Agile implementation in hardware environment

Master Thesis project

Laurens van Driessen

Delft University of Technology

TU Delft

accenture

Agile implementation in hardware environment

by

Laurens van Driessen

to obtain the degree of Master of Science
at the Delft University of Technology,
to be defended publicly on Friday July 14, 2023 at 15:00 PM.

Student number:	4484495	
Project duration:	February 1, 2023 – July 14, 2023	
Thesis committee:	Dr. ir. M. Kroesen,	TU Delft, chair
	Dr. N. Pachos-Fokialis,	TU Delft, first supervisor
	Dr. ir. M. Kroesen,	TU Delft, second supervisor

Preface

This thesis marks the end of my time as a student at the TU Delft. After graduating for the master High-Tech engineering, I finished my second master Management of Technology with this thesis. During my time at the TU Delft, I learned a lot about all kinds of disciplines and met amazing people. I would not have enjoyed my time here without you. Thank you all for that!

First, I would like to thank my first supervisor Nikos for his support during the thesis. The meetings we had were always insightful and inspiring. Besides the meetings, I really appreciate the time you took to give me feedback in a short amount of time. Also, I would like to thank Maarten for being part of my graduation committee and taking the time to give me feedback.

Furthermore, I would like to thank Thomas and Erik from Accenture for their guidance at Accenture. Thomas, I want to thank you for all the weekly meetings we had to talk about the progress and the time and effort you spent to help me finish my thesis on time. Especially the help with planning my interviews and giving advice on how to approach them. Erik, I want to thank you for the weekly informal chats we had about life at Accenture, life outside Accenture, and of course the project. I also want to thank my colleagues from Accenture for their nice talks and (coffee) breaks. For me, it was always a pleasure to go to the office.

Finally, I would like to thank Accenture for giving me the opportunity to write my thesis during the internship. Last but not least, I want to thank all the interviewees for their time and insightful information about the topic.

Laurens van Driessen
Delft, July 2023

Summary

Agile development approaches are becoming more popular the last years. With large success in the software industry, the interest in using agile methods has shifted to the development of physical products. Since agile has been originally developed for the software industry, it does not cover aspects of the hardware development environment. However, some companies do want to switch from a traditional approach to a hybrid or even complete agile approach. Reasons for this can vary from the wish to reduce time-to-market, improve communication and collaboration among the teams, or because they do not want to lack behind their competitors even though they did not know if the method will be effective for their own company. The problem is that the implementation and use of the agile methodology in the traditional approach face some issues concerning effectiveness, performance, and mismatch with expectations. The objective of this research is to look into the implementation and use of the agile methodology in the traditional models in the hardware industry. This has led to the research question:

> **How to effectively implement and use the agile methodology in the hardware environment?**

A literature review has been conducted on the success factors and barriers of agile implementation in the hardware environment. These factors and barriers can be used to find what actions are needed for successful implementation. Furthermore, the benefits and challenges of the use of agile are investigated and an overview is created. For the second part of this research, a qualitative research approach is used. This part will be more exploitative and will dive into the perceptions and best practices by doing a case study with multi-person interviews. This case study is primarily performed in the electrical engineering department of a large high-tech company. The company is focused on the semiconductor industry and has both a large R&D department and a large manufacturing department. Multiple interviews are being executed and compared to each other.

This case study consists of three parts, one with engineers of the software department, one with the engineers of the hardware department, and one with the team leads of the hardware department. The first part is only to verify what the success factors and barriers were with the implementation of the agile methodology in the software environment. This information can be used for the implementation of the agile methodology in the hardware development. The second and third part consists of interviews with engineers and team leads of the electrical engineering department of the Research and Development (R&D) department. This data can be used to design both a framework for the successful implementation and for the successful use of the agile methodology.

The findings of the interview highlighted that the success factors and barriers are difficult to see independently of each other, as they are often linked and can reinforce one another. The factors that play an important role are on the one hand the mindset of the management and the employees, and on the other hand the availability of pilots and training. The mindset has been found to influence the way the new method is adopted and executed in the company. Training enhances the right mindset and helps to get the right skills to perform the implementation. Pilots are trials with a small part of the organization to experiment with the implementation of the new way of working. Pilots are necessary to customize the approach to the company and create a smooth transition for the complete firm.

The benefits of the use have been divided into four categories; people, process, organization, and product. People will get more focus on their tasks since prioritization and planning of the work have been improved. Agile tools improve transparency and allow employees to track dependencies with other teams, therefore collaboration can be improved. Also, the use of agile rituals improves the communication and quality of the work. Feedback provided in retrospective meetings can be used in future situations and for example, daily stand-ups and planning events improve communication due to more frequent, more transparent, and better-structured meetings. The use of early prototypes is beneficial for the process as well. By creating a consistent structure of the teams the communication and collaboration improved.

The agile methodology should be adapted at some points to better match the characteristics of the hardware environment. The adaptions have been categorized into people, process, organization, and

product. For people, there should be more emphasis on creating the right mindset by providing training. The agile way of working is hard to understand for people in the hardware environment since they do not feel the urge to change and want to keep doing their job instead of spending time on meetings. They are used to the old way of working and following agile practices does feel naturally contradictory for hardware developers. A lack of clear communication about the purpose of using agile results in resistance to change. Therefore, there should be a focus on creating the right mindset for employees. On the process level, it is necessary to adjust the rituals to the needs of the teams. Since hardware deals with physical components that have longer lead times, the meeting intensity should be adapted to that. This should be an iterative process, led by the status of the project. Furthermore, there should be more focus on the alignment of tools to match with the other teams/suppliers since in hardware the dependencies on other teams/suppliers are harder to control because of the physical nature of the products. Moreover, from an organizational perspective, it is important that team structures are aligned in the organization. Also, people should no longer be specialists in one specific part but be multi-specialized. This could, however, be hard to achieve since some companies have to deal with a high level of complexity that asks for specialists. On the product level, the focus should be on the iterations and rapid prototyping even if the prototype is not functional. While the method originally state that after each sprint you should deliver something to be tested it is more important to have a prototype to start the conversation.

The way of implementing and using the agile method should match the company and the people. For the implementation, the mindset of the people should be right in place and the pilot can show if the proposed way of working is right. For the use, it is of significant value that besides the people aspect, the tools, and organizational structure are aligned. Also here the rituals, prototyping, training, and way of specialization should match the company and the teams. The findings of this thesis can be used as a guideline to implement and use the agile methodology in large-scale hardware companies.

Generalizability is one of the limitations of this research since the case study is only performed at one company. Also, the amount of interviewees is limited to 12 people. Conducting interviews with more people from different departments or different companies increases this generalizability. The designed framework could also be further validated by conducting interviews on the outcomes of the adaption of the agile framework. Future research could dive into the actual use of this adapted agile framework in the semiconductor industry. It could be valuable to conduct the same research in a couple of years at the same company and check whether the perception of people changed and if new insights on the way of working have been created. Furthermore, it is interesting to investigate the use of agile in other hardware industries.

Contents

Preface **i**

Summary **ii**

1 Introduction **1**

2 Literature review **4**

2.1 Characteristics hardware industry 4

2.1.1 Challenges 5

2.1.2 Traditional model hardware projects 6

2.1.3 Agile method in hardware 9

2.1.4 Hybrid method 14

2.2 Agile implementation 16

2.2.1 Success factors 16

2.2.2 Barriers 18

2.3 Conclusion 20

2.4 Research questions 21

3 Methodology **22**

3.1 Selection of the research approach 22

3.1.1 Research Strategy 22

3.1.2 Population and sampling 23

3.2 Research flow diagram 23

3.2.1 Phase 1: Problem definition 24

3.2.2 Phase 2: Data collection 24

3.2.3 Phase 3: Data analysis 25

3.2.4 Phase 4: Report results 27

4 Results **28**

4.1 Implementation of agile methodology in the hardware environment 28

4.1.1 Management 28

4.1.2 Employees 29

4.1.3 Organization 29

4.2 Use of agile methodology in the hardware environment 31

4.2.1 People 31

4.2.2 Process 31

4.2.3 Organization 34

4.2.4 Product 35

4.2.5 Benefits and Challenges 35

4.3 Adaption of agile method 37

4.3.1 People 37

4.3.2 Process 37

4.3.3 Organization 38

4.3.4 Product 38

5 Discussion **40**

5.1 Implementation 40

5.2 Use 42

5.3 Practical relevance 44

5.4 Limitation and future research 44

6 Conclusion **46**

A Agile values and principles 52

B HREC 53

C Challenges agile development 59

D Success factors and barriers implementation 62

E Interview questions 65

F Interview summaries 70

F.1 Interviewee 1 70
F.2 Interviewee 2 70
F.3 Interviewee 3 71
F.4 Interviewee 4 71
F.5 Interviewee 5 71
F.6 Interviewee 6 72
F.7 Interviewee 7 72
F.8 Interviewee 8 72
F.9 Interviewee 9 73
F.10 Interviewee 10 73
F.11 Interviewee 11 74
F.12 Interviewee 12 74

G Guideline validation adapted framework 75

List of Figures

1.1 Research flow diagram . . . 3

2.1 Hardware development process based on (Drechsler & Breiter, 2003; Arroyo, 2023; Cadence, 2023) . . . 5
2.2 Stage gate model (Jungwirth, 2022) . . . 7
2.3 Waterfall model (Naveen, 2015) . . . 7
2.4 V-model (Gutiérrez Rivas et al., 2012) . . . 8
2.5 Agile model (Javatpoint, 2021) . . . 9
2.6 Mindmap of the challenges of agile in hardware environment (Atzberger & Paetzold, 2019; Schmidt et al., 2018a) . . . 11
2.7 Perceptions on agile development for hardware startups (Nguyen-Duc et al., 2018) . . . 13
2.8 New generation model (Cooper, 2014) . . . 14
2.9 Overlapping activities within and between stages (Cooper, 2014) . . . 15
2.10 Agile framework comparison (Ebert & Paasivaara, 2017) . . . 16

3.1 Simplified conceptual framework . . . 22
3.2 Coding categories implementation . . . 26
3.3 Coding categories use . . . 27

4.1 Factors affecting the implementation of agile sorted by the number of times mentioned 30
4.2 Benefits of the use of agile in hardware environment . . . 36
4.3 Challenges of the use of agile in hardware environment . . . 36
4.4 Adjustment of the agile framework to meet characteristics of hardware environment . . 37
4.5 Framework of the adaption of the agile method in hardware environment . . . 39

5.1 Transition traditional approach to agile/hybrid approach . . . 40
5.2 Successful implementation agile methodology . . . 42
5.3 Framework of the use of the agile method in hardware environment, with adaption and perceived outcome . . . 43

C.1 Challenges use agile(Atzberger & Paetzold, 2019) . . . 60

E.1 Interview structure . . . 65

G.1 Validation adapted framework . . . 76

List of Tables

2.1 Top 3 benefits of using agile for soft and hard criteria (Atzberger et al., 2019) 10
2.2 Highest rate benefits of using agile in hardware environment according to Atzberger et al. (2019) 11

3.1 List of interviewees 23

A.1 Agile values and corresponding principles Beck (2001) 52

C.2 Challenges in agile development with actual and expected impact on a scale of 0-5 (Schmidt et al., 2018a) 59
C.1 Complete list of benefits of use agile in hardware environment categorized in soft and hard critera 60
C.3 Conflicts in agile development with actual and expected impact on a scale of 0-5 (Schmidt et al., 2018a) 61

D.1 Success factors of agile implementation in hardware environment (Dikert et al., 2016; Naslund & Kale, 2020; Kalenda et al., 2018; Senapathi & Srinivasan, 2013; Campanelli et al., 2017; Gupta et al., 2019; Koehnemann & Coats, 2009; Paasivaara et al., 2018; Moravcová & Legény, 2016; Paterek, 2017; Mancin, 2016; Dingsøyr & Moe, 2013; Bjarnason et al., 2011; Pinto & Slevin, 1988; Conboy et al., 2011; Weichbroth, 2022; Hajjdiab & Taleb, 2011) 63
D.2 Barriers of agile implementation in hardware environment (Dikert et al., 2016; Naslund & Kale, 2020; Kalenda et al., 2018; Senapathi & Srinivasan, 2013; Campanelli et al., 2017; Gupta et al., 2019; Koehnemann & Coats, 2009; Paasivaara et al., 2018; Moravcová & Legény, 2016; Paterek, 2017; Mancin, 2016; Dingsøyr & Moe, 2013; Bjarnason et al., 2011; Pinto & Slevin, 1988; Conboy et al., 2011; Weichbroth, 2022; Hajjdiab & Taleb, 2011) 64

1
Introduction

Agile development approaches are becoming more popular the last years. The modern concept of agility was established in 2001, when 17 software developers gathered to agree on a distinct working methodology that would meet the demands and conditions of their industry at the time. Agility is the capacity to continually and swiftly react to and adapt to expected and unexpected changes in a dynamic environment, and to exploit those changes to one's advantage (Böhmer et al., 2015). Agile development enables decentralized situational awareness, continuous improvement (iterative trial-and-error loops), shared ownership, and the ability to adapt to change in a dynamic project environment (Schmidt et al., 2018b). The agile method is a very human-being-oriented management approach (Nerur & Balijepally, 2007; Nerur et al., 2005). The Agile Manifesto consists of 12 principles that were set up around 4 values and form the basis of the agile framework (see Appendix A) (Beck, 2001).

As of this moment, agility has been on the increase ever since, resulting in the emergence of various agile approaches throughout the years. Yet, the core understanding that all agile techniques share is their adherence to the 'Agile Manifesto'. The most extensively employed techniques in the software industry today are scrum, which was first presented in 1995, Kanban, eXtreme Programming (XP), and Crystal (Atzberger et al., 2019).

The benefits of using agile methods have been investigated by several studies by VersionOne (2018) and Rodríguez et al. (2012). They compared the predicted advantages with the actual benefits and found that especially for the criteria that are not quantitatively measurable, such as communication, transparency, or knowledge creation, the results are comparable. For the criteria that are quantitatively measurable, the improvements are rather negligible while companies often have high expectations. In terms of predicted advantages, the TOP 5 reasons for embracing agile development are increased productivity, shorter project lead times, enhanced software quality, and advanced adaptability. Among the TOP 5 actual advantages are enhanced transparency, increased productivity, and improved ability to adapt to changes. The findings show that one of the primary motivations for using agile development is to improve time-to-market, product quality, and project risk management (Schmidt et al., 2018b).

The interest in using agile methods has shifted to the development of physical products (Dingsøyr et al., 2012; VersionOne, 2018). Physical products are products that do not only consist of software and hence bear a physical nature. These products incorporate components of the hardware domain combined with electronic, mechatronic, and mechanical domains (Atzberger & Paetzold, 2019). During projects, the requirements of hardware products could change due to other needs of the consumers. Also, the availability of materials, machines, and suppliers could influence the production flow. Since boundary conditions are frequently changing in today's high-tech industry the agile methodology could be of significant value. Agile development helps businesses to deal with such circumstances more efficiently and effectively by responding fast to expected and unforeseeable changes within dynamic contexts and using those changes to their advantage (Schmidt et al., 2018b).

Since the agile manifesto is originally developed for the software industry it does not cover aspects of the hardware development environment. Nevertheless, some guiding principles and core values can also be applied here (Schmidt et al., 2018b). According to Atzberger & Paetzold (2019), the possibility of transferring the principles of the manifesto from software development to hardware development has been rated 3.9/5 (n=91), suggesting that it should be possible to use the agile methodology in the hardware environment.

Agile product development is similar to agile software development. It focuses on the ability to inherently and rapidly create, embrace and learn from the change with the goal of contributing to the perceived customer value. Uncertainties can be either negative, resulting in risks, or positive, leading to opportunities (Lévárdy, 2006). As a result, agile development is adaptable and seeks to capitalize on changes rather than prevent them (Beck, 2001; Böhmer et al., 2015; Atzberger et al., 2019). Agile product development also covers administrative organization activities besides operative design activities.

Currently, many companies are using more traditional product development approaches like a waterfall or stage-gate processes (Cooper, 2014; Chuang et al., 2014). These methods are especially more commonly used in larger projects. These methods have more formal, bureaucratic processes and emphasize specialist, cooperation, and decision-making by experts. The project's goal will be reached by a well-thought-out approach based on the information that is available at the start of the project (Nerur & Balijepally, 2007).

Some companies do want to switch from this traditional approach to a hybrid or even complete agile approach. Reasons for this can vary from the wish to increase time-to-market, increase communication and collaboration among the team, or because they do not want to lack behind their competitors even though they did not know if the method will be effective for their own company (Schmidt et al., 2018a; Atzberger & Paetzold, 2019; Rodríguez et al., 2012). The problem is that the implementation of the agile methodology in the traditional problem faces some issues concerning effectiveness, performance, and mismatch with expectations. The objective of this research is to look into the implementation and use of the agile methodology in the traditional models in the hardware industry. Success factors and barriers need to be identified, perceived outcomes related to people, process, organization, and product should be found, and the adaption of the agile framework to meet the characteristics of the hardware environment should be emphasized. Therefore, the following research and subquestion have been designed:

RQ: ***How to effectively implement and use the agile methodology in the hardware environment?***

- SUB 1: What are the success factors and barriers in the implementation of the agile methodology in the hardware environment?
- SUB 2: How can the adoption of agile methodologies affect the perceived individual and team outcomes in terms of people, process, organization, and product?
- SUB 3: How can agile methodologies be adapted to suit the unique characteristics and constraints of hardware development?

The company that will be researched is active in the semiconductor industry. It produces hardware and software components for large machines. The R&D department is divided into a hardware, software, and electrical development department. The company already started (partly) implementing the agile methodology. For the software development department, this implementation was successful. The company started implementing agile tools for the hardware development department but is facing some serious issues. Therefore this company is a good fit for the research.

The fact that the company is in the transition phase to a hybrid/agile approach makes it also relevant from a scientific perspective. This transition generates new questions and new knowledge about what is working and what is not. By doing this, the focus of academia can be on the core issues and allow them to determine measures and ways to overcome them. Therefore, other researchers can use this case study to validate and expand their knowledge on agile implementation and use in the hardware environment. Furthermore, the findings of this thesis can be used as a guideline to implement and use the agile methodology in large-scale hardware companies. The final goal is to design a framework of how to effectively implement and use the agile methodology in the hardware environment. This framework can help with the understanding of how the agile methodology should be adapted to suit the hardware characteristics.

From a MOT perspective, this thesis adds to the existing scientific studies in the technology and strategy context. It also touches on research development management since the proposed framework is designed for the R&D department. The project is done from a corporate perspective since the aim is to use the framework to improve the company's processes. Methods and techniques proposed in the course 'Research Methods' are used to perform this research.

This research is divided into four phases. The chapters belonging to that phase are displayed in the phase as well. In Figure 1.1, this visualization is shown.

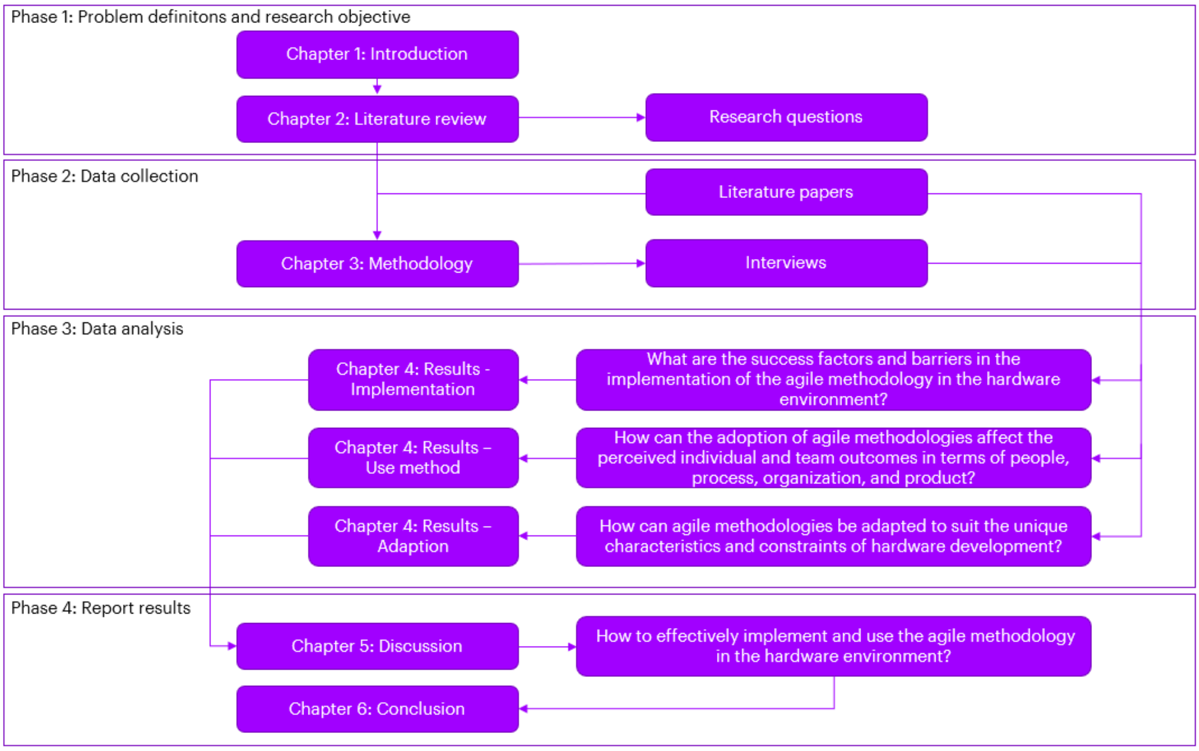

Phase 1: Problem definitons and research objective
Chapter 1: Introduction
Chapter 2: Literature review
Research questions
Phase 2: Data collection
Literature papers
Chapter 3: Methodology
Interviews
Phase 3: Data analysis
Chapter 4: Results - Implementation
What are the success factors and barriers in the implementation of the agile methodology in the hardware environment?
Chapter 4: Results – Use method
How can the adoption of agile methodologies affect the perceived individual and team outcomes in terms of people, process, organization, and product?
Chapter 4: Results – Adaption
How can agile methodologies be adapted to suit the unique characteristics and constraints of hardware development?
Phase 4: Report results
Chapter 5: Discussion
How to effectively implement and use the agile methodology in the hardware environment?
Chapter 6: Conclusion

Figure 1.1: Research flow diagram

The first phase is to define the problem, find the knowledge gap, and come up with the research objective. In chapter 2, literature research has been conducted to find relevant papers about the hardware industry, the use of traditional models, the use of agile models, and the use of hybrid models in hardware development. The focus will be on describing the methodology, followed by the benefits and challenges. The success factors of agile implementation in traditional models are described and the barriers to the implementation are given. The research and subquestions are derived from this literature research.

The second phase consists of data collection. The data is collected from both the literature and from the interviews. Semi-structured interviews are performed with 12 interviewees. The data collected from both sources are simultaneously conducted. In chapter chapter 3, the methodology of the research is described. The chapter gives an overview of how the research is done, how the data is gathered, and how the research questions will be answered.

The third phase is the data analysis. This phase overlaps a bit with the previous phase since the interviews are analyzed directly after they have taken place. The data from the literature and from the interviews are used to answer the 3 subquestions. For all subquestions, both data sources are used. This phase consists of three sections in which every section answers a separate subquestion and can be found in chapter 4.

The fourth phase is the reporting of the results. The analyzed data is discussed in this phase to answer the main research question. This will be described in chapter 5 together with limitations and suggestions for future research. Finally, in chapter 6, the conclusion of the research is given by answering the research questions.

2 Literature review

To get a deeper understanding of the implementation and use of agile in the hardware environment, a literature review is performed. To start this chapter, the literature that is found on the hardware development process is discussed. To get a better understanding of the problems in this environment the development process will be explained first. The challenges in the hardware industry are described and a comparison with the software industry is given. This is relevant to show why agile implementation in the hardware environment is not so straightforward.

In today's hardware industries, traditional methodologies are commonly used. These methodologies are explained to show the current way of working. Also, the challenges they are facing are highlighted to demonstrate what the current limitations are. Hybrid methodologies that are found are discussed as well, to show the current state of the art. In the second part of the literature review, the success factors and barriers of the implementation of the agile methodology into a traditional model are described. These factors are divided into several categories and are of use when implementing the agile framework. The goal of this part is to show what is already known about implementing agile and to use this knowledge to compare it with the results from the case study.

The papers used in this literature review are gathered by using Google Scholar and Scopus. Besides that, the references of the found papers are used to get more information about a specific claim that is made in a paper. If useful, these papers are used in this review as well. Keywords that are used in multiple combinations with each other are 'agile', 'stage-gate', 'hardware', 'high-tech market', 'semiconductor', 'hype agile', and 'challenges'.

2.1. Characteristics hardware industry

The hardware industry refers to the industry with physical components. This can be in different industries like cars, planes, semiconductors, health, computers, etc. The work that takes place can be manufacturing as well as assembling or designing. The classical hardware development process can be simplified into eight stages (Figure 2.1). Each phase in the process has a distinct set of goals, purpose, and exit criteria. The general process from start to finish consists of the following phases:

- Ideation or requirements: This is the initial phase of the whole process with the aim to define the scope of the problem and find proof that the proposed solution will work. It defines the needs and opportunities and ends with a proof of concept. Also, the manufacturing guidelines, technical specifications, testing procedures, and other constraints and conditions are determined in this phase. Everyone involved in the project should have enough knowledge about the hardware product after this stage. (Drechsler & Breiter, 2003)
- Concept: According to Arroyo (2023), the potential solutions found in the previous phase should be optimized in this phase and the different design assumptions should be tested. A user-centric approach is preferred in this phase to guarantee that the designed product matches the user's needs and can be adopted by the user. At the end of this phase, a clearly defined possible solution should be ready to be designed.

- Design / Engineering: Lima et al. (2015) stated that in this phase, the course plan for meeting the requirements is created together with a functional prototype. The design will specify how the purpose will be fulfilled. The phase starts with sketches and ends with a reliable and functional prototype that focuses on the core features of the final product. These prototypes are often made by a 3D printer to secure fast production. Mostly, multiple parts are produced to find the most efficient solution. These prototypes will be tested by users to get feedback on how to improve the design (Cadence, 2023).
- Prototype: In this phase, a real scale fully functional prototype is built that proves that the design could also work on a real scale. Some features need to be removed or redesigned to meet the requirements that were set earlier. Building a prototype is useful to check the feasibility of the product and find problems in an earlier stage. (Arroyo, 2023).
- Testing: Charvat (2003) stated that the real-scale prototype needs to be tested before the product will be produced on large scale. This phase is the final stage in which changes can be made to the product. The performance and functionality of the product will be ensured through a systematic assessment. Also, the manufacturing process is tested by testing the first batch of products and finding ways to optimize this process.
- Production: At this phase full-scale production and assembling take place. This stage guarantees the creation of a reliable product that efficiently and affordably complies with all design specifications. An economic and marketing plan should be ready at this stage to ensure a successful product launch (Arroyo, 2023; Cadence, 2023).
- Use and Maintenance: After production, the products go to the users for their intended applications. However, requirements may change or things might not work as expected. Therefore, maintenance of the product is also a part of the process. Nowadays, hardware products are on average only used for two or three years, due to the fast changes and developments (Drechsler & Breiter, 2003).
- Re-use: Most of the materials that are used in hardware products could be re-used in a way. This should already be taken into account while designing it. Conforming with standards is becoming a key issue due to the demand for more sustainable products (Drechsler & Breiter, 2003).

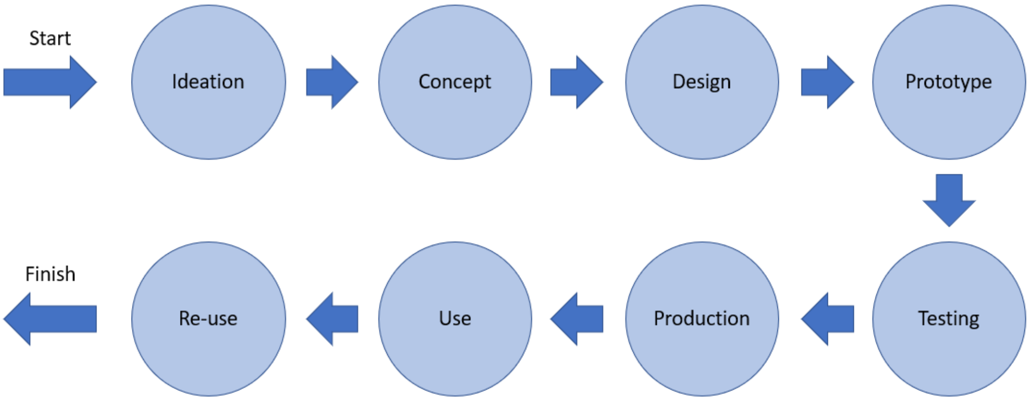


Figure 2.1: Hardware development process based on (Drechsler & Breiter, 2003; Arroyo, 2023; Cadence, 2023)

2.1.1. Challenges

Hardware development projects encounter some problems that are of importance for the decision on the management approach. These problems are different compared to software development for example and should be considered before implementing an approach. According to Drechsler & Breiter (2003), the main problems are the uniqueness of hardware systems, technically oriented management, weak planning, a high number of possible solutions, the individuality of designers, rapid technological changes, missing standards, and status monitoring. Campanelli et al. (2017) stated that since products are often created just once it is a more costs expensive project. The management of hardware projects is very technology-based and therefore often has insufficient management skills. According to Bjarnason et al. (2011), weak planning, weak status monitoring, and bad collaboration could be the results of this management malfunction. Since engineers are considered to be individual people and like to work on themselves the management needs to guide the team to achieve good collaboration. Furthermore, there is not one standard for parts that are used which makes it harder to match everything in the right manner (Drechsler & Breiter, 2003).

According to Brandl et al. (2018), hardware products compared to software products have higher costs for changes, are less malleable, better suited for quantitative changes instead of qualitative ones, have higher upfront design costs, higher lead-time, more expensive to test and validate so happens less. Mosher et al. (2018) adds that also the time to build a prototype takes between days to months while for software it takes on the order of minutes to hours. Also, hardware products mostly use standard parts and consist of large physical parts that cannot easily be changed after manufacturing. A project has high costs if something needs to change at a later stage, also leading to disruptions in the development cycle (Schuh et al., 2016; Mosher et al., 2018).

According to Mosher et al. (2018), team members are likely to be specialized in hardware development projects and skills are often not interchangeable while for software development projects many skills are interchangeable.

Furthermore, there exists a high dependency on suppliers in hardware development and communication is more complex among the development teams (Fuchs & Golenhofen, 2018). Augustin & Schabacker (2019), stated that physical prototypes are required making it more expensive due to more working steps and resources. Moreover, redesign is difficult for hardware development (Fuchs & Golenhofen, 2018).

2.1.2. Traditional model hardware projects

For a long time, only traditional models were used in the hardware development environment. Multiple traditional models are used among different organizations. Although there are some differences between these models they do have at least one thing in common which is the linear behavior, finishing one part first before going to the next part. In this section, multiple traditional models are discussed to illustrate their working principle. To be able to integrate the agile methodology into one of the traditional models, a clear overview of these traditional models should be made. Three models will be discussed here: the stage-gate model, the waterfall model, and the v-model.

Methods

1. Stage-Gate model

   In the 1980s, the stage gate system was created as a result of an in-depth study of successful ”intrapreneurs” within their large businesses. The lessons they learned with their practices of bringing successful new products to the market provided the basis of the early stage-gate model. The stage-gate model has evolved over the years and incorporated many new practices (Cooper, 2014). The Stage-Gate model is a framework for new product development that breaks down the process into distinct stages or phases, with specific gates or checkpoints between them. Each stage involves specific activities, such as market research, concept development, prototyping, and testing, and each gate requires a go/no-go decision before the project can move to the next stage (Jungwirth, 2022) (see Figure 2.2). The advantages of Stage-Gate include increased development speed, better quality, greater discipline, and better overall performance compared to informal development processes (Cooper, 2014). The aim for many of the tools associated with the Stage-Gate process is to decrease iterations by predicting the process from the start (Zhang, 2013; Sommer et al., 2015).

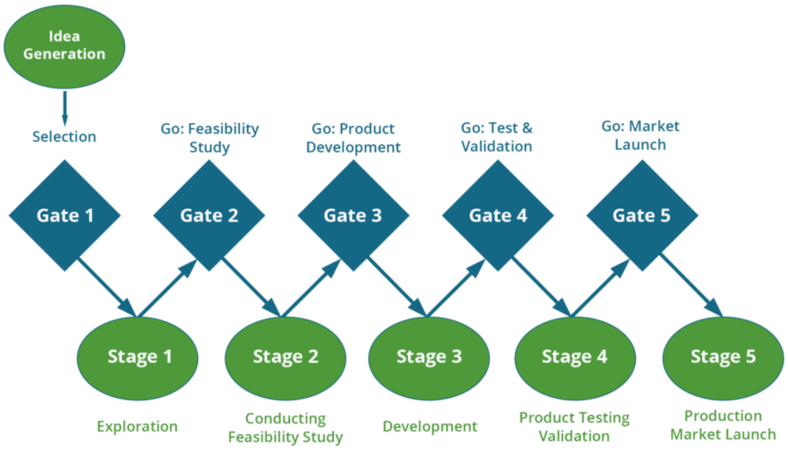


Figure 2.2: Stage gate model (Jungwirth, 2022)

2. Waterfall methodology

   The waterfall model was originally published in 1970 by Winston W. Royce and has been used widely in both the field of hardware and software development (Royce, 1970). In Figure 2.3, the waterfall model is shown to illustrate the different stages. This waterfall model consists of the following stages; requirements, analysis, design, implementation, testing, deployment, and maintenance. In this approach, the project moves on to the next stage only if the previous stage is completed successfully. If the step is fully completed, the project team cannot perform any changes or revert back to the previous stage. The idea behind this approach is that by spending a considerable time in the initial design phase all failures are already captured, so the rest of the process can run smoothly and there is no need to go back anymore. This approach is the most successful if the project is small and the requirements are all very clear. The separate phase can be performed by different teams since every phase is completely finished after it moves on. However, documentation is even more important in that case (Mccormick, 2012; Naveen, 2015). Some more advantages of the use of the waterfall method are the ease of the management of the project, ease to use and follow, cost-effectiveness, and a very small chance of performing rework (Naveen, 2015).

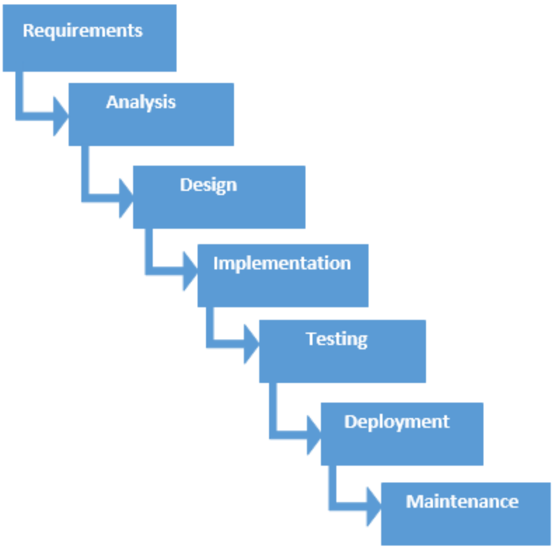


Figure 2.3: Waterfall model (Naveen, 2015)

3. V-Model

The V-model is in a way similar to the waterfall model since the next stage can start after the previous stage is completely finished. However, the v-model has the verification and validation steps after the implementation phase (Ebert & Paasivaara, 2017). An example of a v-model can be seen in Figure 2.4. This process consists of the following phases; the concept of operations, requirements, detailed design, implementation, integration and test, system verification, operation, and maintenance (Gutiérrez Rivas et al., 2012).

A system test plan is created before the development started to ensure the product meets the functionality defined in the requirements. Also, an integration test and component test are made to test both the integration of the whole system and the components on themselves. Due to these testing activities, the defects in the design can be found at an earlier stage resulting in a higher chance of success. Besides that, the approach is simple and easy to use since the phases are clearly defined. This model is most effective for small projects where the requirements are easily understood, clearly defined, and fixed. The needed technical expertise and the ability to use technical resources should be present to ensure the project succeeded. (Ebert & Paasivaara, 2017; Gutiérrez Rivas et al., 2012).

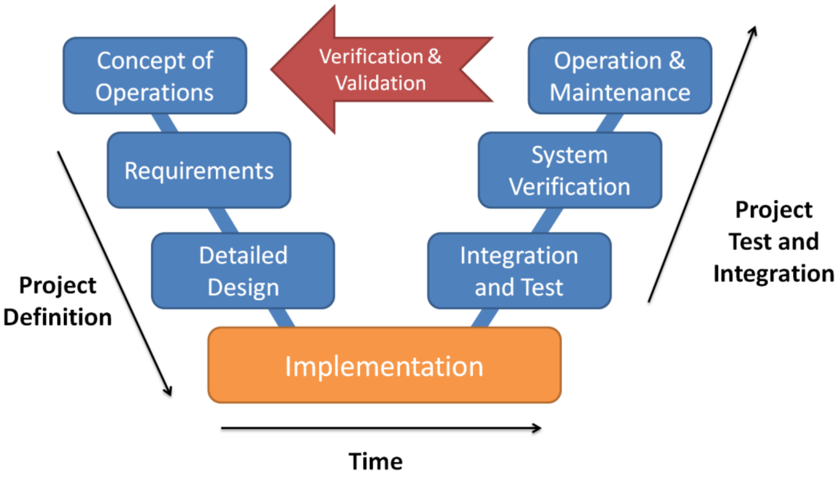


Figure 2.4: V-model (Gutiérrez Rivas et al., 2012)

Challenges traditional models

According to Cooper (2014) and Lenfle & Loch (2010), the stage gate model is often too rigid, linear, financially based, controlling, bureaucratic, loaded with paperwork, structured, context-based, too planned, and too much non-value added work to handle more innovative projects. Furthermore, it does not encourage experimentation since it is not adaptive enough (Cooper, 2014; Lenfle & Loch, 2010). The stage gate approach requires that the project or product is defined before the project moves into development. Fact-based product definition is a fundamental principle of stage gate (Cooper, 2014). However, often clients are not completely clear on what they need or want at the beginning of the project and therefore it is hard to get a 100% accurate product definition at the start.

Naveen (2015) states that also for the waterfall model, disadvantages are that the approach is not useful for large projects, less effective if requirements are not clear at the start of the project, and testing starts late in the process so the risk of finding issues relatively late in the project is higher. It is recommended to use this model only if the requirements are known and changes in the projects are stable.

Regarding the v-model, Ebert & Paasivaara (2017); Gutiérrez Rivas et al. (2012) highlighted that the disadvantages of this approach are that is it relatively rigid, and also test documents should be rewritten if changes take place in the process. This process takes more time resulting in more time to finish the project.

2.1.3. Agile method in hardware

The aim is to implement the agile methodology in one of the current traditional models in the hardware environment explained in the previous section. In this section, the agile methodology will be explained and the current benefits and challenges of the use of the agile methodology in the hardware environment are discussed.

Method

The agile methodology has been first described in 2001 by 17 software developers who agreed on a distinct working methodology that meets the demand and conditions of their industry. They wrote 12 principles based on 4 values in their Agile Manifesto that form the basis of the agile framework (see Appendix A) (Beck, 2001). Agility is the capacity to continually and swiftly react to and adapt to expected and unexpected changes in a dynamic environment, and to exploit those changes to one's advantage (Böhmer et al., 2015). An example of an agile model can be found in Figure 2.5. This model consists of the following phases; plan, design, develop, test, deploy, review, and launch. After the review phase, the cycle starts over again and only if the complete project is finished it moves to the launch phase.

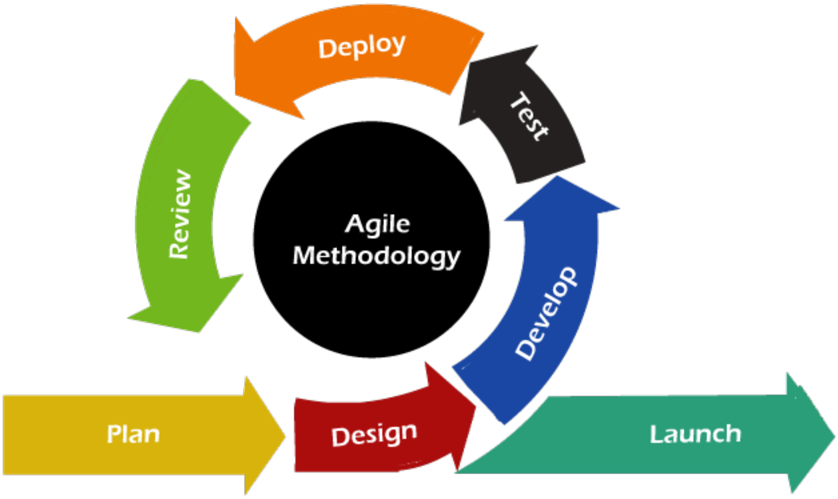


Figure 2.5: Agile model (Javatpoint, 2021)

Benefits of agile in hardware

The use of agile methodology in the hardware environment could result in some benefits. These benefits are divided into soft and hard criteria. Soft criteria are not quantitatively measurable and hard criteria are.

- Soft criteria

  Soft criteria are related to the aspects that are not quantitatively measurable, such as communication, transparency, or knowledge creation. Weichbroth (2022) found that these aspects are usually underestimated in terms of actual benefit due to the implementation of agile development, meaning that the real improvements are higher. Especially improved communication, increased flexibility, increased transparency, reduced reaction time to changes, improved project-related commitment, increased project effectiveness, and increased team morale are often underestimated. According to Naslund & Kale (2020), besides the just mentioned benefits also the risk reduces because of the use of agile since it makes the projects more predictable.

  A survey done by Schmidt et al. (2018a), looked at the expected and actual consequences of benefits, challenges, and conflicts of the agile method. According to the findings, agile development provides benefits such as enhanced communication, decreased reaction time to changes, increased project effectiveness, and quicker time-to-market. The top 5 actual benefits are soft variables such as enhanced communication, decreased reaction time to changes, more transparency, increased flexibility, and higher project-related dedication.

According to Bjarnason et al. (2011), it is easier to deal with a higher number of tasks and a higher number of project goals when using agile practices like sprints, or a scrum board that helps to create a better overview and structure. The sprints could set targets and help the teams to work toward them. Campanelli et al. (2017) adds that also the dependencies are better visualized by the use of these agile tools, and daily stand-ups help to communicate clearly about the dependencies. These stand-ups also improve the coordination between different offices at a number of locations in different time zones.

The most significant improvement can be seen in the improved communication within the team, followed by the reduced reaction time to respond to changes, and the transparency within the company. The lowest improvement was in the alignment with the corporate strategies followed by the improved development processes. In Table 2.1, the list of benefits can be found.

Table 2.1: Top 3 benefits of using agile for soft and hard criteria (Atzberger et al., 2019)

	Soft	**Hard**
1	Communication	Quality
2	Transparency	Time-to-market
3	Commitment	Costs

- Hard criteria

Hard criteria are aspects that are usually well quantifiable, such as costs, lead times, and quality. According to Atzberger et al. (2019), the improvements in hard criteria due to the use of agile hardware development are rather negligible compared to the soft criteria. Many companies have high expectations of reducing costs, shortening time to market, and improving adherence to schedules when implementing agile in their current model, but these aspects appear to be lower in the actual system. Although the improvements are not as high as expected upfront we can still see improvements compared to the situation before implementation. The biggest improvement can be seen in terms of improved quality followed by the improved time-to-market. This is all related to the shorter iterations that happen due to the use of Agile.

Especially in the hard criteria, Schmidt et al. (2018a) stated that a hype about the use of agile development seems to exist. This hype causes frustration when the realization comes that the desired benefits will not be achieved. There is a risk that the approach could be completely abandoned because companies do not see the desired improvement. Although there are some considerable improvements the company does not give value since they use the method with the wrong purpose.

According to Garzaniti et al. (2019), the development process is given pace by Scrum, which successfully boosts the team's productivity over time in terms of activities done. Improvement is assessed with an average of 3-5% on each succeeding Sprint.

A comparison is made between the project management and the research and development (R&D) department. According to Atzberger et al. (2019), there exists a difference in the expected benefits between these two groups. Especially in the soft criteria like communication, project-related commitment of all parties, and transparency in the company. Besides that, the order of highest-rated expected benefits is different for the two groups. While for project management, communication, flexibility changes, commitment, and transparency have the highest rank. The delivery times, flexibility changes, reaction speed, and time to market are the highest for the R & D. Both have the flexibility changes on the second rank, but besides that, the top 4 have different benefits. In Table 2.2, the top 6 benefits are displayed. The full list of benefits can be found in Appendix C.

Table 2.2: Highest rate benefits of using agile in hardware environment according to Atzberger et al. (2019)

	Project Management	**Research & Development**
1	Communication	Delivery dates
2	Flexible changes	Flexible changes
3	Commitment	Reaction speed
4	Transparancy	Time-to-market
5	Reaction speed	Customer satisfaction
6	Delivery dates	Communication

Challenges

Besides the benefits of the use of the agile methodology, there also exist drawbacks. Especially in the hardware environment, it is not so easy to integrate the methodology into the current processes. According to Böhmer et al. (2015), the introduction of agile methods into the field of hardware is approximately ten years behind agile software development. Meaning that there are still some major steps to take before the methodology could function the most optimal. If the capabilities, limitations, and possible adaptions of agile methods are not communicated properly, frustration and refusal could be the results (Atzberger & Paetzold, 2019). Therefore the challenges of the agile methodology in the hardware environment are listed below and will be discussed in more detail.

Schmidt et al. (2018a) looked at the expected and actual challenges of the use of agile in the hardware environment. He found that most investigated challenges are overestimated and companies expect larger impacts of the challenges than they actually cause. Only working incrementally and product modularization are underestimated in his study (Schmidt et al., 2018a). In Appendix C, all the identified challenges with the expected and actual impact are illustrated. In Figure 2.6, all the challenges found in the literature are shown.

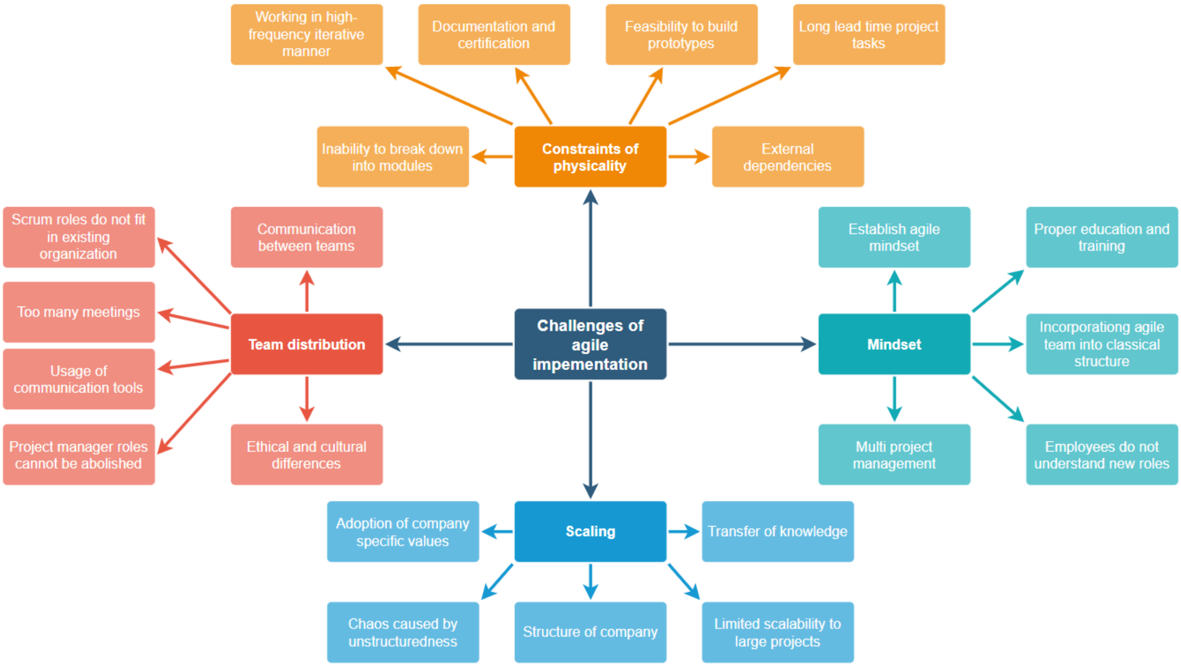

Working in high-frequency iterative manner
Documentation and certification
Feasibility to build prototypes
Long lead time project tasks
Inability to break down into modules
Constraints of physicality
External dependencies
Scrum roles do not fit in existing organization
Communication between teams
Establish agile mindset
Proper education and training
Too many meetings
Team distribution
Challenges of agile impementation
Mindset
Incorporationg agile team into classical structure
Usage of communication tools
Project manager roles cannot be abolished
Ethical and cultural differences
Multi project management
Employees do not understand new roles
Adoption of company specific values
Scaling
Transfer of knowledge
Chaos caused by unstructuredness
Structure of company
Limited scalability to large projects

Figure 2.6: Mindmap of the challenges of agile in hardware environment (Atzberger & Paetzold, 2019; Schmidt et al., 2018a)

Constraints of physicality:

Due to the physical nature of hardware products, several challenges arise that do not exist in the design of virtual products. Oveseen & Dowlen (2012) stated that compared to writing software the building and testing of hardware prototypes is a way more time-consuming activity. Therefore, frequent prototype production is hindered due to longer production times. Besides that, according to Mosher et al. (2018), it could be hard or sometimes impossible to break down the system into different modules. Resulting in larger systems with fewer iteration possibilities. Aiming to have a short sprint is therefore not possible. Furthermore, external dependencies can play a role in the constraints. Since parts in large machines are bought from different suppliers or sometimes even external partners that are developing a separate module, a project team is dependent on other projects as well Schuh et al. (2016). Atzberger & Paetzold (2019) mentioned that there could rise a mismatch between plannings of two projects resulting in an inefficient process. The agile methodology prescribed that the documentation should be minimized as possible. This could however be a bottleneck if multiple people are working on the same project or if someone is leaving the project.

Mindset:
According to Dikert et al. (2016) and Atzberger & Paetzold (2019), the way of working in an agile environment is completely different compared to the traditional models. Therefore, the organizational culture must change accordingly to assure a successful transformation. This also means that the mindset of all employees should be changed. To achieve this, proper education and training should be provided to them. The acceptance and common understanding to change and the long-running process of the maturation of agility in the company is part of this mindset as well (Atzberger & Paetzold, 2019).

Hunter (2020) stated that working with agile means that a lot of things are about people and managing things instead of real technology. The fundamental, underlying technical capability to do engineering sometimes reduces since it is more about getting stuff done instead of necessarily the right stuff. For some engineers, this is not what engineering is about, resulting in a misunderstanding of why they should implement agile.

Also according to Schmidt et al. (2018a), the biggest challenges are to implement an appropriate mindset, nest it into a company that is classically organized (embedding agile teams), and translate the practices from agile software development to hardware development. Although it is hard to change the mindset, all interviewees from his study acknowledged that interpretation and adaption to the agile methods and practices are mandatory to perform effective agile hardware development (Schmidt et al., 2018a). Furthermore, Schmidt et al. (2018a) found that employees on a personal level are more willing to react to change than expected while on an organizational level, the willingness to change does not seem to be large.

Scaling:
Mosher et al. (2018) stated that agility was intended to be used in small project teams so that communication is easy and fast. However, hardware development companies are often larger and therefore in principle not suitable for agility. Inadequate approaches to agile hardware scaling frameworks worsen the issue. Atzberger et al. (2019) found that larger teams require more communication, complex structure, and difficult decision-making. If the limitations, capabilities, and possible adoptions are not communicated properly refusal and frustrations could arise. Challenges arise in terms of the company culture and structures (Atzberger et al., 2019). Also, according to Schmidt et al. (2018a) chaos caused by unstructuredness, and limited scalability to large projects are challenges caused by the use of agile in hardware development.

Team distribution:
Atzberger & Paetzold (2019) and Atzberger et al. (2019) highlighted that the company should change to a more decentralized development where the company is divided into teams that make their own decisions. Good communication between the teams to align the same sprints should be guaranteed to effectively use this agile methodology.

Mosher et al. (2018) and Schmidt et al. (2018a) stated that working with agile requires decentralized decision-making and collaboration rather than centralized decision-making and cooperation. This contradicts with each other and causes challenges since the organizational structures need to be revised completely. On the same note, the scrum roles do not fit in the existing organization and project manager roles are hard to abolish Schuh et al. (2016).

Perceptions on agile

Some researchers stated that using the agile methodology in the hardware environment is nowadays a hype (Schmidt et al., 2018b). Some companies moved to the agile methodology because they did not want to lack behind their competitors even though they did not know if the method will be effective for their own company. There exists a discrepancy between the academic understanding of agile development and the motivations why the industry wants to implement the agile methodology (Schmidt et al., 2018b).

According to Nguyen-Duc et al. (2018), the perceptions of engineers on agile development can be reflected in three categories; principles, practices, and scope. Regarding principles, the agile methodology relates to short-driven evolution, less upfront planning, high speed of prototyping and development, and fast time-to-market according to the interviewees. The focus on internal collaboration instead of already defined processes together with the possibility to respond fast to unexpected changes due to the partnership and full control of development activities are mentioned as well (Nguyen-Duc et al., 2018).

Interviewees mentioned certain practices like sprint planning, product owners, frequent delivery, and Kanban. According to Nguyen-Duc et al. (2018), there does not exist a formal way of adopting practices from the agile methodology but much more customized adoptions. Besides the engineering activities like agile product development and rapid prototyping also business level is mentioned as the scope of agile (Nguyen-Duc et al., 2018). Figure 2.7 gives an overview of the before-mentioned categories and examples that are mentioned by the interviewees.

According to Ochodek & Kopczyńska (2018), the perceived best top-ranked agile practices are 1) Available customers, 2) Establishing shared vision, 3) Organizing everyday meetings, 4) Organizing demo meetings, 5) Providing easy access to requirements, and 6) Make requirements testable. These practices can be summarized into customer collaboration, shared vision, and continuous feedback/verification. Ochodek & Kopczyńska (2018) found that the perception of agile practices differs among the developers and the responsible person for the project. An example of this is that the practices that should support developers like having daily team meetings and negotiating iteration scope with the customer are perceived as less important by the developers than by the project manager/scrum master. Besides that, they found that you could use the frequency of the use of the practice to predict the perceived importance quite well (Ochodek & Kopczyńska, 2018).

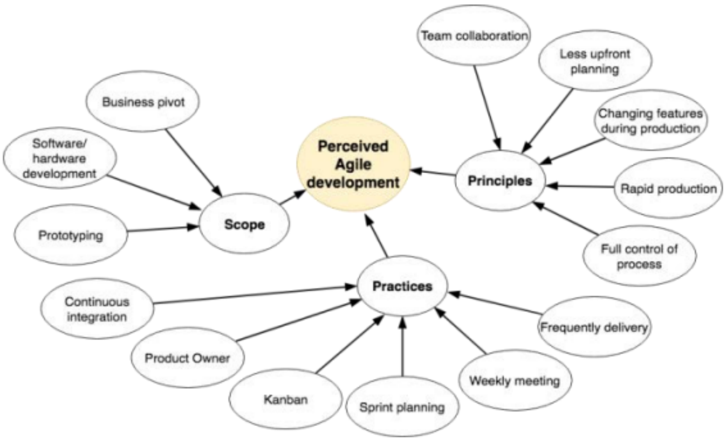


Figure 2: Thematic map of perceived agile development in hardware startups

Figure 2.7: Perceptions on agile development for hardware startups (Nguyen-Duc et al., 2018)

Learn agile by educational games

An effective way to teach people new practices is by the use of educational games. Omidvarkarjan et al. (2023), presented a novel training module that could realistically simulate the implementation of the agile methodology in hardware development projects. This game reflects the challenges of applying agile in the hardware environment by mimicking the processes, tools, and roles in the engineering teams. A positive learning outcome in terms of agreement with the principles could be observed by both inexperienced and experienced engineers. Therefore, using such a game could help companies to smoothly educate their people with the use of the agile methodology.

2.1.4. Hybrid method

Over time methodologies change by adopting different things from other methodologies. Hybrid methodologies combining two methodologies arise and become the new way of working in companies. Some are only compatible with the software development environment and others can be used in the hardware environment as well, although some problems still exist here. There are hybrid methodologies that combine the agile methodology and a traditional methodology and are used in the hardware environment. In this section, two hybrid methodologies used in the hardware environment are highlighted: the next-generation model and the SAFe model.

Methods

1. Next generation model

   Cooper (2014), showed a hybrid solution between the stage-gate method and the agile methodology and called it the ”next-generation model”. In this model stages and gates do still exist but the process and its function are different. The gates are still the moments where decisions are made and the gates where work gets done but this happens in a more dynamic, vibrant, leaner, and faster way. This can be summarized in three main categories; adaptive and flexible, agile, and accelerated (Figure 2.8) (Cooper, 2014).

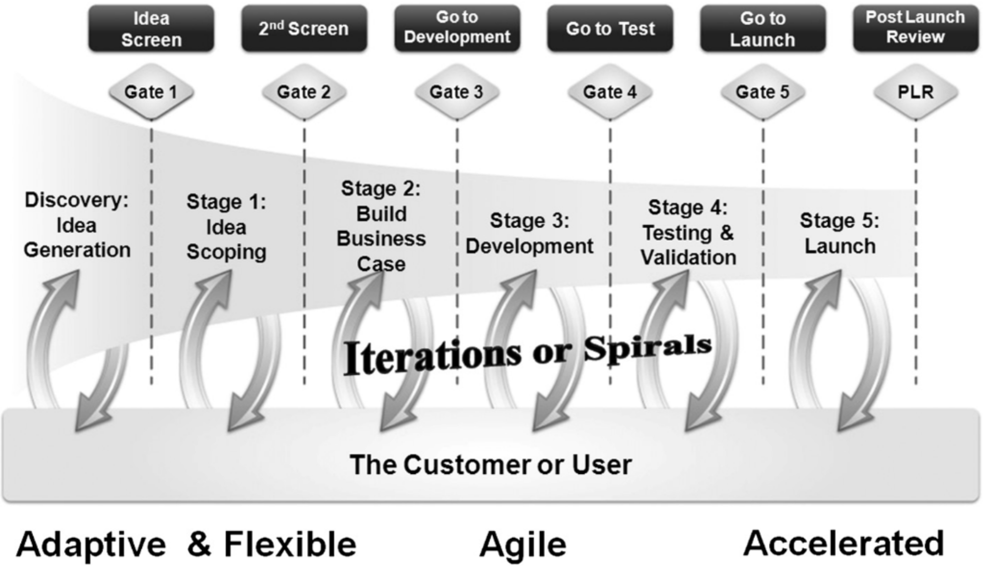


Figure 2.8: New generation model (Cooper, 2014)

- Adaptive and flexible: Incorporating iterations or spiral development to get the product earlier to the customers so they could give feedback and therefore have more build-test-revise iterations.
- Agile: introduce sprints and short time-boxed frames where the output of one frame is something that can be demonstrated to the stakeholder. Only documented things are not a viable output of one sprint. The system moves from one deliverable to the other and is based on the reduction of unnecessary activities, bureaucracy, and waste.
- Accelerated: The focus of the new system is on accelerating the processes by making use of properly resourced and fully staffed cross-functional teams in order to reach the maximum speed to market. The key activities within a stage and even entire stages themselves could overlap. This allows projects to continue if the information is stable and reliable instead of waiting for perfect information. Gates are therefore less important in this model Figure 2.9.

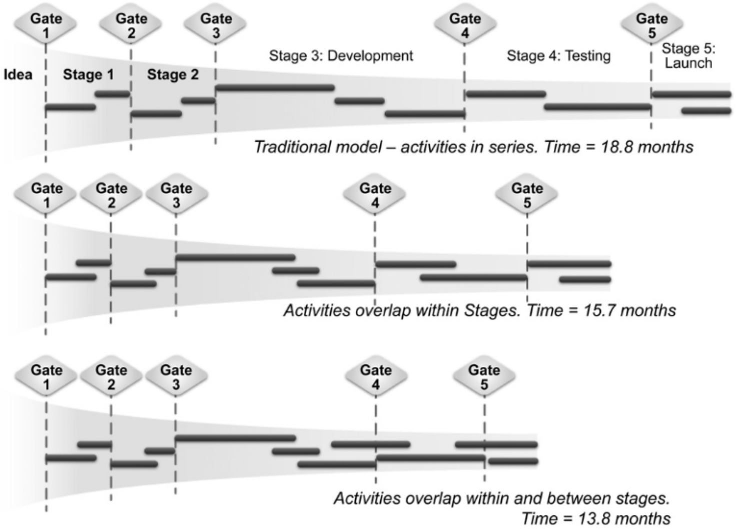


Figure 2.9: Overlapping activities within and between stages (Cooper, 2014)

Using agile in the stage gate model does not mean that the latter should be completely abandoned. Stage-gate can provide good support for the agile processes in terms of decision-making, structure, and communication (Cooper, 2014). Full dedicated cross-functional teams with access to the resources needed are key to maximizing the speed of the project. One major impediment is the lack of focus of people since they are spread over multiple projects and tasks. Also, the spreading of resources among too many projects can hamper fast delivery (Cooper, 2014). This next-generation model has been compared to the traditional model and found to be better. This comparison is done on multiple aspects to get a better overview of the framework. However, this approach has not yet been implemented in any company and is just a theoretical model. No information about the actual benefits can therefore be shown here (Cooper, 2014). Therefore, it could be interesting to test this model in a large company to get insights into the performance of the company while using this methodology.

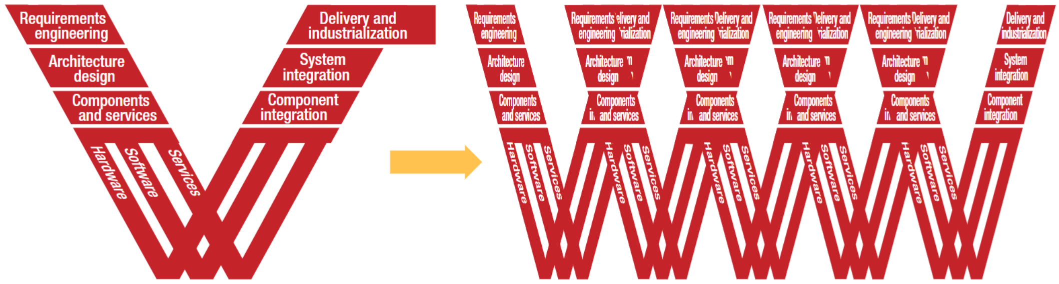
2. SAFe

Another model that is using agile practices in a traditional model is the SAFe (Scaled Agile Framework) methodology. This model is primarily aimed at practices in large organizations. It was developed by Dean Leffingwell and is based on agile principles and values, but scaled up to work in complex enterprise environments (Leffingwell, 2007).

The SAFe methodology provides a framework for coordinating multiple agile teams to work together on a large-scale project. It incorporates agile practices such as iterative development, continuous integration, and frequent delivery of working software/hardware. SAFe also includes additional practices and processes for managing larger teams and coordinating multiple projects (Leffingwell, 2007; Sandur Madhu Murthy, 2020).

The key components of SAFe include (Sandur Madhu Murthy, 2020):

- Agile Teams: These are the different teams that focus on certain areas of the project, each with its own Scrum master and Product Owner.
- Agile Release Trains (ARTs): ARTs are multiple agile teams that collaborate to deliver a larger project increment. The ARTs typically operate in a cadence of 8-12 weeks.
- Program Increment (PI) Planning: This is a planning session at which stakeholders and teams meet to plan and coordinate work for the upcoming period.
- SAFe Lean-Agile Principles: These principles serve as the foundation for the SAFe approach, which emphasizes decentralized decision-making, teamwork, and quick learning.



Classic V model for systems engineering:

- Goal: Collaborative development of safety-critical functional components
- Organically grown synchronization points between hardware, software, and emerging services
- Governance-oriented heavy processes with insufficient flexibility and agility

Agile scaling frameworks:

- Goal: Flexible collaboration across the lifecycle with adaptive architectures and assured quality
- Full vertical abstraction layers with clear interfaces and synchronization points
- Flexibility and agility while pertaining to quality demands

Figure 2.10: Agile framework comparison (Ebert & Paasivaara, 2017)

SAFe is a popular methodology for implementing agile practices in larger organizations because it provides a structured framework for coordination, cross-functional collaboration, and communication across multiple stakeholders, departments, and teams. Besides that it gives a structured framework for planning, executing, and tracking which improves transparency and visibility. Furthermore, it comes with flexibility, adaptability, and rapid learning from the agile methodology which provides a project to act quickly to changes (Ebert & Paasivaara, 2017; Sandur Madhu Murthy, 2020).

Challenges

As mentioned earlier, the next-generation model is only a theoretical model. It has not yet been tested in a company or real-life case. According to Cooper (2014), the proposed approach in which the new phase already starts before the previous gate has been closed could be beneficial, but could also lead to more rework due to changes made in a previous phase. Furthermore, the model does not include how to change the mindset of the employees and how people should be educated. Before this model could be implemented in a real life case these things should be considered first (Cooper, 2014).

Regarding the SAFe model, according to Ebert & Paasivaara (2017), potential drawbacks are the complexity of the framework that requires significant effort and time in understanding and implementing the framework in the current processes. It also comes with costs of implementation since people need to be trained and consulted to use the new approach. Furthermore, Sandur Madhu Murthy (2020) stated that a culture shift needs to take place which can be hard for some companies that have a hierarchical structure. Therefore, resistance against the change can occur in the organization. The heavy emphasis on planning could also be experienced negatively by some employees since it causes a lack of flexibility and could feel like time-consuming practice. Therefore some people think it increases bureaucracy and slows down decision-making and innovation (Ebert & Paasivaara, 2017).

2.2. Agile implementation

This chapter is divided into two subsections. Firstly, the success factor of agile implementation will be discussed and secondly, the barriers of agile implementation are highlighted.

2.2.1. Success factors

Success factors are aspects of a project that should be utilized to achieve success. Traditional examples include, among other things, qualified project team members and clearly stated goals (Pinto & Slevin, 1988). Translating this to the implementation of the agile methodology means that success factors define what elements are useful to obtain in an organization during the implementation. Since every company is different, it is hard to generalize it for all companies. However, some components are returning in every agile implementation.

Naslund & Kale (2020) did research into the critical success factor of the implementation of the agile methodology in traditional approaches and found that the most mentioned success factors are related to ”people” and ”culture”. They are almost all concerned with how to increase the motivation and engagement of the employees through culture shift, mindset change, coaching, and training. Lack of investments, lack of motivation, and change resistance are also mentioned by Dikert et al. (2016) as things that need to be considered when implementing the agile methodology. Management plays an important role in the establishment of the new way of working. Kalenda et al. (2018) found that in an organization where the change has a bottom-up approach, management can become reluctant to change, making it hard to implement the approach among the whole organization. While having a top-down approach could lead to a disconnect between the team members and the management since they feel that it was not a collaborative decision but rather a mandate (Conboy et al., 2011), with a lack of motivation as a result. Thus, training, coaching, engaging, and motivating employees are critical to transformation success (Naslund & Kale, 2020).

The success factors in this analysis are divided into 3 main categories; management, people/employees, and organization. In the following section, we will discuss them one by one and elaborate on the details of the success factor. In Appendix D, the full list of success factors is shown.

Management

According to Dikert et al. (2016), the availability of management support is crucial in the successful implementation of the agile methodology. Kalenda et al. (2018) stated that management should be educated about agile so that they can create a vision, business goals, and strategy for the transformation, resulting in a better understanding of the practices and the processes of the management. Furthermore, according to Naslund & Kale (2020) management should initiate the change, motivate the team members, and make their support visible to the team members.

Also, strong leadership that has the willingness to take risks and could withstand external pressure to follow the traditional processes are also important success factors. According to Senapathi & Srinivasan (2013), the management should communicate that the change is non-negotiable, need to show strong commitment even if problems occur, and convince the employees that agile will bring the desired results. Also, Dikert et al. (2016) stated that the role of change leaders should be recognized and they should be engaged without the baggage of the past.

The way of communicating about the change, both internally and externally, influences the success rate of the implementation (Campanelli et al., 2017; Gupta et al., 2019). Especially in the beginning, management should intensively communicate, arrange social events, and create/communicate positive experiences according to Koehnemann & Coats (2009). Close connections and constant communication between team members and different teams are necessary. Paasivaara et al. (2018) suggested that new communication tools could be implemented such as a new communication flow, and the use of common backlog could increases the success of the agile implementation as well.

Moravcová & Legény (2016) stated that a transparent environment should be created for openness in the team without the fear of feeling observed or judged. It should be possible to discuss problems that occur during the implementation to improve teamwork. This could increase the collaboration and engagement of the team members Paterek (2017); Gupta et al. (2019).

People/Employees

The level of autonomy is a significant factor in the success of the use of the agile methodology. Employees should get the right balance of oversight and autonomy with a healthy level of remaining centralized decision-making according to Mancin (2016) and Paterek (2017). They state that employees should be empowered to make their own decisions and be self-organized.

According to Dikert et al. (2016), the level of knowledge and expertise on agile practices should be increased among the whole organization that is implementing agile. This could be achieved by providing training on the agile method for both management and employees. By involving change agents and agile champions this knowledge exchange could be more efficient. Kalenda et al. (2018) stated that hiring external experts with broad and deep familiarity with agile development could accelerate the process of knowledge exchange. Furthermore, according to Senapathi & Srinivasan (2013), exchange programs of agile coaches with other organizations could help. It is advisable to have systematic training of people through recurring events, presentations of external experts, or communities of practice. In these trainings, it is important to coach the employees by listing and asking questions and not dictating them and forcing them to change. Also, enough coaches should be available in an organization to guarantee good guidance. Naslund & Kale (2020) highlighted that pair coaching, which combines an expert with domain knowledge with an agile expert, proved to provide more efficient and accurate coaching. Enough resources with engineering, business, and agile knowledge to provide these trainings should be present and enough time and space for people should be made available to learn and adapt to the new way of working (Kalenda et al., 2018).

Koehnemann & Coats (2009) stated that a common view on the change is one of the most important success factors of the implementation. Roles, responsibilities, and common definitions should be defined and wrong ideas and misconceptions should be pointed out. Also, according to Dingsøyr & Moe (2013), the importance of the culture shift that is necessary for the agile transformation should be emphasized so that an agile mindset could be created. It should be clear that some existing company rules no longer apply and that the focus will be on agile values. Furthermore, the organization should have a positive attitude towards agile methods and be aligned with those values (Kalenda et al., 2018).

According to Bjarnason et al. (2011) it is important to engage everyone in the organization in the new way of working to achieve a successful implementation. A company can start with agile supporters and use the persons with previous agile experiences to involve the rest. Also, employee buy-in could connect the employees to the company. The motivation of employees needs to be maintained in the new teams and roles by including them in the decisions making in the teams (Pinto & Slevin, 1988; Conboy et al., 2011).

Organization

From an organizational perspective, according to Mosher et al. (2018), the structure of the organization should be aligned to gain success in the transformation. Meaning that the management and teams are using the same approach. The role of middle management in the transformation should be clearly defined upfront. Naslund & Kale (2020) stated that it is advisable to keep the teams small and have development teams at the same location. Existing roles in the company should be reviewed and adapted and be aligned and communicated with the employees. Achieving symbiosis between the formal and informal organizational structure could help to align the employees even better. A common agile framework should be used for the whole organization so that everyone understands how it works and is aligned among the organization. Kalenda et al. (2018) noted that long-established teams should be restructured to create an "awakening" effect, which holds the teams from being stuck in their old habits. The number of projects per employee should also be increased and people should work on 1 team.

It is essential to realize that every company is built differently, uses other techniques, and has other organizational structures therefore there is not only one way to use the agile method (Dikert et al., 2016). As a result, agile practices need to be adapted to fit the individual context and needs to be customized and implemented carefully. According to Weichbroth (2022), the implementation should be a step-by-step process in which continuous learning plays an important role. By mapping the new approach to the old way of working the implementation could be easier. A single approach needs to be chosen to align everyone in the organization and set a clear goal.

Naslund & Kale (2020) proposes to start with a pilot so that the management could gain acceptance of the change. Also, one could gather insights from the pilot that can be of use for the large transformation. Good preparation for this first program increment is therefore important. Moravcová & Legény (2016) stated that after the general direction is set for the transformation, the company should calculate and communicate the expected costs, benefits, and risks of the transformation.

Mancin (2016) highlighted that the availability of decent tools and infrastructure to allow teams to transition their work procedures are helpful in the transition to an agile approach. The use of existing platforms if they can be adapted to future evolving needs is the preferred choice. Investments in system improvements should take place all the time to ensure continuous improvements over time. Also, the tools need to be aligned with each other to ensure a good transition of the organization (Kalenda et al., 2018).

2.2.2. Barriers

Implementing agile in a traditional model in large organizations is complex and comes with several challenges. Large projects need adequate communication and coordination between teams, management of internal and external dependencies, the involvement of other nonagile groups, and the inclusion of relevant people in the process. In this section the barriers of the implementation of the agile method in the hardware environment are highlighted and discussed. The section is divided into 3 categories: Management, people, and organization. In Appendix D, the full list of barriers is shown.

Management

From a management perspective, a challenge could be that the management is not willing to change. According to Dikert et al. (2016), especially in middle management where managers are not expected to manage teams anymore. Besides the confusion about what their new role will be, resistance arises because they are feared that they are not needed anymore. Also, Gupta et al. (2019) noted that managers could change to micromanagement of teams causing several problems like the loss of interest of the team in meetings and therefore stopping attending them, since they were not responsible for communication and synchronization anymore and therefore found the meetings useless. According to Bjarnason et al. (2011), leadership is often stuck in the old bureaucracy and wants to keep to the old system. The top-down mandate also creates resistance among the employees.

Conboy et al. (2011) highlighted that inconsistent communication or miscommunication could be dramatic for the implementation of the agile methodology. Misunderstanding about the agile concepts could arise and cause internal conflict. Also, a lack of guidance from the literature could hamper the organization of implementing the new method (Mosher et al., 2018). Kalenda et al. (2018) stated that having a distributed environment where teams are working at different locations makes close relationships of constant communication and team spirit hard to obtain. If people do not understand the work of others they do not want to rely on that, which makes the work inefficient. Also, all the knowledge and infrastructure need to be built at every location which takes more resources. Furthermore, the interpretation of agile is different between teams so communication about a clear view of the use of agile is important. Moreover, Dikert et al. (2016) noted that high workload and pressure could impede communication in the team, resulting in teams that postpone meetings, resulting in less communication and even more inefficient work.

Weichbroth (2022) highlighted that transparency is one of the main building blocks of agile implementation, a lack of transparency could be a problem. Transparency could be hampered by the distributed teams that are not working at the same location and therefore not able to share all their data. Teams that are unwilling to admit their mistakes could be the problem as well. Teams could not learn from others' failures, resulting in less efficient processes.

People/Employees

From a people perspective, some more challenges with the implementation of agile could arise. Long & Starr (2008) found that resistance among the employees increases due to the new responsibilities that agile development brought to teams. Teams are expected to be self-managed in agile development, but not everybody was pleased. Teams did not want to solve their new problems. Also, according to Dikert et al. (2016)for a lot of people, it is challenging to work according to this autonomous way of working. Therefore, they should learn how to work with this method which takes time. Training is important to teach people how to work with the new methodology. Weichbroth (2022) reported that the lack of available training, knowledge inside the company, or coaching could lead to bad implementation of the agile methodology. Also, the lack of business understanding, and the knowledge about the customer and product could result in less effective implementation (Kalenda et al., 2018). Lack of resources for training, coaching, or other learning activities could also be seen as a barrier to effective implementation. Hajjdiab & Taleb (2011) found that organizations underestimated the difficulty of agile implementation. Some organizations do not have sufficient knowledge in-house and refuse to use external experts because of restricted financial resources.

According to Koehnemann & Coats (2009), it is hard to effectively implement the agile methodology in the hardware environment as the mindsets of employees in an organization are not aligned. People have skepticism towards the new way of working and therefore do not want to change to the new method. Another issue could be that people keep following their old commitments since organizational culture is at odds with agile values. General resistance arises and other functions are unwilling to change to the new method (Weichbroth, 2022).

Dikert et al. (2016) found that the lack of recognition for the effort of employees to for example improve the development process could hamper them from helping further improve the implementation. The lack of commitment and teamwork is disastrous for a smooth implementation. Too much pressure and workload also limit the motivation of employees.

Organization

From an organizational perspective, one of the main challenges is the alignment of the new teams to the existing structure. Kalenda et al. (2018) found that the use of old and new approaches side by side could be difficult since the structure is still not aligned and people do not know where they belong. Therefore, the integration with non-agile parts of the organization is important during the start of the implementation. The gap between long and short-term planning should be made clear and aligned in the whole organization. Mosher et al. (2018) stated that the role of the middle management should be communicated clearly as well as the role of the other managers. Weichbroth (2022) reported that misalignment inside a company could cause problems since teams do not want to rely on agile teams if they do not know whether the agile teams would deliver their work on time. Furthermore, according to Dikert et al. (2016), companies have challenges in rearranging physical spaces if that is needed for the new team setup, which could hamper the fast implementation of the new teams.

Bjarnason et al. (2011) stated that bad implementation happens if the agile method is customized poorly. The method is simply not applicable in the company and people refuse to work with the new method. People tend to refer to the old way of working and therefore stick to that approach. Challenges arise in the adjustment to incremental delivery pace and in the adjustment of product launch activities.

Also, according to Kalenda et al. (2018), the too fast roll-out of the use of agile in the current way of working could cause problems in the effectiveness of the implementation. Since processes are not yet ready to be used it causes problems in communication, alignment, and suitability.

Tools need to be adjusted as well to match the new way of working. Issues like quality assurance, accommodating non-functional testing, and lack of automated testing could occur during the transition to a new method (Weichbroth, 2022). It should be important to recognize these issues and prepare for the implementation of the integration of the method.

Kalenda et al. (2018) found that some companies experienced difficulties with the measurement and monitoring of the progress of the agile implementation. The main problem was to find meaningful results since they did not know what to measure to get this. First, the definition of progress and success needs to be clear. Secondly, a plan of how to measure this progress and success needs to be made upfront so that it can be used during the implementation.

2.3. Conclusion

The use of the agile methodology has a lot of advantages over the traditional development approaches (e.g. stage-gate model, the waterfall model, and the v-model). Especially, communication, transparency, commitment, and flexible changes increased due to the use of agile. Since the agile methodology was originally developed for the software environment it is not yet completely compatible to be integrated into the hardware environment. Because of the physical nature of hardware products, scaling challenges, mindset, and organization structure challenges additional troubles with the implementation of agile methodologies arise. These problems have been summarized in this chapter. Cooper (2014) designed a hybrid methodology that could be used in the hardware environment. This method uses parts of the agile methodology and combines them with the traditional approaches. Also, the SAFe model is used as a hybrid method with parts of agile. The use of these models in large hardware company structures is however not yet so successful. Success factors for the implementation of agile in the hardware environment are essential to be achieved in the implementation. Also, the barriers should be taken into account and tried to be avoided. These success factors and barriers are divided into three categories; management, people, and organization.

Mosher et al. (2018) did research for a company where agile is implemented in the hardware environment in the space industry. This research was primarily focused on the planning of agile sprints and estimations of this work. Also, Weichbroth (2022) published a research around the implementation of agile in the hardware industry. However, a framework that consists of both the implementation part and the effective use of agile is not found in the literature. Also, knowledge about the implementation of agile in the semiconductor industry is not described. Previous scholars focused on agile implementation and barriers but not on how to adapt the methodology to suit the hardware characteristics. Also, the researchers focused on medium size companies with sustainable growth. There is yet no research on a fast-growing company that is in the transition of the implementation of the agile methodology.

Therefore, this research focuses on both the implementation and use of agile in the semiconductor industry, in which a fast-growing company will be used as a case study. To design this framework, the success factors and barriers to the implementation and use of the methodology in a large company should be described together with the perceptions of the engineers and team leads. Researchers could use this framework and compare it with other methodologies. Also, companies could use this framework to help them implement the agile methodology in their existing model.

2.4. Research questions

As mentioned in the introduction the research and subquestion of this thesis will be:

RQ: How to effectively implement and use the agile methodology in the hardware environment?

- SUB 1: What are the success factors and barriers in the implementation of the agile methodology in the hardware environment?
- SUB 2: How can the adoption of agile methodologies affect the perceived individual and team outcomes in terms of people, process, organization, and product?
- SUB 3: How can agile methodologies be adapted to suit the unique characteristics and constraints of hardware development?

The answer of SUB1 gives a better understanding of how to successfully implement the agile methodology in the traditional model in the hardware environment. With implementation, the transition phase from the traditional approach to the agile/hybrid approach is meant. Success factors are defined as aspects of a project that should be utilized to achieve success. Examples are management support, way of communicating, and use of pilots. Barriers are defined as aspects that hinder or limit the smooth implementation of agile.

The answer of SUB2 gives a better understanding of how to use the agile methodology in the hardware environment. It focuses on the benefits and challenges that arise because of the use of agile. Both the individual and team outcomes are discussed. Everything related to people is captured in the people's outcome. This varies from mindset to availability of training, for employees and for management. The process is related to everything that is an outcome of practices that are happening in the company, like communication, meetings, and productivity. The organization aspect is defined as everything that is related to the structure of the organization, like team structures, the structure between management and project teams, and specialization. Product is defined as the things that are related to the output of the company, which is a physical product in this environment.

The answer of SUB3 gives a framework of how the agile methodology should be adapted to suit the characteristics of the hardware environment. The categories of the perceived outcome are linked to the categories of the proposed framework.

The answers to the subquestions are used to finally answer the main research question. This will be done by providing two frameworks, one for the successful implementation and one for the successful use of the agile methodology.

3 Methodology

3.1. Selection of the research approach

3.1.1. Research Strategy

To determine how to effectively implement the agile methodology into the traditional model in the hardware environment several research strategies are required. Literature research is required to find factors that could affect the success of the implementation of the agile method. Also, the barriers to implementing the agile methodology should be found. These factors and barriers can be used to find what actions are needed for successful implementation. For the second part, qualitative research is needed. This part will be more exploitative and will dive into the perceptions and best practices by doing a case study with multi-person interviews. This case study is primarily performed in the electrical engineering department of a large high-tech company. The company is focused on the semiconductor industry and has both a large R&D department and a large manufacturing department. The reason for choosing this company and this department for our research has two components. The first reason is that the company is in the middle of this large transition. Since the software has already made the transition, some hardware teams as well but also some teams have not yet changed it is a very relevant and valuable environment to perform this research. Reason two is that the company is one of the key players in the semiconductor industry meaning that it could be one of the most complex cases to implement the agile methodology. The reason for choosing electrical engineers is a more practical reason. Initially, both mechanical and electrical engineers were asked but because of availability, the decision was made to go for the electrical engineers.

Multiple interviews are being executed and compared to each other. The decision to choose qualitative research instead of quantitative research is because qualitative is more eligible for this kind of exploratory research purposes. This case study consists of three parts, one with engineers of the software department, one with the engineers of the hardware department, and one with the team leads of the hardware department. The first part is only to verify what the success and barriers were with the implementation of the agile methodology in the software environment that can be used for the implementation of the agile methodology in the hardware development. The second and third part consists of interviews with engineers and team leads of the mechanical engineering department of the Research and Development (R&D) department. A case study protocol can help to make the research more repeatable and transparent, according to (Yin, 2003). In Appendix E, the interview protocol is displayed. The sampling method, overview of the case study, and data collection procedures are highlighted in this chapter.

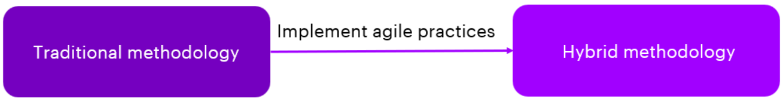


Figure 3.1: Simplified conceptual framework

3.1.2. Population and sampling

The population of this study consists of software and hardware employees in the semiconductor industry in the Netherlands. The semiconductor industry is relevant for this research since it is a fast-growing innovative industry, in which companies are constantly aware of the competition in the market. Since we are based in the Netherlands it is the most convenient to research the Dutch semiconductor industry. Non-probability sampling is used to collect the data. The selection of interviewees is non-random and chosen based on convenience. This decision is made since the research is done in collaboration with one of the clients of Accenture. Sampling bias and selection bias are risks that could occur by using this convenience sample. For this research a single case design study has been chosen, meaning that the case will only focus on one company.

As mentioned before, the company of this study is active in the semiconductor industry. It produces and designs hardware and software components for large machines. The company has a large R&D department as well as a large manufacturing plant. The company already started (partly) implementing the agile methodology. For the software development department, this implementation was successful. For the hardware development department, the company started implementing agile tools but is facing some serious issues. Therefore this company is a good fit for the research. Since there are three units of analysis, the software engineers, the hardware engineers, and the hardware project leads, it is considered to be an embedded single-case design. By choosing three units of analysis a comparison between these three groups can be made.

For this research, 12 people are interviewed, all from the same company but with different functional roles. The interviewees are divided into software engineers (SW engineers), hardware engineers (HW engineer), and hardware team leads (Team lead HW). The reason to go for three different groups is to get a more representative overview of the situation. Since software engineers are already using the agile methodology they could give a good description of the benefits and challenges in the long term. Besides that, they can use their experience to give a more accurate description of the potential benefits of agile in hardware. The hardware engineers are the core of this research and therefore form the largest group of interviewees. They experienced the transition and should work with the new practices of the agile methodology. They could give a clear description of the current situation and what they like and do not like. The team leads have a manager role in the company, they are in charge of the decisions that are made about the use and implementation. Also, they are facing other challenges than the hardware engineers which could broaden the view on the use and implementation. In Table 3.1, the list of interviewees is displayed. The number of years at the company is added for every interviewee to show their experience in the company. Since the start of the implementation of the agile methodology was 3 years ago, almost all interviewees have experienced the transition.

Table 3.1: List of interviewees

Interviewee	Group	Code	Years at company
Interviewee 1	SW Engineer	SW1	24
Interviewee 2	SW Engineer	SW2	1
Interviewee 3	HW Engineer	HW1	15
Interviewee 4	HW Engineer	HW2	7
Interviewee 5	HW Engineer	HW3	2
Interviewee 6	HW Engineer	HW4	5
Interviewee 7	HW Engineer	HW5	15.5
Interviewee 8	HW Engineer	HW6	16
Interviewee 9	Team lead HW	TL1	27
Interviewee 10	Team lead HW	TL2	28
Interviewee 11	Team lead HW	TL3	3
Interviewee 12	Team lead HW	TL4	11

3.2. Research flow diagram

This research is divided into four phases. This has been displayed in Figure 1.1 in the introduction. In this section, the phases are further elaborated.

3.2.1. Phase 1: Problem definition

To identify the problem and come up with the research questions, a literature review has been conducted here. The papers used in this literature review are gathered by using Google Scholar and Scopus. Besides that, the references of the found papers are used to get more information about a specific claim that is made in a paper. If useful, these papers are used in this review as well. Keywords that are used in multiple combinations with each other are ‘agile’, ‘stage-gate’, ‘hardware’, ‘high-tech market’, ‘semiconductor’, ‘hype agile’, and ‘challenges’.

3.2.2. Phase 2: Data collection

The data to answer the sub and research questions is gathered by both a literature review and by doing interviews. Data from both sources are compared with each other.

Literature review

The literature review is done by using Google Scholar and Scopus. The first part of this literature review focuses on the success factors and barriers of the implementation of agile in the hardware environment. Keywords like ”Success factors”, ”Barriers”, ”implementation”, ”Agile”, ”Hardware environment”, ”Hardware development”, ”Hardware” are used to find relevant papers. The data from different papers are combined and divided into three categories: management, people, and organization.

Interview

This research makes use of semi-structured interviews. This means that the questions are already made upfront but there is also room for follow-up questions during the interview. These follow-up questions are used to get more specific information and go more in-depth on interesting topics.

These interviews will be conducted primarily online via MS Teams, due to the ease of planning the interviews. The interviews are audio recorded to allow the researchers to make transcripts of the interview and use that for a summary. The transcript is made by uploading the .mp3 file into MS Word and automatically generating the transcript. This transcript is checked manually and a summary of the transcript is made. This summary will be shared with the interviewee to check for the right interpretation of the answers. The summaries are made anonymous to avoid people from giving socially acceptable answers and be in line with privacy concerns. Interviewees are asked to sign an informed consent form prior to the interview. This form consists of an agreement on the data sharing, data storage, potential risks, and publication of the thesis. The informed consent form has been approved by the Human Ethical Resource Committee (HREC) of TU Delft and can be found in Appendix B.

For the case study, three different groups of people are interviewed:

- Group (1) consists of around 2 software engineers from the company.
- Group (2) consists of around 6 engineers from the electrical department of the R&D department.
- Group (3) consists of around 4 team leads of the R&D department.

During the interview with the software engineers (group 1), I want to get to know what the success factors and barriers were with the implementation of the agile methodology in the software development department. I want to hear their opinion about the possible implementation in the hardware environment and what agile practices they suggest to implement and what not. I want to compare this with the literature to use this information in the interviews with the second and third groups.

During the interviews with the engineers (group 2), I want to get a better understanding of what things need to be done according to them to successfully implement the agile method in the current traditional model and how likely they think the implementation will succeed. I also want to hear the limitation of the implementation with the potential solutions they can think of and if it is possible to implement those solutions immediately. I am curious to know how likely they think this implementation will succeed and what the potential benefits are in their opinion. Secondly, I want to get a deep understanding of the current way of working and want to get to know the strong and weak points of the approach. I want to hear what they need and what they are missing in the current approach. I want to focus here on their perception of the use of agile, the communication between departments and internally, and the achieved performances. I want to ask this both for the individual outcome and for the team outcome. I want to verify whether the time an engineer works at the company influences the perception of the likelihood of success.

During the interviews with the project lead (group 3), I want to verify whether the perception of the project leads matches the perceptions of the engineers. The first part of the interview is therefore similar to the interview with the engineers. First, the identification of the current system with its strong and weak points (success factors and barriers). Followed by the potential solutions for these barriers and the possibilities of implementation, and finally the perception of the success rate and potential benefits (outcome in terms of usage, communication, and performance). The second part of this interview is more on the long-term vision of the use of agile and how to sustain the implemented changes that are proposed. Also, the responsibilities of managing an agile team will be discussed.

The interview consists of five parts that are divided into 4 phases. In the first phase, a short introduction is given about the topic, and background questions are asked to identify the role of the interviewee in the company. The second phase focuses on the success factors and the barriers of the implementation of the agile methodology in the hardware environment. Interviewees are asked how they have experienced the transformation and what they consider to be a success factor. Also, aspects that hinder the implementation are asked. The third phase consists of two parts: the current situation and the future situation. These two parts will be asked simultaneously in this phase. The part is divided into five main subjects, consisting of the way of working, agile tools, agile maturity, collaboration, and performance. The final phase of the interview is the closure. In this part, the interviewee will be informed about the next steps of the research. The summary of the interview will be sent to the interviewee after the interview so the interviewee could check if the answers to the questions are well interpreted. Also, I will thank the interviewee for their participation and effort. This part will be the same for all interviews, so also for the software engineers and hardware team leads. The complete list of the interview questions can be found in Appendix E.

3.2.3. Phase 3: Data analysis

The data from parts of the interviews and findings from the literature review are used to answer the sub-questions:

- SUB 1: What are the success factors and barriers in the implementation of the agile methodology?
- SUB 2: How can the adoption of agile methodologies affect the perceived individual and team outcome in terms of people, process, organization, and product?

As described before, the first sub-question will be answered by using the information from groups 1,2, and 3 together with the findings in the literature. This information will be compared to verify whether groups 2 and 3 see the same success and barriers in the implementation in the hardware environment. The amount of times a success factor or barrier is mentioned in different interviews will give a score on the importance of the success factor/barrier.

For the second sub-question, a comparison is made between the engineers and project leads to find out whether the position of the employee has an influence on the perceptions of the outcome in terms of the use of the agile methodology, communication, and performance. I want to find out what they need and what they are missing and how this could influence both their individual and team outcome. Finally, the last sub-question will be answered by combining all the information gathered from the interviews and the literature. A framework will be made that stated what to include and exclude from the agile methodology in the traditional model. It will give a conclusion about how to effectively implement this and what the limitations are. That last question was:

- SUB 3: How can agile methodologies be adapted to suit the unique characteristics and constraints of hardware development?

The interviews are audio recorded to allow the researchers to make transcripts of the interview and use that for a summary. The transcript is made by uploading the .mp3 file into MS Word and automatically generating the transcript. This transcript is checked manually and a summary of the transcript is made. This summary will be shared with the interviewee to check for the right interpretation of the answers. The summaries are made anonymous to avoid people from giving socially acceptable answers and be in line with privacy concerns. The data from the summaries is used to answer the subquestions.

The data that is obtained will be categorized before it could be analyzed. I will use a thematic research approach to compare the data from the two interviews with each other and draw conclusions on the main research question. According to Yin (2003), data analysis is considered the most difficult when exerting a case study. The transcripts will be coded into some categories that are based on the findings in the literature. The coding process is done manually in Excel. The frequency of the different categories is used as a measure of the importance of the category. Since the codes and categories are based on existing theory, deductive coding is used. In Figure 3.2, the categories that are used for coding for the implementation are shown, and in Figure 3.3, the categories that are used for coding the use of agile are displayed.

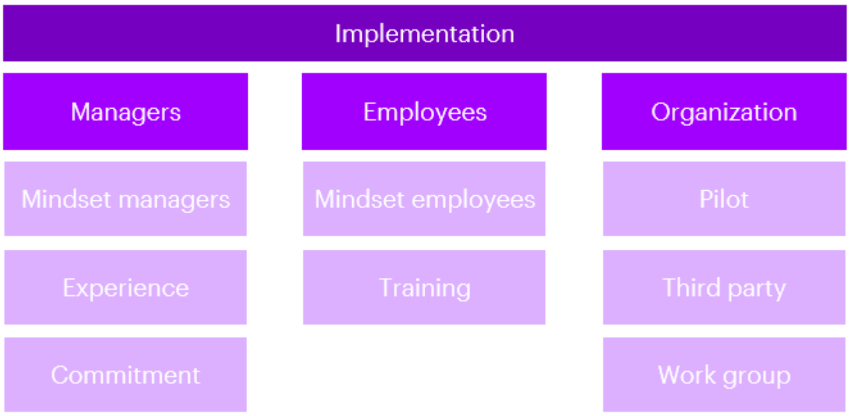


Figure 3.2: Coding categories implementation

The implementation has been divided into three main categories: managers, employees, and organization. The category of managers focuses on the mindset of managers, the experience, and the commitment to change. The mindset of employees and the availability of training for employees belongs to the employee category. From an organizational perspective, the use of pilots, third-party involvement, and work groups are displayed. Third-party involvement is related to the use of external consultants to guide the implementation. Work groups are special groups beyond the normal work activities of people that are investigating one specific topic.

The categories for codes of the use of agile are divided into four categories: people, process, organization, and product (Figure 3.3). The category of product is divided into the use of prototyping, quality of the product, and stability of the product. From a people perspective, commitment, transparency, and training are used. The category of process is divided into prioritization, planning, tools, meetings, communication, and administration. Prioritization is the ease of prioritizing tasks for the project, planning focuses on the punctuality of the planning, and tools are the tools that are used for planning, communication, and collaboration. Meetings focus on the meetings that are held, communication on the way of communication, and administration on the task of reporting. Finally, from an organizational perspective, we look at the structure and stability of the organization.

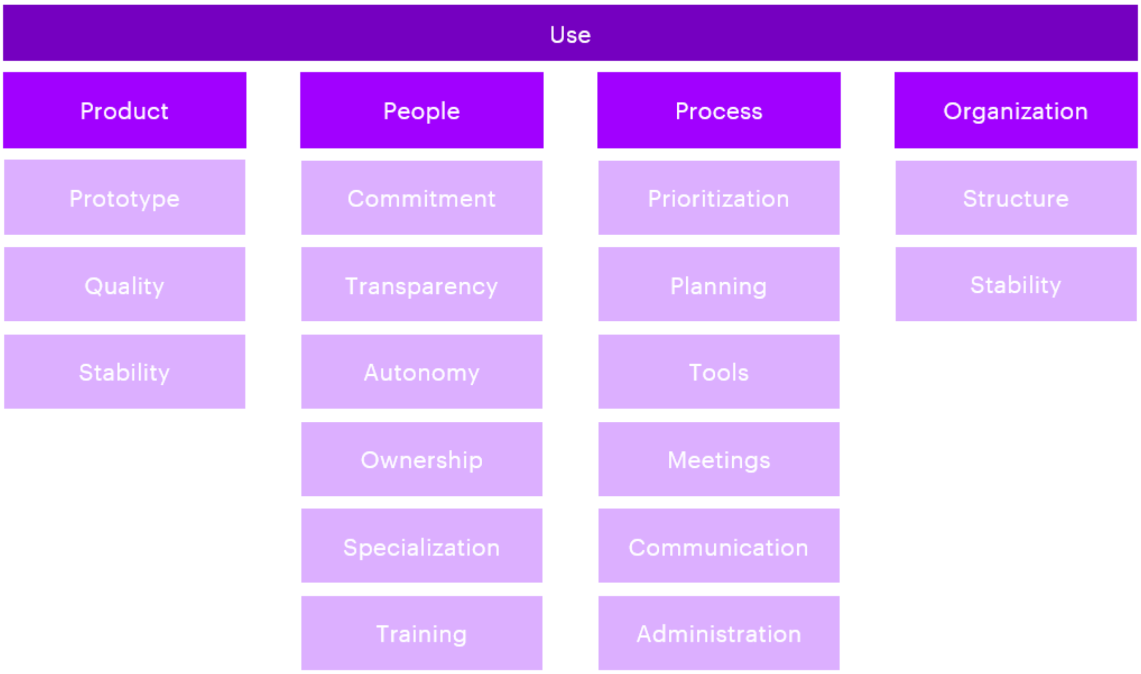


Figure 3.3: Coding categories use

3.2.4. Phase 4: Report results

Using the analysis made in the previous phase the main research question can be answered. In this phase, the results of the three sub-questions will be discussed and a reflection on the findings will be given. The identified success factors and barriers are used as a baseline of how the agile methodology should be implemented in the hardware environment. This will be combined with the benefits and challenges of the use of agile identified in the second question. The framework of the third question shows how the agile method should be adapted to suit the characteristics of the hardware environment.

4 Results

As mentioned in the methodology, 12 interviewees were conducted consisting of 2 software engineers, 6 hardware engineers, and 4 hardware team leads. The first part of this chapter dives into the implementation of the agile methodology in the hardware environment. The second part is about the use of agile and the third part is about how the agile method should be adapted to match the hardware characteristics. At the company, the agile implementation started 3 years ago meaning that [10/12] interviewees have experienced this transition (Table 3.1). Most interviewees, however, have way more experience at the company (average work experience of 12.5 years). Therefore, these interviewees could make a good comparison between the old way of working and the new one.

4.1. Implementation of agile methodology in the hardware environment

The first part of the interview focused on the implementation of the agile methodology in the traditional way of working. This section is divided into three categories: management, employees, and organization. This has been done to allow the researcher to compare the findings in a later stage with the literature.

4.1.1. Management

As mentioned in the intro of this chapter, the company in this research started adopting the agile methodology 3 years ago, meaning that almost all interviewees have experienced this transition at the company [10/12]. HW3 started working at the company when the implementation just started and described the situation as:

> ”I hit the jackpot. I said, there’s absolutely nothing right about what you’re telling me.” - HW3

This emphasizes the lack of experience with the new methodology that was present at the company. [9/12] mentioned the lack of experience as one of the main barriers to a good implementation of the agile methodology in the hardware environment. On the other hand, good guidance from new product owners could lead to a smooth implementation of the agile methodology. HW6 described that the management initially started by giving the teams a lot of freedom to implement it as they want with the idea that the acceptance of the transition increases. However, this led to even more resistance cause the teams felt that they need to do everything themself without any guidance. Hw6 also noted that the message of the management to the employees was that the efficiency will increase by 30% by introducing agile. This number was way too ambitious and people started to resist the change cause they only saw the increase in meetings and overhead and no efficiency increase. In the transition phase, a lot of new scrum master and product owners need to be found. Some people from existing teams got the role of a scrum master without any experience in scrum. This resulted in a lack of structure and control in the teams, and therefore inefficient (scrum)meetings. SW2 mentioned that the management should not enforce the change, but better let people understand why they are making the transition. According to HW5, hiring new people to become scrum master has also been found to be difficult since people need to have at least 6 months to learn how the company works before they understand it and can perform their job properly. TL2 mentioned that they have been trying to make the transition to agile two times, but decided to move back to the traditional way of working after the first try. The reason for that was that it made everything way more complex than easier and there was too much resistance in the organization.

4.1.2. Employees

Offering a training program for the current and new employees is mentioned as a success factor of a smooth implementation [5/12]. The amount of training defers among the employees. HW5, who has the role of product owner, for example, had 10-15 days of training. This training session was about why the new way of working is introduced, and how you should work in the new agile way of working. But also about the consequences of working agile, both positive and negative. On the other hand, according to HW6 some hardware engineers on the work floor got a shorter version of this training session and only had 2 days of training. HW6 also doubts if a lot of hardware engineers actually went to these training sessions, since it was not that well communicated. SW1 noted that he experienced the same. Since the training program is concise people do not get the complete story about why it is beneficial for them and therefore do not get the right mindset afterward. According to SW2, the main message of the training should be to let people understand why we should do this. TL3 noted that in their team everyone got the basic training but that there are still things to win in this field since the picture that is sketched in the training is different than how it is experienced in real life.

> ”If someone comes in fairly new to this role, or without prior experience in the company, well, that won’t help. It did help me, but for others, it could be a problem. [...] Then you need to make sure there’s either a good scrum master or another good buddy with that product owner.” -HW1

It is essential to establish the right mindset of everyone in the company with respect to the use of agile to achieve success. According to TL2, especially some older hardware engineers do not see the added value of using agile practices. In their opinion, it is just again another new flavor of the management structure and simply does not work. They mention often that they just want to do their own work and do not want to spend time on ’useless’ reporting and meetings. Also, TL1 mentioned that he felt resistance because people did not know why this new way of working is better than the previous attempts to implement agile practices. SW1 stated that the hardware engineers should feel the urge to change to the new approach themself by establishing the right mindset. This could be created by making them aware of the continuously changing world and letting them feel the pain of not being able to act on that. This will result in an intrinsic urge to change to the new approach. Also, it could help to introduce agile practices without letting them know that the new practices are agile, to avoid people being skeptical upfront because it is agile. HW5 mentioned that some managers were afraid that they will lose control of the people when using agile. HW6 mentioned that it is really important to introduce agile as a means and not as a goal.

> ”It’s yet another sauce about a management structure, it simply doesn’t work. - TL2”

> ”Agile should be implemented as a means not as a goal. - HW6”

4.1.3. Organization

In some departments, an external party has been hired to lead the transition to the agile way of working. According to HW2, this does not work as planned by the management. Since the external consultants do not understand the company and do not respect the way the processes work in the company the external consultants were not taken seriously by the people in the organization. Resulting in resistance from the people and a wrong mindset, which is disastrous for a good transition. TL3 mentioned that the external consultants at their team really focused on high-performing teams and how to optimize agile practices while the people just want to get help with the use of agile tools and get an explanation about how to effectively plan and use the time in meetings. This mismatch between external consultants and people on the work floor led to more resistance.

These external consultants suggested making a hard transition, meaning that they should move completely from the traditional way of working to the agile way of working. This forces everyone to think of the new way of working but also resulted in a loss of connection with the other parts of the company that does not have switched to the new way of working. Moreover, by changing completely to the new way of working people lack the vision of where they were standing with respect to the master plan that was made up front. According to HW3, this resulted in complete chaos where nobody knew what to do.

Instead of making a hard transition [10/12] interviewees mentioned that the use of a pilot is beneficial for a smooth transition. According to HW3, to achieve a smooth implementation the pilot should be with people from the work floor that are actually going to do the job later on. He adds to the importance of not only having a hypothetical pilot on paper or with agile coaches cause this will always be different than reality. HW4 described that in their team they always work with pilots, so every change they do is executed in a small team first. If something does not work properly, they fix it first before implementing it on other teams. This means a lot of interaction and iteration but will result in better collaboration. TL1 described that during the transition to an agile way of working, they decided to implement it in several teams simultaneously to achieve faster results. This led however to a lot of resistance because it resulted in a chaotic situation. TL3 experienced the same, as their team changed rapidly from approach and there was no long-term planning anymore. HW6, explained that this resistance came from the fact that people felt that they should figure it out themself and there was a lack of guidance. ”Fail fast, learn fast” as they describe it has been mentioned by [4/12] interviewees. TL4 emphasizes that they should do the rollout of the pilot with care and focus, and speed up the process if the progress allows.

Some other interviewees [3/12] mentioned that they started implementing all agile practices that were given to them (daily standups, retrospectives, demo-meeting, sprint planning). After a while, they reflected together with the team on what they liked and do not like. Also, they discuss what parts are useful for the team and what parts are too time-consuming. This feedback is used to change the intensity of the meetings. This way of reflecting is repeated sometimes to keep following the needs of the team.

HW1 mentioned that the company started by working in separate groups (workgroup) to find solutions for a couple of challenges. It works to let a particular group of people intensively focus on one challenge and come up with a solution for that. One of these groups focused on the planning tool, to find if the tool could be extended with more features. Another group focused on the structure of the teams with more product owners/scrum masters. This could later be enrolled in a pilot to one team and be extended to the rest of the company if successful.

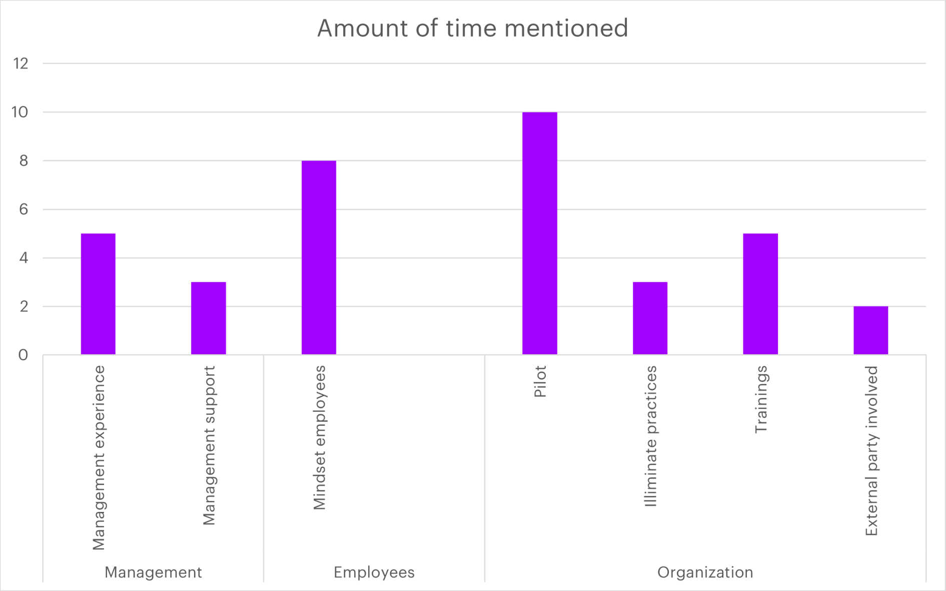


Figure 4.1: Factors affecting the implementation of agile sorted by the number of times mentioned

4.2. Use of agile methodology in the hardware environment

The second part of the interview is about the use of the agile methodology in the hardware environment. This section is divided into four categories: people, process, organization, and product.

4.2.1. People

The use of agile has increased the commitment of engineers to the work they do. According to TL2, engineers have validated the products better and therefore have more trust in the products. Also, they feel more ownership of the products since they have more control over the planning. TL3 added to this that by increasing this control people are end-to-end involved in the project, which increases the ownership. A lot of people like this increase in ownership. However, there are also examples of people that showed resistance since they want to hear what is expected from them instead of finding out themselves. The transparency of what you and your peers are doing has been increased by the use of agile, resulting in a more open working culture. SW1 stated that the agile method also allows people to constantly check whether their point of view matches the reality by the feedback loops. This reduces the chances of misconception about the final results. According to HW2, engineers understand their priorities better and what the company needs to be successful, and therefore they want to take that additional step to work a bit harder. It creates a culture where people want to help each other to achieve more.

There is still some resistance from people to the use of agile. Especially the older employees do not want to change since they like the current way of working. According to HW6, it is important to increase the acceptance of all employees by giving them the message that changing to a new way of working is an investment that should be made for the long term. Important here is that agile should be explained as a means and not as a goal. TL3 mentioned that younger people in a way expect that companies are making the change to agile to encounter the increased complexity and deal with the uncertainties. Therefore, it should be needed to make the change to also satisfy their needs.

Training is provided to learn people how to work with agile. But HW6 mentioned that this training session was very limited. More training with more in-depth information about why and how to use agile should be provided. As mentioned in the previous section, there is a lot of variety on the quantity and level of the training that is provided. TL3 got training on how to use SAFe in the hardware environment while others did not get the opportunity to attend that session.

4.2.2. Process

Using agile helps to set clear priorities and make proper planning. A backlog is created and during PI meetings the amount of work is divided along the capacity per team member. Since priorities are better set, people can focus on what they actually should deliver, resulting in faster deliveries. The idea of agile is that the planning is made bottom-up. Meaning that the engineers should make the planning and should communicate this to the higher management. Currently, according to HW2 there is some resistance since the management does not accept the choices. Therefore, it should be more useful if the management should help with making the decisions on the priorities according to HW2. Priorities between teams could however also lead to a conflict since every team has its own priority which could for instance not match the priority of another team. According to HW5, this prioritization results in more clarity and better discussions on what is important for the greater goal instead of the short wins. Also, TL1 agreed on this and added that it makes the challenges better visible for the teams.

Currently, most teams only plan one quarter ahead according to HW1. This should be extended to a longer period in order to know upfront what resources should be available in the future. If more people are necessary, they should start educating them at least 6 months ahead. TL3 also mentioned the necessity to plan far ahead since in their team they experienced a lot of times that they were too late with ordering stuff since they did not plan well ahead. HW5 noted that the problem is that currently there is no budget to plan for 6 quarters ahead. This makes it hard to know how much can be spent in the upcoming period. According to HW2, the current planning is already 50% above the capacity for 11/13 teams. If something goes wrong this will shift further and result in even more work in the future. HW3 stated that in his team there is a strict policy that new work could only be registered at one central point. Therefore, there exists a clear overview of all the tasks. HW5 mentioned that by using agile planning project managers can see the capacity of the team members up front and can act if they expect that people are acting above their capacity. HW1 explained that in the past it was doable to help people last minute from other teams but because of the strict planning, you should incorporate this request for help already in your planning meaning that you should think ahead on what you need from others.

All team members need to make estimates about the tasks they expect to finish in the upcoming sprint. According to HW1, people tend to have difficulties with making the right estimation resulting in unfinished tasks at the end of the sprint. SW2 explained that the reason for that often is the fact that they start the project from scratch resulting in a lot of uncertainties that could affect the estimations. HW1 noted that by the use of agile the amount of right estimations has increased, also because of the prioritization. The suggestion is made to have a training to look back at your estimates and find out if you met your goals and if not what the reasons are. By doing so, one could learn from the past and apply it to future situations. HW3 noted that the estimations of his team are always made together with the group, resulting in better estimations.

HW1 mentioned that following the standard procedure you should plan the time in one week for 80% and keep 20% of your time for overhead and other urgent requests from fellow colleagues. It depends however on your role in the organization if that is feasible since some roles ask for more time on those requests.

Right now multiple planning tools coexist. For the agile teams, JIRA is used and allows teams to have a clear overview of their tasks and planning. This planning is however not aligned with other teams, making it difficult to act on changing circumstances. The integration step between the teams is missing making it hard to display the dependencies between the teams. HW1 stated that right now people should call/mail someone from another team to communicate if things change. This person should then change it in their own system, making it sensitive to errors and resulting in a time-consuming activity. It should be more reliable and efficient if this could be managed automatically. TL4 noted that although it is useful to align the planning of the teams it should not become the manner to communicate with other teams. The tool should be a place where the appointments are displayed but not a communication tool. If dependencies are noted in time and communicated properly there should be no problem, but since this does not happen every time some teams move from escalation mode to escalation mode according to HW2. TL3 noted that every now and then there should be a formal moment to coordinate the dependencies with other teams. Some teams use another planning tool to incorporate higher-level planning, while other teams already integrate their planning with another team. This is especially useful if two teams work closely together and are highly dependent on each other (HW3). HW5 mentioned that at their team the planning tool is not aligned resulting in double the administration work. HW4 stated that the use of agile does not change the way they deal with dependencies of external parties since it is still out of their own control. HW5 mentioned that it should be valuable to design a tool that combines team planning of several teams and make a combined project plan. TL1 agreed on this and adds that a supporting team should be set up that builds this system and keeps it up and running. TL2 noted that in his teams people designed the structure to show dependencies on other teams themselves which is working fine for them but could not be enrolled in the complete company. To ensure that rollout, a better structured and organized platform should be built. HW6 stated that the collaboration with teams that have not yet made the transition to agile is even more complex. Also, the higher management is still working in the traditional way of working meaning that the planning tools also do not overlap. As a solution, the structure of the teams should be consistent and aligned to ensure reliable planning.

Sometimes people forget to report in the tool what they did, resulting in a wrong overview of the work that has been done. This makes it hard to make an accurate estimation of the work in the future. Strict control of the tool should therefore be performed preferably. Besides that, Hw2 stated that a common way of working with the tool is missing, therefore everyone has their own manners and the structure is missing. This results in the fact that people could not independently understand each other's tasks and could not help another person easily. Also, interchanging information between teams is difficult, since for example, the definition of a story point differs. According to HW3, this is because the management led the teams to decide how to use the tool themself instead of giving them a standard way of working. This makes the collaboration between teams harder. A disadvantage of Jira according to HW3 and HW5 is the speed of the program. Right now it takes way to much time to load the program, also making changes takes too long. The perspective of the use of the tool should be clear to every one according to HW4.

> ”The tool is not going to solve issues that are content related, it can only help to make things more clear” - HW4

Hw5 added that currently the tool is mostly used as a ”to-do” list and not really as a planning tool. This should however be a great improvement if you could use it for that. Integration of more programs like Excel would be a nice addition to the tool according to HW4. By doing that you could make a central point where you could also share data. HW6 said that for their team it does not matter what program they will use if at least everyone is using the same program and see the program as a means that helps them instead of a goal to become more agile. According to TL2, it should be of great value if the tool could also be used as a dashboard to show how well the team is performing so that the feedback can be used for the new planning. Also, other teams can use that data to increase their efficiency.

Also, dependencies on external suppliers are not possible to incorporate in the own planning. One should ask the supplier when they could deliver the product and asks frequently for updates about the delivery time. TL4 mentioned that if the system of the supplier could be linked with the system of the company this will no longer be necessary. Since suppliers are also using JIRA this should be possible according to TL4.

The amount of meetings is varying for every team. Some teams have 2 daily stand-ups per week, some have 3, other 4. Most of the teams discuss after a certain period how valuable the meetings are and if people are happy with the amount of meetings. The feedback from those meetings is used to change the intensity of the daily stand-ups. HW3 mentioned that one of the improvements of doing these standups is that people communicate better with each other. Also, people can help each other better and everyone is aware of the things other team members are doing. In the past, when there were only meetings once in two weeks it was possible to talk to nobody in a month’s time if you miss one single meeting. This new transparent way of working is therefore good for collaboration in a team. HW6 indicates that it is important to note that the daily standups are not going to become technical reviews/ discussions. Otherwise, it will result in time-consuming meetings that are missing the goal of the standup, which should be to provide a short status update. Also raising blocking issues can be discussed shortly according to SW2, making it really helpful for the progress of the project. In the beginning, there was some resistance to the standup meeting cause people felt that they were judged if they did not meet the planning. This changes however over time when people got a better understanding of the goal of the standup meetings according to TL1. SW2 proposed that before doing a standup meeting people could write in a chat if they have any important updates and if nobody has one they could skip the standup meeting.

Retrospective meetings are performed in all teams. Most of them [9/12] describe them as positive. However, some also experience that not all engineers are evenly satisfied with these meetings. Some engineers indicate that they just want to do their job and do not want to talk about how they feel or what they should do better since they already know how they are performing. HW3 stated that if you manage to do the retrospective meetings in the right cadence, so not too often, people tend to like these meetings. Some retrospective meetings are used for the feedback of the last sprint and others are used to look at the complete past quarter. This depends on the needs of the teams.

Demo meetings are also held once in two weeks for most of the teams. Although some teams do not have a prototype ready every two weeks, most of the engineers do like these meetings. TL2 indicates that these prototypes can also just be cardboard prototypes to show the working principle. Prototyping enhances the discussion about the product and makes it more tangible. HW2 said that in his team once every two weeks a daily standup is used to discuss the overall planning, the changes that have been made, and how the next quarter will look like.

TL1 mentioned that he also has alignment meetings with other Team leads to align their work with the other teams. This occurs once every quarter.

HW1 highlighted the importance of speaking the same language across teams. Some teams call Epics a feature and the other way around. This generates confusion and miscommunication among teams. In the last couple of years, a lot of new people have joined the company. Also for them, it is important that they get familiar with a common way of communication. Since the company has employees from all around the world it makes it even more significant to clearly execute one standard.

By using agile it is easier to inform your stakeholders since it is more standardized. HW6 stated that through improved communication also the quality of the products increases. Also, TL2 mentioned that the communication in the teams improved resulting in a better understanding of the content and goals. SW2 highlighted that by frequently interacting with the stakeholder one could better understand the hidden requirements, resulting in a better product.

A drawback of the use of agile is the increased administrative work and overhead. Some engineers complain about the amount of time they need to spend on reporting. They do not see the added value of all this reporting and mark it as a waste of time. Good agreements should be made about what needs to be reported and what not to reduce the amount of administrative tasks. Especially when they need to follow the time of one sprint and need to deliver something after this sprint it is most of the time an update on the report. People do not like to do all this reporting and just want to continue with their job. The same holds for the overhead, people prefer to not spend their time on all these rituals and meetings but can work on the things they are hired to do.

4.2.3. Organization

The reason for the company to use agile was to create a more stable organization according to SW1. With a stable organization, a well-structured organization is meant in which people know where they belong and what teams are doing. The need for a more stable organization came from the need to handle the pace of growth of the company. According to SW1, in the past, every new employee got a buddy. Right now that is not possible anymore since otherwise buddies are continuously teaching new employees the tricks of the job and could not do their own work anymore. The teams should be a place where people could learn the appropriate knowledge in this fast-growing company. The structure of the teams defer among the company. Some teams have a separate scrum master and product owner and in some other teams, these two roles belong to one person. On a higher level, the company tried to set up a lead product owner that is responsible for multiple product owners. Hw5 explained that this plan did not make it since the product owners simply does not have the capacity to fulfill this role. Currently, the top management still follows a waterfall approach meaning that planning is made upfront with deadlines in between where the process needs to continue to the new phase. While the teams work in agile and are promoted to come up with their own planning, a conflict arises. The management sometimes dictates that x amount of work needs to be finished while the teams indicate that they could do less work in the available time. Managers tend to micromanage the process instead of giving a clear final goal and trust the teams to plan the work themself according to HW6. If you do this right, and keep teams working together for a longer time they got more ownership of the product resulting in more commitment to the project. You need to give the teams space and build trust in the team and let them deliver what they promise to create empowerment in the team. The management and teams should meet in the middle and an Epic owner should synchronize the planning of the two teams according to TL2.

Agile is currently most often used as a way to organize the company better, while on the design level of the organization, it is not regularly used. SW2 highlighted that the agile way of working should match the organizational structure and therefore should be adapted accordingly. This is an iterative process that takes time. Small iterations are often not possible and people do not like to make prototypes because they have to. HW6 stated that they were already working agile in a way since trial and error is often part of the design process. Especially since it is often not clear upfront what the engineers should deliver as a final product the use of agile is relevant.

Teams get the freedom to work on their own project with their own planning. They should act autonomously if they finally meet the set goals by the management. This increases ownership and works the most efficiently since teams can optimize what works best for them. By pushing teams to deliver what you want them to deliver you only create resistance to the management according to HW6.

TL2 emphasizes that if the company manages to make the transition for all the teams to an agile way of working problems with the available office space could arise. If every team has a pi-planning meeting on the same day to be aligned with each other there are lack of available rooms. This should be taken into account when deciding to move all the teams.

Hardware development teams consist of engineers that are often specialists in one specific thing. This increases the productivity of that specific part but makes it harder to get help from other people. Other people simply do not have the right skills to take over the job. This is in fact against the agile way of working where every role is interchangeable according to TL1. SW1 suggested that it should be useful to work more in a function group than in a project group to ensure that people could help each other better and are interchangeable if something occurs. TL3 explained that a better way to tackle this problem is by making people more multifunctional. This could be arranged by letting people learn from their peers and change roles so now and then. By doing this, you allow people to help each other and if someone is not able to work on the project that person can be replaced by someone else.

4.2.4. Product

Agile is now primarily used at the higher level of the organization. However, in some teams, the agile way of working is already introduced on the design level. Especially the shorter iteration cycles are used and the early production of prototypes. TL2 mentioned that some teams are now forced to make these prototypes. Some engineers do have resistance to doing this since in their eyes they do not have any risks and therefore it is not necessary to do. They are just doing it because they have to, resulting in a waste of resources (both time and materials). They prefer to make virtual 3D models that can show how it should work. HW6 mentioned an example in which it should have been beneficial to have an early prototype since they found a mistake in the design in a late state, with a lot of costs and a waste of time as a result. TL3 explained that in their team they work with short iteration cycles to come up with the final solution. Prototyping is part of that process. Furthermore, TL4 mentioned that the use of 3D printing really helps with the production of rapid prototypes. SW2 stated that also from a hardware point of view, it should be useful to create prototypes in an early stage of the project. Although the product is not fully functional you can get feedback from the team/customer/project lead if you are heading in the right direction.

Some engineers found it hard to divide the final product into different segments. If they do so, they can only deliver a report on the status of the product which felt like a waste of time for some engineers, according to TL2. For other smaller parts, it is possible to align the segments to the sprints. In that case, they do not find any problems. TL3 mentioned that in their team they had a slicing workshop to learn how to divide the product into different segments and allow you to come up with a minimal viable product.

According to TL2, the stability of the product increases due to the use of agile. The ”one time right” has been increased, meaning that when the product is delivered to the customers the product is working instantly properly. However, TL2 stated that the time to deliver a product has increased a little bit. The time to market has not been significantly increased [7/12]. According to HW3, the quality of the work increased due to the use of agile because the engineers could focus better on their job resulting in more careful work. TL2 added that you could undermine risks earlier and therefore design and manufacture higher-quality products.

> ”The quality of the product is improving due to the clearer separation of tasks and earlier risk detection. Efficiency is increasing, and the stability of the deliverables is also improving.”
> -TL2

4.2.5. Benefits and Challenges

Using the analysis of the use of the agile methodology the benefits and challenges are identified. These benefits and challenges are also categorized into product, people, process, and organization.

Figure 4.2 shows the identified benefits of the use of agile in the hardware environment. From a product perspective, the increased quality and stability of the product are mentioned 3 and 2 times, respectively. From a people perspective, increased commitment (5), increased transparency (5), increased ownership (2), and increased autonomy (4) are mentioned as a benefit of the use of agile. From a process perspective, increased communication (10), increased prioritization (4), improved planning/estimations (9), increased productivity (5), and increased time-to-market (3) are mentioned. This category has the most and also often mentioned benefits of the use of agile. Finally, from an organizational perspective, the increased organizational stability is mentioned 3 times as a benefit of the use of agile.

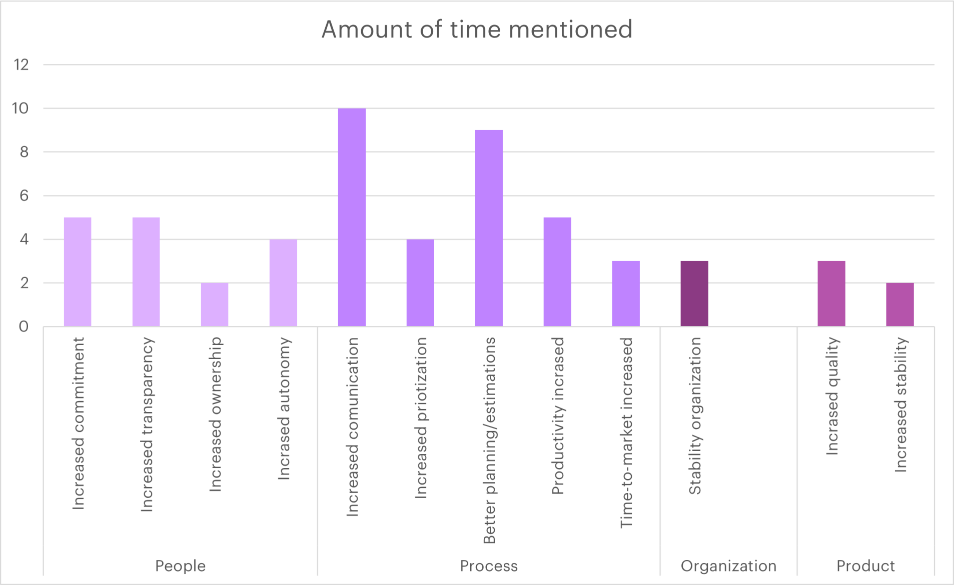


Figure 4.2: Benefits of the use of agile in hardware environment

In Figure 4.3 the challenges that arise from the use of agile are displayed. From a product perspective, 7 interviewees mentioned that the quality of the product does not increase by using the agile methodology while they expect it to be. From a people perspective, the resistance of people is still the biggest challenge (mentioned 7 times). Besides that, the fact that the availability of people with the same skills in one team is very limited was found to be a challenge (mentioned 3 times). Since people are too specialized they could not help others in their teams which could delay the process. From a process perspective the large amount of meetings (9), tools not sufficient (7), too much administration (5), bad collaboration with other trains (10), productivity does not increase (5), time-to-market does not increase (7) are mentioned as challenges. Finally, from an organizational perspective, the discrepancy between the structure of the management and teams is mentioned as a challenge. The structure does not match since the management is still using waterfall while the teams are working in agile. This leads to a mismatch in communication and collaboration.

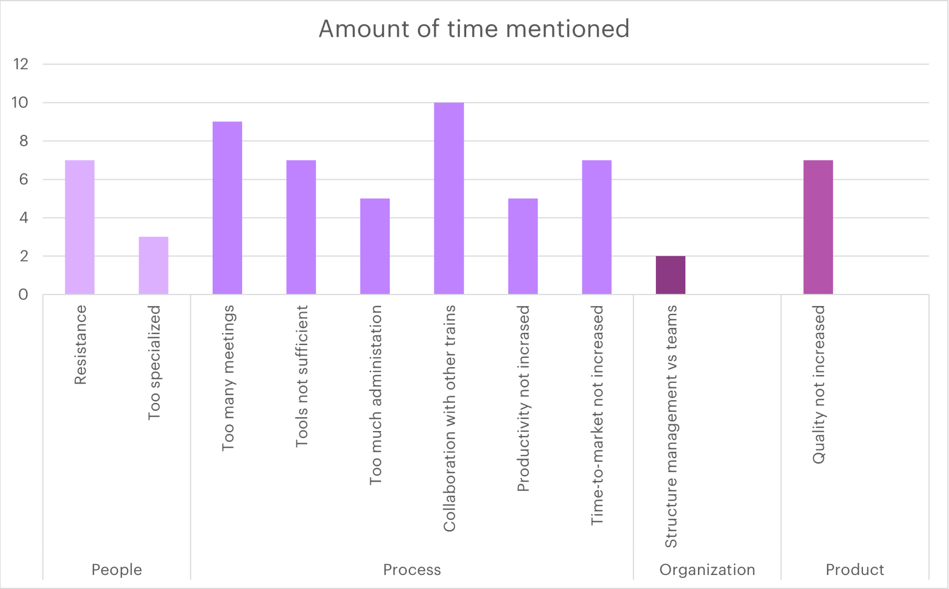


Figure 4.3: Challenges of the use of agile in hardware environment

4.3. Adaption of agile method

The agile method should be adjusted on several aspects to meet the requirements of the hardware environment. The 4 pillars on which we changed the original agile methodology are ’People’, ’Product’, ’Process’, and ’Organization’. Figure 4.4 shows a visualization of the adjustment of the agile framework.

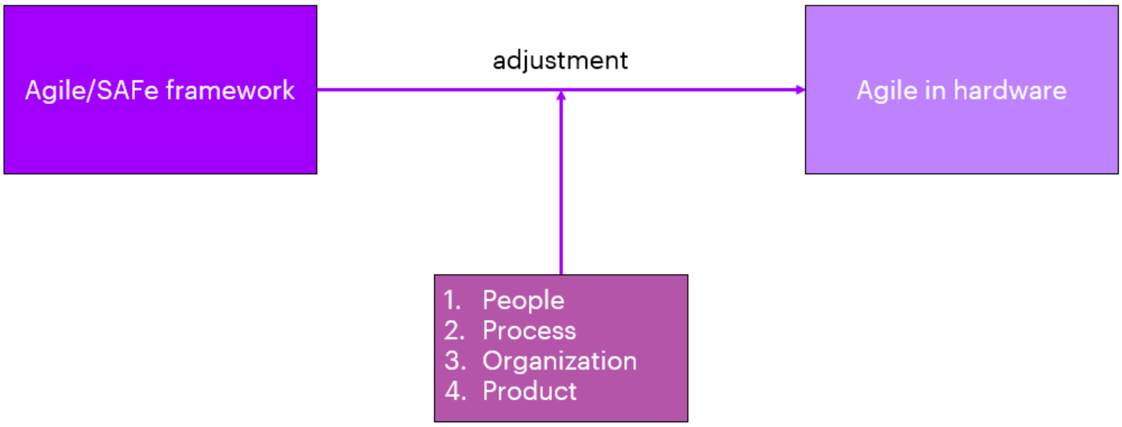


Figure 4.4: Adjustment of the agile framework to meet characteristics of hardware environment

4.3.1. People

From a people perspective, it is even more important to establish an agile mindset. Since hardware engineers are used to working with traditional approaches this change of approach requires a whole new way of thinking. Even during studies, a waterfall method is mostly used in design projects. Also, people do not like to change something that is working properly in their eyes.

> ”We are humans. We like something that’s really familiar. That’s how the brain is wired, our brain like familiarity.” -SW2

According to HW6, there should be some kind of intrinsic motivation to change. This motivation can be established by means of offering training. During these trainings, a good description of why the use of agile is beneficial for hardware engineers should be given. This makes them aware of the positive consequences of using agile. TL2 mentioned that the focus needs to be on how to use agile practices to ensure that everyone can work efficiently and get the most out of the practices. These two things together result in less resistance against the use of the methodology. HW5 stated that one should respect that this gain of acceptance takes time and is received differently for every person. Therefore, personal attention should be given to shape the training to the need of the engineers. According to HW2, by using the agile methodology the people in the team can make their own planning. This results in more commitment for the team since they make the estimations themselves and feel more responsible. In fact, the autonomy of the team increases which results in more ownership of the team. TL2 mentioned that people are more supportive of the product and have more confidence in a successful outcome. Focus on the increase of ownership is even more relevant for hardware engineers. Since some are working on a small piece of a machine the ownership could easily disappear. Making people feel ownership will result in more willingness to work harder and be committed to doing their job. Also, transparency increases due to use of the planning tools. Project leaders could immediately see the status of the project. Furthermore, as a consequence of the increased acceptance and established mindset the commitment increases, which will also enhance the ownership.

4.3.2. Process

From a process perspective, according to TL3 the rituals such as daily standups, retrospective, and sprint times should be adjusted to the wishes and needs of the teams. While in software it is easy to have consistent sprint periods and iterate every sprint, in hardware this is more complex. This varies with the product and the phase the team is in at the time. A more dynamic way of using the method should be favorable. Also, the amount of retrospective and daily standup meetings could be adjusted to what the team needs. For some teams, it helps to give an update more often than for others. Furthermore, it should be noted that the most important thing is that the meetings are useful and efficient. TL1 mentioned that they changed the frequency of the meeting in order to get the right balance for the right moment of the project. This is in order to comply with everyone's wishes.

> ”Initially, there was some resistance, which led to adjusting the frequency so that you can find a good balance and also consider the project's timing” -TL1

If the scrum master or somebody else from the team noticed that this is no longer the case, the frequency of the meetings should be adopted. The alignment of planning tools is even more important since teams in hardware are often more dependent on other teams, or suppliers. By adjusting the rituals the communication between the team members will increase. The planning and estimations become better and people are better able to make the right estimations.

4.3.3. Organization

From an organizational perspective, it is of significant value that the structure of the teams is consistent so that the trains are better aligned according to HW5. This alignment should be created since in hardware the dependencies on other teams are high and collaboration between teams is often necessary. TL2 mentioned that often the management of hardware companies is still working in a waterfall approach while the teams are working in agile. Conflicts arise if the planning does not match or if things on the project change in the meantime. A consistent structure in the company allows the teams and management to better collaborate and align the planning. TL2 suggested that also the introduction of an EPIC owner that is responsible for the synchronization between the management and agile teams could help.

> ”Somewhere halfway, you come across each other because you have the waterfall from the top and agile teams from below. They converge somewhere, and ideally, you want an EPIC owner who is responsible for the synchronization between them.” -TL2

By making a consistent structure of the teams the stability of the organization increases as a consequence. According to SW2, this is needed to continue supporting the significant growth of the company. Furthermore, TL3 stated that in hardware teams we often see engineers being highly specialized in their own expertise which results in more efficient work but on the other hand makes it harder to ask for help from other engineers. Also, if someone is not able to work this person can not be replaced by somebody else, resulting in delays in the project. TL3 suggested that it is therefore advisable to let people work on different expertise to make them interchangeable.

4.3.4. Product

From a product perspective, the focus should be more on minimal viable prototype production. Since prototyping helps to discover risks in an earlier stage it improves the project according to TL3. While it is not always possible to make a working prototype it is still of significant value to produce something to start the conversation. Also, SW2 mentioned that with the prototype you could show the team or customer your direction and already get feedback if it matches the expectations. As a result the quality of the product increases. Also, since risks are earlier detected the stability of the product increases.

> ”So what happens in agile is that whenever we do something, although it will not be a complete product, we make a small feature and we show it to the customer. we know for sure that we are not showing them a complete product. We ask them if they can explain us or give us feedback if we are heading in the right direction” - SW2

The reporting process should also be adjusted to the wishes and needs of the project team. SW1 stated that in their team they were shown how to reduce the administrative tasks since there was resistance to the increased workload. By reducing the administration the added value of the things that are reported was much higher.

Using the above mentioned results, a framework of the aspects of the agile method that should be adapted is designed (Figure 4.5). The framework is divided into four categories: people, process, organization, and product. These categories are the same as those used in the interviews. For each category, the adaption for that category is mentioned. This framework can be used as a guideline on what should be adapted to the agile methodology to suit the hardware characteristics.

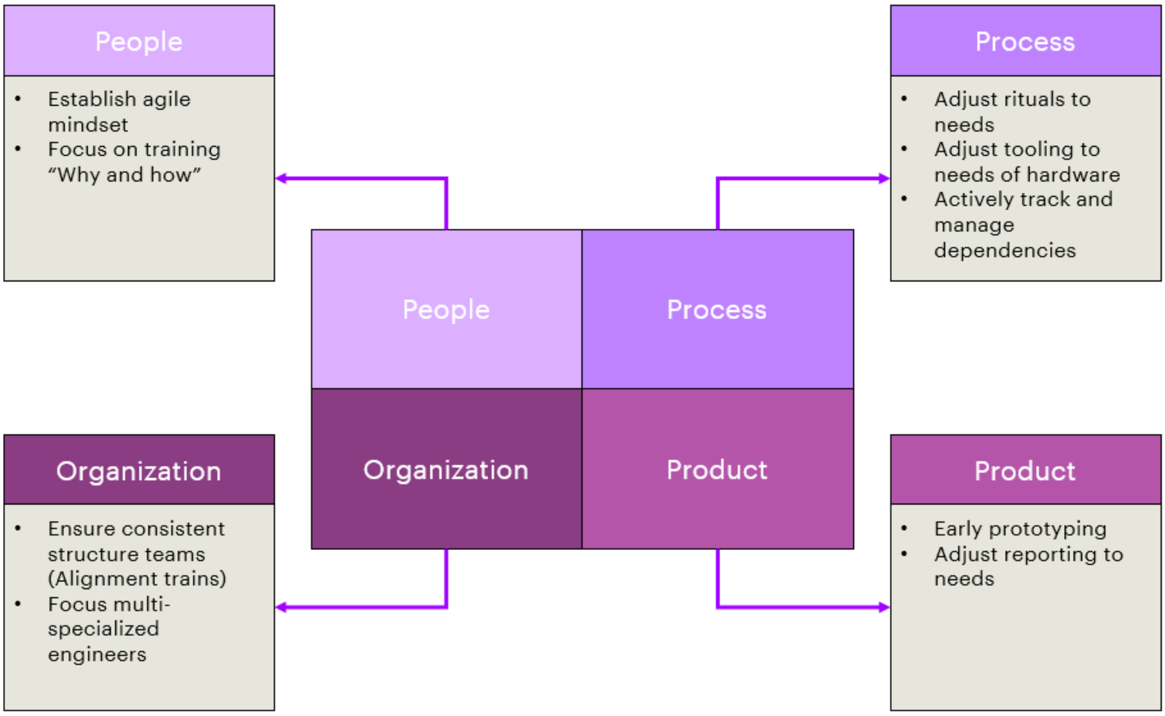

People
Establish agile mindset
Focus on training "Why and how"
Process
Adjust rituals to needs
Adjust tooling to needs of hardware
Actively track and manage dependencies
People
Process
Organization
Product
Organization
Ensure consistent structure teams (Alignment trains)
Focus multi-specialized engineers
Product
Early prototyping
Adjust reporting to needs

Figure 4.5: Framework of the adaption of the agile method in hardware environment

5 Discussion

In order to make the transition from the traditional approach to an agile/hybrid approach two main phases need to be completed. The first phase is the successful implementation phase, which is the starting point of the transition and ends after the method is actively utilized. The second phase is the successful use phase, in which the adaptions of the method that are proposed in the result chapter are being executed resulting in better perceived outcomes. When finishing these phases, the company successfully transitions to the agile/hybrid approach. Figure 5.1 shows a schematic overview of this transition.

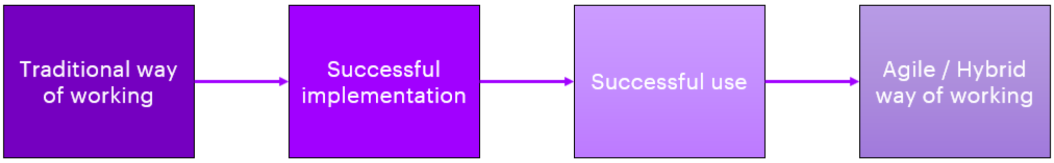

Traditional way of working
Successful implementation
Successful use
Agile / Hybrid way of working

Figure 5.1: Transition traditional approach to agile/hybrid approach

5.1. Implementation

In the literature, a list of barriers and success factors for the implementation of the agile methodology in the hardware environment has been identified. These barriers and success factors are categorized into three categories: management, employees, and organization. The success factors and barriers are difficult to see independently of each other, as they are often linked and can reinforce one another. As shown by previous scholars (Dikert et al., 2016; Naslund & Kale, 2020; Kalenda et al., 2018), one of the main success factors is management support. They should initiate the change, motivate the team members, and make their support visible to the team members. Furthermore, the management should communicate that the change is non-negotiable, need to show strong commitment even if problems occur, and convince the employees that agile will bring the desired results. Looking at the interview, similar results can be seen. [7/9] interviewees highlighted the importance of the role of the management. Both the support and experience of the management are mentioned as barriers/success factors.

The second category is the employees. Mancin (2016) and Paterek (2017) noted that employees should be empowered to make their own decisions and be self-organized. Also, the level of knowledge and expertise in agile practices should be increased. Long & Starr (2008) found that resistance among the employees increases due to the new responsibilities that agile development brought to teams. Teams are expected to be self-managed in agile development, but not everybody was pleased. Teams did not want to solve their new problems. Also, according to Dikert et al. (2016) for a lot of people, it is challenging to work according to this autonomous way of working. According to Koehnemann & Coats (2009), it is hard to effectively implement the agile methodology in the hardware environment as the mindsets of employees in an organization are not aligned. People have skepticism towards the new way of working and therefore do not want to change to the new method. In the interviews, we recognized the same pattern. People tend to resist the change because it is not clear to them what the benefits are of the new way of working. Also, they felt that they have to discover how to work with the new method themself without any guidance. The autonomous way of working does not match their expectations. During the interviews, the importance of the availability of enough training is highlighted [3/9] times. Also in literature, we found that training is of significant value to create the right mindset Naslund & Kale (2020). Kalenda et al. (2018) stated that hiring external experts with broad and deep familiarity with agile development could accelerate the process of knowledge exchange and transformation. During the interviews, the advice of external parties on the transformation has been experienced as a barrier. The external party does not understand the processes in the company and therefore the connection got lost with the employees and people did not take the external party seriously anymore.

The third category is the organization. Dikert et al. (2016) stated that it is essential to realize that every company is built differently, uses other techniques, and has other organizational structures therefore there is not only one way to implement and use the agile method. As a result, agile practices need to be adapted to fit the individual context and needs to be customized and implemented carefully. According to Weichbroth (2022), the implementation should be a step-by-step process in which continuous learning plays an important role. A single approach needs to be chosen to align everyone in the organization and set a clear goal. [7/9] interviewees mention the availability of good pilots as a crucial success factor for the implementation of the agile methodology. Also, the importance of adjusting the practices to the company has been mentioned by the interviewees. To achieve a smooth implementation the pilot should be with people from the work floor that are actually going to do the job later on. Also the importance of not only having a hypothetical pilot on paper or with agile coaches has been mentioned cause this will always be different than reality. Some interviewees have already good experiences with pilots and mention that with every change they do it is executed in a small team first. If something does not work properly, they fix it first before implementing it on other teams. This means a lot of interaction and iteration but will result in better collaboration.

The findings of the interview and literature are used to design an implementation framework (Figure 5.2). This framework consists of the 3 categories: management, employees, and organization. The organization category is divided into two aspects, pilots and available training. The framework shows the relation between the four components to achieve smooth implementation of the agile methodology. The mindset of the management is enhanced by following training about the ”why and how” of agile. By having the right mindset management also is willing to participate in these trainings. For the engineers, this process works the same. By having more knowledge of the agile methodology they tend to accept the change better. They should however have the willingness to participate in the training program. As discussed earlier, pilots are of significant value for the smooth implementation of the agile methodology. A pilot will help with creating the right mindset both for the management and for the engineers. Already having the right mindset will help the success of the pilot. Also, a more critical attitude could help to improve the pilot even more. Note that this framework can be used during the start of the transition phase, which is the first moment when starting to implement the agile methodology in the traditional approach in the hardware environment.

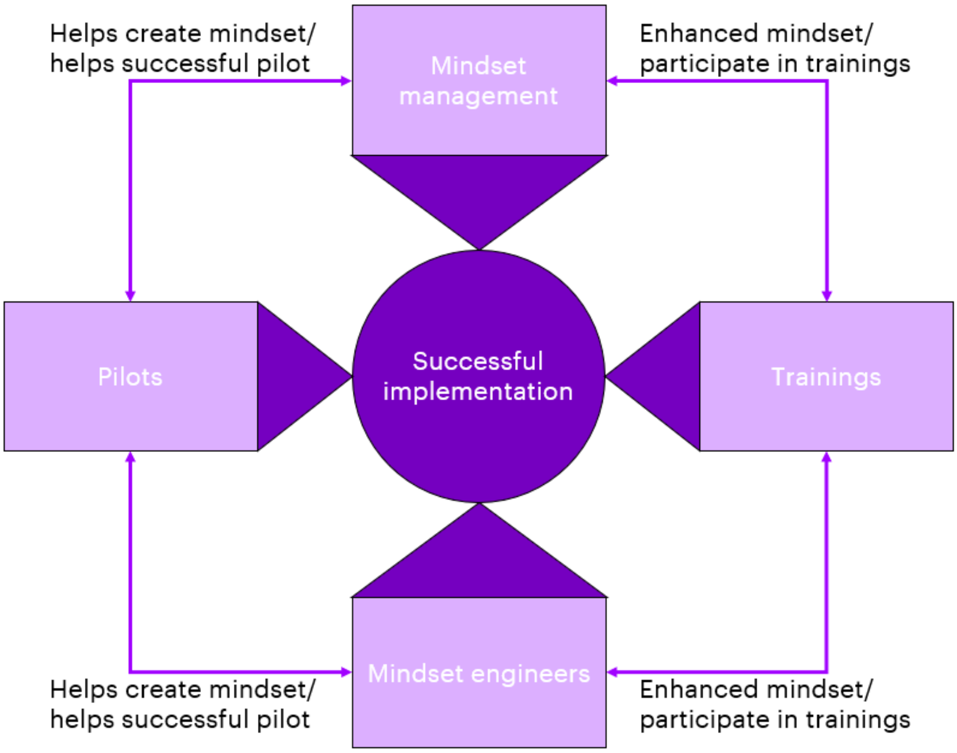


Figure 5.2: Successful implementation agile methodology

5.2. Use

In the literature benefits and challenges of the use of the agile methodology in the hardware, environment are highlighted. The benefits have been divided into soft and hard criteria. Firstly, looking at the soft criteria. The top 3 soft criteria identified by Atzberger & Paetzold (2019) are communication, transparency, and commitment. From the interviews, these three criteria have been found as well. For the hard criteria, the top 3 identified by Atzberger & Paetzold (2019) are quality, time-to-market, and costs. Both Quality and time-to-market are mentioned by the interviewees, but the costs aspect is not described as one of the benefits of agile.

Looking at the way how to effectively use the agile methodology the interviewees highlighted several points. These are categorized into four categories; people, process, organization, and product.

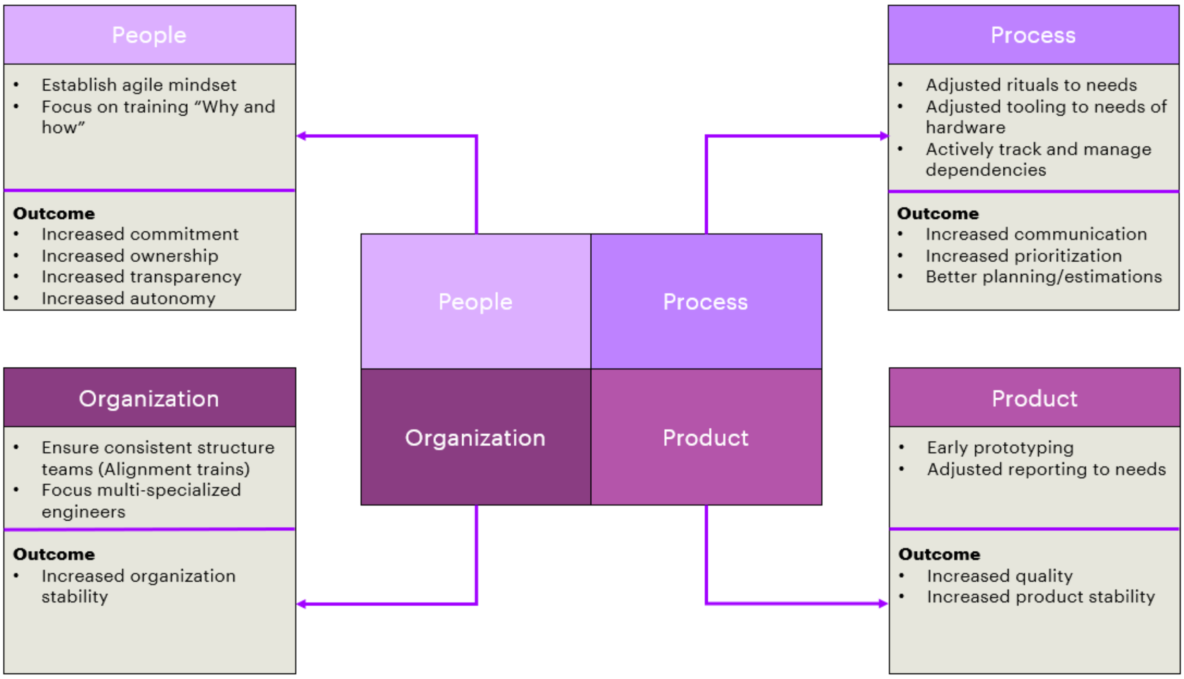
From a people perspective, an agile mindset should be established by providing training and guidance. This mindset is of significant value to effectively use the agile methodology. From the interviews, we found that it is important to increase the acceptance of all employees by giving them the message that changing to a new way of working is an investment that should be made for the long term. Important here is that agile should be explained as a means and not as a goal. This has been emphasized by Koehnemann & Coats (2009), who stated that a common view on change is one of the most important success factors of the use of agile. Roles, responsibilities, and common definitions should be defined and wrong ideas and misconceptions should be pointed out. Also, according to Dingsøyr & Moe (2013), the importance of the culture shift that is necessary for the agile transformation should be emphasized so that an agile mindset could be created. It should be clear that some existing company rules no longer apply and that the focus will be on agile values. Furthermore, the organization should have a positive attitude towards agile methods and be aligned with those values (Kalenda et al., 2018).

From a process perspective, the agile rituals help to set clear priorities, make proper planning and use feedback to improve the quality of the work. According to the interviewees, daily stand-ups can be used to discuss a status update. Retrospective meetings are useful to provide feedback on the work that has been done and how it can be improved. Demo meetings allow teams to show the prototypes they have created and enhance the collaboration on the project. Sprints can help to structure the work and set clear goals. In literature, according to Bjarnason et al. (2011), it is easier to deal with a higher number of tasks and a higher number of project goals when using agile practices like sprints, or a scrum board that helps to create a better overview and structure. The sprints could set targets and help the teams to work toward them. Campanelli et al. (2017) adds that also the dependencies are better visualized by the use of these agile tools, and daily stand-ups help to communicate clearly about the dependencies. Retrospective or sprint review meetings are of value as well since the feedback from the meeting can be executed earlier and implemented during the project. Augustin & Schabacker (2019) stated that sprints need to be flexible to tackle delays that can occur due to the waiting time for physical prototypes because of external dependencies.

From an organizational perspective, it is favorable if the whole organization works with the agile methodology. By doing so, the communication is well structured and collaboration becomes easier. If it is not possible to let the whole company work in an agile manner for one reason, at least all the project teams should be aligned. Management should respect the fact that planning is made bottom up and teams have the freedom to work independently. Kalenda et al. (2018) found that the use of old and new approaches side by side could be difficult since the structure is still not aligned and people do not know where they belong. Weichbroth (2022) reported that misalignment inside a company could cause problems since teams do not want to rely on agile teams if they do not know whether the agile teams would deliver their work on time. Also here it is important to align the teams with each other.

From a product perspective, short iterations and rapid prototyping should be applied to work most effectively. According to the interviewees, the quality of the work increased due to the use of agile because the engineers could focus better on their job resulting in more careful work. Risks could be earlier undermined and therefore design and manufacture higher-quality products. Also, early prototypes can make mistakes visible in an earlier stage, which could otherwise result in higher costs of change and a waste of time. According to Augustin & Schabacker (2019), when you incorporate prototypes into your design process, you will have a better sense of whether you are solving the real challenges in the most effective way for your consumers. It will also generate more constructive feedback from team members and consumers since you have a physical object instead of a simulation. The use of iterative prototyping techniques will therefore help your team find faster, better, and higher-quality solutions.

In Figure 5.3, a framework for the use of the agile method in the hardware environment is shown. This framework is based on the framework of the adaption of the agile method shown in Figure 4.5. The suggested adaptions are translated into effective use of the agile method. In this framework, the use of the method is shown on top and the perceived outcome is displayed below.


People

- Establish agile mindset
- Focus on training "Why and how"

Outcome

- Increased commitment
- Increased ownership
- Increased transparency
- Increased autonomy

Process

- Adjusted rituals to needs
- Adjusted tooling to needs of hardware
- Actively track and manage dependencies

Outcome

- Increased communication
- Increased prioritization
- Better planning/estimations

Organization

- Ensure consistent structure teams (Alignment trains)
- Focus multi-specialized engineers

Outcome

- Increased organization stability

Product

- Early prototyping
- Adjusted reporting to needs

Outcome

- Increased quality
- Increased product stability

Figure 5.3: Framework of the use of the agile method in hardware environment, with adaption and perceived outcome

5.3. Practical relevance

Hardware companies can use the framework proposed in this research to help with the implementation and use of the agile methodology in their company. The company at which this research is based can use the framework to further enroll the use of agile among the firm and adjust the current practices so it will better match with the hardware environment.

Besides the company at which this research is executed, other companies in the same industry could use the findings as well. They could get a better understanding of the challenges and benefits occurring at this company. By using this case, they could avoid some challenges and focus on the success factors. This will lead to a smoother implementation. Especially for firms that are using or planning to use the SAFe framework this thesis could be relevant. Nevertheless, for companies that are not going to use SAFe the findings in the literature about the success factors, barriers, benefits, and challenges of the agile methodology are still valid. Also, the results of the framework can be used for other methods of agile as long as it is interpreted with care. Since the rituals and tools can be different they could not be directly used.

This research is performed at a company in the semi-conductor industry but also in other industries this framework can be applied. Managers of the investigated company can use the findings to give more attention to several agile practices and improve the way of working. Furthermore, managers of teams in the company that are not using agile at the moment can use this thesis as a guideline for a smooth implementation of the agile methodology. Since the industry might work differently some changes should be made to adjust the characteristics of that industry.

The identified success factors and barriers of agile implementation can be used by the investigated company to reflect on their process and can be used in the future at the next step of the transition. Also, the findings in the literature can be used as fundamental for the reflection. Other companies can use these findings to compare with their own transition or use it for a future transition. Also, they are already aware of the challenge and barriers at the start of the use of the method, they could ensure that the process will run smoothly.

The identified benefits and perceived outcomes can be used by the company as proof to convince employees about the need to implement agile practices. The challenges can be used to develop a plan to tackle these and become even more effective. Other companies can use these findings as an example of the potential benefits they could gain if they make the transition as well. The same holds here, it could function as proof to convince the management to make the transition.

5.4. Limitation and future research

There are some limitations in the research in this thesis, due to the way of data collection and the time constraints of this project. This affects the generalizability of the results, making it more difficult to state that the results are valid for the whole hardware environment. These limitations can however give a direction for further research, in which these limitations can be included.

Firstly, case studies tend to have an issue with generalizability since the number of interviewees is limited due to time constraints. Besides that, it could result in researcher bias because the researcher's own subjective feelings may influence the case study. The repeatability of the research is also low since doing other case studies may lead to other results. This should all be taken into account when evaluating the results of this study. One of the main limitations causing the lack of generalizability is that the interviews have been conducted only with engineers from one company. This could lead to a bias since in other companies it could be experienced differently. For future research, it would therefore be interesting to check whether the same results are found at other companies. This company is using the SAFe methodology which limited the scope to only one agile methodology. For future research, it would be interesting to check whether the implementation and use of other agile methodologies would result in the same findings. This could be at the same company if they change from methodology or at other companies that are using another agile methodology. It would be interesting to investigate whether and on what scale the size of the company influences the effectiveness of the implementation and use of these methods in the hardware environment. Moreover, research on the differences between companies that are fabricating different kinds of hardware could be interesting as well.

Another limitation is the bias of the researcher or the interviewee. Since a semi-structured interviewee method has been used not all the questions were prepared upfront. There could arise a bias in the way these unprepared questions are asked. Furthermore, since not all interviewees got the same questions some interviewees might have had the opportunity to talk more thoroughly than others. Also, some questions were built upon the previous question, and dependent on the answer the focus of the next question differs. Resulting in a difference in the in-depth answers of the different interviewees. Since all interviews are executed online via MS Teams it was hard to read body language which could have influenced the results.

Besides that, the amount of interviews is limited due to time constraints. More interviews could result in a wider perspective on the way of working making it more generalizable. Only 2 software engineers are interviewed making it hard to draw conclusions on that. Also, the sample size of the hardware engineers (6) and team leads (4) is limited. Increasing the number of interviewees will give more certainty about the validity of the results. For future research, it should therefore be advisable to extend the number of interviewees. Furthermore, having more interviews with people from the same company but from different departments could give more insight into the whole organization and if differences exist in the company. Moreover, it would be interesting to perform the same research in a couple of years to check whether the perception of people changed and if new insights on the way of working have been created. Also, the company is growing really fast, people are working on the agile way of working internally, and a lot of things are taking place at the same time. It could be interesting to see how this evolves over time, and what lessons we can learn from that.

Furthermore, the results of the framework are not fully validated by the interviewees. The framework is not evaluated on certain criteria to measure the outcome of the use of the new framework. For future research, it should be interesting to validate this model and make some improvements if needed. In Appendix G, the guidelines for how to validate the adapted framework are shown.

The company of research is in the middle of the transition, which means that the perception of people could be dependent on the current stage of the transition. Furthermore, it will also mean that the long-term effects of the transition can not yet be evaluated. For future research, it should therefore be interesting to investigate the long-term benefits and challenges of the use of agile methodology.

Finally, since the company at which the interviews have been conducted is a client of Accenture it could be possible that interviewees gave the politically correct answer. Although the interviewees are anonymous in this thesis there could still be a threat that their identity could be exposed. Furthermore, since qualitative research is used it is not possible to draw objective conclusions on the improvement in numbers. This makes it hard to compare the results with existing or future research.

6 Conclusion

Agile development approaches are becoming more popular the last years. With large success in the software industry, the interest in using agile methods has shifted to the development of physical products. Since agile has been originally developed for the software industry, it does not cover aspects of the hardware development environment. However, some companies do want to switch from this traditional approach to a hybrid or even complete agile approach. Reasons for this can vary from the wish to reduce time-to-market, improve communication and collaboration among the team, or because they do not want to lack behind their competitors even though they did not know if the method will be effective for their own company. The problem is that the implementation and use of the agile methodology in the traditional problem face some issues concerning effectiveness, performance, and mismatch with expectations. This has led to the research question:

> **”How to effectively implement and use the agile methodology in the hardware environment?**

To answer this question, three subquestions have been formulated. These focus on the success factors and barriers to the implementation of the agile methodology, the perceived benefits of the use of agile, and how the agile method could be adapted to meet the characteristics of the hardware environment.

chapter 2 and chapter 4 highlighted that the success factors and barriers are difficult to see independently of each other, as they are often linked and can reinforce one another. The factors that play an important role are on the one hand the mindset of the management and the employees, and on the other hand the availability of pilots and training. The mindset has been found to influence the way the new method is adopted and executed in the company. Training enhances the right mindset and helps to get the right skills to perform the implementation. Pilots are necessary to customize the approach to the company and create a smooth transition for the complete firm.

The benefits of the use have been divided into four categories; people, process, organization, and product. People will get more focus on their tasks since prioritization and planning of the work have been improved. Agile tools improve transparency and allow employees to track dependencies with other teams, therefore collaboration can be improved. Also, the use of agile rituals improves the communication and quality of the work. Feedback provided in retrospective meetings can be used in future situations and for example, daily stand-ups and planning events improve communication due to more frequent, more transparent, and better-structured meetings. The use of early prototypes is beneficial for the process as well. By creating a consistent structure of the teams the communication and collaboration improved.

The agile methodology should be adapted at some points to better match the characteristics of the hardware environment. The adaptions have been categorized into people, process, organization, and product. For people, there should be more emphasis on creating the right mindset by providing training. The agile way of working is hard to understand for people in the hardware environment since they do not feel the urge to change and want to keep doing their job instead of spending time on meetings. They are used to the old way of working and following agile practices does feel naturally contradictory for hardware developers. A lack of clear communication about the purpose of using agile results in resistance to change. Therefore, there should be a focus on creating the right mindset for employees. On the process level, it is necessary to adjust the rituals to the needs of the teams. Since hardware deals with physical components that have longer lead times, the meeting intensity should be adapted to that. This should be an iterative process, led by the status of the project. Furthermore, there should be more focus on the alignment of tools to match with the other teams/suppliers since in hardware the dependencies on other teams/suppliers are harder to control because of the physical nature of the products. Moreover, from an organizational perspective, it is important that team structures are aligned in the organization. Also, people should no longer be specialists in one specific part but be multi-specialized. This could, however, be hard to achieve since some companies have to deal with a high level of complexity that asks for specialists. On the product level, the focus should be on the iterations and rapid prototyping even if the prototype is not functional. While the method originally state that after each sprint you should deliver something to be tested it is more important to have a prototype to start the conversation.

The way of implementing and using the agile method should match the company and the people. For the implementation, the mindset of the people should be right in place and the pilot can show if the proposed way of working is right. For the use, it is of significant value that besides the people aspect, the tools, and organizational structure are aligned. Also here the rituals, prototyping, training, and way of specialization should match the company and the teams. The findings of this thesis can be used as a guideline to implement and use the agile methodology in large-scale hardware companies.

References

Arroyo, A. (2023). *Hardware development process: from idea to production.* Retrieved from https://wikifactory.com/+wikifactory/stories/hardware-development-process-from-idea-to-production

Atzberger, A., Gerling, C., & Schrof, J. (2019). *Evolution of the hype around agile hardware development.*

Atzberger, A., & Paetzold, K. (2019). Current challenges of agile hardware development: What are still the pain points nowadays? In (Vol. 2019-August, p. 2209-2218). Cambridge University Press. doi: 10.1017/dsi.2019.227

Augustin, L., & Schabacker, M. (2019, 07). Combining scrum and design thinking for a highly iterative and user-centric hardware development project. *Proceedings of the Design Society: International Conference on Engineering Design*, *1*, 2189-2198. doi: 10.1017/dsi.2019.225

Beck, K. (2001). Manifesto for agile software development.
doi: agilemanifesto.org

Bjarnason, E., Wnuk, K., & Regnell, B. (2011). A case study on benefits and side-effects of agile practices in large-scale requirements engineering. In *Proceedings of the 1st workshop on agile requirements engineering.* New York, NY, USA: Association for Computing Machinery. Retrieved from https://doi-org.tudelft.idm.oclc.org/10.1145/2068783.2068786 doi: 10.1145/2068783.2068786

Brandl, F. J., Kagerer, M., & Reinhart, G. (2018). A hybrid innovation management framework for manufacturing – enablers for more agility in plants. *Procedia CIRP*, *72*, 1154-1159. Retrieved from https://www.sciencedirect.com/science/article/pii/S2212827118305389 (51st CIRP Conference on Manufacturing Systems) doi: https://doi.org/10.1016/j.procir.2018.04.022

Böhmer, A., Beckmann, A., & Lindemann, U. (2015, 12). Open innovation ecosystem -makerspaces within an agile innovation process..

Cadence. (2023). *The phases of the hardware product development lifecycle.* Retrieved from https://resources.pcb.cadence.com/blog/2022-the-phases-of-the-hardware-product-development-lifecycle

Campanelli, A., Bassi, D., & Silva Parreiras, F. (2017, 05). Agile transformation success factors: A practitioner's survey. In (p. 364-379). doi: 10.1007/978-3-319-59536-8_23

Charvat, J. (2003). Project management methodologies : selecting, implementing, and supporting methodologies and processes for projects. Wiley, Hoboken, NJ.

Chuang, S.-W., Luor, T., & Lu, H.-P. (2014). Assessment of institutions, scholars, and contributions on agile software development (2001–2012). *Journal of Systems and Software*, *93*, 84-101. Retrieved from https://www.sciencedirect.com/science/article/pii/S0164121214000697 doi: https://doi.org/10.1016/j.jss.2014.03.006

Conboy, K., Coyle, S., Wang, X., & Pikkarainen, M. (2011, 07). People over process: Key challenges in agile development. *IEEE Software*, *28*. doi: 10.1109/MS.2010.132

Cooper, R. G. (2014). What's next? after stage-gate. *Research Technology Management*, *57*, 20-31. doi: 10.5437/08956308X5606963

Dikert, K., Paasivaara, M., & Lassenius, C. (2016). Challenges and success factors for large-scale agile transformations: A systematic literature review. *Journal of Systems and Software*, *119*, 87-108. Retrieved from https://www.sciencedirect.com/science/article/pii/S0164121216300826 doi: https://doi.org/10.1016/j.jss.2016.06.013

Dingsøyr, T., & Moe, N. B. (2013, aug). Research challenges in large-scale agile software development. *SIGSOFT Softw. Eng. Notes*, *38*(5), 38–39. Retrieved from https://doi-org.tudelft.idm.oclc.org/10.1145/2507288.2507322 doi: 10.1145/2507288.2507322

Dingsøyr, T., Nerur, S., Balijepally, V., & Moe, N. (2012, 06). A decade of agile methodologies: Towards explaining agile software development. *Journal of Systems and Software*, *85*, 1213-1221. doi: 10.1016/j.jss.2012.02.033

Drechsler, R., & Breiter, A. (2003). *Hardware project management-what we can learn from the software development process for hardware design?* Retrieved from https://www.researchgate.net/publication/200503044

Ebert, C., & Paasivaara, M. (2017). *Scaling agile a look at some frameworks.* IEEE Software: Software Technology.

Fuchs, C., & Golenhofen, F. (2018). *Mastering disruption and innovation in product management: Connecting the dots.*

Garzaniti, N., Briatore, S., Fortin, C., & Golkar, A. (2019). Effectiveness of the scrum methodology for agile development of space hardware. In *2019 ieee aerospace conference* (p. 1-8). doi: 10.1109/AERO.2019.8741892

Gupta, R., Jain, S., Singh, B., & Jha, S. K. (2019, 02). Key factors in scaling up agile team in matrix organization. In (p. 1-5). doi: 10.1145/3299771.3299793

Gutiérrez Rivas, J., Berthing, J., Fernández García-Valdecasas, D., & Díaz, J. (2012, 09). Safety-critical platform model based on certification standards..

Hajjdiab, H., & Taleb, A. (2011, 09). Adopting agile software development: Issues and challenges. *International Journal of Managing Value and Supply Chains*, *2*, 1-10. doi: 10.5121/ijmvsc.2011.2301

Hunter, T. (2020). *Early evangelists of agile engineering wouldn't mind watching it die.* Retrieved from https://builtin.com/software-engineering-perspectives/lean-agile-methodology-software-engineering

Javatpoint. (2021). *Difference between agile and waterfall model.* Retrieved from https://www.javatpoint.com/agile-vs-waterfall-model

Jungwirth, K. (2022). *Product development with the stage-gate® process (part 3): Procedure and structure.* Retrieved from https://www.inloox.com/company/blog/articles/product-development-with-the-stage-gate-r-process-part-3-procedure-and-structure/

Kalenda, M., Hyna, P., & Rossi, B. (2018). Scaling agile in large organizations: Practices, challenges, and success factors. *Journal of Software: Evolution and Process*, *30*(10), e1954. Retrieved from https://onlinelibrary.wiley.com/doi/abs/10.1002/smr.1954 (e1954 smr.1954) doi: https://doi.org/10.1002/smr.1954

Koehnemann, H., & Coats, M. (2009). Experiences applying agile practices to large systems. In *2009 agile conference* (p. 295-300). doi: 10.1109/AGILE.2009.59

Leffingwell, D. (2007). *Scaling software agility: Best practices for large enterprises (the agile software development series)*. Addison-Wesley Professional.

Lenfle, S., & Loch, C. (2010). *Lost Roots: How Project Management Came to Emphasize Control Over Flexibility and Novelty* (Post-Print No. hal-00557549). HAL. Retrieved from https://ideas.repec.org/p/hal/journl/hal-00557549.html

Lima, G. L. B., Ferreira, G. A., Saotome, O., da Cunha, A. M., & Dias, L. A. V. (2015). Hardware development: Agile and co-design. *2015 12th International Conference on Information Technology - New Generations*, 784-787.

Long, K., & Starr, D. (2008, 09). Agile supports improved culture and quality for healthwise. In (p. 160-165). doi: 10.1109/Agile.2008.61

Lévárdy, V. (2006). *"model-based framework for the adaptive development of engineering systems*. Technische Universität München.

Mancin, E. (2016, 01). Make your enterprise agile transformation initiative an awesome success. In (Vol. 422, p. 191-202). doi: 10.1007/978-3-319-27896-4_16

Mccormick, M. (2012). *Waterfall vs. agile methodology.* Retrieved from http://mccormickpcs.com/images/Waterfall_vs_Agile_Methodology.pdf

Moravcová, B., & Legény, F. (2016). "agile adoption" in it companies - building a change capability by qualitative description of agile implementation in different companies. In T. Borangiu, M. Dragoicea, & H. Nóvoa (Eds.), *Exploring services science* (pp. 251–262). Cham: Springer International Publishing.

Mosher, T., Kolozs, J., & Wilder, E. (2018, 09). Agile hardware development approaches applied to space hardware.. doi: 10.2514/6.2018-5233

Naslund, D., & Kale, R. (2020, 10). Is agile the latest management fad? a review of success factors of agile transformations. *International Journal of Quality and Service Sciences*, *ahead-of-print*. doi: 10.1108/IJQSS-12-2019-0142

Naveen. (2015). *What is waterfall model in software testing and what are advantages and disadvantages of waterfall model.* Retrieved from https://testingfreak.com/waterfall-model-software-testing-advantages-disadvantages-waterfall-model/

Nerur, S., & Balijepally, V. (2007, mar). Theoretical reflections on agile development methodologies. *Commun. ACM*, *50*(3), 79–83. Retrieved from https://doi-org.tudelft.idm.oclc.org/10.1145/1226736.1226739 doi: 10.1145/1226736.1226739

Nerur, S., Mahapatra, R., & Mangalaraj, G. (2005, may). Challenges of migrating to agile methodologies. *Commun. ACM*, *48*(5), 72–78. Retrieved from https://doi-org.tudelft.idm.oclc.org/10.1145/1060710.1060712 doi: 10.1145/1060710.1060712

Nguyen-Duc, A., Weng, X., & Abrahamsson, P. (2018, 10). A preliminary study of agility in business and production: Cases of early-stage hardware startups. IEEE Computer Society. doi: 10.1145/3239235.3267430

Ochodek, M., & Kopczyńska, S. (2018). Perceived importance of agile requirements engineering practices – a survey. *Journal of Systems and Software*, *143*, 29-43. Retrieved from https://www.sciencedirect.com/science/article/pii/S0164121218300955 doi: https://doi.org/10.1016/j.jss.2018.05.012

Omidvarkarjan, D., Hofelich, M., Conrad, J., Klahn, C., & Meboldt, M. (2023). Teaching agile hardware development with an open-source engineering simulator: An evaluation with industry participants. Retrieved from https://doi.org/10.3929/ethz-b-000598611 doi: 10.3929/ethz-b-000598611

Oveseen, N., & Dowlen, C. (2012). *The challenges of becoming agile-experiences from new product development in industry and design education.*

Paasivaara, M., Behm, B., Lassenius, C., & Hallikainen, M. (2018, 10). Large-scale agile transformation at ericsson: a case study. *Empirical Software Engineering*, *23*. doi: 10.1007/s10664-017-9555-8

Paterek, P. (2017, 09). Agile transformation in project organization: Knowledge management aspects and challenges..

Pinto, J., & Slevin, D. (1988, 06). Critical success factors across the project life cycle. *Project Management Journal*, *19*, 67.

Rodríguez, P., Markkula, J., Oivo, M., & Turula, K. (2012). Survey on agile and lean usage in finnish software industry. In *Proceedings of the acm-ieee international symposium on empirical software engineering and measurement* (p. 139–148). New York, NY, USA: Association for Computing Machinery. Retrieved from https://doi-org.tudelft.idm.oclc.org/10.1145/2372251.2372275 doi: 10.1145/2372251.2372275

Royce, W. (1970). *Managing the development of large software systems.* IEEE WESCON.

Sandur Madhu Murthy, S. (2020). Qualitative study of scaled agile framework: An organizational approach. *ProQuest Dissertations and Theses*, 201. Retrieved from https://www.proquest.com/dissertations-theses/qualitative-study-scaled-agile-framework/docview/2476539749/se-2 (Copyright - Database copyright ProQuest LLC; ProQuest does not claim copyright in the individual underlying works; Last updated - 2023-03-07)

Schmidt, T., Weiss, S., & Paetzold, K. (2018a). *Agile development of physical products - an empirical study about motivations, potentials and applicability*.

Schmidt, T., Weiss, S., & Paetzold, K. (2018b). Expected vs. real effects of agile development of physical products: Apportioning the hype. In (Vol. 5, p. 2121-2132). Faculty of Mechanical Engineering and Naval Architecture. doi: 10.21278/idc.2018.0198

Schuh, G., Schröder, S., Lau, F., & Wetterney, T. (2016). Next generation hardware development: Requirements and configuration options for the organization of procurement activities in the context of agile new product development. In *2016 portland international conference on management of engineering and technology (picmet)* (p. 2583-2591). doi: 10.1109/PICMET.2016.7806809

Senapathi, M., & Srinivasan, A. (2013). Sustained agile usage: A systematic literature review. In *Proceedings of the 17th international conference on evaluation and assessment in software engineering* (p. 119–124). New York, NY, USA: Association for Computing Machinery. Retrieved from https://doi-org.tudelft.idm.oclc.org/10.1145/2460999.2461016 doi: 10.1145/2460999.2461016

Sommer, A. F., Hedegaard, C., Dukovska-Popovska, I., & Steger-Jensen, K. (2015, 1). Improved product development performance through agile/stage-gate hybrids: The next-generation stage-gate process? *Research Technology Management*, *58*, 34-44. doi: 10.5437/08956308X5801236

VersionOne. (2018). *12th annual state of agile report*.

Weichbroth, P. (2022, 08). A case study on implementing agile techniques and practices: Rationale, benefits, barriers and business implications for hardware development. *Applied Sciences*, *12*, 8457. doi: 10.3390/app12178457

Yin, R. (2003). *Case study research: Design and methods*. SAGE Publications.

Zhang, L. (2013, 10). Managing project changes: Case studies on stage iteration and functional interaction. *International Journal of Project Management*, *31*, 958-970. doi: 10.1016/j.ijproman.2012.11.014

A

Agile values and principles

Table A.1: Agile values and corresponding principles Beck (2001)

Agile values	Agile principles
Inidividual and interaction	Business people and developers work together
	Motivated individuals
	Face to face communication
	Self organizing team
	Reflection
Working software	Working software
	Technical excellence and enhanced agility
Customer collaboration	Satisfy customer
Responding to change	Welcome to changing requirements
	Frequent delivery
	Simplicity
	Sustainable development

B

HREC

Since human resources are involved in this research, approval from the Human Research Ethical Committee needs to be obtained. To get this approval I will complete the checklist, complete the informed consent materials and make a data management plan. I will also get a data processing agreement from the faculty data steward (Nicolas Dintzer). This application process will start immediately after the kick-off meeting to make sure the HREC has enough time to approve and the interviews can start on time.

Informed Consent Form – Master Thesis Management of Technology – Agile implementation in the hardware environment

You are being invited to participate in a research study titled Agile implementation in hardware environment. This study is being done by Laurens van Driessen from the TU Delft, in collaboration with Accenture NL. The role of Accenture NL is to provide access to the available knowledge inside the firm, make the connections with the high-tech company, and provide additional coaching during the project. Accenture is not in charge of the research and only has a facilitating role. The interest of Accenture is to use the outcome of the research (final thesis) as a basis that can be used in the agile transformation in the hardware environment for the client.

The purpose of this research study is to provide a framework how to effectively implement the agile methodology in the traditional model in the hardware environment and will take you approximately *60* minutes to complete. The data will be used for the foundation of the proposed solutions, compared with the literature to identify differences, and compared to other interviews to identify differences among the company. The data will be presented in my master thesis and published on the TU Delft repository. We will be asking you to participate in this interview and give explanation about the success factor and barriers you are facing in the current situation, your perceived individual and team output in terms of usage, communication, and performance, and how likely the proposed solutions will succeed in your opinion.

The interview will be either audio-recorded or videorecorded, based on your preference and whether the interview takes place online or face-to-face.

As with any data collection activity the risk of a breach is always possible. To the best of our ability your answers in this study will remain confidential. We will minimize any risks by gathering and storing as little personal data as necessary and storing the data on a secure and private Accenture OneDrive location, only accessible by the Accenture research team. The data from the interviews will be anonymised before publication in the thesis report. The research team has access to the following data: name, email address, audio/video recordings, transcripts, anonymous summary. People of Accenture outside this research team does not have access to any of the above data except for the anonymous summary that will be shared in my final thesis. The final thesis report will be published in the TU Delft thesis repository. Your participation in this study is entirely voluntary and you can withdraw at any time. You are free to omit any questions.
A summary of the interview will be provided to you after the interview. You are free to request removal of parts of the summary before publication in the thesis report. All other personal data collected (audio/video recording, notes, participant list) will be deleted at the latest 1 month after the completion of the project.

For any questions or complaints regarding the research, feel free to contact us:
Laurens van Driessen (Researcher), l.n.vandriessen@student.tudelft.nl, or Nikos Pachos-Fokialis (responsible researcher), N.Pachos-Fokialis@tudelft.nl

PLEASE TICK THE APPROPRIATE BOXES	Yes
A: GENERAL AGREEMENT – RESEARCH GOALS, PARTICPANT TASKS AND VOLUNTARY PARTICIPATION	
1. I have read and understood the study information dated 17-04-2023, or it has been read to me. I have been able to ask questions about the study and my questions have been answered to my satisfaction.	☐
2. I consent voluntarily to be a participant in this study and understand that I can refuse to answer questions and I can withdraw from the study at any time, without having to give a reason.	☐
3. I understand that taking part in the study involves: • An audio-recorded or video-recorded interview. • A summary of the interview based on the recording.	☐
4. I understand that the study will end in July 2023.	☐
B: POTENTIAL RISKS OF PARTICIPATING (INCLUDING DATA PROTECTION)	
5. I understand that taking part in the study also involves collecting specific personally identifiable information (PII) [name, email address] and associated personally identifiable research data (PIRD) [audio or video] with the potential risk of my identity being revealed.	☐
6. I understand that the following steps will be taken to minimise the threat of a data breach, and protect my identity in the event of such a breach anonymisation of the summary of the interviews before publication, secure data storage on the Accenture OneDrive, and access to the recordings is limited to the Accenture project team.	☐
7. I understand that personal information collected about me that can identify me, such as my name and email address, will not be shared beyond the study team.	☐
8. I understand that the (identifiable) personal data I provide will be destroyed the latest 1 month after the finish of the project.	☐
C: RESEARCH PUBLICATION, DISSEMINATION AND APPLICATION	
9. I understand that after the research study the de-identified information I provide will be used for a master's thesis report, which will be published in the TU Delft thesis repository. In the published report, only the summaries will be published.	☐
10. I agree that my responses, views or other input can be quoted anonymously in research outputs	☐
D: (LONGTERM) DATA STORAGE, ACCESS AND REUSE	
11. I give permission for the summary of the interviews that I provide to be archived in TU Delft repository so it can be used for future research and learning.	☐
12. I understand that access to this repository is open.	☐

Signatures

____________________________	___________________________	_________
Name of participant	Signature	Date

Study contact details for further information:

Laurens van Driessen
[redacted]
l.n.vandriessen@student.tudelft.nl

Date 21-Apr-2023

Contact person Dr. Cath Cotton, Policy Advisor
Academic Integrity

E-mail c.m.cotton@tudelft.nl

Human Research Ethics
Committee TU Delft
(http://hrec.tudelft.nl)

Visiting address
Jaffalaan 5 (building 31)
2628 BX Delft

Postal address
P.O. Box 5015 2600 GA Delft
The Netherlands

Ethics Approval Application: Agile implementation in the hardware environment
Applicant: Driessen, Laurens van

Dear Laurens van Driessen,

It is a pleasure to inform you that your application mentioned above has been approved.

In addition to any specific conditions or notes, the HREC provides the following standard advice to all applicants:

- In light of recent tax changes, we advise that you confirm any proposed remuneration of research subjects with your faculty contract manager before going ahead.
- Please make sure when you carry out your research that you confirm contemporary covid protocols with your faculty HSE advisor, and that ongoing covid risks and precautions are flagged in the informed consent with particular attention to this where there are physically vulnerable (eg: elderly or with underlying conditions) participants involved.
- Our default advice is not to publish transcripts or transcript summaries, but to retain these privately for specific purposes/checking; and if they are to be made public then only if fully anonymised and the transcript/summary itself approved by participants for specific purpose.
- Where there are collaborating (including funding) partners, appropriate formal agreements including clarity on responsibilities, including data ownership, responsibilities and access, should be in place and that relevant aspects of such agreements (such as access to raw or other data) are clear in the Informed Consent.

Good luck with your research!

Sincerely,

Dr. Ir. U. Pesch
Chair HREC
Faculty of Technology, Policy and Management

C

Challenges agile development

Table C.2: Challenges in agile development with actual and expected impact on a scale of 0-5 (Schmidt et al., 2018a)

	Actual	Expected
Establishing agile working attitude	2.95	2.8
Embedding agile development project teams in classically organized company	2.75	2.95
Interpreting agile practices from software industry to the development of physical products	2.4	2.8
Modularization of the product	2.25	1.9
Limited scalability to large projects	2.15	2.5
Long lead time project tasks (e.g. Procurement of machine tools, lang-term measurements)	2.1	2.75
Scrum roles do not fit to existing organization	2.05	2.7
Working incrementally	2.02	2
Internal process models need to be edited	2	2.4
Working in an high-frequency iterative manner	1.9	2.1
Project manager roles cannot be abolished	1.85	2.25
Employees do not understand scrum roles	1.8	2.2
External dependencies (e.g. To development partners and sservice providers)	1.75	1.8
Scrum requires many team meetings	1.7	2.05
Building prototypes quickly	1.65	2.1
Choosing appropriate kinds of prototypes (e.g. Documents, paper, models cad models, laboratory setups)	1.45	1.75
Changes of well-established software tools	1.4	1.7
Building prototypes at low costs	1.2	1.6
Changes (claim management) are revenues sources	1.2	1.8
Norms / certification do not allow agile working styles	1.1	1.8
Chaos caused by unstructuredness	1.05	1.85

Table C.1: Complete list of benefits of use agile in hardware environment categorized in soft and hard critera

	Soft criteria	**Hard criteria**
1	Improved communication	Improved product quality
2	Reduced risks in the project	Reduced development costs
3	Increased transparancy in the company	Early customer benefits
4	Improved product alignment with corporate strategies	Improved adherence to delivery dates
5	Improved integration of the customer and/or user	Shortened product development (time-to-market)
6	Improved learning processes and knowledge geeration	
7	Increased porject-related commuitment of all parties	
8	Improved customer understanding	
9	Improved control of complexity	
10	Increased flexibility to respond to changes	
11	Higher chances of product acceptance by the market	
12	Increased productivity of the development project	
13	Increased exploitation of emerging opportunites	
14	Incrased reaction speed to changes	
15	Increased effectiveness in the development project	
16	Improved customer and internal stakeholder satisfaction	
17	Improved development processes	

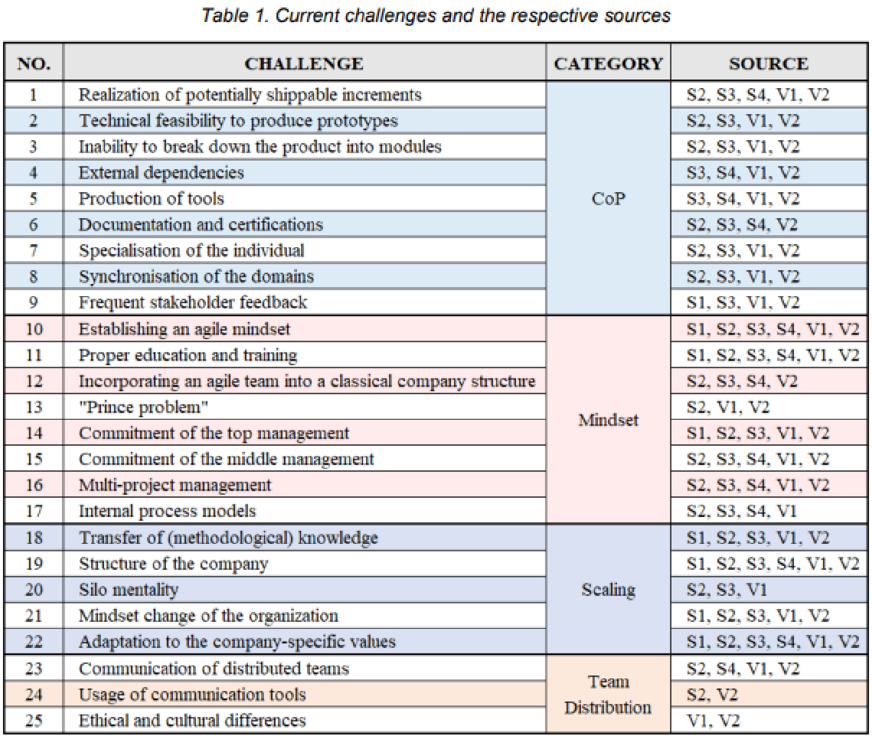


Table 1. Current challenges and the respective sources

NO.	CHALLENGE	CATEGORY	SOURCE
1	Realization of potentially shippable increments	CoP	S2, S3, S4, V1, V2
2	Technical feasibility to produce prototypes		S2, S3, V1, V2
3	Inability to break down the product into modules		S2, S3, V1, V2
4	External dependencies		S3, S4, V1, V2
5	Production of tools		S3, S4, V1, V2
6	Documentation and certifications		S2, S3, S4, V2
7	Specialisation of the individual		S2, S3, V1, V2
8	Synchronisation of the domains		S2, S3, V1, V2
9	Frequent stakeholder feedback		S1, S3, V1, V2
10	Establishing an agile mindset	Mindset	S1, S2, S3, S4, V1, V2
11	Proper education and training		S1, S2, S3, S4, V1, V2
12	Incorporating an agile team into a classical company structure		S2, S3, S4, V2
13	"Prince problem"		S2, V1, V2
14	Commitment of the top management		S1, S2, S3, V1, V2
15	Commitment of the middle management		S2, S3, S4, V1, V2
16	Multi-project management		S2, S3, S4, V1, V2
17	Internal process models		S2, S3, S4, V1
18	Transfer of (methodological) knowledge	Scaling	S1, S2, S3, V1, V2
19	Structure of the company		S1, S2, S3, S4, V1, V2
20	Silo mentality		S2, S3, V1
21	Mindset change of the organization		S1, S2, S3, V1, V2
22	Adaptation to the company-specific values		S1, S2, S3, S4, V1, V2
23	Communication of distributed teams	Team Distribution	S2, S4, V1, V2
24	Usage of communication tools		S2, V2
25	Ethical and cultural differences		V1, V2

Figure C.1: Challenges use agile(Atzberger & Paetzold, 2019)

Table C.3: Conflicts in agile development with actual and expected impact on a scale of 0-5 (Schmidt et al., 2018a)

	Actual	Expected
Acting according to te manifesto is difficult (not only applying practices)	2.6	2.8
Decentralized decision authority / loss of power for managers	2.4	2.4
Employees feel overconstrained	2	1.95
Attract qualified employees	1.8	1.85
Managers worry about their jobs	1.25	1.45
Teams are torn apart	1.2	1.55
Employees worry about their jobs	0.6	0.9

D

Success factors and barriers implementation

Table D.1: Success factors of agile implementation in hardware environment (Dikert et al., 2016; Naslund & Kale, 2020; Kalenda et al., 2018; Senapathi & Srinivasan, 2013; Campanelli et al., 2017; Gupta et al., 2019; Koehnemann & Coats, 2009; Paasivaara et al., 2018; Moravcová & Legény, 2016; Paterek, 2017; Mancin, 2016; Dingsøyr & Moe, 2013; Bjarnason et al., 2011; Pinto & Slevin, 1988; Conboy et al., 2011; Weichbroth, 2022; Hajjdiab & Taleb, 2011)

Management/Leadership	
Management	Make management support visible
	Educate management on agile
	management initiated the change
	engagement of executive helps since they better understood the practices and the processes
	Top management buy-in and support
	Top management plays in motivating employees for the agile transformation and help them grow
	Create a vision, business goals, and strategy for the transformation
Leadership	Convince employees that agile will bring the desired results
	Willingness to take risks
	Withstand external pressure to follow traditional waterfall processes
	Recognize the importance of change leaders
	Engage change leaders without baggage of the past
	Show strong commitment even if problems occur
	Communicate that change is non-negotiable
Communication	Communicate the change intensively, internally and to external stakeholders
	Create and communicate positive experiences in the beginning
	close connections and constant communication between teams and team members are necessary for successful agile development
	Build a common backlog
	Arrange social events
	Implement new communication tools and flow
Transparent	establish a transparent environment for openness in the team without fear of discussing problems to improve teamwork
	not feel observed because of transparency as the environment was very open-minded and relaxed.
	Increased collaboration and make the change transparent
People / employees	
Autonomy	Empower employees to make their own decisions
	Give employees the right balance of oversight and autonomy with a healthy level of remaining centralized decision-making
	Allow teams to self-organize
Training / Coaching	Management is educated on agile
	Involving change agents and agile champions
	Provide training on agile methods
	increase the level of knowledge and expertise on agile practices
	hire an external expert with broad and deep familiarity with agile development
	used exchange programs of agile coaches with other organizations and was also collaborating with universities
	a systematic training of coaches and other people inside of an organization (recurring events, presentations of external experts or communities of practice)
	ensures enough available coaches in an organization
	pair coaching, which combines agile experts with an expert with domain knowledge to enable more efficient and accurate coaching, proved to work well
	coached people by listening and asking questions, not dicatating and forcing them to change
	Teams need time and space to learn and adapt to the new way of working and change their habits
	Ensure availability of resources with engineering, business, and agile knowledge
Mindset / Alignment values	positive attitude toward agile methods
	Align the values among the organization
	define a common view on the change
	define roles, their responsibilities, and common definitions properly
	point out wrong ideas and misconceptions
	Change organizational culture/create an agile mindset
	emphasizing the importance of the culture shift necessary for the agile transformation
	introduction of agile breaks existing company rules and structures,
	Concentrate on agile values
Organization / structure	
Structure	Middle management has a clearly defined role in the transformation
	keep teams small
	Define, align and communicate new roles for employees
	the need to review and adapt existing roles and positions, often leading to personal expectations of roles, which conflict with agile ideas
	Having development teams at the same location (no remote communication necessary)
	Integrate team management within the team
	Achieving symbiosis between formal and informal organizational structures
	Restructure long-established teams to create an "awakening" effect
	Use a common agile framework for the whole organization
	Decreased number of projects per one employee
Customize agile approach	Customize the agile approach carefully
	Conform to a single approach
	Map to old way of working to ease adaptation
	Adapt agile practices to fit the individual context
	Continuous, evolutionary, step-by-step implementation and continuous learning
Piloting	Start with a pilot to gain acceptance
	Gather insights from a pilot
	Preparing well for the first program increment planning event
	After the general direction for the transformation is set, the company should set out to assess the expected costs, benefits and risks of the transformation
Tools	Invest in system improvements
	Make deels tools and infrastructure available which allow teams to transition their work procedures
	Use existing platforms if they can be adapted to future evolving needs
	tools must be aligned
Incentives and measures	Align measures with evolving practices throughout the transformation
	Develop consistent measures throughout the organization
	Establish personal performance evaluation by the scrum team
	Establishing incentives to adopt agile methods
	Measure the transformation according to the desired outcomes
	Besides the implementation of adapted performance measures, providing development and growth opportunities are another method to evoke the desired behavior

Table D.2: Barriers of agile implementation in hardware environment (Dikert et al., 2016; Naslund & Kale, 2020; Kalenda et al., 2018; Senapathi & Srinivasan, 2013; Campanelli et al., 2017; Gupta et al., 2019; Koehnemann & Coats, 2009; Paasivaara et al., 2018; Moravcová & Legény, 2016; Paterek, 2017; Mancin, 2016; Dingsøyr & Moe, 2013; Bjarnason et al., 2011; Pinto & Slevin, 1988; Conboy et al., 2011; Weichbroth, 2022; Hajjdiab & Taleb, 2011)

Management/Leadership	
Management support	top down mandate creates resistance
	management unwilling to change
	management in waterfall mode
Leadership	keeping the old bureaucracy
Communication	Misunderstanding agile concepts
	lack of guidance from literature
	Interfacing between teams difficult
	interpratation of agile different between teams
Transparent	unwilling to admit mistakes and learn from delivery failure
People / employees	
Autonomy	autonomous team model challenging
Training / Coaching	Lack of coaching
	lack of trainings
	Lack of knowledge, coaching, and training
	lack of business (customer, or product) understanding
	Lack of resources for trainings
Mindset / Alignment values	skeptcism towards the new way of working
	general resistance to change
	old commitments kept
	organizational culture at odds with agile values
	other functions unwilling to change
Engagement	Lack of commitment and teamwork
	Too much pressure and workload
	excessive ethusiasm
Organization / structure	
Structure/alignment	challenges in rearranging physical spaces
	global distribution challenges
	achieving technical consistency
	inconsistent processes and practices across teams
	Distributed environment
	using old and new aproaches side by side
	Integration with nonagile parts of the organization
	Middle managers’ role in agile unclear
	requirement refinement challenging
	creating and estimating user stories hard
	gap between long and short term planning
	pervasiveness of traditional development methods
Customize agile approach	agile customized poorly
	revering to old way of working
	challenges in adjusting to incremental delivery pace
	challenges in adjusting product launch activities
Piloting	Too fast roll-out
Tools	rewarding model not teamwork centric
	Quality assurance issues
	accommodating non-functinal testing
	lack of automated testing
	requirements ambiguity affects QA
	fragmented tooling and project-related data/measurements;
Incentives and measures	Measuring progress

E
Interview questions

In Figure E.1, the structure of the interview with hardware engineers is shown. This structure is similar to the structure of the interviews with the software engineers and the hardware team leads.

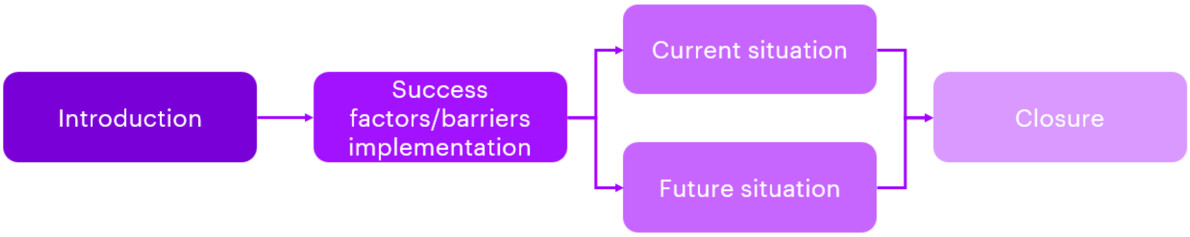


Figure E.1: Interview structure

The interview consists of five parts that are divided into 4 phases. In the first phase, a short introduction is given about the topic, and background questions are asked to identify the role of the interviewee in the company, how many years the interviewee works at the company, and how much experience the person has with agile. This part will be used to compare the answers of the interviewees with each other and to identify if the background has an influence on the answers of the interviewee. For the software engineers and hardware team leads the same questions will be asked.

The second phase focuses on the success factors and the barriers of the implementation of the agile methodology in the hardware environment. Interviewees are asked how they have experienced the transformation and what they consider to be a success factor. Also, aspects that hinder the implementation are asked. For the software engineers and hardware team leads the same questions will be asked. For the team leads I added a question about the decision of the implementation of the agile approach.

The third phase consists of two parts: the current situation and the future situation. These two parts will be asked simultaneously in this phase. The part is divided into five main subjects, consisting of the way of working, agile tools, agile maturity, collaboration, and performance. The questions about the way of working focus on the advantages and limitations of the current way of working and what could be improved in the future situation by using more agile practices. A question about the added value of agile tools will follow this. The next question will be about the adoption of the agile mindset/culture and how mature they are. The questions about collaboration focus on the way of communication within the team, outside the team, and with external suppliers, and on the dependencies that the people experience to other teams. The final part is on the productivity and quality of the way of working in the current system and how this can be improved in the future situation. For the software engineers, this part is about the current way of working with the agile methodology and how they think the way of working could be beneficial for the hardware environment. I also want to know how agile practices should be changed in their opinion to work properly in the hardware environment. For the interview with hardware team leads, I added a question about the manager's skills that are needed to lead an agile team. I also changed the other questions so it is more focused on a manager's perspective.

The final phase of the interview is the closure. In this part, the interviewee will be informed about the next steps of the research. The summary of the interview will be sent to the interviewee after the interview so the interviewee could check if the answers to the questions are well interpreted. Also, I will thank the interviewee for their participation and effort. This part will be the same for all interviews, so also for the software engineers and hardware team leads.

Interviews “Agile implementation in hardware environment”

Software engineers:

Background

1. What is your experience at ASML? Could you briefly explain what your position/role is, years of experience, and what your primarily tasks/responsibilities are?
2. Do you have experience with using Agile? If yes, how many years?

Current situation

3. What are the advantages of working with agile?
4. What are the disadvantages of working with agile?

Future hardware (brainstorm based on experience in software)

5. Do you think the agile way of working could be beneficial in the hardware environment?
6. How should the agile methodology be changed to work properly in the hardware environment?
7. What are success factors for the implementation of the agile methodology?
8. What factors hinder the implementation of the agile method? How can these be solved?

Hardware engineers

Background

1. What is your experience at ASML? Could you briefly explain what your position/role is, years of experience, and what your primarily tasks/responsibilities are?
2. Do you have experience with using Agile? If yes, how many years?

Implementation

Success factors & barriers

3. What do you consider a success factor for the implementation of the agile methodology?
4. Are there factors that hinder the implementation of the agile method? How can these be solved?

Current situation

Way of working

5. What do you consider to be an advantage regarding the use of agile in your current way of working?
6. What are limitations in the current way of working?
7. How do you think your way of working could be improved with the integration of more agile practices?

Agile tools

8. How do you see the added value (sprints, visualization, kanban, prioritization) of agile tools (e.g. Jira)?

Agile maturity

9. Do you think the agile mindset/culture is well adopted among the team? If no, how could this be achieved?

Collaboration

10. How does agile affect the way of communicating within the team, outside the team and with external suppliers? Could this be improved?
11. How do you experience the dependencies to other teams and suppliers? If issues with dependencies occur, how do you solve them?

Performance

12. How do you experience your own and team performance in terms of productivity and quality (re-work, iterations) in the current way of working? Could that be further improved?

Hardware Team leads

Background

1. What is your experience at ASML? Could you briefly explain what your position/role is, years of experience, and what your primarily tasks/responsibilities are?
2. Do you have experience with using Agile? If yes, how many years?

Implementation

Success factors & barriers

3. *Why was Agile chosen to use as a method?*
4. What do you consider a success factor for the implementation of the agile methodology?
5. Are there factors that hinder the implementation of the agile method? How can these be solved?

Current situation

Way of working

6. What do you consider to be an advantage regarding the use of agile in your current way of working?
7. What kind of managers skills are needed in an agile team?
8. What are limitations in the current way of working?
9. How do you think the way of working of your team could be improved with the integration of more agile practices?

Agile tools

10. How do you see the added value (sprints, visualization, kanban, prioritization) of agile tools (e.g. Jira) as a team lead?

Agile maturity

11. Do you think the agile mindset is well adopted among the team? If no, how could this be achieved?

Collaboration

12. How is the current way of communicating within the team, outside the team and with external suppliers? Could this be improved?
13. How do you experience the dependencies to other teams and suppliers? If issues with dependencies occur, how do you solve them?

Performance

14. How do you experience your team performance in terms of productivity and quality in the current way of working? Could that be further improved?

F
Interview summaries

F.1. Interviewee 1

The interviewee mentions that the implementation of agile is still in progress and estimates that it will take about five years to fully transition. They explain that initially, project leaders were skeptical of the agile approach but eventually realized the benefits of having more frequent feedback loops and the ability to influence priorities every two weeks.

The interviewee also talks about the challenges of applying agile practices in hardware development. They mention the difficulty of tracking progress in hardware projects, as well as the need for clear communication between hardware and software teams. They highlight the importance of early feedback and continuous alignment among team members.

The interviewee emphasizes that the agile approach should be adapted to suit the hardware environment and that each team may have its own unique implementation of agile practices. They mention the significance of practices such as continuous integration and continuous deployment in software development and acknowledge that some practices may be more applicable to software than hardware.

The interviewee mentioned the value of agile methodologies in improving collaboration, feedback loops, and adaptability in both software and hardware development. They highlight the importance of aligning teams and continuously checking and adjusting their worldviews to keep up with the complex and ever-changing environment.

F.2. Interviewee 2

The interviewee emphasizes the importance of test-driven development, where code is tested throughout the development process to accommodate changing requirements. The interviewee also highlights the need for constant interaction with stakeholders to ensure the delivery of the right solution. They believe that agile communication is superior to the traditional waterfall model as it allows for continuous feedback and avoids misunderstandings.

Regarding challenges, the interviewee mentions the difficulty of estimating unknowns during the project's initial stages and the risk of scope creep. They also address the debate about delivering incomplete products, acknowledging that while it may not be ideal, it allows for feedback and ensures that the project is heading in the right direction. They further discuss the applicability of agile practices in hardware development, suggesting that modifications can be made to adapt to the longer timeframes and complexity of hardware projects.

In terms of specific practices, the interviewee finds value in stand-up meetings as a platform for raising and resolving blocking issues rather than just providing status updates. They also emphasize the importance of backlog refinement meetings to discuss upcoming tasks and provide a broader overview of the team's work.

Regarding the implementation of agile in a hardware environment, the interviewee suggests starting with a trial period to allow teams to experience the benefits firsthand. They caution against forcing adoption and instead recommend helping individuals understand the advantages and encouraging them to make an informed decision. The challenges in implementing agile primarily stem from the natural resistance to change and the attachment to familiar ways of working. The interviewee advises against enforcing agile practices dictatorially, emphasizing the importance of management supporting the transition and allowing teams to explore agile methodologies willingly.

F.3. Interviewee 3

Several aspects of the implementation of Agile in the hardware environment were discussed. Firstly, the interviewee mentioned the importance of everyone speaking the same language within the company and understanding their roles. In the past, there were different terms used for the same thing which caused confusion and made communication difficult.

Next, the interviewee talks about the challenges faced in implementing Agile in the hardware environment. They mentioned that it is crucial to have experienced individuals in the roles of product owner and scrum master. Besides that, it is important to look ahead. Nowadays, sometimes they only look one quarter ahead, while hardware design requires more. Iterations and deliverables need to be carefully managed to ensure timely completion.

The interviewee emphasizes the role of management in the agile implementation. The management team sets priorities and decides which tasks should be tackled first. They also mentioned the need for accurate estimates to plan and allocate resources effectively. However, the interviewee notes that estimating the effort required for each task accurately is challenging and often leads to underestimation.

In terms of challenges, the interviewee highlights the need for improved communication between teams and the management of dependencies. They mentioned that there is currently no system in place to track and manage dependencies between different teams and projects. This can lead to delays and integration issues when different teams need to work together.

F.4. Interviewee 4

The interviewee mentioned that the company has undergone significant changes over the years and has hired many new employees. With the increase in projects and changes, transparency became crucial. Previously, there was no clear overview of project statuses and progress, but now there is a backlog and a focus on priorities. The interviewee emphasized the importance of transparency in managing multiple projects with a large team.

During the transition, there were challenges such as a lack of understanding and experience, as well as the initial abandonment of waterfall planning without a suitable alternative. External consultants were initially relied upon, but they lacked understanding of the company's existing processes and were not taken seriously by the organization. The speaker also mentioned resistance to change within the company, as some employees were accustomed to traditional methods and were skeptical of Agile.

The interviewee highlighted the benefits of Agile, particularly the improved transparency and prioritization, which helped engineers better understand project goals and deadlines. The interviewee mentioned that Agile created a culture of collaboration and willingness to help each other. However, there were still some challenges, including the resistance against increased administrative tasks for engineers. Overall, the interviewee acknowledged the positive impact of Agile in their organization but also recognized areas where improvements could be made.

F.5. Interviewee 5

The interviewee mentioned that initially, they had difficulties understanding and accepting the Agile approach, but with the support of experienced project leaders and management, they gradually embraced it. The interviewee emphasized the importance of having a plan, even though it doesn't need to be set in stone, and the advantage of regularly reviewing and adjusting priorities in Agile work. They mentioned that the implementation at the company was chaotic and lacked proper management and guidance. They highlighted the need for good management and practical pilot projects to facilitate a successful transition.

Currently, they are using agile practices such as daily stand-up meetings, sprint planning, and backlog reviews. They mentioned the importance of involving the team in planning and estimation, and the flexibility to adapt plans based on changing circumstances. They acknowledged that the speed of their system, particularly in terms of creating roadmaps, could be improved. However, they observed increased productivity, quality of work, and clear boundaries within the team. They also mentioned that Agile methods have positively impacted time-to-market, although challenges arise due to the involvement of many new team members and various projects. The communication within and outside the team has improved. Also, the estimations of work and time management improved. They emphasized the importance of commitment and realistic estimation, as well as the flexibility to adjust plans as needed. Overall, they believe that Agile has brought positive changes to their work environment.

F.6. Interviewee 6

The interviewee mentioned that initially, only one or two teams used Scrum, but later the company enrolled it in more teams. The transition to Agile was smooth but required time to explain the approach to outsourcing teams. Challenges included reducing administrative time on Jira, aligning project features with the quarterly schedule, and managing scope changes and dependencies are mentioned. Scrum Masters and product owners intervened to address delays and ensure timely responses.

Improvements were suggested, including the ongoing project of adding Jira Align and integrating with Excel for better collaboration. However, it was noted that Agile couldn't solve hardware or supply chain issues. Agile practices improved communication within teams and with stakeholders, but communication with other departments and handling dependencies remained challenging. Engineers appreciated improved communication, but some team members had concerns about the administration. According to the interviewee, agile likely increased engineers' productivity, but the impact on work quality was not notable. Agile allowed customization and continuous improvement, fostering a positive working culture.

F.7. Interviewee 7

The interviewee highlights the main challenge during the transition from waterfall to Agile as resistance from the sponsors, who desired more involvement. This resulted in conflicts and adjustments during management reviews. The lack of familiarity with capability-based increases and the shift in power dynamics contributed to the friction. Overall, the transition to Agile was smooth, especially with the adoption of Jira for planning and tracking. However, integrating upstream processes and collaborating across teams posed challenges, indicating a need for improved alignment and functionality in Jira.

Initially, there was resistance from the work floor due to additional administrative work, but once they embraced the new approach, they found it beneficial for clarity and early issue detection. Training was provided to adapt to the new way of working, with the interviewee personally receiving around 10-15 days of training. Some employees preferred a small guide or manual for reference. A different mindset and thinking were crucial in transitioning from a hierarchical to a train-based project organization. Line managers may struggle with this change but should understand that both entities can coexist.

At the organizational level, challenges involved increased overhead and hiring/training suitable personnel for new roles, particularly Product Owners. Internal training was found to be faster and more effective than external hiring. Effective communication was sometimes challenging due to varying roles and responsibilities. Communication primarily occurred through meetings. Communication with other teams and suppliers was limited, with work packages transferred only when necessary. Dependencies on suppliers could cause delays, emphasizing the importance of clear agreements and deadlines.

The interviewee expressed skepticism regarding significant improvements in productivity or time to market. Although collaboration and information sharing improved, the expected increase in productivity was not observed.

F.8. Interviewee 8

The interviewee express mixed feelings about the successful implementation of the agile methodology. They believe there are some good aspects to Agile, but they feel that the implementation was driven more by management's desire to introduce something new rather than as a means to achieve specific goals. The expectations of becoming 30% more efficient with Agile were not met, and there was resistance among some people in the organization.

The interviewee notes that Agile has improved communication and collaboration within teams, but it also requires time for coordination, resulting in additional overhead. They express doubt that the expected efficiency gains will be achieved and that people still perceive Agile as a way to work faster. They mention that the implementation approach seemed to be ”fail fast, learn fast” and focused on trial and error. By allowing teams more freedom to experiment the management supported the transition according to the interviewee. However, there was a disconnect between management's perspective and the teams' perspective, as the teams felt the lack of clear guidance and the need to figure things out themselves.

The interviewee indicates that Agile is primarily used for planning rather than being applied in a phased manner or for faster iterations and demos. They note that in hardware projects, the traditional approach is still followed, while Agile is more prominent in software projects

F.9. Interviewee 9

The interviewee has worked as both a team leader and a group leader and has witnessed the transition firsthand. They highlight the challenge of agile teams where everyone is expected to be able to take over each other's work, which is not feasible in the hardware environment due to the specialization required. They explain that they had difficulties in the beginning when working with other teams because it was not easy to exchange team members.

The interviewee mentions the challenges of demonstrating progress in the hardware, where lead times are longer and it's not possible to show tangible results as frequently as in software development. They mention the importance of prioritizing and planning, and how the Agile method allows for better visibility and understanding of the work being done. They also mention the need to allocate time for quality items and the shift towards better planning and predictability.

The interviewee acknowledges the initial resistance from their team members. They mention that some team leaders were initially skeptical about the changes and had concerns about the increased visibility of their work. However, over time, they have seen improvements in how teams function and the clarity it brings to their work. They mention that the implementation was done gradually, with some teams starting earlier and others following later. They note that it took time for the teams to adapt and for the implementation to become smoother. The interviewee mentions that each team has implemented Agile in its own way, which may have required better organization and coordination from management. They mention that different teams had different methods and approaches, which led to some resistance and the need for structuring. They discuss efforts to streamline the different approaches and create a more cohesive structure, such as ensuring that each team has a product owner and a scrum master.

F.10. Interviewee 10

The interviewee mentions that they went through multiple transitions, including switching between classic SAFe, scrum, and kanban. The interviewee explains that working on software products allows for more flexibility with scrum, but when dealing with external suppliers and complex products like those in hardware, it becomes more challenging to align expectations and deliverables. They discuss the difficulties of integrating third-party deliveries and not having full control over the development process.

The interviewee goes on to explain that the transition to SAFe was initially done through several pilots, starting with rituals and lean SAFe before moving to a full implementation. However, they also mention that they have developed their own methodology, adapting SAFe to their specific needs. They emphasize the importance of managing risks, dependencies, and overall planning in their approach.

When discussing applying agile practices to hardware development, the interviewee acknowledges the value of certain rituals and feedback sessions. However, they point out that the nature of hardware development, with long timelines and complex processes, makes it challenging to fully implement agile methods. The interviewee mentions that some tasks within hardware development can be done using scrum, but overall, the process is more suited for a traditional design methodology where engineers work on a project for an extended period.

The interviewee expresses the difficulty of applying agile to hardware development and the limitations of using prototypes as intermediate deliverables. They mention that prototypes are often created for the sake of the PI event and may not add significant value to the actual development process. They discuss the trade-off between mitigating risks and delivering the final product in a timely manner. They also mention that their processes are already automated to address risks, reducing the need for additional prototypes.

Finally, the interviewee highlights that while prototypes can have some value, they do not fully represent the end product. They mention the use of 3D printing for mechanical components but caution against making assumptions based solely on prototypes. They conclude that their approach to hardware development may not align perfectly with agile principles but emphasizes the importance of adapting methodologies to suit the specific needs and constraints of the domain.

F.11. Interviewee 11

The interviewee mentions that there is generally positive acceptance of the transition among team members, although some individuals may have initial reservations. Most team members are open to trying out the new approach and are curious about its implementation.

In previous teams, which were part of large multidisciplinary trains, the implementation of SAFE was relatively smooth. The team members adapted well to the daily stand-ups, planning with JIRA, and PI events. However, some resistance was observed from external coaches brought in to focus on creating high-performing teams. These coaches went too deep into methodologies to elevate performance but did not always meet the specific needs of the teams.

The interview touches on the application of Agile principles at the low level of hardware design. Some teams successfully applied Agile practices, such as slicing workshops and iterative development, to design hardware solutions. However, the interviewee highlights the challenge of applying Agile principles in their own hardware development process, which traditionally follows a waterfall approach. They mention the struggle of breaking down the large deliverables into smaller manageable tasks while adhering to the one-time-right mindset.

The interviewee also discusses the difficulties faced during the transition, such as the need to shift ownership to designers, who may resist taking on more responsibility. Additionally, they mention challenges related to forward planning, as the previous waterfall approach did not adequately account for longer lead times and delayed ordering of necessary components.

Regarding tools, the interviewee acknowledges that JIRA is generally seen as a positive tool for tracking progress and managing tasks, although some older team members may struggle with its usage. They mention the need for a unified tool that combines both Agile functionalities and long-term planning, as currently, they rely on two separate tools (JIRA and Project Online) that are not integrated effectively.

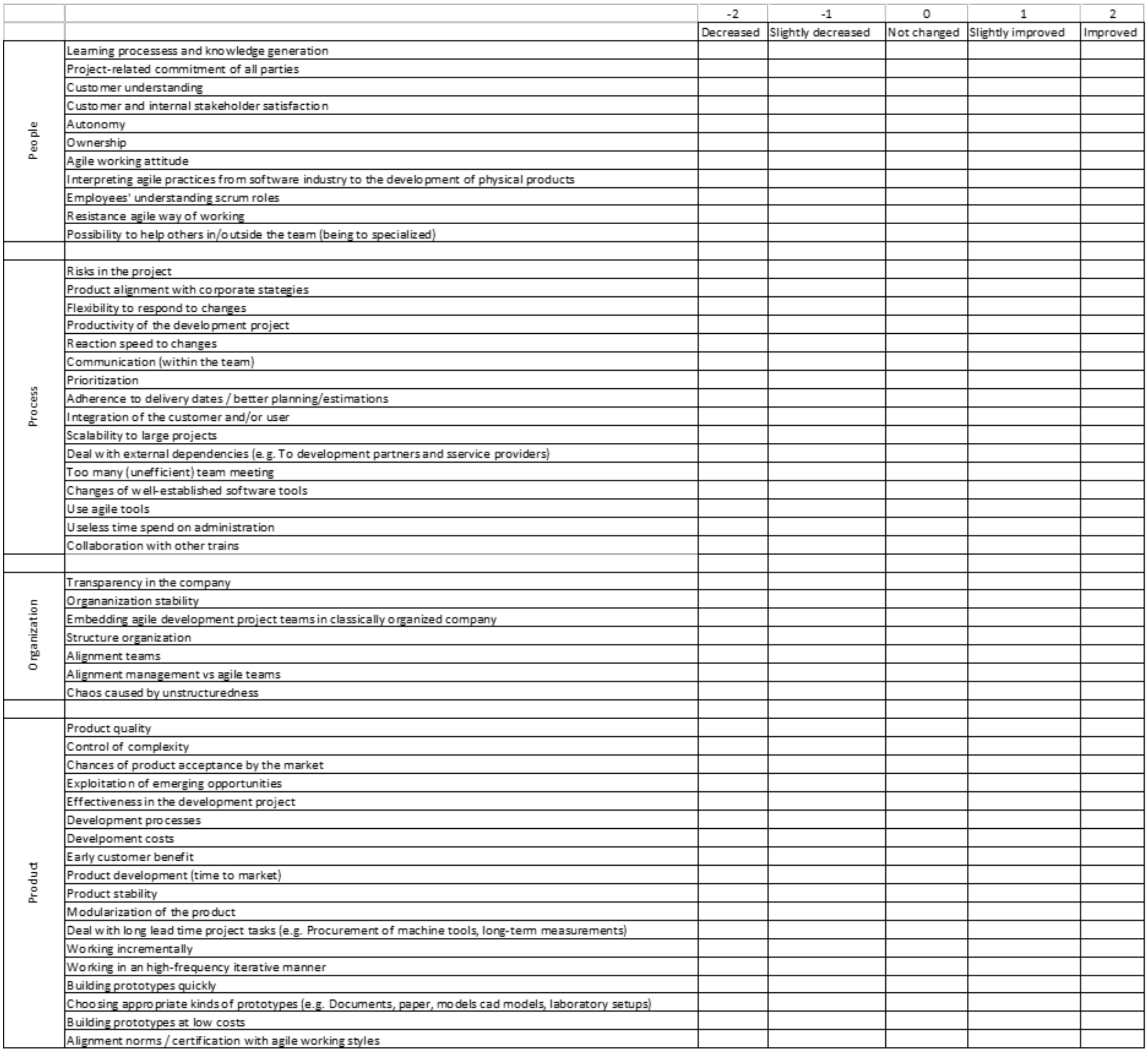
F.12. Interviewee 12

The interviewee mentions that upon returning to the company, Agile was already being implemented. However, they note that the team did not fully embrace the transition and struggled to adapt to the new processes and way of working. The interviewee mentions the challenges of integrating with the larger development teams and the difficulty in finding the right approach for organizing and coordinating work. They also mention the importance of product owners in their work and the limited interaction they have with other teams. The interviewee acknowledges that there was resistance from some people due to the newness of Agile and the lack of clear communication about its benefits in the past. They discuss the differences in mindset between electronic and mechanical teams when it comes to Agile, highlighting the waterfall way of thinking in the mechanical teams. The interviewee believes that the implementation could have been smoother and emphasizes the need for hands-on experience and learning by doing when adopting Agile practices. They suggest that it is important to involve everyone in the process and gradually build understanding and adoption.

The interviewee mentions that the productivity remains the same, but there are advantages in terms of risk management. They emphasize the importance of addressing risks and mitigating them consciously. The interviewee believes that the organization has improved in terms of organizing and communicating risks with clients, resulting in a better product delivered on time. However, they do not think that the time to market has necessarily improved. They discuss the implementation of agile as a good move from a risk management perspective but acknowledge that its success depends on how quickly they can overcome certain challenges. The interviewee also mentions that people on the team are increasingly viewing the use of agile as a positive development, and meetings like retrospectives are valued by the team. They express confidence that agile brings positive changes from an organizational perspective as well.

G Guideline validation adapted framework

The outcome of the adapted framework should be further validated to prove the relation between change in the model and the perceived outcome. This could be done by executing multiple interviews with agile experts and/or with hardware engineers and team leads that are using the adapted framework. These interviewees should rate the new way of working with the adaptions suggested by the framework on several criteria. In Figure G.1, a list of criteria is shown that could be used. This list is a compulsion of the benefits and challenges identified in both the literature and during the interviewees. One could use the results of this interviewee to determine whether the benefits have been improved or the challenges decreased. Also one could verify whether there are actually benefits or that the adapted framework does not results in any benefits.



		-2	-1	0	1	2
		Decreased	Slightly decreased	Not changed	Slightly improved	Improved
People	Learning processess and knowledge generation					
	Project-related commitment of all parties					
	Customer understanding					
	Customer and internal stakeholder satisfaction					
	Autonomy					
	Ownership					
	Agile working attitude					
	Interpreting agile practices from software industry to the development of physical products					
	Employees' understanding scrum roles					
	Resistance agile way of working					
	Possibility to help others in/outside the team (being to specialized)					
Process	Risks in the project					
	Product alignment with corporate stategies					
	Flexibility to respond to changes					
	Productivity of the development project					
	Reaction speed to changes					
	Communication (within the team)					
	Prioritization					
	Adherence to delivery dates / better planning/estimations					
	Integration of the customer and/or user					
	Scalability to large projects					
	Deal with external dependencies (e.g. To development partners and sservice providers)					
	Too many (unefficient) team meeting					
	Changes of well-established software tools					
	Use agile tools					
	Useless time spend on administration					
	Collaboration with other trains					
Organization	Transparency in the company					
	Organanization stability					
	Embedding agile development project teams in classically organized company					
	Structure organization					
	Alignment teams					
	Alignment management vs agile teams					
	Chaos caused by unstructuredness					
Product	Product quality					
	Control of complexity					
	Chances of product acceptance by the market					
	Exploitation of emerging opportunities					
	Effectiveness in the development project					
	Development processes					
	Develpoment costs					
	Early customer benefit					
	Product development (time to market)					
	Product stability					
	Modularization of the product					
	Deal with long lead time project tasks (e.g. Procurement of machine tools, long-term measurements)					
	Working incrementally					
	Working in an high-frequency iterative manner					
	Building prototypes quickly					
	Choosing appropriate kinds of prototypes (e.g. Documents, paper, models cad models, laboratory setups)					
	Building prototypes at low costs					
	Alignment norms / certification with agile working styles					

Figure G.1: Validation adapted framework