent a cause of procrastination and so reduce the risk of procrastination occurring. A complete overview of the sub-themes and codes is given in Figure 10. The following paragraphs will describe countermeasures of procrastination per sub-theme.

4.5.1. Feedback Mechanisms

Several feedback mechanisms can be used to lower the impact of procrastination. An important instrument is short development cycles, a typical characteristic of Agile. *“If you work cyclically, for example, in two weekly sprints, the impact of procrastination will never be very great”* – interviewee 6. According to most interviewees, executing Scrum ceremonies, like the daily stand-up, review, retrospective, and planning session should lead to less procrastination. Another feedback tool is actively monitoring by using metrics, for example team velocity, and making things visual, for example through a Scrum board or burndown chart. The typical Agile feedback loops mentioned in this paragraph aim to identify procrastination as quickly as possible and limit its impact. An

example related to detecting procrastination is presented by interviewee 13: *“If there are already two stand-ups talking about a 3-point user story, then you can signal that the velocity is not right and that someone is procrastinating”*.

4.5.2. Backlog and Priorities

Interviewee 7 indicates that they *“use a rule for teams that they can spend 20% of their time on maintenance, to avoid technical debt”*. This rule is called capacity allocation and ensures that all types of work are performed. This could be a percentage, but it could also be one sprint, for example, the innovation and planning sprint which originates from the SAFe method. Capacity allocation is used to prevent the risk of postponing technical debt, code refactoring, personal development, and the refinement of stories. As mentioned in Section 4.3.1., people prefer small, easy, and short tasks. Interviewee 4 mentioned *“breaking everything up into small items makes it easier to get started,”* making work more manageable and preventing procrastination from occurring. In

principle, if the backlog is transparent and the priorities are clear, *“whatever is at the top of the to-do list should also be pickup first”* - interviewee 15. Having a clear backlog can also prevent the incorrect estimation of workload and too much work in a sprint, two causes of procrastination. All these codes relate to the Agile way of working. Conversely, one interviewee gave a tip not to be transparent about the technical debt work. *“A little bit of hiding in things, so the business does not notice we are doing it because it is part of this thingy or that thingy. We are not making an epic of technical debt, but we will just fix it here and there to brush it off”* – interviewee 3.

4.5.3. Feedback Culture

In paragraph 4.4.1. several feedback mechanisms are mentioned. However, maybe equally important is a steady feedback culture. Eleven of the eighteen interviewees mentioned transparent communication in a way. This includes identifying procrastination, addressing if someone is procrastinating, discussing with the procrastinator, interacting and communicating with each other. Transparent communication creates a constant feedback loop where procrastination is identified in time. Therefore, the impact of procrastination can be restrained. Interviewee 13 thinks it is essential to *“discuss with each other that you run into something or that you notice that you are procrastinating. I think the main thing is that it is not talked about and only comes to light when it is too late”*. Transparency is one of the pillars in Scrum and matches this sub-theme very well. The constant feedback loop in a feedback culture will also enable people to learn and develop so that the personal causes of procrastination will be limited in the longer term and reduce the risk of procrastination occurring. This can be accomplished by strengthening the feedback culture where people feel comfortable speaking up. Interviewee 14 mentioned that *“it will also be a culture thing that you can say you do not have the knowledge”*. Interviewees also mentioned that people should be self-learning, so learn from their own mistakes instead of having the scrum master solve everything for them.

4.5.4. Team

This countermeasure relates to a mature, competent, and engaged Agile team. A competent and engaged product owner and scrum master who can coach the team will positively influence the risk of procrastination occurring and will decrease its impact. Another countermeasure of procrastination could be working together;

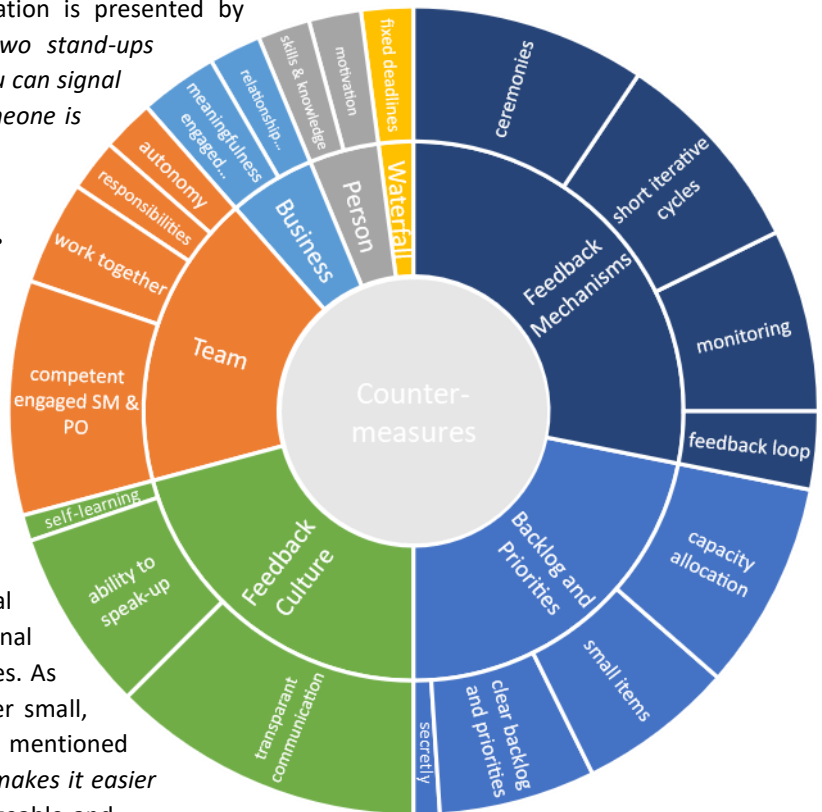


Figure 10: Overview of theme countermeasures

at the office, as buddies, as pairs, or in having code reviews. *“Procrastination is quickly spotted by code reviews”* – interviewee 17, so procrastination is noticed earlier, and measures can be taken to reduce the impact of procrastination. Improving a team’s autonomy and having clear responsibilities could, in the long run, ensure that a team does not procrastinate. Interviewee 3 explained: *“you empower people to do things themselves, and that ultimately ensures that they do not procrastinate”*.

4.5.5. Business

“Stakeholders who can clearly indicate how important the work is to them and how happy they are when they get deliveries, those are things that make people run fast” – interviewee 3. Engaged stakeholders and meaningful work can counteract the risk of procrastination as it reduces the causes of motivation failure, boring tasks, and lack of urgency. Improving stakeholder relationships can make people less likely to postpone tasks with stakeholder dependencies, like acceptance tests.

4.5.6. Person

A countermeasure that affects the person is to increase motivation so that the risk of procrastination occurring can be reduced. Interviewee 2 explains this as follows: *“I am convinced that if you put a procrastinator on a project that suits him very well, it can prevent that person from procrastinating”*. Additionally, increasing people’s skill set and knowledge could also prevent people from postponing difficult work, one of the causes of procrastination.

4.5.7. Waterfall

In contrast with Agile characteristics that could prevent procrastination, some interviewees indicate that waterfall methods are less noncommittal and that fixed deadlines can be beneficial for procrastination. However, this countermeasure is not widely supported.

4.6. Theoretical Saturation

In total there are four themes, twenty sub-themes, and 76 unique codes. Figure 11 provides an overview of the newly found codes per interview. Appendix 6 shows which codes were found in which interviews. In the first interviews, the most newest codes were found. The following interviews mostly confirmed the insights given in the first interviews. In the first 50% of the interviews, 92% of the codes were already found. This could indicate that there is theoretical saturation.

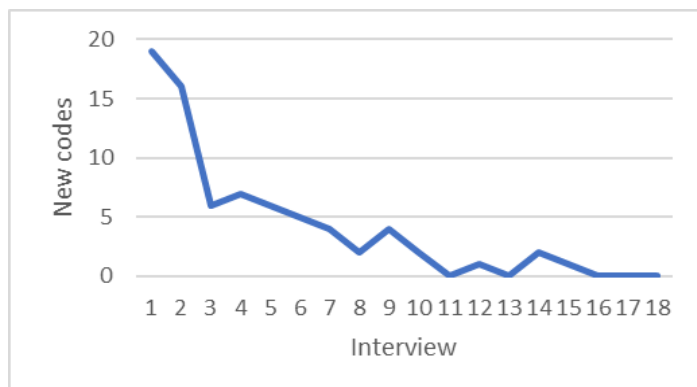


Figure 11: Saturation over interviews

4.7. Summary and Propositions

In this section, the results of this research will be summarized, and propositions will be generated. The propositions are based on the results of this chapter, numbered and bold.

The interviewees mentioned developers, scrum masters, and product owners as procrastinators of various types of work to the end of a sprint or PI. Despite much variation about the magnitude of the problem, almost all interviewees indicate that procrastination is a problem in Agile IT projects. As stated in Chapter 2, procrastination harms time, costs, functionality, and quality in (waterfall) projects (Chen et al., 2019; Steel, 2007; Wu et al., 2014). Procrastination in Agile IT projects can result in late deliveries, failure to meet deadlines, long lead times, increased costs, more mistakes, more technical debt, a smaller product, fewer features, and irrelevant features.

Proposition 1: Procrastination in Agile IT projects is widely spread and is a problem that has a negative influence on the timing, costs, quality, and functionality of the project, just as in (waterfall) projects.

Various causes of procrastination in Agile IT projects emerged in the interviews. For the sub-themes person, tasks, and environment, it is questionable whether it is a specific problem in Agile projects as they also occur in waterfall projects (McCrea et al., 2008; Steel, 2007).

Proposition 2: Task averseness, personal characteristics, business influence, team composition and maturity, Agile characteristics, and environment are causes of procrastination in Agile IT projects.

The impact of procrastination in Agile IT projects can be reduced by countermeasures that relate to Agile working. Feedback mechanisms, like ceremonies, short iterative cycles, and monitoring, help an Agile team detect procrastination early and allow the team to take action to reduce procrastination and thus limit the impact of procrastination. These feedback mechanisms are common in Agile projects. As described in Chapter 2, the expectation was that Scrum artifacts and iterative cycles would reduce the risk of procrastination occurring, but the interviewees indicate that these artifacts or mechanisms are more likely to reduce the impact of procrastination. Transparency is one of the pillars in Scrum (Cervone, 2011) and matches the countermeasure of an open feedback culture very well. Transparency in communication and the ability to speak up can ensure that procrastination can be identified quickly and that the impact of procrastination can be limited. An Agile team with an engaged and competent product owner and scrum master and where collaboration is the norm, procrastination will be noticed earlier, and measures can be taken to reduce the impact of procrastination.

Proposition 3: The Agile countermeasures feedback mechanisms, an open feedback culture, and a competent and mature Agile team make it possible to detect procrastination in Agile IT projects early so that actions can be taken to reduce the impact of procrastination.

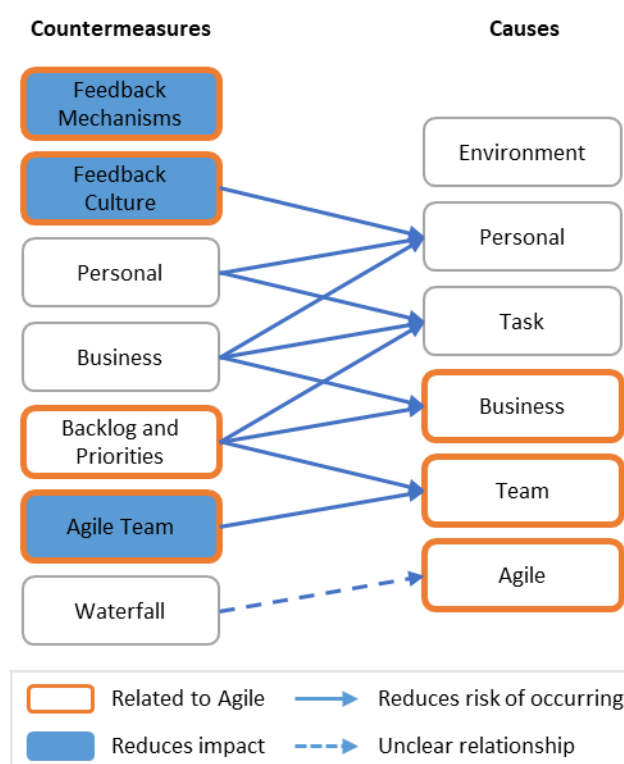


Figure 12: This figure shows (1) whether a countermeasure or cause is related to Agile and (2) how a countermeasure works against procrastination. A countermeasure can reduce the impact of procrastination by detecting procrastination in an early stage, or it can prevent a cause of procrastination and so reduce the risk of procrastination occurring.

The sub-themes feedback culture and a mature and engaged Agile team can reduce the impact of procrastination, but also prevent procrastination from occurring by mitigating personal causes of procrastination and unclear authority in the longer term. A clear backlog and transparent priorities relate to Agile and allow capacity allocation, breaking up into small and clear items and pick-up items at the top of the backlog. These countermeasures will prevent task, business, and team-related causes and reduce the risk of procrastination. Countermeasures related to business and personal characteristics, like meaningfulness, improving stakeholder relationships, increasing personal motivation, and improving skills and knowledge, are not explicitly related to the Agile way of working but could prevent procrastination in Agile IT projects. Figure 12 shows whether a countermeasure or cause is related to Agile and how a countermeasure works against procrastination.

Proposition 4: The countermeasures feedback culture, a clear backlog and transparent priorities, competent and mature Agile team, business stakeholders, and personal characteristics can prevent procrastination in Agile IT projects from occurring as it prevents causes of procrastination.

Agile projects can be more prone to technical debt because the stories with direct business value often have higher priority. Thereby, less visible stories like refactoring can easily be postponed to the next sprint because “it does not really fit right now” – interviewee 6. On the one hand, interviewees indicate that resolving technical debt is postponed more often, but deferring tasks that should prevent technical debt, such as refactoring and documentation, also leads to more technical debt and thus lower the quality of the product or service. Capacity allocation and clear priorities could reduce the risk of postponing resolving and preventing technical debt. In Figure 13, the relations between procrastination, technical debt, and capacity allocation are visualized.

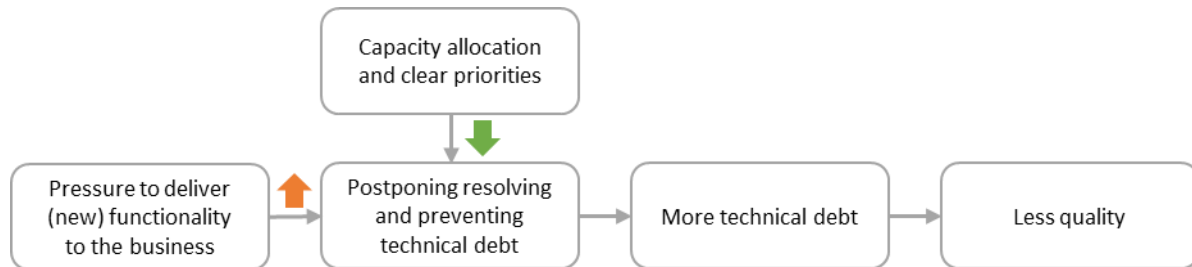


Figure 13: Relations between pressure to deliver, technical debt, quality, and capacity allocation

Proposition 5: Postponing resolving and preventing technical debt is more common in Agile IT projects because of the high pressure to deliver functionality, leading to even more technical debt. Capacity allocation and clear priorities could mitigate this.

Some interviewees mentioned that Agile characteristics like high flexibility and plenty of freedom could cause procrastination in Agile IT projects. Thereby, a few interviewees indicated that waterfall methods are less noncommittal and that fixed deadlines can be beneficial for procrastination. However, there is not much support for this. Most interviewees indicate that Agile characteristics reduce the impact of procrastination and its risk of occurring.

Proposition 6: The Agile way of working reduces the impact of procrastination and its risk of occurring in Agile IT projects.

5. Discussion, Conclusions, and Recommendations

This final chapter contains a discussion of the outcomes of this research. The first section will reflect on the results, the research process, and the research method. In Section 5.2, the research conclusion will be given by answering the research (sub-)question(s). Thirdly, the recommendations for practice will be given. Section 5.4. will give recommendations for further research based on the limitations and results of this research. Finally, the main conclusions are summarized in Section 5.5.

5.1. Discussion and Reflection

5.1.1. Reflection on the Results

In this section, the expectations derived from theory will be discussed, interpreted, and compared with the empirical findings of this research and contribute to the existing scientific knowledge. Project escalation is a major issue in IT projects and has been occurring for decades (Keil & Mann, 1997; Project Management Institute, 2018; The Standish Group International, 1995, 2015, 2020). Cognitive biases, like procrastination, can cause project escalation (Keil, 1995). Project success of waterfall projects can be harmed by procrastination due to much focus on details, long phases, and the heavily weighted method (Chen et al., 2019; Great Britain. Office of Government Commerce, 2002; Steel, 2007; Wu et al., 2014). The interviews results show that procrastination is also a widely spread problem in IT projects that work with the relatively new and considerably different Agile method. Procrastination in Agile IT projects negatively influences the timing, costs, quality, and functionality of the project. This research did not clarify whether procrastination occurs more often in waterfall or Agile projects.

Interviewees most often mentioned resolving or preventing technical debt as postponed work in Agile IT projects. So procrastination could result in the accumulation of technical debt. For example, delaying code

refactoring or postponing documentation can lead to major quality issues. Procrastination in Agile IT projects can also not be technical debt related. Postponing development work does not directly lead to technical debt but rather to delivering less functionality. Procrastination and technical debt have similarities as they both originate due to overvaluing short-term benefits (Rios et al., 2018; Wu et al., 2014). This makes them both susceptible to similar causes. The pressure to deliver (new) functionality to the business fast and frequently is one of the causes of procrastination and the accumulation of technical debt in Agile IT projects. This is in line with the conclusion of Kruchten et al. (2012), which argues that technical debt can accumulate much faster in Agile projects because most Agile projects deliver products and services frequently in a fluid and fast-changing environment. According to Sleesman et al. (2018), this dynamic and fast-changing environment is typical for IT projects and makes them prone to biases. In addition, it can be noted that focusing on short successes instead of solving technical debt for the long term corresponds to the construal-level theory of McCrea et al. (2008), where events far in time tend to be more abstractly represented than events close in time. This is also one of the causes of procrastination. In summary, procrastination can lead to the accumulation of technical debt, and they can have similar causes.

The literature and interviews show that task averseness and personal characteristics, like motivation failure, focusing too much on details, and insecurity, are causes of procrastination that arise in both waterfall and Agile IT projects (McCrea et al., 2008; Steel, 2007). Additional causes of procrastination in Agile IT projects that were not reflected in the literature but derived from the interviews are related to team composition and team maturity. According to the literature, task averseness occurs in IT projects that lack the novelty of discovery-oriented projects (Wu et al., 2014). Therefore, it is crucial to pay attention and recognize task averseness in (Agile) IT projects. Korzaan and Morris (2009) described that behavioral issues are common causes of problems in software development. This can be confirmed by the fact that personal-related issues are one of the most cited causes of procrastination in Agile IT projects.

The interview results show that Agile feedback mechanisms like short iterative cycles, scrum artifacts, and scrum activities are important countermeasures of procrastination. This is in line with the expectations of the theoretical framework and relates to intermediate deadlines mentioned by Ariely and Wertenbroch (2002). An addition to the body of knowledge is the insight that feedback mechanisms, an open feedback culture, and an engaged and mature Agile team will aim to notice procrastination early so that measures can be taken to reduce the impact of procrastination. A fourth Agile countermeasure to procrastination is a clear backlog and transparent priorities. This is in line with the expectations that a concrete sprint backlog act as a countermeasure against procrastination in Agile IT projects. A concrete addition to the body of knowledge and practical very relevant is the fact that capacity allocation and clear priorities could reduce the risk of postponing resolving technical debt and postponing tasks to prevent technical debt. Although technical debt occurs more often in Agile projects (Kruchten et al., 2012), this countermeasure is a powerful argument for working in an Agile way. Capacity allocation and clear priorities are precious in the dynamic and rapidly changing environment typical of IT projects (Rios et al., 2018). The interviews do not reflect the expected countermeasure of concrete subgoals.

Additions to the existing knowledge of preventing procrastination are business and personal related countermeasures, like meaningfulness, improving stakeholder relationships, increasing personal motivation, and improving skills and knowledge. These countermeasures are not explicitly related to the Agile way of working but could prevent procrastination in Agile IT projects. Schedule pressure, diverse project teams, and team competition are countermeasures in (waterfall) projects (Van Oorschot et al., 2018; Wu et al., 2014) but are not reflected by the interviewees in Agile IT projects.

5.1.2. Reflection of the Process and the Method

This explorative research is based on literature research and semi-structured interviews. The overall research and interview process went smoothly. The interviewees asked all questions from the interview protocol, sometimes deviated from the proposed order, and asked additional questions. Although the aim was to interview experts with at least two years of experience working with Agile, one of the participants had only one and half years of Agile experience. Although it had no significant consequences for this study, better screening could have prevented it. Due to timing and practical constraints, the results are not sent back to the participants

to confirm their accuracy. This can be seen as a limitation of the research. All information of the interviewees is anonymized. None of the interviewees made use of the possibility to withdraw.

A coding scheme is set up based on the thematic analysis to enhance reproducibility, increase transparency, and reduce the risk of subjectivity. Despite countering researcher bias by creating an interview protocol, transcribing the interviews, and sharing information about the bias, it cannot be entirely prevented that one interviewer may ask more questions about his/her research and bias. By collaborating with other interviewers, interviewees from three different organizations could be interviewed. Despite these organizations being in different sectors, they are all Dutch organizations. This may reduce the international generalizability. The spoken language during the interviews was Dutch, and so were the transcripts of the interviews. This can pose interpretation issues in translating the transcripts.

Cognitive biases, like procrastination, are systematic human errors and might be subconscious (Keil et al., 2007). So it is questionable whether interviews are the best method because the interviewee may not be aware of procrastination. Although most research about cognitive biases is quantitative, the qualitative approach in this study made it possible to explore the role of procrastination in Agile IT projects, which is the objective of this research. The semi-structured interviews made it possible to structure the interview questions somewhat to allow multiple interviewers to conduct the interviews but still be flexible enough to be explorative. Also, Nuijten et al. (2020) used interviews in their exploratory research about cognitive biases. On the other hand, interviewees indicate that it is difficult to estimate what percentage procrastination is a problem. This can be seen as a limitation and argument for conducting quantitative research in the future. See also Section 5.4 for recommendations for further research.

5.2. Conclusions

This study has the following problem statement: not much is known about the role of procrastination in Agile IT projects, which might lead to a blind spot in project management and could be a risk in recognizing the bias and preventing or reducing the impact of procrastination. The objective of this research is to get a better understanding of the role of procrastination in Agile IT projects. This paragraph will summarize the outcomes of this research and answer the five sub-questions formulated in Chapter 1. Answering these sub-questions will answer the main research question: What role does procrastination play in Agile IT projects?

1. How does procrastination manifest itself in Agile IT projects?

Procrastination is a problem in Agile IT projects and manifests itself across different Agile roles and various types of work. Postponing resolving and preventing technical debt is more common in Agile IT projects.

2. What are the causes of procrastination in Agile IT projects?

An essential cause of procrastination is the high pressure of the business to deliver functionality. Personal characteristics, nature of the task, business influence, team composition and maturity, Agile characteristics, and environment are other causes of procrastination in Agile IT projects.

3. What effect can procrastination have on time, costs, functionality, and quality in Agile IT projects?

Procrastination may only harm the quality and functionality of a project since the timing and budget of an Agile IT project are fixed. Nevertheless, this study shows that procrastination can negatively influence the timing, costs, quality, and functionality of an Agile IT project. Moreover, the accumulation of technical debt is more likely to emerge as a negative quality effect of procrastination in Agile IT projects.

4. What are possible countermeasures against procrastination in Agile IT projects?

A clear backlog and transparent priorities could reduce the risk of procrastination occurring and reduce the risk of technical debt in Agile IT projects. The Agile countermeasures feedback mechanisms, an open feedback culture, and a competent and mature Agile team enable to detect procrastination in Agile IT projects at an early stage so that actions can be taken to reduce the impact of procrastination. In addition, the Agile countermeasures feedback culture, a competent and mature Agile team, and the more general

countermeasures related to business stakeholders and the person could prevent causes of procrastination and thus prevent the risk of procrastination occurring in Agile IT projects.

5. What is the difference between the role of procrastination in Agile IT projects compared to traditional waterfall IT projects?

Procrastination manifests itself in both waterfall and Agile IT projects. This study does not clarify in which method procrastination is a more significant problem. Personal characteristics and task-related issues cause procrastination in traditional waterfall and Agile IT projects. Postponing and preventing technical debt is more common in Agile IT projects because of the high pressure to deliver functionality to the business. Schedule pressure, diverse project teams, and team competition are countermeasures in general projects but are not recognized in Agile IT projects. Although some interviewees mention that Agile characteristics could cause procrastination and waterfall methods can be beneficial for procrastination, most interviewees indicate that Agile-related countermeasures reduce the impact of procrastination and the risk of it occurring.

5.3. Recommendations for Practice

An increasing amount of IT projects are managed by Agile project management to prevent them from failing (Holz, 2019), which is a common issue (Johnson & Mulder, 2020). Procrastination is a cognitive bias that can negatively affect project success. Clarity on the role of procrastination in Agile IT projects could adapt the way of working in Agile IT teams to deal with procrastination to make IT projects successful. In addition, technical debt is more common in Agile projects (Kruchten et al., 2012) so understanding countermeasures is relevant for practice.

This research comes with several practical recommendations. Firstly, a good understanding of the causes of procrastination is needed to recognize procrastination in a project team or department. The most critical causes relate to task averseness, personal characteristics, and business influence. Secondly, a clear backlog with transparent priorities will mitigate the risk of technical debt. Especially capacity allocation, a method to allocate resources among each type of work, is precious in Agile IT projects for preventing and solving technical debt, personal development, and refinements. Thirdly, countermeasures against procrastination that relate to Agile can prevent procrastination and lower its impact. Having an open feedback culture, a mature and competent Agile team, and a clear backlog and priorities can prevent causes of procrastination and so reduce the risk of procrastination occurring. It is a myth that procrastination can be prevented entirely, so it is crucial to detect procrastination as soon as possible to be able to act upon it. Feedback mechanisms, an open feedback culture, and a mature and competent Agile team all focus on detecting procrastination in an early stage and thereby limiting the impact of procrastination. These countermeasures are typically for the Agile way of working and, therefore another solid argument for adopting the Agile way of working. A fourth recommendation is to adopt personal and business-related countermeasures, namely meaningfulness, improving stakeholder relationships, increasing personal motivation, and improving skills and knowledge. Although these countermeasures do not directly relate to Agile, they could mitigate the causes of procrastination in Agile IT projects and reduce the risk of procrastination.

In summary, this research suggests that working with Agile project management methods can reduce the risk of procrastination occurring and reduce the impact of procrastination in Agile IT projects. For teams that already work Agile, it would be good practice to take a closer look at the pressure from the business, personal development, team composition, backlog and priorities, and feedback loops in the team. For IT teams struggling with procrastination, it would be worthwhile to work Agile and pay particular attention to feedback mechanisms, clear backlog and transparent priorities, feedback culture, and a competent and engaged Agile team.

5.4. Recommendations for Further Research

Based on the limitations of this research, several recommendations for further research will be given. Firstly, only participants from Dutch organizations were interviewed. The Netherlands has one of the most direct and open business cultures globally (Meyer, 2014). This study shows that an open feedback culture is one of the most important countermeasures of procrastination. To minimize the risk that these results only apply to the specific business culture of The Netherlands, it is recommended to do a follow-up study in several countries and

sectors with various business cultures. The Agile maturity of the interviewees and their teams is not included in this study. One of the practical recommendations is to adopt the Agile way of working fully. The question is to what extent the findings of the interviewees are reliable if the interviewees' teams do not fully work Agile. Future research should shed more light on the relationship between the Agile maturity of an IT team and procrastination.

Two closed questions were asked during the interview to understand better how big the problem is and what effect has the most impact on procrastination. The outcomes of these questions are only descriptive and are not statistically tested. In addition, procrastination could be subconscious, so people might not be aware of their level of procrastination. Moreover, interviewees indicate that it is difficult to estimate what percentage procrastination is a problem. To sum up, interviews might not be the perfect method to measure procrastination quantitatively. Although interviews made it possible to explore the role of procrastination in Agile IT projects (the goal of this research), it is also recommended to design quantitative research to test the assumptions found in this research on a bigger sample group. This could be done by (verified) questionnaires, like the General Procrastination Scale (GPS), see Appendix 7. GPS is a scale often used to measure procrastination and can also be used in non-academic and non-student contexts (Klein et al., 2017; Sirois, Yang, & van Eerde, 2019).

This study resulted in several interesting findings that warrant further research. Some interviewees indicate that Agile characteristics could cause procrastination, and waterfall methods could be beneficial for procrastination. However, most interviewees indicate that Agile characteristics reduce the impact of procrastination and the risk of it occurring. Further research is required to understand this apparent contradiction. Procrastination can result to technical debt. In addition, procrastination and technical debt can have similar causes. Since technical debt is more common in complex environments, such as Agile IT projects, it would be interesting to study the exact relationship between procrastination and technical debt in more detail. This is relevant to determine whether causes, effects, and countermeasures are related to procrastination, technical debt, or both. Most interviewees indicate that the impact of procrastination is most significant on the timing and costs of a project. Nevertheless, some interviewees state that procrastination has no impact on the timing and costs because these factors are fixed in Agile projects. Further research should explore the relationship between Agile teams and the timing and costs of a team. This research does not provide insight into whether procrastination occurs more often in waterfall projects or Agile projects. A quantitative study can provide a definite answer to this. Lastly, some interviewees indicate that working from home could negatively influence procrastination in Agile IT projects. Not much is known about procrastination in combination with working from and this research did not focus on this topic either. New research could focus specifically on procrastination in Agile IT projects that work (partly) from home.

5.5. Main Conclusions

Procrastination is a widely spread problem in Agile IT projects and negatively impacts the timing, costs, quality, and functionality, just as with waterfall projects. Postponing and preventing technical debt is more common in Agile IT projects, because of the high pressure of the business to deliver functionality. Capacity allocation can mitigate this. Understanding the causes of procrastination (task averseness, personal characteristics, and business influences) is relevant to recognizing and acting upon procrastination. Agile feedback mechanisms, open feedback culture, and an engaged and mature Agile team enable detecting procrastination early and limiting the impact. This research suggests that Agile project management methods can counteract procrastination in IT projects.

References

- Ariely, D., & Wertenbroch, K. (2002). Procrastination, deadlines, and performance: Self-control by precommitment. *Psychological Science*, 13(3), 219-224.
- Battoia, N. (2019). Innovation in Agile and Waterfall project management: a qualitative analysis.
- Benschop, N., Nuijten, A. L., Keil, M., Rohde, K. I., Lee, J. S., & Commandeur, H. R. (2020). Construal level theory and escalation of commitment. *Theory Decision Sciences*, 1-17.
- Cadena, X., Schoar, A., Cristea, A., & Delgado-Medrano, H. M. (2011). *Fighting procrastination in the workplace: An experiment*. Retrieved from
- Cervone, H. F. (2011). Understanding agile project management methods using Scrum. *OCLC Systems Services: International digital library perspectives*, 27, 18-22. doi:10.1108/10650751111106528
- Chen, Z., Lan, Y., Zhao, R., & Shang, C. (2019). Deadline-based incentive contracts in project management with cost salience. *Fuzzy Optimization Decision Making*, 18(4), 451-473.
- Cuellar, M. J., Keil, M., & Johnson, R. D. (2006). The deaf effect response to bad news reporting in information systems projects. *e-Service Journal*, 5(1), 75-97.
- Great Britain. Office of Government Commerce. (2002). *Managing Successful Projects with PRINCE2*: Stationery Office.
- Holz, B. (2019). Results Summary: Agile in the Enterprise. [https://circle.gartner.com/Portals/2/Resources/pdf/Agile%20in%20the%20Enterprise%202019%20-%20Results%20Summary%20\(updated\).pdf](https://circle.gartner.com/Portals/2/Resources/pdf/Agile%20in%20the%20Enterprise%202019%20-%20Results%20Summary%20(updated).pdf)
- Jani, A. (2008). An experimental investigation of factors influencing perceived control over a failing IT project. *International Journal of Project Management*, 26(7), 726-732.
- Janis, I. L. (2008). Groupthink. *IEEE Engineering Management Review*, 36(1), 36.
- Johnson, J., & Mulder, H. (2020). Endless Modernization.
- Keil, M. (1995). Pulling the plug: Software project management and the problem of project escalation. *MIS quarterly*, 421-447.
- Keil, M., Depledge, G., & Rai, A. (2007). Escalation: The role of problem recognition and cognitive bias. *Decision Sciences*, 38(3), 391-421.
- Keil, M., & Mann, J. (1997). *Understanding the nature and extent of IS project escalation: Results from a survey of IS audit and control professionals*. Paper presented at the Proceedings of the Thirtieth Hawaii International Conference on System Sciences.
- Klein, E. M., Beutel, M. E., Müller, K. W., Wölfling, K., Brähler, E., & Zenger, M. (2017). Assessing procrastination. *European Journal of Psychological Assessment*.
- Korzaan, M., & Morris, S. A. (2009). Individual characteristics and the intention to continue project escalation. *Computers in human behavior*, 25(6), 1320-1330.
- Kruchten, P., Nord, R. L., & Ozkaya, I. (2012). Technical debt: From metaphor to theory and practice. *Ieee software*, 29(6), 18-21.
- McCrea, S. M., Liberman, N., Trope, Y., & Sherman, S. J. (2008). Construal level and procrastination. *Psychological Science*, 19(12), 1308-1314.
- Meyer, E. (2014). *The Culture Map*. New York: PublicAffairs.
- Nuijten, A., Benschop, N., Rijsenbilt, A., & Wilmink, K. (2020). Cognitive Biases in Critical Decisions Facing SME Entrepreneurs: An External Accountants' Perspective. *Administrative Sciences*, 10(4), 89.
- Nuijten, A., Keil, M., & Commandeur, H. (2016). Collaborative partner or opponent: How the messenger influences the deaf effect in IT projects. *European Journal of Information Systems*, 25(6), 534-552.
- Park, C., Im, G., & Keil, M. (2008). Overcoming the mum effect in IT project reporting: Impacts of fault responsibility and time urgency. *Journal of the Association for Information Systems*, 9(7), 1.
- Pawar, R., & Mahajan, K. (2017). Benefits and issues in managing project by PRINCE2 Methodology. *International Journal Of Advanced Research In Computer Science Software Engineering*, 7(3), 190-195.

- Project Management Institute. (2018). Success in Disruptive Times; Expanding the Value Delivery Landscape to Address the High Cost of Low Performance. <https://www.pmi.org/-/media/pmi/documents/public/pdf/learning/thought-leadership/pulse/pulse-of-the-profession-2018.pdf>
- Rios, N., de Mendonça Neto, M. G., & Spínola, R. O. (2018). A tertiary study on technical debt: Types, management strategies, research trends, and base information for practitioners. *Information Software Technology, 102*, 117-145.
- Rozental, A., Bennett, S., Forsström, D., Ebert, D. D., Shafran, R., Andersson, G., & Carlbring, P. (2018). Targeting procrastination using psychological treatments: A systematic review and meta-analysis. *Frontiers in psychology, 9*, 1588.
- Rozental, A., & Carlbring, P. (2014). Understanding and treating procrastination: A review of a common self-regulatory failure. *Psychology, 5*(13), 1488.
- Sabherwal, R., Sein, M. K., & Marakas, G. M. (2003). Escalating commitment to information system projects: findings from two simulated experiments. *Information Management, 40*(8), 781-798.
- Saunders, M. N. K., Lewis, P., & Thornhill, A. (2019). *Research methods for business students*. Harlow, United Kingdom: Pearson.
- Sirois, F. M., Yang, S., & van Eerde, W. (2019). Development and validation of the General Procrastination Scale (GPS-9): A short and reliable measure of trait procrastination. *Personality individual differences, 146*, 26-33.
- Sleesman, D. J., Lennard, A. C., McNamara, G., & Conlon, D. E. (2018). Putting escalation of commitment in context: A multilevel review and analysis. *Academy of Management Annals, 12*(1), 178-207.
- Sneed, H. M. (2014). *Dealing with Technical Debt in agile development projects*. Paper presented at the International Conference on Software Quality.
- Steel, P. (2007). The nature of procrastination: a meta-analytic and theoretical review of quintessential self-regulatory failure. *Psychological bulletin, 133*(1), 65.
- The Standish Group International. (1995). The CHAOS report.
- The Standish Group International. (2015). CHAOS Report 2015. https://www.standishgroup.com/sample_research_files/CHAOSReport2015-Final.pdf
- The Standish Group International. (2020). CHAOS 2020 Beyond Infinity.
- The University of Sheffield. (2019). GPS-9-2019. https://sheffieldpsychology.eu.qualtrics.com/jfe/form/SV_0Ukzjib70klHw5D
- Turetken, O., Stojanov, I., & Trienekens, J. J. (2017). Assessing the adoption level of scaled agile development: a maturity model for Scaled Agile Framework. *Journal of Software: Evolution process, 29*(6), e1796.
- Van Oorschot, K. E., Sengupta, K., & Van Wassenhove, L. N. (2018). Under pressure: The effects of iteration lengths on agile software development performance. *Project Management Journal, 49*(6), 78-102.
- Watson, D. C. (2001). Procrastination and the five-factor model: A facet level analysis. *Personality individual differences, 30*(1), 149-158.
- Wu, Y., Ramachandran, K., & Krishnan, V. (2014). Managing cost salience and procrastination in projects: Compensation and team composition. *Production Operations Management, 23*(8), 1299-1311.

Appendix 1: Interview Protocol

Protocol interviewees

Introduction

You are invited to participate in an interview about cognitive biases in Agile IT projects. This interview is part of three studies conducted by students of the Open University in the Netherlands. Three topics will be discussed in the interview. It concerns the role of procrastination, the role of the mum-effect, and the role of self-efficacy in Agile IT projects. The definitions of these cognitive biases and the questions that will be asked during the interview are included in the appendix. Below are some preparation questions that I would like to receive an answer from you via email. In addition, I would like to inform you that all information from the interview will be anonymized. No judgment is made on the projects or products you are working on. It's about your experiences with cognitive biases in Agile IT projects, regardless of the company or project you are currently working on. This interview is voluntary, you can withdraw at any time.

Preparation questions

As a preparation for the interview, could you answer the following questions via mail?

1. What is your current role in Agile IT projects? And what other Agile roles did you fulfilled in the past?
2. How many years of experience do you have working in Agile IT projects? And how many years of experience do you have working with traditional waterfall methods like PRINCE2?
3. Which Agile method do you currently work with? And which Agile methods have you worked with before?
4. To what extent do you think your team/organization works Agile? Think of:
 - a. Iterative cycles with continuous feedback
 - b. Short-cycle deliveries
 - c. Scrum events (such as day start, refinement, sprint planning, sprint review, retrospective)
 - d. Prototyping, for example, by using a minimum viable product (MVP)

Definitions

Procrastination

Procrastination refers to the tendency of people to delay the beginning or completion of an intended task, despite the expectation that they will be worse off because of the delay. Procrastination occurs, for example, with developers who consistently put off most of their work until deadlines are approaching. Procrastination also occurs when teams (or product owners) push issues forward, leading to technical debt. Technical debt is the phenomenon whereby people focus on short-term solutions and postpone complex (technical) issues, which results in extra work later in the development process. Procrastination can affect projects' timing, costs, functionality, and quality.

The mum-effect

The mum effect in this study concerns the failure to report bad news about the status of (the activities in) the IT project, with the possible consequence that the project ultimately escalates (overruns in planning, goes over budget, deviates from the scope, the product is not of the desired quality and/or does not offer the desired functionalities) or even completely fails. The main question of this research is: how does the mum effect manifest itself in Agile IT projects?

Self-efficacy

Everyone has some degree of self-efficacy, it's not a negative trait. It is your belief in your own abilities how well you can perform a particular task or action. The level of your self-efficacy can increase and decrease, this is due to experiences from previous tasks in the past. High self-efficacy can result in overconfidence (high self-efficacy) or self-underestimation (low self-efficacy). This can create a distorted picture of reality, which we call bias. Biases can lead to erroneous decisions.

Feedback is received within Agile projects. For example, estimates of how much time work costs. In retrospect, the estimates can be just right, too low, or too high. Feedback can also follow on work performed, for example, feedback is that the work delivered meets expectations, or is below or above expectations.

Interview questions

Below you can find the questions that will be asked during the interview.

Procrastination

1. Have you observed procrastination in Agile IT projects? What did this look like, and how did you notice this?
2. What do you see as causes of procrastination in Agile IT projects?
3. What could be the impact of procrastination on costs, time, functionality, and quality of an Agile IT project?
 - a) Could you compare the impact of procrastination on costs, time, functionality, and quality? In other words, which impact is the greatest, the second-largest, etc.?
4. What do you see as possible countermeasures against procrastination in Agile IT projects to either prevent it or reduce its impact?
5. Which typical Agile characteristics can prevent/decrease or cause/increase the impact of procrastination?
6. How do you see the relation between procrastination and technical debt?
7. What percentage of Agile IT projects do you think procrastination is a problem? (0-100%)

The mum-effect

1. What is your experience with the mum effect in Agile IT projects?
2. What consequences do you know or can you imagine caused by the mum-effect in Agile IT projects?
3. On a scale of 1 to 5, how big was the impact on (or can you imagine the impact on) 1) the finances, 2) the planning, 3) the scope (functionality), and 4) the quality?
 - a. Can you also rank the consequences from largest to smallest?
4. What causes of the mum effect in Agile IT projects do you know or can you imagine?
5. What countermeasures do you know or can you imagine that can be used against the mum effect in Agile IT projects?
6. Which Agile characteristics do you know that (possibly) play a role in how the mum-effect manifests itself in Agile IT projects?
7. In what percentage of Agile IT projects do you think the mum effect occurs?

Self-efficacy

1. Do you know of an example of a situation where within an Agile team, or with an individual, there was overconfidence or self-underestimation in relation to the work/tasks that had to be performed?
 - a. What do you think was the cause of this?
 - b. Were there any negative impacts on time, budget, quality, and functionality?
 - i. Can you rank these consequences from high impact (1) to low impact (4)?
 - c. Have countermeasures been taken to prevent this from happening in the future?
2. Do you know of an example of a situation within an Agile IT project where feedback influenced the belief in one's own capabilities (trust)?
 - a. Have you experienced that in the event of positive or negative feedback, mistakes were made afterward, for example, due to overconfidence or self-underestimation?
3. Suppose there is overconfidence or self-underestimation within the Agile team, what do you think are measures within Agile that can prevent errors as a result?
 - a. Do you have suggestions for new additional measures?
4. What typical Agile characteristics do you think can cause overconfidence or self-underestimation?
5. In what percentage of IT projects do you think that self-underestimation or overconfidence occurs (0-100%)?

Extra information interviewers

This research is part of a broader research, and questions regarding other biases will also be included in the interviews. Three different researchers will each interview six experts in the field about three biases. The following information and Table 2 will be sent to the researchers so that they have a broader sense of the bias procrastination.

Procrastination is a complex bias where people voluntarily delay the beginning or completion of an intended action, despite the expectation that they will be worse off because of the delay. Procrastination can occur with developers who consistently put off most of their work until deadlines are approaching. Procrastination can also occur when teams (or product owners) push issues forward, leading to technical debt. Procrastination can affect projects' timing, costs, functionality, and quality.

The expectation is that the role of procrastination is different in waterfall projects than in Agile projects. Waterfall projects are heavily weighted, focus much on details, and have lengthy phases, resulting in a higher risk for procrastination and negatively influencing the timeliness, costs, quality, and functionality of a project. Conversely, Agile projects have short iterative cycles, concrete subgoals, and use Scrum artifacts and Scrum activities, reducing the cause of procrastination and reducing the risk of procrastination occurring. On the other hand, technical debt occurs more often in Agile projects than in waterfall projects, resulting in delays, cost overruns, quality issues, and less functionality. In Table 2 the effects, causes, and countermeasures of procrastination in (waterfall) projects and Agile projects are summarized.

Appendix 2: Interview Protocol (Dutch)

Protocol geïnterviewden

Introductie

U bent uitgenodigd om deel te nemen aan een interview over cognitieve biases in Agile IT-projecten. Dit interview maakt deel uit van drie onderzoeken die worden uitgevoerd door studenten van de Open Universiteit. In het gesprek komen drie onderwerpen aan de orde. Het betreft de rol van uitstelgedrag, de rol van het mum-effect en de rol van self-efficacy in Agile IT-projecten. De definities van deze cognitieve biases en de vragen die tijdens het interview zullen worden gesteld zijn bijgevoegd in de bijlage. Hieronder staan een aantal voorbereidende vragen die ik graag van u beantwoord via de mail terug zou willen ontvangen. Daarnaast wil ik u mededelen dat alle informatie uit het interview wordt geanonimiseerd. Er wordt geen oordeel geveld over de projecten of producten waaraan u werkt. Het gaat om uw ervaringen met cognitieve biases in Agile IT-projecten, dit is onafhankelijk van het bedrijf of het project waar u nu werkt. Dit interview is vrijwillig, u kunt zich altijd terugtrekken.

Vorbereidende vragen

Zou u ter voorbereiding op het interview de volgende vragen via de mail kunnen beantwoorden?

1. Wat is uw huidige rol in Agile IT-projecten? En welke andere Agile rollen heeft u in het verleden vervuld?
2. Hoeveel jaar ervaring heeft u met het werken in Agile IT-projecten? En hoeveel jaar ervaring heeft u met het werken met traditionele watervalmethodes zoals PRINCE2?
3. Met welke Agile methode werkt u op dit moment? En met welke Agile methoden heeft u al eerder gewerkt?
4. In hoeverre vindt u dat uw team/organisatie Agile werkt? Denk aan:
 - a. Iteratieve cycli met continue feedback
 - b. Kort-cyclische opleveringen
 - c. Scrum events (zoals dagstart, refinement, sprintplanning, sprintreview, retrospective)
 - d. Prototyping, bijvoorbeeld door gebruik te maken van een minimal viable product (MVP)

Definities

Uitstelgedrag

Uitstelgedrag verwijst naar de neiging van mensen om het begin of de voltooiing van een voorgenomen actie uit te stellen, ondanks de verwachting dat ze slechter af zullen zijn vanwege de vertraging. Uitstelgedrag treedt bijvoorbeeld op bij ontwikkelaars die consequent het grootste deel van hun werk uitstellen tot de deadlines naderen. Uitstelgedrag treedt ook op wanneer teams (of product owners) problemen voor zich uit schuiven, wat kan leiden tot technical debt. Technical debt is het fenomeen waarbij men focust op korte termijn oplossingen en moeilijke (technische) vraagstukken uitstelt, wat later in het ontwikkelingsproces resulteert in extra werkzaamheden. Uitstelgedrag kan de timing, kosten, functionaliteit en kwaliteit van projecten beïnvloeden.

Het mum-effect

Het mum-effect betreft in dit onderzoek het niet melden van slecht nieuws over de status van (de werkzaamheden in) het IT-project, met als mogelijk gevolg dat het project uiteindelijk escaleert (uitloopt in planning, over het budget gaat, afwijkt van de scope, het product niet van gewenste kwaliteit is en/of niet de gewenste functionaliteiten biedt) of zelfs volledig faalt. De hoofdvraag van dit onderzoek is: hoe manifesteert het mum-effect zich in Agile IT-projecten?

Self-efficacy

Iedereen heeft een bepaalde mate van self-efficacy, het is geen negatieve eigenschap. Het is jouw geloof in je eigen capaciteiten hoe goed je een bepaalde taak of actie kan uitvoeren. De hoogte van jouw self-efficacy kan toenemen en afnemen, dit komt door ervaringen (feedback) van eerdere taken uit het verleden. Een hoge self-efficacy kan resulteren in overmoedigheid (veel self-efficacy) of zelfonderschatting (weinig self-efficacy).

Hierdoor kan een vertekend beeld van de werkelijkheid ontstaan, dit noemen we bias. Biases kunnen leiden tot foutieve beslissingen.

Binnen Agile-projecten wordt feedback ontvangen. Bijvoorbeeld inschattingen van hoeveel tijd werkzaamheden kosten. De inschattingen kunnen achteraf precies goed, te laag, of te hoog zijn. Ook kan er feedback volgen op geleverde werkzaamheden, feedback is bijvoorbeeld dat het geleverde werk aan de verwachtingen voldoet, of beneden of boven verwachting is.

Interview vragen

Hieronder vind u de vragen die tijdens het interview gesteld zullen worden.

Uitstelgedrag

1. Heeft u wel eens uitstelgedrag in Agile IT-projecten waargenomen? Hoe zag dit eruit en waar merkte u dit aan?
2. Wat ziet u als oorzaken van uitstelgedrag in Agile IT-projecten?
3. Wat kan de impact zijn van uitstelgedrag op kosten, tijd, functionaliteit en kwaliteit van een Agile IT-project?
 - a. Kunt u de impact van uitstelgedrag op kosten, tijd, functionaliteit en kwaliteit vergelijken? Met andere woorden, welke impact is het grootst, de op één na grootste, enz.?
4. Wat ziet u als mogelijke maatregelen tegen uitstelgedrag in Agile IT-projecten om het ofwel te voorkomen of de impact te verminderen?
5. Welke typische Agile-kenmerken kunnen de impact van uitstelgedrag voorkomen/verlagen of veroorzaken/vergroten?
6. Hoe ziet u de relatie tussen uitstelgedrag en technical debt?
7. In welk percentage van de Agile IT-projecten denkt u dat uitstelgedrag een probleem is? (0-100%)

Het mum-effect

1. Wat is uw ervaring met het mum-effect in Agile IT-projecten?
2. Welke gevolgen kent u of kunt u zich voorstellen die worden veroorzaakt door het mum-effect in Agile IT-projecten?
3. Op de schaal van 1 tot 5, hoe groot waren de gevolgen op (of kunt u zich voorstellen dat de gevolgen zijn op) **1**) de financiën, **2**) de planning, **3**) de scope (functionaliteit) en **4**) de kwaliteit?
 - a. Kunt u de gevolgen ook rangschikken van groot naar klein?
4. Welke oorzaken van het mum-effect in Agile IT-projecten kent u of kunt u zich voorstellen?
5. Welke tegenmaatregelen kent u of kunt u zich voorstellen die (kunnen) worden ingezet tegen het mum-effect in Agile IT-projecten?
6. Welke kenmerken van Agile kent u die (mogelijk) een rol spelen op hoe mum-effect zich manifesteert in Agile IT-projecten?
7. In hoeveel procent van de Agile IT-projecten denkt u dat het mum-effect zich voordoet?

Self-efficacy

1. Kent u een voorbeeld van een situatie waarbij binnen een Agile team, of bij een individu, sprake was van overmoedigheid of zelfonderschatting in relatie de werkzaamheden/taken die moesten worden uitgevoerd?
 - a. Wat was de oorzaak volgens u hiervan?
 - b. Waren er negatieve gevolgen voor de tijd, kosten, kwaliteit en functionaliteit?
 - i. Kunt u de deze gevolgen rangschikken van veel impact (1) naar weinig impact (4)?
 - c. Zijn er tegenmaatregelen genomen om dit in het vervolg te voorkomen?
2. Kent u een voorbeeld van een situatie binnen een Agile IT-project waarbij feedback invloed had op het geloof in eigen capaciteiten (vertrouwen)?
 - a. Heeft u meegemaakt dat bij positieve of negatieve feedback hierna fouten werden gemaakt, bijvoorbeeld door overmoedigheid of zelfonderschatting?

3. Stel dat binnen het Agile-team sprake is van overmoedigheid of zelfonderschatting, wat zijn dan volgens u maatregelen binnen Agile waarmee fouten als gevolg hiervan kunnen worden voorkomen?
 - a. Heeft u suggesties voor nieuwe aanvullende maatregelen?
4. Welke typische Agile-kenmerken kunnen overmoedigheid of zelfonderschatting volgens u veroorzaken?
5. In hoeveel procent van de IT-projecten komt zelfonderschatting of overmoedigheid voor denkt u (0-100%)?

Extra informatie interviewers

Dit onderzoek maakt deel uit van een breder onderzoek en vragen over andere biases zullen ook in de interviews worden opgenomen. Drie verschillende onderzoekers zullen elk zes experts in het veld interviewen over drie biases. De volgende informatie en Tabel 2 zullen naar de onderzoekers worden gestuurd, zodat ze een breder beeld hebben van de bias uitstelgedrag.

Uitstelgedrag is een complexe bias waarbij mensen de neiging hebben om vrijwillig het begin of de voltooiing van een geplande actie uit te stellen, ondanks de verwachting dat ze slechter af zullen zijn vanwege de vertraging. Uitstelgedrag treedt bijvoorbeeld op bij ontwikkelaars die consequent het grootste deel van hun werk uitstellen tot de deadlines naderen. Uitstelgedrag kan ook optreden wanneer teams (of product owners) problemen voor zich uit schuiven, wat kan leiden tot technical debt. Technical debt is het fenomeen waarbij men focust op korte termijn oplossingen en moeilijke vraagstukken uitstelt, wat later in het ontwikkelingsproces resulteert in extra werkzaamheden. Uitstelgedrag kan de timing, kosten, functionaliteit en kwaliteit van projecten beïnvloeden.

De verwachting is dat de rol van uitstelgedrag bij watervalprojecten anders is dan bij Agile projecten. Watervalprojecten hebben veel overhead, richten zich veel op details en hebben lange fasen, wat resulteert in een hoger risico op uitstelgedrag en een negatieve invloed hebben op de tijdigheid, kosten, kwaliteit en functionaliteit van een project. Omgekeerd hebben Agile-projecten korte iteratieve cycli, concrete subdoelen, maken ze gebruik van Scrum-artefacten en Scrum-activiteiten, wat de oorzaak van uitstelgedrag zou kunnen verminderen en daarmee het risico op uitstelgedrag. Aan de andere kant komt technical debt vaker voor in Agile-projecten dan in watervalprojecten, wat waarschijnlijker zal leiden tot vertragingen, kostenoverschrijdingen, kwaliteitsproblemen en minder functionaliteit. In Tabel 2 zijn de effecten, oorzaken en tegenmaatregelen van uitstelgedrag in (waterval)projecten en Agile-projecten samengevat.

Appendix 3: Interviewees Characteristics

Table 3: Overview of interviewees characteristics. Interviewees are numbered one to eighteen.

Nr	Corporate sector	Current role	Current Agile method	# years Agile	# years waterfall
1	High tech	Chief product owner	SAFe	9	20
2	High tech	Release train engineer	SAFe + Scrum	3	2
3	High tech	Product owner + architect	SAFe + Scrum	4	12
4	High tech	Release train engineer	SAFe + Scrum	4	8
5	High tech	Agile coach	SAFe	7	0
6	High tech	Release train engineer + scrum master	Scrum	1,5	2
7	Financial services	Product owner + business analyst	SAFe	5	3
8	Financial services	Product owner + developer	SAFe + Scrum	4,5	11
9	Financial services	Developer	SAFe	6,5	18
10	Financial services	Product manager	SAFe	6	2
11	Financial services	Scrum master + developer	SAFe	6	6
12	Financial services	Product owner + architect	SAFe	8	4
13	Business services	Business analyst	Scrum	3,5	0
14	Business services	Product manager	Other Agile method	10	20
15	Business services	Scrum master	Scrum	5	0
16	Business services	Agile coach	Other Agile method	3	0
17	Business services	Scrum master	Scrum	9	6
18	Business services	Developer	Other Agile method	7	0

Appendix 4: Coding Scheme

This coding scheme will clearly describe how themes, sub-themes, and codes are related. This appendix will clearly describe themes, sub-themes, and codes. This includes transparent rules regarding whether a piece of text belongs to a code or not. This coding scheme aims to increase transparency, reduce the risk of subjectivity, and increase the reproducibility of this research. The first section will describe the themes and sub-themes. The second section will describe the codes in more detail.

Themes and Sub-themes

Manifestation

The theme manifestation focuses on clarifying who, what, when, and how much people procrastinate in Agile IT projects.

- **Who:** describes who procrastinates on an individual level and group level.
- **What:** describes what is postponed by the procrastinator.
- **When:** describes when procrastination occurs.
- **How much:** describes how often procrastination is a problem in Agile IT projects. This subject is not reflected by a code in the coding scheme.

Causes

The theme causes describe the identified origins of procrastination in Agile IT projects. Causes related to business, person, tasks, team, agile, and environment are branched into sub-themes.

- **Business:** deals with the causes related to the interaction between IT and non-IT departments and the influence of the internal customer.
- **Person:** focuses on causes that relate to personal characteristics and his/her behavior.
- **Tasks:** includes the causes that relate to the type of task that is postponed.
- **Team:** relates to the team composition and team maturity.
- **Agile characteristics:** focuses on causes that originate specifically from Agile characteristics.
- **Environment:** describes the causes that relate to the environmental issues of an organization.

Effects

The theme effects are related to the consequences of procrastination in Agile IT projects.

- **Time:** relates to all timing and deadline consequences of procrastination.
- **Quality:** relates to the quality effects of procrastination.
- **Functionality:** relates to the delivered functionality as an effect of procrastination.
- **Costs:** relates to the costs and budget of the project as an effect of procrastination.
- **Other:** relates to all other consequences of procrastination.

Countermeasures

This theme attempts to describe countermeasures that reduce the impact of procrastination or reduce the risk of procrastination.

- **Feedback mechanisms:** countermeasures of procrastination related to instruments, tools, processes, and rituals to improve transparency and enhance the feedback loop.
- **Backlog and priorities:** countermeasures of procrastination related to transparent prioritizing of work and backlog refinement.
- **Feedback Culture:** countermeasures of procrastination related to the degree of an open feedback culture in the team and organization.
- **Team:** countermeasures of procrastination related to a mature, competent, and engaged Agile team.
- **Business:** countermeasures of procrastination related to the relationship and way of working with the business.
- **Person:** countermeasures of procrastination related to personal characteristics and skills.

Code details

In Table 4 the code description and frequency per code, sub-theme, and theme are given. The code description can be seen as a coding rule; it will elaborate when a text belongs to the code and, if applicable, when not. The frequency of a code indicates the number of times this code appears across all interviews. If a code occurs several times in one interview, this code is counted only once. All the codes refer to Agile IT projects, which is not explicitly mentioned in the code description.

Table 4: Coding Scheme

Theme	Sub-theme	Code	Code Description	Freq.
Mani- festation	What	Resolving or preventing technical debt	This code is assigned when observing the postponement of technical debt or the postponement of tasks that could prevent technical debt. This includes postponing refactoring of code and resolving performance issues.	13
		Development work	This code is assigned when observing the postponement of development work. This includes work that is related to new functionality. This code is also assigned to the text where interviewees talk about tasks, user stories, and features without explicitly mentioning what kind of work.	10
		Not recognized	This code is assigned if interviewees do not recognize procrastination.	4
		Migration	This code is assigned when observing the postponement of migration-related activities.	3
		Strategical and architectural decisions	This code is assigned when observing the postponement of strategic and architectural decisions.	2
		Refinement	This code is assigned when observing the postponement of preparation work and refinement of work.	2
		Completing tasks	This code is assigned when observing the postponement of completing tasks instead of starting tasks.	2
		Stakeholder verification	This code is assigned when observing the postponement of stakeholder verification or user acceptance test.	1
		Personal development	This code is assigned when observing the postponement of personal development, like postponing courses to become more T-shaped.	1
		Operational tasks	This code is assigned when observing the postponement of operational work instead of development work.	1
		Communication	This code is assigned when observing the postponement of communication, for example, towards stakeholders.	1
	Who	Developer	This code is assigned when observing that developers procrastinate.	7
		Product owner	This code is assigned when observing product owners or chief product owners procrastinating.	4
		Individuals	This code is assigned when observing individuals procrastinate.	2

Theme	Sub-theme	Code	Code Description	Freq.
		Scrum master	This code is assigned when observing scrum masters or release train engineers procrastinating.	1
		Group	This code is assigned when observing people in groups or teams procrastinate.	1
		Architect	This code is assigned when observing that architects procrastinate.	1
	When	End sprint	This code is assigned when observing people who procrastinate until the end of a sprint is approaching.	3
		End PI	This code is assigned when observing people who procrastinate until the end of a product increment is approaching.	1
Causes	Tasks	Preference easy small short	This code is assigned when the cause of procrastination is attributed to the preference for easy, small, and short tasks. This includes the postponement of difficult, complex, and big tasks	8
		Unclear	This code is assigned when the cause of procrastination is attributed to unclear and vague tasks.	5
		Dependencies	This code is assigned when the cause of procrastination is attributed to tasks with dependencies with other teams or other externals.	4
		Boring	This code is assigned when the cause of procrastination is attributed to not challenging or boring tasks. This includes people who cherry-pick, so working on preferred tasks first.	4
	Person	Incompetent	This code is assigned when the cause of procrastination is attributed to the person's incompetence. The person does not have the right skills or knowledge or feels insecure about his/her skills.	6
		Introvert	This code is assigned when the cause of procrastination is attributed to the introversion of people. This includes being afraid to speak up, give pushback, or ask for feedback.	5
		Human nature	This code is assigned when the cause of procrastination is attributed to the human nature of people. This code is assigned if no specific personal cause is mentioned, but just because it is human to procrastinate.	5
		Motivation	This code is assigned when the cause of procrastination is attributed to the lack of motivation.	2
		Too detailed	This code is assigned when the cause of procrastination is attributed to people who are too focused on details.	1
	Business	Pressure to deliver functionality	This code is assigned when the cause of procrastination is attributed to the business pressure to deliver functionality. This includes the pressure to meet business requirements and the priority of new functionality over resolving technical debt.	14

Theme	Sub-theme	Code	Code Description	Freq.	
		Lack of urgency	This code is assigned when the cause of procrastination is attributed to a lack of business urgency. This relates to the lack of stakeholders in review sessions and the lack of feedback. This also includes projects that are more innovative in nature, without a clear business goal.	2	
	Team	Unclear authoritative	This code is assigned when the cause of procrastination is attributed to unclear authority in teams so if it is unclear who should make decisions.	3	
		Too much work in sprint	This code is assigned when the cause of procrastination is attributed to too much work in a sprint.	2	
		Low capacity people	This code is assigned when the cause of procrastination is attributed to a low capacity of the right people.	2	
		Incorrect estimation workload	This code is assigned when the cause of procrastination is attributed to the incorrect estimation of workload by a team or team members.	2	
	Agile	High flexibility	This code is assigned when the cause of procrastination is attributed to the high flexibility of the Agile method. This includes the lack of fixed deadlines and the lack of go/no go moments, which are available in waterfall projects.	3	
		Abuse of freedom	This code is assigned when the cause of procrastination is attributed to the abuse of freedom and autonomy. This includes people who do not take responsibility.	3	
	Environment	Working from home	This code is assigned when the cause of procrastination is attributed to (mandatory) working from home.	3	
	Effects	Time	Longer lead time	This code is assigned when a longer lead time of work is seen as an effect of procrastination. This includes if people have to make more working hours.	8
			Later deliveries	This code is assigned when later deliveries are seen as an effect of procrastination. This includes not meeting the sprint goals and not meeting deadlines.	8
Fixed			This code is assigned when no effect on timing is expected due to procrastination, as there is a fixed team and a fixed number of sprints.	3	
Overtime			This code is assigned when overtime is seen as an effect of procrastination. This includes employees who work extra in the evenings or on weekends.	2	
Quality		Technical debt	This code is assigned when technical debt is seen as an effect of procrastination. This includes unstable and not future-proof solutions. Technical debt also refers to less documentation.	10	

Theme	Sub-theme	Code	Code Description	Freq.	
		More mistakes	This code is assigned when making more mistakes is seen as an effect of procrastination.	3	
		No effect	This code is assigned when no effect on the quality of the product is expected due to procrastination.	1	
	Functionality	Smaller product	This code is assigned when the delivery of a smaller product or a product with less functionality is seen as an effect of procrastination.	4	
		No other features	This code is assigned when no other features can be delivered due to procrastination is seen as an effect of procrastination. This includes blocking other teams who can not deliver other features.	4	
		No effect	This code is assigned when no effect on the product's functionality is expected due to procrastination.	3	
		Changing	This code is assigned when the risk of changing functionality or a changing environment over time is seen as an effect of procrastination.	2	
		Not in production	This code is assigned when the product is not deployed in production is seen as an effect of procrastination.	1	
	Costs	More working hours	This code is assigned when more working hours result in more costs is seen as an effect of procrastination.	7	
		Fixed	This code is assigned when no effect on the costs is expected due to procrastination, as there is a fixed team and a fixed number of sprints.	3	
		Less efficiency	This code is assigned when less efficiency resulting in more costs is seen as an effect of procrastination.	1	
	Other	Personal health	This code is assigned when a decrease in personal health is seen as an effect of procrastination. This includes burnout and lack of personal development.	2	
		Unsatisfied stakeholders	This code is assigned when unsatisfied stakeholders are seen as an effect of procrastination.	1	
		Reputation	This code is assigned when the deteriorating reputation of a team or department is seen as an effect of procrastination.	1	
	Counter-measures	Feedback mechanisms	Ceremonies	This code is assigned when a countermeasure of procrastination is identified as correctly executing Agile ceremonies. This includes daily stand-ups, reviews, retrospectives, and planning sessions.	9
			Short iterative cycles	This code is assigned when a countermeasure of procrastination is identified as having short iterative cycles. This includes having short deadlines.	8
Monitoring			This code is assigned when a countermeasure of procrastination is identified as implementing	7	

Theme	Sub-theme	Code	Code Description	Freq.
			monitoring. This includes formulating metrics and visualizing progress.	
		Feedback loop	This code is assigned when a countermeasure of procrastination is identified as implementing a continuous feedback loop between stakeholders and the team.	3
	Backlog and Priorities	Capacity allocation	This code is assigned when a countermeasure of procrastination is identified as capacity allocation. This includes making a certain percentage of the workload available for resolving technical debt. Introducing an innovation and planning sprint is also seen as a form of capacity allocation.	8
		Small items	This code is assigned when a countermeasure of procrastination is identified as dividing large tasks into smaller tasks that can easily be picked up.	6
		Clear backlog and priorities	This code is assigned when a countermeasure of procrastination is identified as having a clear backlog with clear priorities. This includes clear priorities according to technical debt.	6
		Handle technical debt secretly	This code is assigned when a countermeasure of procrastination is identified as handling technical debt secretly, not informing stakeholders.	1
	Feedback culture	Transparent communication	This code is assigned when a countermeasure of procrastination is identified as transparent communication. This includes identifying procrastination, addressing if someone is procrastinating, discussing with the procrastinator, interacting and communicating with each other.	12
		Ability to speak-up	This code is assigned when a countermeasure of procrastination is identified as the ability to speak up in a team.	7
		Self-learning	This code is assigned when a countermeasure of procrastination is identified as learning from own mistakes so that the team becomes self-learning.	1
	Team	Competent engaged SM and PO	This code is assigned when a countermeasure of procrastination is identified as having a competent and engaged scrum master and/or product owner. This includes the right coaching skills.	9
		Work together	This code is assigned when a countermeasure of procrastination is identified as working together. This includes having a buddy, pair programming, and doing code reviews.	4
		Clear responsibilities	This code is assigned when a countermeasure of procrastination is identified as having clear responsibilities. This includes a clear owner of a user story.	2
		Autonomy	This code is assigned when a countermeasure of procrastination is identified as increasing team autonomy.	2

Theme	Sub-theme	Code	Code Description	Freq.
	Business	Meaningfulness and engaged stakeholders	This code is assigned when a countermeasure of procrastination is identified as clear meaningfulness. This includes the clarity of the importance of tasks and engaged stakeholders.	3
		Improve relationship	This code is assigned when a countermeasure of procrastination is identified as improving the relationship with stakeholders.	2
	Person	Increase skills and knowledge	This code is assigned when a countermeasure of procrastination is identified as increasing the skills and knowledge of people.	2
		Increase motivation	This code is assigned when a countermeasure of procrastination is identified as increasing people's motivation.	2
	Waterfall	Fixed deadlines	This code is assigned when a countermeasure of procrastination is identified as implementing fixed deadlines instead of the flexible Agile way of working.	2
	4	20	76	

Appendix 5: Overview Thematic Analyses

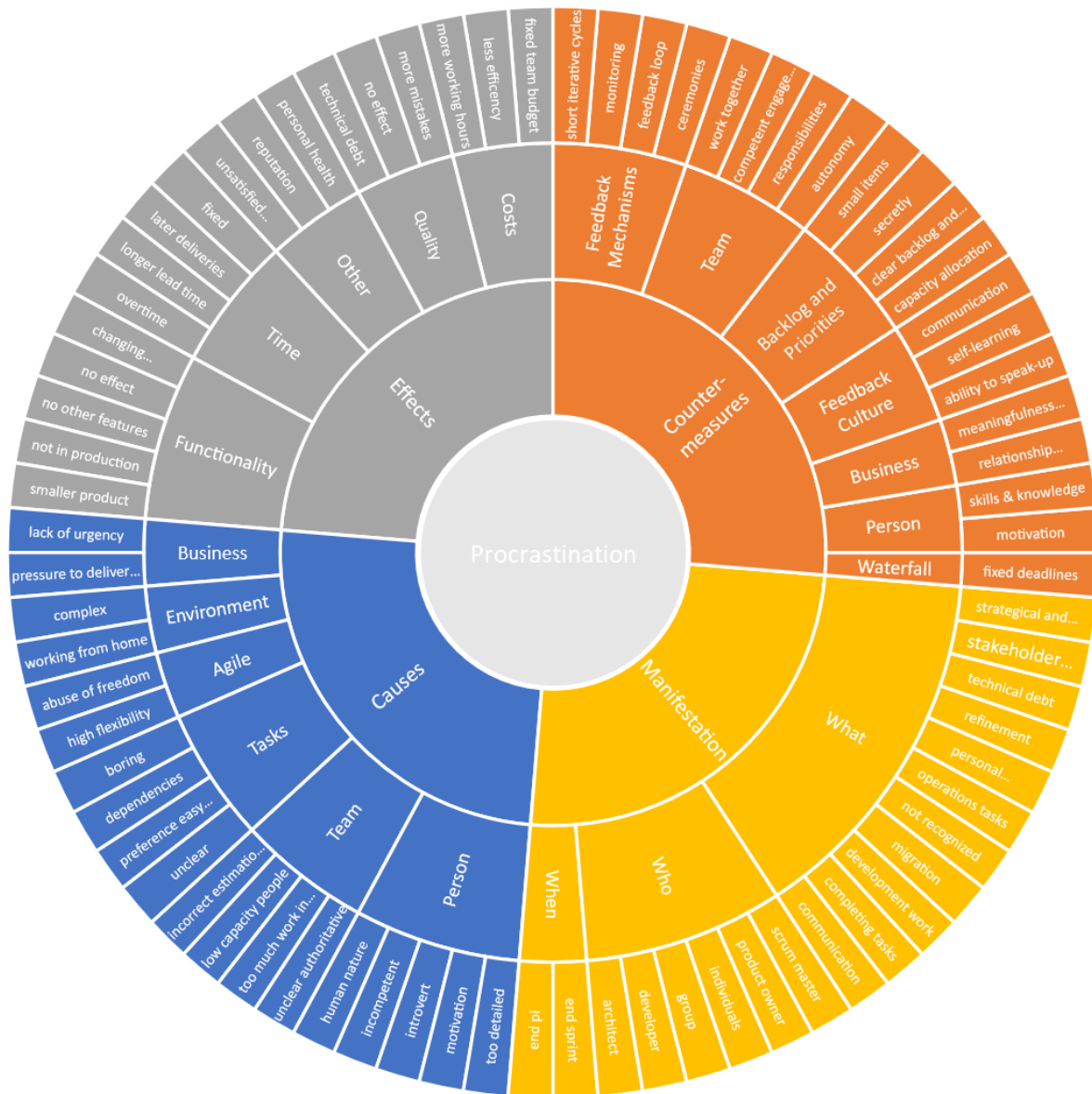


Figure 14: Overview of all themes, sub-themes, and codes.
(All codes have the same size in this visualization)

Appendix 6: Overview of Codes per Interview

Table 5: Overview of codes per interview

Theme	Sub-theme	Code	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	Total	
Manifest.	What	communication				■															1	
Manifest.	What	completing tasks		■														■				2
Manifest.	What	development work		■		■		■	■		■	■	■	■	■		■			■	10	
Manifest.	What	migration						■	■							■					3	
Manifest.	What	not recognized					■		■	■											4	
Manifest.	What	operations tasks														■					1	
Manifest.	What	personal development					■														1	
Manifest.	What	refinement									■					■					2	
Manifest.	What	resolv. or prev. technical debt	■		■		■	■	■	■			■	■	■	■	■	■	■	■	13	
Manifest.	What	stakeholder verification			■																1	
Manifest.	What	strategical & arch. decisions						■						■							2	
Manifest.	Who	architect												■							1	
Manifest.	Who	developer		■	■		■	■	■		■			■							7	
Manifest.	Who	group		■																	1	
Manifest.	Who	individuals		■											■						2	
Manifest.	Who	product owner					■	■				■	■								4	
Manifest.	Who	scrum master															■				1	
Manifest.	When	end pi		■																	1	
Manifest.	When	end sprint						■				■		■							3	
Causes	Tasks	boring					■	■	■		■										4	
Causes	Tasks	dependencies		■				■			■						■				4	
Causes	Tasks	preference easy small short		■			■	■		■		■		■	■	■					8	
Causes	Tasks	unclear		■	■																5	
Causes	Person	human nature			■	■	■											■	■		5	
Causes	Person	incompetent									■			■	■	■	■		■		6	
Causes	Person	introvert		■	■	■	■				■										5	
Causes	Person	motivation			■													■			2	
Causes	Person	too detailed		■																	1	
Causes	Team	incorrect estimation workload					■					■									2	
Causes	Team	low capacity people										■				■					2	
Causes	Team	too much work in sprint									■							■			2	
Causes	Team	unclear authoritative						■						■						■	3	
Causes	Business	lack of urgency			■	■															2	
Causes	Business	pressure to deliver funct.		■		■	■	■	■	■	■	■	■	■	■	■	■	■	■	■	14	
Causes	Env.	working from home							■					■		■					3	
Causes	Agile	abuse of freedom					■					■		■							2	
Causes	Agile	high flexibility				■		■	■												3	
Effects	Time	fixed		■										■	■						3	
Effects	Time	later deliveries					■	■	■	■	■	■	■	■	■	■					8	
Effects	Time	longer lead time			■	■		■	■			■		■	■					■	8	
Effects	Time	overtime								■										■	2	
Effects	Costs	fixed team budget		■										■	■						3	
Effects	Costs	less efficiency					■														1	
Effects	Costs	more working hours			■		■	■	■				■		■					■	7	
Effects	Funct.	changing functionality					■		■												2	
Effects	Funct.	no effect			■															■	3	
Effects	Funct.	no other features		■				■										■		■	4	
Effects	Funct.	not in production						■													1	
Effects	Funct.	smaller product		■	■							■		■							4	
Effects	Quality	more mistakes								■		■		■							3	
Effects	Quality	no effect								■											1	
Effects	Quality	technical debt		■		■	■	■	■	■			■	■					■	■	10	
Effects	Other	personal health						■												■	2	
Effects	Other	reputation									■										1	
Effects	Other	unsatisfied stakeholders								■											1	

Theme	Sub-theme	Code	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	Total
Counterterm.	Backlog & Prio.	capacity allocation		■			■	■	■			■	■						■	■	8
Counterterm.	Backlog & Prio.	clear backlog and priorities										■	■				■			■	6
Counterterm.	Backlog & Prio.	handle technical debt secretly			■																1
Counterterm.	Backlog & Prio.	small items		■			■							■		■			■	■	6
Counterterm.	Team	autonomy			■	■															2
Counterterm.	Team	responsibilities									■									■	2
Counterterm.	Team	competent engaged SM & PO	■		■	■		■							■		■	■	■	■	9
Counterterm.	Team	work together								■				■		■					4
Counterterm.	Person	motivation		■															■		2
Counterterm.	Person	skills & knowledge													■			■			2
Counterterm.	Waterfall	fixed deadlines			■				■												2
Counterterm.	Business	relationship stakeholders			■													■			2
Counterterm.	Business	meaningful engaged stakeh.	■		■														■		3
Counterterm.	Feedback Cult.	ability to speak-up		■			■				■			■		■	■	■	■	■	7
Counterterm.	Feedback Cult.	self-learning				■															1
Counterterm.	Feedback Cult.	transparent communication		■		■	■	■		■				■	■	■	■	■	■	■	12
Counterterm.	Feedb. Mech.	ceremonies	■	■		■	■			■				■						■	9
Counterterm.	Feedb. Mech.	feedback loop		■		■	■														3
Counterterm.	Feedb. Mech.	monitoring				■	■					■		■	■			■		■	7
Counterterm.	Feedb. Mech.	short iterative cycles		■	■	■	■	■										■		■	8

Appendix 7: General Procrastination Scale

Participant code: _____

GPS-9

Please be as honest and accurate as you can throughout. Try not to let your response to one statement influence your responses to other statements. There are no "correct" or "incorrect" answers. Answer according to your own feelings, rather than how you think "most people" would answer.

	1 False	2 Not usually true for me	3 Sometimes false/true for me	4 Mostly true for me	5 True of me
1. I often find myself performing tasks that I had intended to do days before.					
2. Even with jobs that require little else except sitting down and doing them, I find they seldom get done for days.					
3. I generally delay before starting work I have to do.					
4. In preparing for some deadlines, I often waste time by doing other things.					
5. I often have a task finished sooner than necessary.					
6. I usually buy even an essential item at the last minute.					
7. I usually accomplish all the things I plan to do in a day.					
8. I am continually saying I'll do it tomorrow.					
9. I usually take care of all the tasks I have to do before I settle down and relax for the evening.					

Scoring: Reverse code items 5, 7, and 9; then take the average of all items for the total score.

Figure 14: General Procrastination Scale – 9 (The University of Sheffield, 2019)