

Contents

| | |
|---|-----------|
| 1. Introduction..... | 5 |
| 1.1 Background | 5 |
| 1.2 Problem | 5 |
| 1.3 Purpose | 6 |
| 1.4 Delimitations..... | 6 |
| 1.5 Definitions | 6 |
| 2. Theoretical framework..... | 8 |
| 2.1 Agile methodologies | 9 |
| 2.1.1 Scrum..... | 9 |
| 2.1.2 XP | 11 |
| 2.1.3 Kanban..... | 11 |
| 2.2 Transition challenges | 11 |
| 2.2.1 Communication challenges | 11 |
| 2.2.2 Project Manager role | 13 |
| 2.2.3 Change in mindset | 14 |
| 2.2.4 Organizational agility..... | 15 |
| 2.2.5 Decision making | 16 |
| 2.2.6 Documentation | 17 |
| 2.2.7 Tools..... | 18 |
| 3. Method | 21 |
| 3.1 Case study..... | 21 |
| 3.2 Data Collection..... | 21 |
| 3.3 Data analysis..... | 23 |
| 3.3.1 Quality evaluation of the research design | 25 |
| 3.5 Research ethics | 27 |
| 4. Results | 28 |
| 4.1 Communication- Encourage communication | 28 |
| 4.1.1 Demand social skills..... | 29 |
| 4.2 PM role- Facilitate change | 29 |
| 4.2.1 Involve planning | 30 |
| 4.2.2 Grow transparency..... | 30 |
| 4.3 Change in mindset- Provide training and support | 30 |
| 4.3.1 Clear requirements..... | 31 |
| 4.3.2 Eliminate trust issues..... | 31 |
| 4.3.3 Mitigate chaos | 32 |
| 4.4 Organizational agility- Change processes first | 32 |
| 4.4.1 Transform upside- down..... | 33 |
| 4.4.2 Persuade middle management | 33 |
| 4.5 Decision making- Encourage collaboration..... | 33 |
| 4.5.1 Self- organize..... | 34 |
| 4.5.2 Discuss issues | 34 |
| 4.6 Documentation and maintenance- Turn documentation to communication | 35 |
| 4.6.1 Need of documentation..... | 35 |

| | |
|--|-----------|
| 4.6.2 Innovate documentation | 35 |
| 4.7 Tools and technologies- Self- explore..... | 35 |
| 4.7.1 Self- explore..... | 36 |
| 4.8 Themes Developed in the Content Analysis of the Interviews | 36 |
| 5. Discussion..... | 41 |
| 5.1 Results discussion..... | 41 |
| 5.2 Implications for practice | 43 |
| 5.2.1 Methods discussion | 44 |
| 6. Conclusion..... | 45 |
| 6.1 Limitations | 45 |
| 6.2 Suggestions for future research..... | 45 |
| References | 46 |
| Figure 1: Communication participants in Agile project..... | 53 |
| Figure 2: Concept model of holistic overview of transitional challenge | 54 |
| Figure 3: Concept model of themes and sub- theme dependencies..... | 55 |
| Table 1: Summary of sub- concepts..... | 56 |
| Table 2: Summary of interviewee positions and data collection..... | 57 |
| Table 3: Developer challenges | 58 |
| Table 4: Manager challenges | 58 |
| Table 5: Summary of themes and sub- themes | 59 |
| Table 6: Extent of challenges..... | 60 |
| Table 7: Summary of challenges from literature compared to results | 60 |
| Appendices | 61 |
| Appendix A: Interview guide for managers..... | 61 |
| Appendix B- Interview Questions for managers | 63 |
| Appendix C: Interview guide for development team..... | 64 |
| Appendix D- Interview Questions for development team | 66 |
| Appendix E- Interviews summary | 67 |

Abstract

Agile software development transition has numerous traps that companies fail to jump. Researchers have studied difficulties that might be faced and have contributed suggestions on the evolution of agile development when it becomes a part of a company family. My thesis provides a holistic overview of the biggest challenges that organizations face during an agile methodologies transition.

To have a better overview on the phenomenon at hand, my thesis includes a case study which investigates the challenges during the transition from traditional to agile methodologies. The study traces different experiences about companies transitions to agile methodologies by interviewing agile development team members and project managers from various countries and backgrounds. There are challenges that developers and managers are not aware of and the analysis section sheds light on them to prevent eventual pitfalls during the transition to agile methodologies.

The findings are useful for managers who have a task to deploy a transition to agile methodologies but are unaware of the difficulties. The study will also help companies who work with traditional methodologies, like waterfall methodology, but want to reach agility and revolutionize the workflow from within. Finally, developers will get useful insights on how to handle this change, if they do not have any previous agile development experience.

The research reflects that agile methodologies are sustainable solutions for software development practices and more and more companies are open to the transition despite the potential risks.

Keywords: Agile methodologies transition, challenges in agile development transition, deployment of agile methodologies

Acknowledgements

Writing a thesis is a challenge which many people face in their lives. I saw many sides of me during this process for the first time, and I learned a lot and not only about the topic at hand. I learned that I am persistent, and I can achieve what I am after when I really want it, but also that getting help is not something to fear.

Achieving this amazing opportunity to study IT Management and Innovation in Sweden would be impossible without the help and support of my parents. Together they gave me a chance to achieve everything I want and become the best version of myself.

My heartfelt gratitude to Florian Meinert, who stood beside me in times of hopelessness and in times of happiness and bliss. I am forever grateful for his intangible and mental support.

This amazing journey would be impossible without my teachers - Christina Keller and Andrea Resmini, and my thesis supervisor - Asif Akram. I am thankful that they dedicated their time and knowledge to me and my classmates.

Finally, I am grateful for the support of my family and friends back home.

1. Introduction

1.1 Background

Since software development came up four decades ago, there have been many methodologies that help software development companies to operate it. Some of these companies rely on heavy documentation, strict planning and are thoroughly traditional (Cho, 2010), such as waterfall methodology, Spiral and Rapid Software development. My study will focus on agile methodologies which is in contrast with the traditional ones. Companies choose agile methodologies because it uses as least as possible documentation so that the developers can focus on the development process in order to finish a project faster. Traditional methods have failed to bring bigger value to companies because of heavy documentation, extensive planning and designing up front, but are preferred due to their “straightforward and structured nature” (Cho, 2010). Another reason to choose agile methodologies is that planning is kept to a minimum when the project starts but it happens throughout the whole process. Since companies want to be more efficient in their software development, they are more motivated to implement agile methodologies. To have competitive advantage and positive customer collaboration, the customers want to make the development quicker and faster. This means that companies who use agile methodologies should adopt to changes faster and get accustomed to new plans in a constantly developing and changing environment.

In traditional methodologies like waterfall, the plan and requirements are defined in the beginning of the project (Heeager, 2012). The method requires heavy documentation and no changes can be made during the process, meaning that waterfall methodology cannot cope with the changing environment, affecting the company (Cho, 2010). Because of the rapid market changes, software development needs to change as well and find ways to adapt to the ecosystem (Ngo-Ye & Ahsan, 2005). The agile methodologies give a great opportunity for companies to be competitive as a result of the advantages that they have, compared to traditional methodologies (Dybå and Dingsøyr, 2008). Most companies choose agile development because the customer needs are most likely to be met, the team can face changing requirements and overall the company business objectives align better with IT. I will trace the factors that can be challenging to the company and understand how they affect the development team and their connection to the management. Researching the problems at hand will help future project managers of agile development teams understand the challenges before the transition to agile methodologies and avoid problems by having in mind the existing barriers. This will give value to the public by delivering a stable and working software that will satisfy the actors.

1.2 Problem

Agile development is a topic which grows more every day and requires a vast quantity of research. Even though there is a great deal of articles on the topic (Boehm & Turner, 2005; Cho, 2010; Nerur, Mahapatra, & Mangalaraj, 2005), we can rarely find empirical evidence on the challenges that organizations face with an agile methodologies transition.

My study will provide evidence of the existing gap that companies should be aware of (Taylor, 2015) by reviewing the theory based on the issue and provide practical challenges. The

knowledge gap in the literature is connected to a missing description of agile methodologies transition challenges for both managers and developers coming from a traditional background in the field of software project management. In the study at hand, this phenomenon will be identified as “non-agile development experience.” The researches have acknowledged some obstacles of the transition for organizations, but none of them has reviewed them separately for management and developers, having in mind their different background. Further research is suggested in this field and recent studies asked to look deeper into it (Pikkarainen, Haikara, Salo, Abrahamsson, & Still, 2008). In this line of research, Laanti and Abrahamsson (2011) suggest that a holistic view on the transition challenges is required.

1.3 Purpose

The purpose of my thesis is to provide a holistic overview of the challenges on developers and project managers that agile transition causes, because I want to find out how those challenges vary according to different experience. With my study, I want to help my readers understand what the differences in theory are compared to the practical point of view.

The following research questions are built upon the given purpose of the study:

RQ1: What are the challenges during the transition to agile methodologies for managers and developers?

RQ 1.1: To what extent do the challenges differentiate?

1.4 Delimitations

The study is not a quantitative one because of the required number of respondents and the time limit affecting the gathering of results and analysis.

The thesis does not create a “step- by- step” guide of agile methodologies transition because there is no right way to do it and, after all, this is not a framework that anyone can simply implement. Every company is different and can choose the most suitable way that this transition can fit and benefit everyone.

My study does not aim to offer solutions for problems and issues that are caused as a result of the challenges, and it also does not aim to fix them. It offers concepts, considered important and possible suggestions on how to prevent them in advance and to have them in mind when making an agile methodologies transition. This does not necessarily mean that they apply to all sizes of companies, but it is beneficial to consider them as eventual problems in the future.

Several companies are used to study the phenomenon at hand which means that the study does not focus on only one organization.

The respondents of the interviews are not chosen by any other specific characteristics except experience in agile development. Instead, they represent various cultures, age, experience and geolocation.

1.5 Definitions

My thesis introduces the following definitions:

- *Transition* - A transition to agile methodologies describes the process of changing from one methodology to another, in this case to agile methodology. It involves all development practitioners and is considered as problematic and challenging, because

it also involves changes in all organizational aspects (Gandomani, Zulzalil, Azim, & Ghani, 2014).

- *Non-agile background*- Experience in traditional (waterfall) methodologies (Laanti, Salo, & Abrahamsson, 2011).
- *Agile methodology* - A definition of agile methodology varies through different authors. Kennaley (2010, p.34) sets one of the definitions as follows: *“An iterative and incremental (evolutionary) approach to software development which is performed in a highly collaborative manner by self-organizing teams within an effective governance framework with” just enough” ceremony that produces high quality software in a cost effective and timely manner which meets the changing needs of its stakeholders”*.
- *Agile practices* - Agile practices is a definition that combines all methodologies within agile Software development, for example Scrum, XP (Extreme Programming), Kanban and Lean (Jalali & Wohlin, 2010). In their essence, agile development practices consider changes throughout the workflow and require *“close collaboration between customers and developers and delivering software within time and budget constraints”* (Jalali & Wohlin, 2010, p.1). The practices rely on informal communication compared to a detailed documentation and the processes are iterative and adaptive.

2. Theoretical framework

This chapter aims to synthesize past knowledge on the topic, starting with a history and background of agile methodologies. Later, I perform a critical analysis and a systematic review of the articles and spot the gaps in the literature followed by proposing directions for future research.

My study uses a systematic search approach based on Webster and Watson “*Analysing the past to prepare for the future: Writing a literature review* “. The authors claim that “*A systematic search should ensure that you accumulate relatively complete census of relevant literature.*” (Webster & Watson, 2002, p.4; Levy & Ellis, 2006, p.5). The aim is to retrieve the most relevant theory from articles and book chapters. The keyword search has been performed through several databases: Scopus, JU library, ProQuest, IEEE, Research Gate, Elsevier and Google Scholar.

Literature review

The contents of this section focus on the theoretical framework of the thesis. The current overview and situation of the challenges in the agile development transition are studied based on literature review. All aspects of those concepts are researched which helps companies to have in mind the various challenges that can occur in a transition to agile methodologies and practices.

Concerning the quality of all the articles and book chapters in my study, only peer-reviewed papers have been used as Levy and Ellis (2006) suggest. Magazines, newspapers and web-pages have not been used due to lack of theoretical background. All articles are chosen by reading the abstract, introduction and conclusion or by following relevant references in the literature itself. Many articles are chosen exactly because they are referenced in other studies who are important and useful. In total 98 articles are downloaded to create a relevant and qualitative literature review. According to Webster and Watson (2002), a necessary part of a study is the prior review of the literature in the field of study. This gives our work a better understanding of the problem at hand and reveals current and future research directions. The research in my thesis is performed with the help of various databases and searching with keywords. This research is possible with the help from several databases like JU library, Scopus, ProQuest, Google Scholar and a wide variety of key words such as: *transition challenges, agile deployment, barriers in agile development, agile transition challenges*. After collecting extensive range of research, the keywords are narrowed down to: *Scrum , XP, PM role in agile methodologies, agile development history, deployment of agile methods, agility*.

Description, history, predecessors

Agile software development has raised a great deal of discussions within the software development community. Some companies prefer agile methodologies, others fancy traditional approaches and a third category try to mix them. To better understand why and when the right time to transit to agile methodologies in an organization is, the management should be aware of the agile development history and what the methodologies are all about.

A predecessor of agile development is the Iterative Incremental Development (IID) whose history can be traced back to the early seventies. This process is followed by the traditional approach, where each next stage can be executed only if the previous one is completed. Traditional approaches rely on documentation and are characterized as “heavyweight”. When a project is compassed by traditional methodologies, it is necessary to plan and document the whole set of requirements and plan every step of the project. As it turns out

in the mid 1990s, managers and developers found this step challenging and unnecessary (Highsmith, 2002). As both sides were delivering projects late and customers were not satisfied, the community developed the agile methodologies in order to embrace change instead of denying it. The official beginning of agile methodologies starts in 2001 on a conference in Utah attended by 17 process experts, where the phrase “*agile methodologies*” comes from (Larman & Basili, 2003). “*Agile with a capital “A” refers to a project management style*” (Taylor, 2015). The agile methodologies, also characterized as “*lightweight*”, consist of short iterative cycles, they rely heavily on customer collaboration and teamwork, there is constant feedback from the client and early product delivery is highly valued (Koskela, 2003). The success of the methodologies is so high, that term “*agile*” is now a synonym of “*flexible manufacturing practices*” (Cockburn & Williams, 2003) guaranteeing client satisfaction and quality. The founders wrote the Agile Manifesto (<http://agilemanifesto.org/>), where the four core values can be found:

- individuals and interactions over processes and tools
- working software over comprehensive documentation
- customer collaboration over contract negotiation
- responding to change over following a plan.

There are several methodologies that give managers the opportunity to implement agile development in the companies successfully. In the next section, we will explore deeply the most known and used methodology - Scrum, followed by a short description of Extreme Programming (XP), Kanban and Lean.

Agile methodologies

Nowadays, a great deal of organizations state that they have the “agile thinking” or show enthusiasm in transitioning to agile development in their companies. According to a study from Forrester, the IT industry admits that there are numerous benefits of the methodologies and report positive aftermath (Schwaber, 2007). When organisations adopt new agile development practices, managers have to face a large number of challenges when they have to shift from traditional methodologies (Boehm & Turner, 2005). However, it is claimed that it is not uncommon, that some companies are not aware of how broad the transition is and how big changes that must be performed inside the organization. As it turns out, this can be a challenging task (Svensson, 2005). The agile methodologies consist of short iterative cycles whose aim is to prioritize and optimise the actor requirements by counting on the developer team skills and knowledge more, than focusing on documentation. In their core, agile development practices undergo a given number of iterative cycles where the team tests the software several times before delivering a potentially shippable product. The team sets the way of work, the principles and embraces changes instead of performing strict planning. An important part of the project is to grasp the changes by perceiving them as an integral part of the work and how interrelated they are to the constantly changing environment instead of avoiding them and being afraid to accept them. Change in a project should be a motivator to create better software, deliver a stable product and react to fluctuations in the ecosystem for the sake of bringing a greater value to the customer.

2.1.1 Scrum

In Project Management terms, Scrum is identified and differentiated from traditional heavyweight methodologies as “lightweight” and an agile process whose aim is to facilitate software product development in a constantly changing software ecosystem (Cervone, 2011).

A distinguishing part of Scrum is its iterative development whose aim is to control all chaotic aspects that emerge in a team, fix errors, improve communication and coordination. A final goal of Scrum is delivering a stable product faster with improved quality compared to traditional methodologies.

The methodology consists of ceremonies, artefacts and roles (Schwaber, 2004). The roles are the Product owner, Scrum Master and the team itself. The ceremonies include Daily Scrum Meeting, the Daily Scrum of Scrums Meeting, the Sprint Review Meeting and the Sprint Planning Meeting (Cho, 2010). Finally, the artefacts include the Product backlog, Sprint Backlog and Burndown chart. The process starts by reviewing the ROI (Return on investment) and figuring out the milestones of the project, having in mind that changes will come throughout the project. The items with priority are separated as isolated tasks during the Sprint planning meeting which go to the backlog in the end of the meeting. All items in the backlog are done one by one when the sprint starts, and this is repeated through all iterations.

A great advantage for companies when managers use Scrum is its simplicity. They implement necessary factors for success such as communication, iteration, efficiency and great productivity. By decreasing as much as possible the unnecessary bureaucracy and achieving more practicability in terms of management of the project, it is possible for all team members to do a meaningful and productive work (Cervone, 2011).

Even though the methodology is widely used and preferred in the software development community, Scrum has challenges that every manager must be aware of before performing the transition. For all team members, it is better to have a description of what is being done for new-coming developers and share equal knowledge for every member, instead of only one. It is known that the methodology relies on little, and possibly none, documentation. Some developers even consider the code itself as a document, which leads to more comments inside the program used for development (Cho, 2010, pp. 191-192).

Communication is well supported by introducing the Daily Scrum Meetings as it is well known that supporting it is essential for success (Parnas, 2006). Nonetheless, if a company consists of several teams, communication can be a challenging factor for the team and the management. The consequences from lack of communication can be numerous, including duplicated code and unfulfilled requirements, as a result of the lacking feedback from the clients. This also implies that customers should be more involved in the decision-making process from the beginning until the deployment of the product. It is essential that the clients must be aware of what they want and have a vision of the final product so that the developers could work more effectively and efficiently. It is the manager role to involve the clients as much as the team needs to deliver a stable product, otherwise the team loses time because of the lack of communication and information.

Above all, the distinguishing Scrum ceremonies help the team to avoid most of those challenges and be up to date with the development of the product. The daily stand-up meetings might be unnecessary for some developers, a waste of time or too long for others, but it is the manager task to keep the meetings long enough to produce value to the result (Cho, 2010).

2.1.2 Extreme Programming (XP)

Beck (2000) describes XP as a perfect example of embracing change in an organization while transitioning to agile practices is EXtreme Programming (XP). This methodology is mostly about forgetting old habits which are easily adapted in traditional methodologies. XP is about being able to realize the capabilities that developers have and putting them in use the best way possible. This methodology focuses on excellent understanding and application of programming skills which require perfect communication within the team including constant feedback. This methodology is different than others by its short cycles and continuous feedback, an overall plan which changes during the project, adopting new concepts according to the resulting business needs and constant tests which nurture progress in the development (Beck, 2000).

2.1.3 Kanban

Kanban is a Japanese approach, which dates to 1950s when it was used in the car industry. In its essence, Kanban is system made for scheduling within the manufacturing (Ahmad, Markkula, & Oivo, 2013). In software development, Kanban was first used in 2004 at Microsoft. The purpose is to give a better visualisation of the workflow and minimize the Work in Progress (WIP) (Kniberg, 2009). This methodology allows customers to review the released software, while the developers focus on the work at hand with the help of “shorter feedback loops”. Kanban has the following principles:

- Visualize the workflow
- Limit Work in Progress
- Make process policies explicit
- Improve collaboratively (using models and the scientific method)

The feedback and results from companies that use Kanban are positive and supportive of the methodology (Ahmad et al., 2013). As a part of the agile development family, this methodology is based on iteration and adapting to changing requirements through its characteristic possibility to visualize all processes and nurturing collaboration and communication (Kniberg, 2009). By limiting Work in Progress, Kanban supports a stable workflow and increases the work performance within the team (Anderson, 2010).

1.2 Transition challenges

In this section I give an overview of the most commonly met transition challenges in the literature that organizations face when they decide to transit to agile methodologies. I will not provide solutions to solve the existing barriers but will highlight them to make companies aware of them in their future agile development transition.

2.2.1 Communication challenges

This section focuses only on the communication barriers between managers and software development teams, excluding communication between managers and customers or developers and customers.

As we discovered in the previous section, agile methodologies like Scrum and XP are preferred by companies, because they adapt fast to the rapidly changing business environment. Nevertheless, there is little research and a knowledge gap on how they affect

communication (Pikkarainen et al., 2008). In most organizations, agile methodologies are preferred because of the fast development of the product and a higher quality when it is been delivered (Holmström et al. 2006). The studied methodologies are also being chosen by managers because they improve collaboration and communication in fast- developing situations where change is critically important for competitive advantage (Anderson, 2003). Malone and Crowston (1994, p. 62) propose a definition of communication which describes the term as “*managing relationships between producers and consumers*” which in our case is between the management and development team. When we talk about change in development projects, it is essential that we consider communication as a necessary aspect for succeeding (Stelzer and Mellis, 1998).

Although agile methodologies imply that communication is an essential part in the project work, Turner (2003) argues that the actors might give the informal communication a bigger meaning than they should. The author reminds us that after all there are also formal ways of communication, including source codes, test cases and a modest chunk of documentation which are inevitable for every project. Cohn and Ford (2003) suggest that the lack of communication between the project actors, leads to project failure due to the growing void inside companies that transitioned to agile development methodologies. The methods Scrum and XP suggest practices to overcome the communication barriers in the organization through interaction and communication between the stakeholders (Pikkarainen et al., 2008). In their case study, which includes highly skilled software developers and company managers, Coyle, Conboy and Wang (Conboy, Coyle, & Wang, 2010) identify a number of challenges that come between the actors of an agile development project. A key problem in the study that affects the communication between management and development team, is the reliance on social skills. The interviewed developers are undoubtedly talented and skilled in their work but in terms of presentation and communication skills, their performance is uncertain and shaky. All managers support face-to-face communication and collaboration between the actors, but it comes forth, that those high expectations can dwindle most developers productivity.

The literature review acknowledges that the communication channels within an agile development project can go within the development team, project manager and clients. To understand the communication challenges in depth, there are three communication channels that expose the obstacles related to communication. They are separated as following:

- Type A: Development team and clients
- Type B: Development team and Project managers
- Type C: Project managers and clients

My study focuses only on communication type B to give a better understanding of this major challenge. Type A and C do not cover the focus of my study. Thus, we must acknowledge that more information and literature is required in this direction.

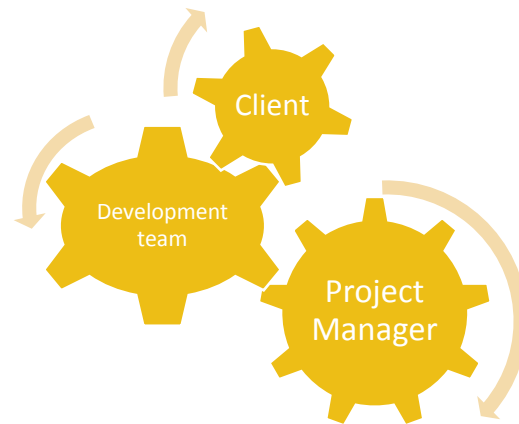


Figure 1: Communication participants in agile development project

As shown in Figure 1, the participants in an agile development project are interrelated and they depend on each other, meaning that all three should be present in a project. My study examines that there is a connection between development team, development team and project managers and project managers and clients. Nevertheless, I will only focus on the relationship (B) between Development team and Project managers.

Type A: Development team and clients

To have a better understanding, I will delve into a short description of the Type A relationship, because I consider it also important for my study.

Agile development works with short iterative life cycles, the development team relies on customer collaboration and learns how to adapt in rapid changes. Communication within development team and clients is a great challenge for software development (Damian et al. 2000). After the end of an iteration the development team delivers a version of the product to the client so that they can test it (Khalil & Khalil, 2016). This means that the following feedback is used in the next iteration and so on until the product is ready to be delivered.

Type B: Development team and Project managers

Communication within software development team is a crucial point for success, especially the one between the management and development team (Pikkarainen et al., 2008). Lack of trust between team members and project managers can be prevented by regular communication which results in a more efficient software development (Paasivaara & Lassenius 2003).

Being aware of these challenges, managers can determine which communication problems they face in the agile methodologies transition and try to prevent them in time.

2.2.2 Project Manager role

The literature on the Project Manager role in an agile development transition is scarce, thus we can only acknowledge the existence of the problem. I would propose a future research direction that focuses on the challenge of finding the manager role in an agile methodologies transition.

As we understood so far, historically, first there are the traditional software development practices (Fitzgerald, 1996) whose values are completely in contrast with those of agile methodologies and IID. Consequently, it is thought that Project Managers (PMs) focus less on planning and support and motivate the team. A great share of the literature on agile development focuses on the comparison between traditional methodologies with iterative

incremental ones, mostly supporting the latter ones (Geschi et al., 2005). What has been spotted as a gap in the literature is whether the PM role is the same in both cases.

The role of the PM is to facilitate and aid a project and not dictate, but as the newer methodologies have developed more and support self-organization, the PM role starts to diminish (Cobb, 2011). As the managers work towards empowering the team, there can be several challenges. The function of a PM in traditional methodologies seems to be undermined due to the shifting requirements in agile methodologies which include a more expert knowledge inside the team (Taylor, 2015).

The shifting role of the manager during a transition, can be very discouraging and can lead to a few challenges. Traditional managers are prone to giving orders, so when they meet agility, where planning is not a necessity, they are discouraged to manage the project. This is also valid when the company goes through a transition and middle managers are scared that they might lose their positions. I identified the following sub-concepts to help understand this challenge.

Type A- Balancing control and agility

Competitive advantage can be hurdled by heavyweight and bureaucratic product development. That is a reason why organizational control is needed but overdoing it might hamper the adaptation to business goal thus, failing the agility (Cobb, 2011). In today's business, companies should be able to adapt to changes fast, and being flexible is needed as much as being able to control business processes. According to Cobb (2011), a project is not successful if a good market window is missed due to a not aggressive enough schedule, but the focus is meeting the costs. Failing to be flexible and adapting to new business changes, but meeting the costs and schedule also leads to a possibility to diminish the projects and lose managerial agility.

Type B: Leadership and collaboration

In agile development, the management focuses on leadership and collaboration within projects (Gandomani, Zulzalil, Ghani, Sultan, & Nafchi, 2013). This is in contrast with traditional methodologies where the management plays a more controlling role and does not focus on team collaboration as much as in agile methodologies. This is a reason why managers should be aware that teams that work within heavyweight methodologies have issues coping with the new management roles and techniques.

2.2.3 Change in mindset

A relatively often met barrier in agile development transition is the change of the company mindset. In its essence, agile development is a mindset that the team must understand and nurture by being aware of the values that support it. The agile transition can be successful, only if the mindset is also implemented in the cultural organization of the company. A big challenge for managers is indeed changing the mindset of the teams when preparing for a transition.

In order for the mindset to be cultivated into the company, there are some changes that need to be implemented (Pikkarainen et al., 2008). The team should understand that failure is not scary and should be understood as an opportunity to learn and get more knowledge. A very important part in the agile methodology mindset is embracing the changes that happen in the company and adapting to them. This makes the workflow lighter and the product more stable and efficient (Nerur et al., 2005). An important challenge when managers want to change the team mindset is developing a wider use of knowledge sharing. This needs to happen to help the team realize different problems during the transition and help each other

deal with them. It can be problematic to change the mindset of the developers if their experience is connected mainly to a traditional and documentation-driven way of working and they have to face the agile methodology mindset (Heeager, 2012). This change comes with requirements of different skillset which is difficult for the managers (Mahanti, 2004).

Changing the mindset is a necessary factor that needs to be reviewed from two points of view - those of the organization and the team. The following sub-concepts that I identified are related to my study for the sake of giving a better understanding of this major challenge.

Type A: Organizational

The aim of Management in IT is to improve efficiency, flexibility and organizational adaptability, which is a bigger sign of success than operational performance (Ngo-Ye & Ahsan, 2005). Efficiency takes a large part in understanding how agile development works and changing an organizational mindset can be challenging. It relates to the fact that the delivered software should be optimized and efficient and building an “agile enterprise IT application system” (Ngo-Ye & Ahsan, 2005). Sambamurthy et al. (2003) compares the degree of organizational agility to the possibility of changing the mindset of the company, since they are interrelated.

Type B: Team

Changing to agile development from traditional development is often a big challenge for developers (Begel & Nagappan, 2007). According to the case study of Begel and Nagappan (2007) the developers often deny working, unless the methodology is understood by everyone in the team. This is especially hard in situations when the project includes a big number of teams where some teams use agile development and others waterfall methodology and scheduling becomes a large obstacle.

2.2.4 Organizational agility

The transition to agile methodologies in an organization who is not familiar with it or is used to “heavyweight” methodologies, can be challenging not only to the development team. It affects also other departments, other development teams and mostly, it affects the management (Cohn & Ford, 2003). As Unisys (2004) predicted, agility in a company will be more necessary than efficiency. In the IT field, agility is necessary to streamline processes and build “inter-organizational relationships” (Agarwal & Sambamurthy 2001). Inconsistent organizational relationship to the newly implemented agile methodologies, might lead to challenges and obstacles like technical debt, conflicting and contrary communication and isolated work (Nord & Brayer, 2013).

When developers make a transition from heavyweight methodologies, there might be challenges concerning their view of validity and trustworthiness of the new process. According to Cohn and Ford (2003) it is essential to make the given transition gradually which will ease the work of everyone involved with the project. When the team feels comfortable with the recent changes, the workflow gets faster, which in some cases means that slow developers are left behind, compared to the faster ones. This is the reason why agile methodologies follow the top talent principle from Barry Boehm (1981) “*use better and fewer people*” meaning that these processes require fast thinking which is the basis of agile methodologies thinking. Agile practices promise impressive results and delivering stable products, but the transition itself can slow down the developers while they are introduced to new techniques and this might take time, which can also be challenging to the management.

Organizational agility is one of the major challenges that managers must face. This field requires an in- depth study since the identified sub- concepts require a deep understanding of what agility is and how a company can be agile. The main challenges of making an organization agile must deal with market capitalizing and operational adjustment. Once those challenges are overpassed, agility is accepted together with developing with agile methodologies. Making a change this big, effects the whole organization and touches all processes. This is a reason why this obstacle is considered as one of the most challenging. The sub concepts connected to this challenge are as follows:

Type A: Market capitalizing

The agility of market capitalizing relates to the ability of the organization to adapt to rapid market changes by improving the product according to customer feedback and monitoring the further development and maintenance of the product (Lu & Ramamurthy, 2011). Its main aim is to improve the product itself and the company service which accompanies it by embracing constant change and growth, followed by critical decision making and customer feedback (Sambamurthy et al. 2003; Volberda 1996; 1997).

Type B: Operational adjustment

This kind of agility relates to the physical ability to adjust to market and demand changes from the internal business processes in the organization (Dove, 2001; Sambamurthy et al. 2003). An important foundation of this form of agility is the flexibility and swift way of responding to operations in order to open the way of innovative push during the changes that happen within the company (Lu & Ramamurthy, 2011).

2.2.5 Decision making

The managers of agile development projects have to face “critical decisions” that lead to either success or failure which shape the team decision making further on (Drury, Conboy, & Power, 2011). Delivering a product after several iterations and a value bringing software are the results of frequent “short-term” decisions (Fitzgerald, 2006). A challenge that managers must face is that their decision-making role gets reduced (Lindstrom & Jeffries, 2004) while the roles inside the team shift, resulting to developers taking decisions out of their responsibility guiding them to the fact that they must take even more decisions (Beck, 2000). Addressing these issues require defined roles for the managers and the team so that the most optimal decisions within a team lead to delivering a stable product (Drury et al., 2011).

According to the scarce amount of literature provided, whenever a decision should be made or not, developers rely on their experience and then think about the options concerned the decision (Zannier & Mauer, 2007). Research has concluded that in general, it is better for teams to make effective group decisions because their knowledge and information in a project should be interrelated (Russo and Schoemaker, 1989; Schmidt et al., 2001; Wheeler and Valacich, 1996).

For this challenge, I identify two sub-concepts that are related to my study to give a better understanding and try to improve future important decisions within an agile development project.

Type A: Strategic

These kinds of decisions are related to the long- term prosperity of the company by including the development team (Chandler, 1997).

Type B: Tactical

They relate to daily activities that are responsible to maintain efficient and continuous activities and operations in order to have a functional and stable software (Drury et al., 2011).

2.2.6 Documentation

Numerous organizations are trying to implement agile methodologies, but a great challenge is the documentation and maintenance of the project work (Knippers, 2011). Research is pointing to the importance of this issue, since 80% of the money in a project are spent on maintenance (Jones, 2000; Jones & Bonsignour, 2011). If defects are found at a late point of the project when the product is almost done, it costs ten times more to fix it, especially in heavyweight methodologies like waterfall (Jones, 2000). The Manifesto suggests “working software” comes before “comprehensive documentation” (Manifesto for Agile Software Development, 2001) meaning that it is one of the most important characteristics of an agile development project. Compared to waterfall methodology, the pre-coding phases, like designing and planning, do not exist in agile development, because the developers strive for a better working software (Knippers, 2011). Added to this, there is as little documentation as possible and this combination drove Yahoo to fail the transition of Scrum thanks to poor, incompetent maintenance and absent documentation (Larman & Vodde, 2010). There is a lack of research in this direction because maintenance is often overlooked and cumbersome for the developers who prefer to work with new and exciting technology. But research is positive that documentation is the first thing to be left out when the projects start and use the allocated resources and time.

It is important not to trust agile methodologies blindly just because everyone is using it. Therefore, we must be critical and know when the agile methodologies are appropriate to use and give managers a better understanding of the challenges that the lack of documentation and inadequate maintenance might bring to an organization. When compared, waterfall methodology is mainly being preferred due to the fact that the team delivers a stable product, but cannot respond to change (Heeager, 2012) where agile practices are highly flexible but cannot work with complex products (Dahlbom & Mathiassen, 1993). Pikkarainen & Mantyniemi (2006) suggest that a combination between the both methodologies might be useful and produce the best outcome and use the strengths to achieve the best result (Galal - Edeen & Seyam, 2007).

In order to be aware of future drawbacks, two sub- concepts are identified that are related to my study, aiming to give a better understanding of this major challenge.

Type A: Lack of documentation

Research shows (Larman & Vodde, 2010) that lack of documentation can lead to failing projects. Mostly, the waterfall model is closely connected to a documentation-driven way of work (Heeager, 2012). This is also a way for managers to see if they would choose this methodology, because it is known that heavyweight methodologies use an excessive amount of documentation and bureaucracy, but the total lack of those can be diminishing to the projects.

Type B: Incompetent maintenance

Whenever a software product is developed, it needs maintenance after the final delivery, because it shows the clients that they can trust the organization and work with them again. This means that effective maintenance is an essential part of the successful work of developers and the competitive advantage of the company. Furthermore, this means that an

incompetent maintenance will lead to large costs for the company to prevent making mistakes (Jones, 2000; Boehm, 1987)

2.2.7 Tools

By working with software developing teams, managers stumble upon another challenge concerning the usage of tools. Changing to feature- based and iterative development from a life cycle model of work is a great challenge that managers must face. This change requires adjustments to roles of people, communication, tools and technologies (Nerur et al., 2005). The constantly shifting varieties of tools can cause problems such as required new programming languages and skillsets that the developers might not have and need time to learn. According to previous research (Sircar & Nerur, 2001) the changes in the software development processes are interrelated to changes in the organization itself and just changing the old tools and technologies with new ones will not make it happen faster and better. The tools used to help developers vary and they change often. Using dubious and frequently changing requirements combined with emergent tools and technologies make the agile methodologies transition uncertain and unpredictable (Sutherland, Viktorov, Blount, & Puntikov, 2007). In their study, Sutherland and Viktorov (2007) suggest that the best practice in implementing agile methodologies like Scrum is to use as few tools as possible, and once the right tool is found, all tasks and sprints can be maintained easily. The choice of the right tools helps managers improve time to market and the transparency of the newly implemented agile methodologies in the organization (Leffingwell & Muirhead, 2004).

Choosing the right tools is a critically crucial point that managers must have in mind when they implement agile methodologies, because they play a main role in the successful transition. Therefore, companies who are willing to take this step should invest in tools that support the iterative development and all the techniques that agile requires. But in order to achieve this success the tools need to be operated by the right trained people in order to use them correctly (Nerur et al., 2005).

One sub- concept is identified that is related to this challenge, to give a better understanding of this major challenge.

Type A: Organizational changes

Changes in software development require also strong organizational changes. These organizational level changes require tools that the development team is familiar with and changing the current tools is not the best practice (Nerur et al., 2005). These changes affect the structure and management of the company and choice of tools to help these changes to be successful is crucial.

In table 1 follows a summary of the main challenges represented as concepts and the corresponding sub-concepts.

| Concept | Sub- concept | |
|-------------------------------|-----------------------------|------------------------------|
| <u>Communication</u> | Clients- dev. Team | Dev. Team- PM |
| <u>PM role</u> | Balance control and agility | Leadership and collaboration |
| <u>Organizational agility</u> | Market capitalizing | Operational adjustment |
| <u>Decision making</u> | Strategic | Tactical |
| <u>Change in mindset</u> | Organizational | Team |
| <u>Documentation</u> | Lack of documentation | Incompetent maintenance |
| <u>Tools</u> | Organizational changes | X |

Table 1. Summary of concepts and sub- concepts

The table shows each concept with a corresponding sub-concept. The sub-concepts show the challenges that have been identified in the literature that correspond to each major transition challenge.

In my study, the communication challenges can be seen from the perspectives of development team, project manager and clients. The study focuses mainly on the communication challenges within Type B, meaning that the data collection will focus on the communication challenges only between development team and the project manager.

When there are challenges concerning the project manager role, we have to think how the roles of a project manager shift when they become a part of the agile development team. This situation is considered as diminishing for the managers, since their responsibilities become more restricted. With a Type A challenge, the PMs have a substantial challenge to balance control and agility, where too much control and scheduling might decrease the company agility and possibility to adopt to changes. Within type B, a challenge with leadership and collaboration comes with the circumstance that an agile development team needs a facilitator and a leader, where in traditional methodologies the management focuses on control.

Coming to a very important factor - organizational agility, a great challenge from Type A is capitalizing the market where the company must deal with market changes to be agile, meaning that an organization should be agile and adaptable to market fluctuations. Type B relates to the physical challenges that the company must undergo.

Decision making can have strategic (Type A) and tactical (Type B) challenges concerning the management and the development team. They relate to the question which decision making type is more suitable to a specific conflicting situation to deliver a stable software and reach the project goals successfully.

Change in mindset comes on two levels as well: organizational (Type A) and team (Type B). Change in mindset on organizational level can be a difficult and challenging task which requires a lot of time and effort. Not all departments might agree in adapting the agile methodologies, meaning that companies are not agile when they must compete on the market. The team challenges are connected to the fact that not all developers have worked with the methodologies and this can be a drawback in a project.

When it comes to documentation, the challenges are connected to lack of documentation (Type A) and incompetent maintenance (Type B). Type A is questioning if no documentation is helpful for a project and Type B shows how bad maintenance of the product might ruin the deliverable. Organizational changes (Type A) are a great challenge for choosing the right tools in an agile development team. Working with not the right software might slow down the project or deliver an unstable product.

The reviewed concepts and the corresponding sub- concepts have been grouped together for a better overview of the holistic picture (See [Figure 2](#)).

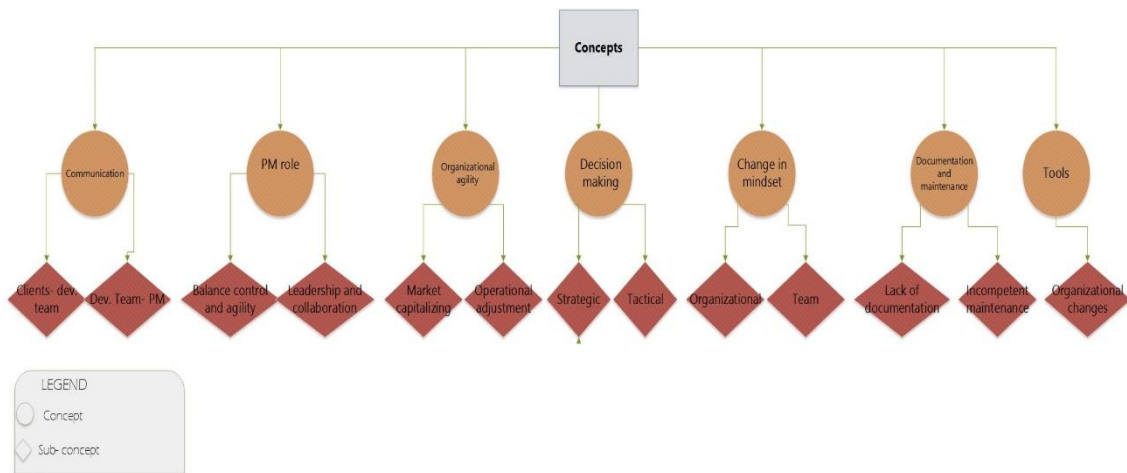


Figure 2: Concept model of holistic overview of transitional challenges

3. Method

3.1 Case study

The purpose of my study is to gain insights on the manager and developer challenges of agile methodology transition. The purpose of my thesis is leading to an exploratory, single embedded case study (Yin, 2003). This means that the research questions can be the type of “What?” questions that suggest propositions for further inquiry, and as Yin suggests, a case study can be of an exploratory nature as any other research strategy. Case studies also favour using the “How?” and “Why?” questions.

According to Saunders (2012), deduction deals with reviewing a theory and then reviewing the data, where induction will deal first with data and then with theory. In the given study, the transition challenges are studied first, which leads us to a better understanding of the barriers for organizations.

The focus of the case study is to investigate the phenomenon of how different developers and project managers/Scrum masters, deal with the transition to agile methodologies, namely Scrum. The respondents include people with and without previous experience with traditional methodologies as well, namely waterfall methodology. I ask them questions related to the identified obstacles from literature and what else is hard for them. The respondents reveal what is challenging for them when they are introduced to agile methodologies for the first time. They are asked to explain how the problems are dealt and what more can be prevented in the future. This case study allows us to trace all these processes from a practical point of view and it gives us a better understanding of what can be challenging for both managers and developers.

3.2 Data Collection

By following the guides of Schultze (2011), while doing the interviews I ask follow-up questions and contacts the interviewees again to make sure the interpretation is right. The purpose is to obtain a relevant information that affects the team and manager challenges in the agile methodologies transition within the companies. The focus is to obtain information about the change in people mindset, communication obstacles between the management and the team and the changing role of the manager. I will interview Scrum masters, agile methodologies coaches and developers with various background and different cultures.

Step one - The data collection process starts by researching the available sources of evidence provided by Yin (2003). The chosen method of data collection is an interview combined with documentation to provide a stable “chain of evidence” (Yin, 2003). An interview is a guided conversation between two or more people. The person guiding the interview is called an interviewer and the respondents are called interviewees. The purpose of the interview is to get relevant insight into the phenomenon of the study and understand the interviewee opinion on a given topic. The interviews can be either formal or structured, by using a given set of predefined questions, or they can also be done in an informal and unstructured matter. The interviews can be divided into several categories: structured, semi-structured, in-depth or unstructured (Kumar, 2011; Saunders, Lewis, & Thornhill, 2009). They are often used as a primary data source within IS research (Schultze & Avital, 2011).

For my study, structured interviews were chosen for 14 out of 15 respondents and one semi-structured interview. The purpose was to get a practical opinion out of managers and

developers who work with agile development to understand how they are challenged from the transition. Furthermore, they were expected to provide as much insight as possible.

Step two - The construction of the interview questions was based on an interview guide (See [Appendix A](#) and [Appendix B](#)). The guide was constructed separately for both managers and developers since the questions for those two groups differ. The purpose of creating this guide was to allow for replication for future researchers.

Step three - The data collection would be impossible without constructing the interview questions based on the interview guide. This helps to build more consistent questions by giving a clear overview and being detailed at the same time. Since the case study has an exploratory purpose, the questions can start with “How?”, “Why?” and “What?”. These kinds of questions are asked in regard to “*a contemporary set of events over which the investigator has little or no control*” (Yin, 2003, p.9).

Step four - After creating two sets of interviews, I started to look for respondents in the field of agile development. The interviewees are approached through e- mails, social media inquiries (Facebook, LinkedIn, Quora) and after they agree, the corresponding interviews are sent. To give them a better understanding of the phenomenon of study, a short description of the questions is provided in the beginning of the interviews.

For managers, it states: “*The following questions are guiding us through a transition process of agile methodologies from the point of view as a manager. These interview questions are connected to seven main challenges that the author identified in the literature. Your answers will have a great contribution to develop a holistic view of the transition challenges and will give more trustworthiness and reliability to the study. This structured interview starts with a short introduction, followed by more detailed questions about each challenge and ends with a few closing questions*”.

For developers, it states: “*The following questions are guiding us through a transition process of agile methodologies from the point of view as a developer from the agile development team. These interview questions are connected to seven main challenges that the author identified in the literature. Your answers will have a great contribution to develop a holistic view of the transition challenges and will give more trustworthiness and reliability to the study. This structured interview starts with a short introduction, followed by more detailed questions about each challenge and ends with a few closing questions.* “

One of the interviewees had a different approach than the others. This respondent has a great influence in scaled agile development transformation in big organizations and I attended a lecture and workshop that the respondent organized in collaboration with Bremen University, Germany. Conducting a semi-structured interview was the only option to retrieve valuable information from such an influential person in the agile development world.

Step 5- After sending out the interviews to each respondent, it took them average of one week for each respondent to answer. Some of them are detailed and others give short answers but with punctual information. The delays in answering can be challenging for my motivation, but patience and persistence are needed in this moment of the development of a thesis. After collecting all the interviews, a summary of their role and data collection method are presented in Table 2.

| <i>Interviewee</i> | <i>Role in organization</i> | <i>Data collection</i> |
|--------------------|---------------------------------------|---------------------------------|
| A | Scrum master | Structured interview |
| B | Software developer | Structured interview |
| C | Web developer | Structured interview |
| D | Agile development coach/ Scrum master | Structured interview |
| E | Agile development coach/ Scrum master | Semi-structured interview |
| F | Project manager | Structured interview |
| G | Software engineer | Structured interview |
| H | Computer science student | Structured interview |
| I | Software engineer | Structured interview |
| J | Software developer | Structured interview |
| K | computer science engineer | Structured interview |
| L | Software developer/ Scrum master | Structured interview |
| M | Software engineer | Structured interview, Documents |
| N | Scrum Master | Structured interview, Documents |
| O | Scrum Master | Structured interview, Documents |

Table 2: A summary of interviewee positions and data collection

3.3 Data analysis

To have a better understanding of the case study and increase the chance of reaching a relevant conclusion, the analysis method of content analysis was preferred due to its depth and an overall better understanding and creation of categories and themes. I focus on several

companies which means that an exploratory single embedded case study was performed. (Yin, 2003).

Content analysis uses systematic coding and categorizing techniques, used to analyse large chunks of textual data in order to form patterns of words, codes, themes and their relationship (Vaismoradi et al., 2013). Content analysis trails a qualitative analysis of data, but at the same time allows to quantify the data (Gbrich, 2007).

Following the guideline of Graneheim and Lundman (2004) for qualitative content analysis there are several steps to conduct the data analysis.

First, I identified the unit of analysis, which in this case is a whole interview. All interviews were spiralling around the transition challenges of agile adoption and therefore the unit of analysis is a whole interview (Graneheim & Lundman, 2004). Each one was read through three times before starting to trace codes. After identifying the meaning units in the text, a process of abstraction was made to create condensed meaning units. I labelled them with categories in order to view the data from a different angle. Based on commonalities in the data and mutually exclusive categories, the categories were created by answering the question “What?” (Graneheim & Lundman, 2004). Since almost all the interviews were structured and not performed face-to-face, I decided to focus on the manifest content. Finally, the themes were formed by interpreting the latent content of the categories by answering the question “How?” and combining the sub-themes. [Table 1](#) also guides me to look deeper into each challenge and extract more obstacles that might have been absent in the literature.

Transcription of interview

With the respondent consent, one of the semi-structured interviews was audio recorded to have a better understanding of the interviewee answers which also allows the interviewer to go back and listen to the interview again in case there are ambiguities. Consequently, the interview was transcribed verbatim.

Response summary

After having all responses, summaries of the answers were created by highlighting the main points of each interview. Microsoft Excel helped to create a better organization of each response. In my study, the analysis of the interviews was done through content analysis in the following order (Graneheim & Lundman, 2004):

- Selection of meaning units - The selection started by reading the interview several times to get a general understanding of the content. While reading, I made notes that could be useful during the analysis and underlined parts of the sentences. After reviewing again, the highlighted parts of the text were chosen as meaning units. Their length was not too big (like whole sentences or paragraphs) and not too short (no less than 5 words). The meaning units contained just enough information which gave enough meaning without too many words, but still provides the main idea of the sentence.
- Creating condensed meaning units - After identifying the meaning units, the next step was condensing the meaning units. They were condensed as closely as possible but still preserving the “core” of the text.
- Identification of codes - The following step includes a process of abstraction. In this process, I used interpretation of the latent and manifest content of the text to group

those codes into “higher logical levels” which were helpful for the next stages. The creation of codes for my study was done by choosing the most appropriate word that fits a description of the condensed meaning unit.

- Aggregation of codes into categories - The categories for my study share analogous meaning. The codes helped to form categories which were mutually exclusive and none of the data stood between several categories, nor fitted it to several categories. Not all categories were mutually exclusive, because some respondents shared opinions instead of experiences. The categories in the study answer the question “What?” and revealed the manifest content of the text.
- Construction of themes and sub-themes - In my thesis, the themes revealed the experience of the respondents. To form the themes, I used the question “How?” and used my interpretation based on the condensed meaning units, codes and categories. The task of the themes was to express the latent content of the text and in some cases, I used several categories that fitted into several themes. After creating the themes, I divided them into sub-themes by using verbs and making them sound like I am persuading them to perform an activity. This way the sub-themes would have more impact on the readers.

3.3.1 Quality evaluation of the research design

Participant validation

After agreement to the interviews, the participants were sent an e-mail in contemplation to receive verification in addition to the questions as an attachment to the e-mail. It stated the following:

Attached to this e-mail, you will find the interview questions about the transition challenges that you experienced. Do not hesitate to ask me if you have any further questions.

After receiving an affirmative response, the process of analysis is initiated. In all cases the answers are positive, so the analysis of the transcribed data starts.

Construct validity

Following Yin (2003) test to provide construct validity, three elements are needed: multiple sources of evidence, chain of evidence and case study report to be viewed by key informants. In my study, one source of evidence was used to provide knowledge about transitional challenges of agile development, namely interviews. After they were conducted, content analysis was performed to retrieve categories and themes.

The chain of evidence was supported by all the resources in the literature review which are used throughout the thesis. The choice of resources is described in the Methodology section. After the analysis of each interview, a summary was made and checked again from the same interviewees to assure accurate interpretation and no bias from me. This method supported data triangulation, suggested by Yin (2003).

The respondents were asked if the summary of their answers was adequate so that they could verify that there was no error in my interpretation. Every interview was analysed only with the consent of each interviewee. *External Validity*

This test “deals with the problem of knowing whether a study findings are generalizable beyond the immediate case study” (Yin, 2003, p. 36) and since statistical generalization does not apply in this research, the case study is using analytical generalization. This means that the results of transition challenges for developers and managers are generalized to a “broader theory” (Yin, 2003, p. 36). The theory was provided by an extensive literature review which contributed to

form the frame of reference from where the interview questions are designed. This allows the analytical generalization of the same literature and results made by other researchers. This process is called “replication logic”.

Reliability

Reliability can be established mainly if a researcher can undergo the procedures of the given case study and reaching the same results and conclusions in the end (Yin, 2003, p. 36). For this purpose, interview guides for both groups were used in order to ensure that the interviews were made under the same circumstances. All supporting procedures that ensured reliability are discussed in this chapter, backed up by evidence, provided in [Appendix A](#) and [Appendix B](#) meaning that bias and errors are eliminated.

In addition to above, to achieve trustworthiness, Graneheim and Lundman (2004) suggest three concepts that assure it: credibility, dependability and transferability.

Credibility addresses the focus of the study, data analysis and choice of participants. In the case of my thesis, the participants are chosen randomly, without restriction of age, gender and territory. They have various experience with agile and traditional methodologies, starting from university students to world-renown agile methodologies coaches. The credibility of the findings can be evaluated by the selection of the right amount of the meaning unit. They are not too broad (paragraphs for example) nor too narrow (single word for example). The meaning is not lost during the condensation and abstraction process because I managed to identify the necessary information and analysing it without losing its core meaning.

Trustworthiness can be achieved also through dependability, which reflects the “*degree to which data change over time and alterations made in the researcher decisions during the analysis process*” or in other words, the longer the period of collecting the data- the bigger the chance of having “*inconsistencies*” from the author (Graneheim & Lundman, 2004). The positive side of my structured interviews is that they allow dependability since all interviewees are asked the same questions. The data collection extends to a maximum of two months, which barely allows for the opinion of the respondents to change. Nevertheless, future researchers who want to replicate my study, might have different results due to “*changes over time*” (Graneheim & Lundman, 2004).

Transferability (Graneheim & Lundman, 2004) deals with “*the extent to which the findings can be transferred to other settings or groups*” (Polit and Hungler, 1999, p. 717). Transferability is achieved through a clear and detailed description of the characteristics of the respondents, data analysis and data collection. This was also strengthened by adding quotations from the interviewees in the analysis of the responses.

After creating the themes of the content analysis, I provided two more tables showing how each respondent felt according to a corresponding challenge. The symbol “X” symbolises that the given challenge is NOT an obstacle for the given interviewee. The symbol “✓” symbolises that the given challenge IS an obstacle for the given interviewee. After summarizing the positive challenge for each obstacle, we can see visualized that the sum shows which obstacle is the most challenging for each group.

- For developers, most challenging are the concepts of PM role and Change in mindset.
- For managers, most challenging are the concepts of PM role, Change in mindset and Organizational agility.

We can see a more detailed visualisation of the results in the following tables.

| Interviewee | Communication | PM role | Change in mindset | Organizational agility | Decision making | Documentation | Tools |
|-------------|---------------|---------|-------------------|------------------------|-----------------|---------------|-------|
| 1- J | ✓ | X | ✓ | X | X | X | X |
| 2- I | X | ✓ | ✓ | ✓ | X | ✓ | X |
| 3- C | X | X | X | X | X | ✓ | X |
| 4- B | X | ✓ | X | X | X | X | X |
| 5- H | X | X | X | X | ✓ | X | X |
| 6- G | X | X | ✓ | X | X | ✓ | X |
| 7- K | X | ✓ | X | X | X | X | X |
| 8- M | ✓ | ✓ | ✓ | X | X | X | X |
| Total: | 2 | 4 | 4 | 1 | 1 | 3 | 0 |

Table 3: Developer Challenges: ✓-is a challenge; X- not a challenge

| Interviewee | Communication | PM role | Change in mindset | Organizational agility | Decision making | Documentation | Tools |
|-------------|---------------|---------|-------------------|------------------------|-----------------|---------------|-------|
| 1- F | X | X | X | X | X | ✓ | ✓ |
| 2- D | X | ✓ | ✓ | ✓ | X | X | X |
| 3- A | ✓ | ✓ | ✓ | ✓ | X | X | ✓ |
| 4- E | ✓ | ✓ | ✓ | ✓ | ✓ | X | X |
| 5- L | ✓ | ✓ | ✓ | X | X | ✓ | X |
| 6- N | ✓ | ✓ | ✓ | ✓ | X | ✓ | X |
| 7- O | X | X | X | ✓ | X | X | X |
| Total: | 4 | 5 | 5 | 5 | 1 | 3 | 2 |

Table 4: Manager Challenges: ✓-is a challenge; X- not a challenge

3.4 Research ethics

Each respondent of the interviews was first requested to take part in this research. The e-mail explained the purpose of the study, the nature of the questions and the kind of interview questions they would receive corresponding to their role in agile development. In the case of the unstructured interview, I asked for the consent of the respondent to participate in an audio recorded interview. I introduced myself, stated in the beginning of the interview the purpose of the study, the nature of the questions and why this interviewee was valuable for the given study. After confirmation, the interview was transcribed and analysed accordingly. Each of the respondents names were replaced with letters from the alphabet to ensure their anonymity.

I conducted 15 interviews, one of which semi-structured and the rest structured. The respondents were given alphabetical letter corresponding to the time at which the interview is taken. Then they were divided into two groups, one of developers and the other one of managers/Scrum masters.

4. Results and analysis

The analysis of the interviews is done based on the identified concepts in the literature review. Currently, most the interviewees were witnessing agile development in their companies as it was, meaning that they did not participate in the transition process. This means that the methodologies were adopted long ago and that the ones without traditional experience had to learn to work with agile development without prior working habits with traditional methodologies. Only a few of my interviewees had a traditional background that meant that agile was, so far, a preferred methodology. As stated in the literature, agile development requires fast thinking and swift reactions to changes, which my interviewees had to deal with not only once. But a general note is that waterfall methodology elements were still kept or there were mixed methodologies which brought more chaos than certainty in the choice of a methodology. In the following text, the results from the interviews are presented by category and themes.

Communication - Encourage communication

The results from the interviews show a gradual change from lack of problems with the developers to larger obstacles for the managers. One extreme shows the support of “*less talk, more data*” and another extreme shows the highly informal environment where communication is the foundation of collaboration. Social skills are important in all cases, especially to define the capabilities of the teams and involve the stakeholder participation. It is highly important to acknowledge that communication between team members and managers is not all there is - we should not forget the clients. Even though their opinion is somewhat leading when they choose to go agile development, when it comes to the organizational transition it is more valued that the teams and management feel comfortable with it. When it comes to communication during a transition, the communication with clients is less important than the one during a project. In general, agility requires social skills and an open mind.

When a company is going through a transition it is important to communicate the change, meaning that agile development must be well understood by everyone. This happens through trainings, meetings and presentations for the company that is going through a transition so that later there are no misunderstandings and most of all - everyone should be transparent. Finding the right way to reach everyone could be challenging, but in the end, it is all worth it.

The way to communicate could be formal or informal. Most of the time both are required due to several reasons. Formal communication is important to maintain between the team and clients and in any sort of management systems. On the other hand, informal communication is highly preferred to have between the team members and support more openness, visibility and transparency. Effective communication inside the team can be achieved through having social skills, but shyness can hinder a successful transition and project work.

Without being able to determine these factors, there will be a lot of inaccuracies in the products and questions will follow from the stakeholders, which will slow down the work and deliver the project over budget and past deadline.

4.1.1 Demand social skills

This sub-theme represents the importance of having social skills in an agile development team. Some interviewees emphasize on the value this concept can bring. Interviewee J for example said:” *We should be able to communicate our thoughts...*”. The respondent is pointing out that communication is necessary to share our thoughts on the project, because collaboration is essential. Respondent F and I agree on this topic by stating consequently that” *...social skills are vital...*” and” *Social skills are a” must have”*. Finally, respondent D confirms the immense value communication brings and that” *social skills are very important*”. We can say that an important part of an agile development team is communication and having social skills is a big advantage and necessity.

PM role- Facilitate change

A challenge for developers when the role of a manager change is the resistance to involve all team mates who have different knowledge about the methodologies or have different skillsets. This brings us to the challenge of identifying the maturity level of the team which a non-agile methodology manager might not be aware of. Identifying the skills of the team in advance for a manager in a new agile methodology environment will help prevent failure of the project which involves duplication of code or having developers that are way behind others. If there are more teams, coordination between managers is necessary and essential. A great challenge that management faces is the inability of facilitating the idea of being agile because most of the older companies rely on traditional methodologies. Managers must be aware that changing and losing control might be scary because this also includes changing of roles and responsibilities.

The biggest challenge here is that in some traditional methodologies, managers are used to giving orders and controlling the work done by the employees. When those managers must transit to agile methodologies, things can get hard and challenging for them because their role is changed in their face. Agility gives an unfamiliar perspective and different values that they have not had the chance to explore. In agile development, people have much more freedom to discuss decisions together and decide on features that the manager does not necessarily have to agree upon. The grant of freedom is often a big problem for traditional managers and this can only be improved by changing the mindset. As it turns out, persuading middle management to transit to agile methodologies is the hardest task for agile development coaches. Their task and role are to help companies understand the values that the agile methodologies represent, by helping the organizations embrace the mindset and facilitate the teams (Home - Agile Coaching Institute, 2016). Top management is easily persuaded by showing them presentations and having a good agile methodology coach and showing them that they need this change. Usually once they are on board with this transition, the team is also ready to accept it, but middle management are afraid that changing their role in a new setting, might lead to losing their job. Traditional organizations need more coordination to work properly and a good project manager is needed to push this initiative through. The middle managers have a lot of power and suddenly there is a new flow in the organization, what people are working on is now more important than who is giving the orders. They feel that their positions are at risk and they are feeling threatened and if there is any resistance, it is usually from middle management. In general, they do not want to give away this power that they have and they are thus not taking agile methodology seriously.



4.2.1 Involve planning

One emergent challenge in agile development is the lack of planning of the requirements and specifications upfront. This is a part of a waterfall methodology approach, but most of the interviewees do not have any experience with traditional methodologies. Respondent I complained that “... *it is hard to pin down a solid delivery date...*” and “...*that some items originally scheduled for delivery may not be complete in time.*”, which can be an emerging issue to the clients and the calculation of expected costs of the project. She also shared that “...*not having a definitive plan...*” can make the final vision of the product look different. The team of respondent M had no experience with either of the methodologies, and “*The general understanding of Agile in my team was that it is a somewhat chaotic development without a priority for upfront planning or documentation and detached management control.*”. This means that the first impression of agile development for this team is that it is disordered. Respondent K emphasized the relationship between the previous sub-theme where social skills are important and how they help for a better and earlier delivery of the project: “*You need to be communicative, sociable kind of and a lot rigorous because of the constant delivery deadlines.*”.

4.2.2 Grow transparency

Transparency is an emergent issue which can be considered a great barrier for some developers. Not everyone who start to work with agile methodologies are used to share all their activities in a project, especially if they come from an individual working scheme like waterfall methodology. Opening all doors can be positive but also at a given point it can be discouraging. Nevertheless, this is what is special about agile methodologies and developers should take their time to learn how to use this concept for their advantage. Interviewee C says that “... *we need to be on the same page and know what the other one is doing...*” which shows again that social skills are necessary for the commitment of this task and that collaboration is crucial. The successful transition to agile development can be done through transparency which is essential to know what the other teammates are doing so that everyone is “...*on the same track...*”. Respondent J confirms that transparency can be new and scary but also necessary and fundamental: “*When I started working in an agile environment the one thing that struck me most was the transparency - all of a sudden every member of the team knew exactly what everyone else was doing right now*”. Interviewee D points out that when a transition happens from waterfall methodology, some teams might have a bigger problem adjusting to transparency and not dealing with it can lead to the exclusion from the company: “*Several people have problems with the transparency (this is also the most often problem we have to face when we help our customers implementing agile methodologies... But a few colleagues have also left the company)*”. Respondent I concludes that “...*we started to have more meetings... and everyone was more aware of what the other team members are currently working on.*”.

4.3 Change in mindset - Provide training and support

One of the most challenging steps in agile methodologies transition turns out to be the change of mindset. This is the foundation that needs to be achieved to have a successful transition and without it, agile methodologies values might be misunderstood and neglected. It is often perceived that this change might be impossible if the projects do not deliver positive results. So, to prove them wrong, agile methodologies coaches need to provide evidence that this adjustment is leading to better results if they want this change to survive. The general perception of the companies before transitioning is that agile methodologies are something fancy and luxurious, but only when they clash the transition process they can see

that this change is difficult to make and it is transforming every process in the company. To have this successful transition, there are also a lot of resources needed to make it possible, so the company needs to make an informed decision lead by someone with enough knowledge in this field.

Changing the mindset is a great challenge for managers and agile methodologies coaches and it is also related to the changing role of the project managers discussed earlier. Transparency takes a substantial portion of this challenge because not everyone is used to having an open environment where everything is visible. For developers, it is easier to accommodate this change by short iterative cycles and planning because it is easier to go back in the reprioritize the backlog which gives a lot of freedom for them and a better acceptance. Accepting changing requirements, being open to discussion and relying on collaboration within the team are some of the skills needed to accept the mindset easier.

In some companies this might be harder than expected due to the fact that the working environment before agile methodology does not accept failure and trying out new things is scary and not an option. Recognizing the problems right now but not doing anything about them because “now is not the time to change things” can hinder the acceptance of the mindset. People who have a traditional experience gain a lot of experience in this field and workflow and they already mastered given techniques. They have learned to trust their own skills and the tools that help them to manage the processes. To ask them to stop using those tools and forget these skills and to start using something quite different without trying to forecast the future through extended project plans, is a challenge. For developers, it might not be comfortable to work in an agile methodologies context in the beginning because this means that now they work with product development and must take ownership of their activities. Some change their minds when they get the chance to be involved in the new workflow, some do not. It is good for them to be open minded combined with willingness to accept change and to learn constantly. It would be also beneficial to excel at reading the environment so that they can understand the positive results of this transition and newly changed system.

4.3.1 Clear requirements

The issue of not having clear goals in a project is one of the greatest challenges in an agile development transition. Most teams are used to having to know what they are supposed to create upfront and this unclear vision could be misleading. Like interviewee I company: “...*the company relies on entirely iterative approach... continuous improvement and flexibility.*”, where iteration is a great part of a project, it is not necessary that this is the case with other companies. For a junior developer like respondent K, that” ... *you do not have clear and detailed specifications...*” can be also misleading, but, on the other hand, this iterative attitude could help a young and fast mind to create better ideas and improve the product.” *Another issue is the lack of design and solid requirements upfront...*” and” ... *lack of upfront solid requirements...*” is what the team of respondent M is challenged against. Also” ...*lack of... central leadership figure...*” which is also connected to the shifting role of the project manager.

4.3.2 Eliminate trust issues

Having trust in a newly transitioned agile development team is an essential part to achieve successful transformation. This opinion is supported by respondent M who claims that” ... *gaining trust in the team members is a crucial part of...*” a successful transition and the delivery of an outstanding project. agile development teams rely highly on trust between members and Scrum master O adds that” ... *you can not implement this methodology without the trust from management, team and clients.*”. This means that proving the company and clients as well that

this methodology is the right one for this project, is crucial for its successful transition. So, eliminating the trust issues is a fundament of changing the mindset of companies and clients to transit to agile methodologies. This is supported by interviewee N who claims that” *...trust within the team is a crucial part of the successful transformation...*”.

4.3.3 Mitigate chaos

Disorder could be somewhat demotivational when a team transits to agile development. One of the flaws of agile methodologies is when introduced to a team it can be chaotic related to the lack of planning, as one of the concepts commented above. Chaos is something that can scare away potential companies that would like to work agile methodologies and mitigating it can give a chance to the methodology to thrive in new organizations. Interviewee M shares his concerns about the opinion of his team in the beginning of their agile methodologies journey, followed by the first experience of respondent O:” *... in my team was that it is a somewhat chaotic development*”;” *... in the beginning, it was so messy and cluttered...*”. In this sense, interviewee N suggests to transitioning companies that” *...chaos can be crucial, so everyone should do their best to eliminate it.*”. Scrum master D concludes that” *The biggest issue could be the change of the mindset. In waterfall methodology project managers are used to give orders and control the work finished by their employees.*”.

4.4 Organizational agility - Change processes first

Depending on the organizational needs and capabilities, transition to agile development can happen inside-out or upside-down, but no matter which way they choose, the transition should always affect the whole organization and not just some teams or some projects.

When a company decides to go inside-out, the change starts from the team itself. It takes small steps, but to start this transformation, changing the mindset is a great necessity which can affect the acceptance either positively or negatively. The organizational change which starts from the teams must be initiated by education about the methodology and what follows. Sometimes it is easier for teams to group by interest and development capabilities or also by personal interest. After recreating the first project, it is important to give a retrospective view of what has been done and to include every department in this project. Similarly, we can see this agile methodology transformation as a project which is essential for the success of this company and relies on the participation of every part in this company.

When doing an upside-down transformation, it is important to make everyone in the company aware of this new methodology. This can be possible by hiring professional coaches which can “take the company by the hand” and lead them through every process. The choice of the transformation of course depends on the range of integration of their products and if they are creating diverse products or single ones. Interviewee E shares that in some cases, the transition can start with a pilot test, which is a European way of initiating change. Compared to the USA for example, a change like this is done in a way that the organization is going fully in, once. This is related to the culture of the managers who are used to delivering successful results very rapidly in a short time to reach targets. In contrast, the European way of doing transition like this, uses softer skills in a way that they do first half first, they learn from it, explore the results and if it is not satisfying, they roll out of this idea.

Nevertheless, many companies do not understand that this change is huge, and it touches all aspects and people of the organization, when it is upside-down. Not having an educated manager specifically for agile methodologies might hinder this transition and crash the agility of the company. This jump is often undervalued and employees do not realize that it requires time, hard work and resources. To reach organizational agility, no matter the approach, a change in mindset and full understanding of the agile values are a great necessity.

4.4.1 Transform upside-down

This challenge is one of the most important in my study since I understood that a successful transition starts upside-down. The only way this process could be fulfilled successfully is if it includes the whole organization. Many companies practice the way of combining an agile methodology with a traditional one, but in fact this turns out to be not only wrong, but entirely in contrast with all agile methodologies values. So, transitioning to agile development touches all components of an organization and as the agile coach E states, “...*this change is huge...it touches tools, people, organizational management, it touches full transformation, it is upside-down.*”. What this respondent is trying to emphasize and give a bigger meaning is that “...*organizations do not understand how vital this change is and how large it is...*”. To implement such a profound change, it must start with the mindset of top- management and really transform the whole organization from there on. Most companies do not have an idea how big the transformation is and that mixing agile methodologies with traditional methodologies is a course that will destroy the values it bears. The transition “...*is undervalued and it is really hard work.*”. In the company of agile development coach D, the transition started in a different approach.” *We started the transformation in the project teams. The change of the mindset has then led to a change in the organization.*”. We can see that change in the mindset is starting point of every transition no matter if it starts top-to-bottom or inside-out.

4.4.2 Persuade middle management

When a company decides to transit to agile development there are several levels of people that must be convinced that this is the best choice for their company. According to the broad experience of respondent E, “... *most challenging is the middle management...they have a lot of power.*”. Going through a successful transition means that not the top- management, but the middle one is the hardest to convince. This challenge is also related to the changing role of the project manager very deeply because it proves the essence of it. Respondent D adds: “*In waterfall methodology project managers are used to give orders and control the work finished by their employees. In agile projects, people have more freedom and ideally a much stronger commitment to the project. This grant of freedom is often a big problem for traditional managers.*”. Traditional methodologies require much more control on management level, and once agility brings uncertain roles in the team, it threatens the position of a manager. This means that agile methodologies could lead to loss of jobs for those who do not have the proper change in mindset. As respondent E claims: “...*they are feeling that their positions are at risk and they are feeling threatened...* and “...*you need to have the top management with you...*” we can conclude that management positions from middle management are important to persuade when we want to have a successful transition. Respondent O adds: “...*the first barrier is to get through middle- management, the rest comes step by step, but not having the managers might completely jeopardize the transition.*”

4.5 Decision making- Encourage collaboration

Making decisions together is an essential part of every project. A challenge identified in the literature is how decisions can be taken from different entities in a team, and in practice we see that decision making is a collaborative process which is highly supported by social skills and highly communicative environment.

Decision making is a part of every agile development project, because the methodology requires collaboration and common decision on a given matter. When a company makes a transition, the agile methodologies values should be taught well to the developers because now they have more freedom and making decisions together should be profoundly

encouraged. This can be done by daily meetings that give free space to the developers to discuss issues as a group and make a common decision.

In more traditional organizations, decision making is not really a part of the process. The decisions are made in the planning part of the project, so developers are supposed to do what is already predefined and decided for them. When they switch to agile development, decision making might be unknown concept from group work point of view. Some companies start this transition by changing the mindset first and then the processes, others start with processes first and the mindset will follow, but in any case, decision making is part of both ways.

We can conclude that decision-making and communication are concepts that are highly related. A balance between formal and informal communication is required in this case, because having formal and informal communication channels will result in less tension between the teammates and the right decision will be made through collaboration.

4.5.1 Self- organize

The ability to self-organize within a team is a part of achieving the core of agility. This means that teams should be able to communicate between each other and the outcome could prevent excessive documentation which in some cases might hinder or slow down an agile methodology project. To achieve the agile development transition, a team needs to be able to work as one but at the same time it is necessary to be able to spread responsibilities. There are cases in which some developers lack specific skills for a given project that others have and achieving a good collaboration can increase their agility and be able to adapt to changes swiftly and quickly. Manager O shares that “...an agile team should be adaptive to the environment and the correlation between each team member should be strong...”. Self-organization proves to be an important and necessary skill and as manager N adds, “it can be crucial for the time when a company makes a transition... self- organization proves that the team is ready to take a step in an agile environment.”.

4.5.2 Discuss issues

Decision making is a vital point of a successful transition to agile development because it encourages team work and cooperation. It requires social skills, improves transparency and increases trust among team mates. Those components are highly encouraged to have to reach agility. Having undiscussed problems can lead to mismatched requirements, duplicated code and many more, but talking about them is essential. For interviewee M even the introduction of meetings is challenging: “I was not able to see the importance of having regular meetings and discussing the status of my progress.”. After getting used to it things started to change for him and “Each problem was discussed within the team at the weekly meetings and a decision was taken upon agreement.”. Decision making in general is supported as a shared group activity among most of the interviewees and a common decision is taken after a conversation among the team. Rarely, in some companies that have a traditional influence it can be witnessed that a decision is made only if the manager agrees. In cases like this, it should be considered the general opinion and discussed why a given decision is not taken.

4.6 Documentation and maintenance - Turn documentation to communication

Documentation can be either too less or too much. Less documentation will result in more questions from stakeholders and this leads to slowing down the project and delivering it past deadline due to lack of customer feedback. Also, future maintenance of the product requires documentation that the developers can follow to track down mistakes. Too much of documentation also slows down the processes and brings down the development team, as identified in the literature review. A challenge is to identify when the documentation is at the right amount, when do they achieve balance and how is efficiency achieved in this case. Or there can be more ways to have sufficient information in a lesser amount. For team and managers that face a transition, it is better to find an innovative way to discover the right balance. The most efficient way though, is to turn documentation into communication and use simple agile methodologies approaches to still have enough documentation. Certain roles now have more responsibilities of knowing certain things and documenting them. For example, the product owner is now responsible of knowing what the content is and communicating it right and frequently.

For developers, documentation is not a challenge, since the agile methodologies values do not require substantial amounts of documentation.

4.6.1 Need of documentation

Documentation is one of the points where we can clearly see traces of traditional influence. Interviewee J is relying on agility in documentation, meaning that *“...if it changes while the project grows, it gives you a lot of room to be agile.”* Respondent F as a manager has a different opinion on the amount of documentation and *“...generally speaking, not documenting enough... we try to keep all documentation up-to-date as priorities change...”*. The logic behind this decision is that documentation as detailed as it is, can help testers and developers in the future if the product needs maintenance and upgrades. Respondent K agrees on an opposite direction, namely that *“...the outcome is better than by doing a huge documentation...”* so, *“Less documentation brings more flexibility”*.

4.6.2 Innovate documentation

Documentation does not necessarily need to be mundane and a cumbersome process. There are ways to innovate it in a way that it less in amount but equally informative and explanatory as if it is kept all through the project. Respondent K explains the process of documenting features but preserving their importance with less documentation: *“First, we describe the user stories, then we split them into tasks...”* and then respondent D confirms that *“...most teams are using user stories.”*

4.7 Tools and technologies- Self-explore

The change in workflow when a transition is done, leads to change in tools. Things like built tools, automation of process steps and containerization of software are becoming more and more important for agile development teams to release fast and often and to have an infrastructure which could be changed easily. In agile development, tools are used to track down the progress of the projects. This brings reliability and shows that there should not be too many tools that might hamper the work. In my study the choice of tools turns out to be the least challenging. Adopting new tools can be done in several ways. A pilot of the new tool can be done in order to test how good the developers feel with it. Experts say, that if a team is agile, then the developers should be able to choose by themselves. It is important to

maintain transparency and visibility through new and updated tools, which depends on the level of integration in the company. It is good to have a set of tools that the team can choose from and try out, then do a pilot test on it and choose the right one.

Choosing the right tool to support task management for example, requires time, practice and patience. So, choosing the right tool is essential to not waste resources and time that can be put in better use.

4.7.1 Self-explore

The choice of tool is the least challenging for developers and managers. Every company has a different approach when it comes to having the right tools and there are no conflicting points for both sides. With interviewee E it is highly recommended that developers research on their own and select the ones that fit their needs the best: “...teams should be able to select the tools by themselves...”. Respondent I agrees that as a developer, tools “... are only a matter of choice.”. Interviewee D as a manager, has a different approach to choose tools, mainly “*Inspect and adapt. Learn about the tools and keep using the best ones.*”. Respondent K is “... doing the same as agile does, iterating to test the tool and if it fits to the good development of the product, then it is surely a good tool.”. In general, we can conclude that choosing tools is not a clashing point and not challenging for both sides.

Below, we can see a visual representation of the discussed sub-themes and their relations and dependencies between each other. The table gives a clear overview of how one sub-theme can influence another (See [Figure 3](#)).

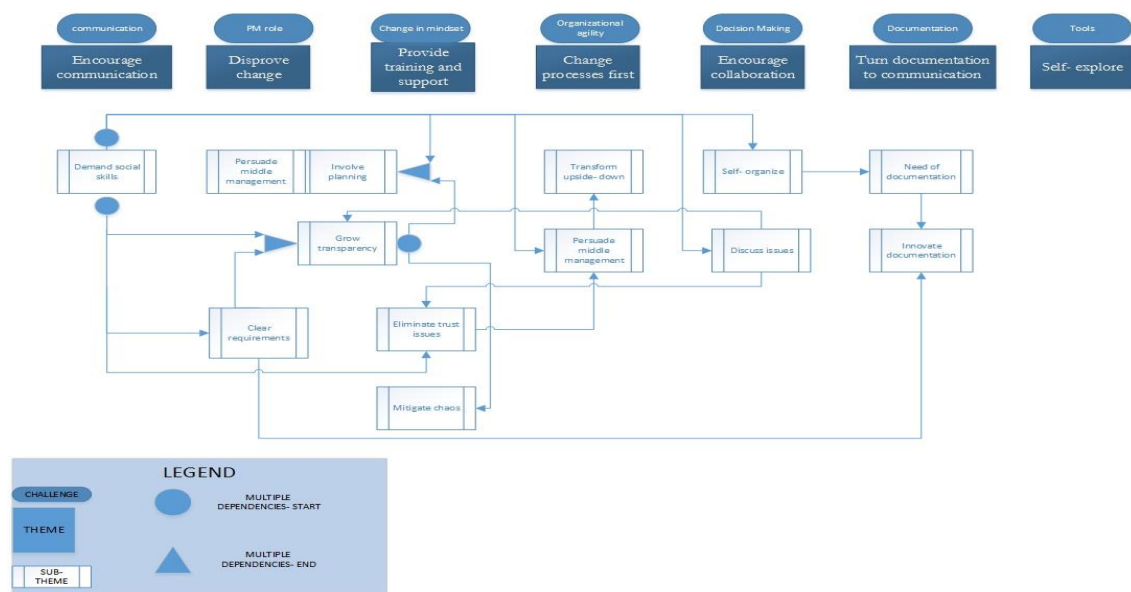


Figure 3: Concept model of themes and sub-theme dependencies

Themes Developed in the Content Analysis of the Interviews

The process of gathering the data started by doing two interview guides (Appendix [A](#), [B](#)) for managers and teams. Then the interviews are conducted and transcribed. Before I started the analysis phase I had get familiar with the various methods. After careful review of the literature (Graneheim & Lundman, 2004) I have chosen to do a content analysis. The unit of analysis is a whole interview and I will focus on the manifest content of the interview, since

they are structured interviews and I do not have access to the latent meaning of the respondents. The preparation phase started by reading each interview at least 3 times to get the whole idea and make notes which I can use later. From there I started abstraction by highlighting the meaning units as a first step and copy/ pasted them in the table. A shorter version of the meaning units is made into the condensed meaning units through the condensation process. Finally, I aggregated the codes and grouped into categories, which exclude each other (Krippendorff, 1980), and made general themes and sub-themes.

The themes are:

- Encourage communication - Communication is a vital part of every transition and change in life. When a company chooses to work with agile development, the management should be aware that every part of this transition should be communicated and implement this process in the following projects and make it a main part of the team. Encourage communication for better results and collaboration for the team members.
- Facilitate change - Change is the ground of a transition to agile development. It can be seen as a synonym to this process, because change affects people, processes and the whole organization. Accepting change and facilitating it once the transition starts, will help the organization to understand the agile values and create successful projects.
- Provide training and support - A big and important change can be done with the help of professionals like agile methodologies coaches. This great organizational transition is important and providing support will only benefit to the company. Administering support through this process will only benefit to the company because having an expert in this field will help to create an effective and flourishing agile environment.
- Change processes first - Transitioning to agile methodologies can be done in several ways, but changing the processes first, proves to be preferred by professionals. This can be done in case that the company does not agree initially to the agile values, but the transition needs to be done anyway. When facing hard conditions of this process, change to processes first and the mindset will follow.
- Encourage collaboration - Collaboration is an integral and inevitable part of an agile development project. It requires communicating all issues and reaching a common decision. To have a successful transition it means to also encourage collaboration.
- Turn documentation to communication - Coming from a non- agile development background can be heavy on team members and managers when they have to face a new way of workflow. To better accept and integrate the new way of working it is encouraged to turn documentation into communication. This means that there are other ways of documenting features without losing time and encouraging communication, trust and transparency at the same time.
- Self- explore- Self-exploration of tools is a great responsibility for the developers. They now have the power to look for tools that reflect best to their skills and use them for their advantage. This gives freedom and more responsibilities that some non- agile developers might face for the first time.

| Challenges | Theme | Sub- theme |
|------------------------|--|-----------------------------------|
| Communication | <i>encourage communication</i> | <i>demand social skills</i> |
| PM role | <i>facilitate change</i> | <i>involve planning</i> |
| | | <i>grow transparency</i> |
| | | <i>clear requirements</i> |
| Change in mindset | <i>provide training and support</i> | <i>grow transparency</i> |
| | | <i>clear requirements</i> |
| | | <i>eliminate trust issues</i> |
| Organizational agility | <i>change processes first</i> | <i>transform upside-down</i> |
| | | <i>persuade middle management</i> |
| Decision making | <i>encourage collaboration</i> | <i>self-organize</i> |
| | | <i>discuss issues</i> |
| Documentation | <i>turn documentation to communication</i> | <i>need of documentation</i> |
| | | <i>innovate documentation</i> |
| Tools | <i>self- explore</i> | <i>X</i> |

Table 5: Summary of themes and sub- themes

After a careful and detailed analysis, I narrowed the results for both managers and developers into 3 stages starting from the least challenging factors to the most challenging ones.

Level of challenges for developers

When we talk about least challenging concepts, in my thesis it means the smallest number of interviewees that consider the given concept as challenging. In this case, developers consider the choice of tools as not challenging at all, also that organizational agility is not challenging, and decision making is something natural and necessary in the team. All developers use an agile methodology way of thinking when they must choose a tool. They iterate, meaning that a tool is tested out and if it does not fit their needs it can be changed with something else. Since most of the participants have not gone through a transition, they do not think that organizational agility is challenging. Decision making is a team activity in an agile development project and is not considered challenging. Everyone discusses the opinions about the projects and they reach a common decision together.

Second to least challenging are the middle level challenges which cover communication and documentation for developers. Some of the respondents consider the need for social skills to be challenging for them, since not all people have the need to be sociable. This is an important fact because communication is one of the most important concepts in an agile development team, according to the findings in my study. Documentation is also important for some participants because having detailed documentation is a necessity for some projects and getting rid of it can be confusing and challenging when a transition is made.

Lastly, the most challenging concepts for developers are the Project manager role and changing the mindset. When a transition is happening in a company, so is the role of the project manager. Developers must adapt fast to the changing positions and requirements and this can be faced with negative attitude to some developers. Fear of change is a great issue which reflects the general behaviour towards the transition and this also means that the responsibilities will grow for everyone. An issue in this matter is the separation of the tasks. Developers should integrate a system to deliver smaller part of requirements and codes for a shorter time and more frequently instead of delivering the product at once. Changing the mindset for developers who transit to agile methodologies can be one of the hardest steps of a transition. The reason is that this huge change comes from within and it will be uncomfortable for people who are not used to the agile values. They must leave their comfort zone and create a new workflow. Not all people adapt to changes as fast as others and this can be discouraging for developers who cannot be agile and flexible.

Level of challenges for managers

The interviewed managers consider decision making and choice of tools as least concerning. As previously discussed, decision making is a team activity which should be nurtured and taken as a natural part of a team. An agile methodology manager or a Scrum master must facilitate decision making and incorporate it during a transition. This means that everyone must be on the same opinion before a decision is made, especially during a transition process when new changes should be implemented. When the transition happens, and the projects require new tools, the managers must let the team decide for itself whether the given tool fits the needs and skills of everyone. Usually this is done through testing and iterating and finally using the tools that corresponds perfectly to the projects and skills of the developers. Middle level challenges for managers are communication and documentation, which are the same results as the ones for developers. Managers and Scrum masters must get used to new ways of work when a company transits to agile methodologies and this includes supporting and facilitating communication and social skills. Coming from a non- agile development experience, some managers might find this challenging because before transition, not everyone from the teams are used to communicating everything that happens in the projects. This is also highly connected to the newly introduced transparency that can be scary.

Documentation is a big part of all non- agile development projects, and removing this from the normal workflow during a transition can cause pressure. Managers now face the fear of not being able to document requirements and calculating deadlines is now harder than before. Not being able to document and plan is challenging for newly transitioned managers. The respondents from the manager group, mark organizational agility, change in mindset and PM role as most challenging during a transition to agile development. As it turns out, the most common challenges for both groups are PM role and change in mindset. Middle level managers are influenced the most when a company transitions to agile development. This happens because most often, they change their position, for which they do not have the proper training, and now their authority is lower. In non- agile development projects, mostly in traditional ones, managers decide every part of the projects and make the orders, whereas in agile development the decisions are made by the team and the manager only helps to move project forward. They cannot put a deadline to deliver the final product because there can be more iterations than expected which includes more changes as well. When the company moves to the transition, managers must get used to the fact that their role is different, and their power is not as high as before. This is interrelated to changing the mindset in a way that there can be resistance first from the middle management and second for the whole company. When doing a transition from non- agile development workflow, the top management that leads this change can meet a great deal of antipathy toward the idea of being agile. This is a profound change that affects everything and accepting values that are foreign, can be one of the biggest challenges. This is also where the transition starts- by changing the mindset of the people, because they lead the change and they are the change. If the management fails at this stage, it will fail for the rest of the transition and possibly there will be none. Accepting agile values will help to reach organizational agility as well. This is the step that touches and affects all processes and people in the company and is different from changing a mindset. This is the “technical” part of the transition which is indeed inevitable if a company wants to work with agile development. To make it in a successful way is one of the most challenging obstacles that management level can face.

The results are concluded and narrowed down for a better understanding, in the following table.

| Challenge degree | Developer challenges | Management challenges |
|--------------------|--|--|
| Least challenging | Tools, Organizational Agility, Decision making | Decision making, Tools |
| Middle challenging | Communication, Documentation | Communication, Documentation |
| Most challenging | PM role, Change in mindset | PM role, Change in mindset, Organizational agility |

Table 6: Extent of challenges

5. Discussion

The purpose of my study is to provide a holistic overview of the challenges that developers and managers face during a transition to agile development. To fulfil this purpose, I conducted interviews and made a case study on this phenomenon. The challenges are in the following dimensions: communication, PM role, change in mindset, organizational agility, decision making, documentation, tools. The analysis from the previous section, shows that in line with previous research, most challenging both for developers and managers are: changing the mindset and the PM role, with addition of organizational agility for the managers. My findings confirm that tools and decision making are not challenging. The results also show some additional challenges that are not defined in the literature. I filled the knowledge gap by contributing additional transition challenges in the following dimensions: transparency, planning and middle management.

5.1 Results discussion

The number of total interviews for my study is 15 of which: Seven are Scrum masters or agile methodologies coaches and eight are developers who currently work with an agile methodology, namely Scrum. Most of the interviewed developers have no previous experience with traditional or agile methodologies and they are introduced to agile development in their companies as it is. Nevertheless, everyone knows the aspects of waterfall method and what is expected in traditional workflow and how it differs from the agile development workflow.

There are numerous articles in the field of agile development, but a few are in relation to the topic of my thesis (Boehm & Turner, 2005; Highsmith, 2002; Laanti et al., 2011; Nerur et al., 2005). The best peer-reviewed articles that other researchers have talked about, also inspire some concepts in my study (Drury, Conboy, & Power, 2012; Iivari & Iivari, 2011). After conducting interviews and analysing them as well as documentation, there are seven themes and 14 sub-themes. The thesis also goes in more details to some of the suggested notions and has contributed additional knowledge to them. Additionally, some of my findings oppose other findings from the literature that inspired the main challenges that I identified. In addition, I also contributed with new concepts that are suggested as further research in one of the studied articles.

While the literature has identified communication, PM role, organizational agility, decision making, change in mindset, documentation and tools as challenges during transition to agile methodologies, my study has identified transparency, planning and middle management as additional challenges. In the following paragraphs, we can see the findings of my thesis against the main articles that are used. A detailed description is provided with explanations how and with what the articles inspired my study.

A number of studies have found challenges related to communication during the transition to agile development (Gandomani et al., 2013); (Conboy et al., 2010). The existing literature discusses the challenges between developers and clients and developers and project managers (Pikkarainen et al., 2008; Damian et al. 2000). In the existing research, the authors acknowledge obstacles between these groups when companies transit to agile development from traditional methodologies. My study confronts these findings and is in contrast with the findings in the existing literature. The results from my thesis show that communication is necessary for agile development and that social skills are important. A successful transition

requires communicative team members and managers. The respondents in my study do not find communication as a challenge, but more as a requirement.

In relation to the project manager role, the literature shows that there are several aspects that are considered as challenging, for example that the transition to agile methodologies leads to loss of control that managers are used to in traditional settings. There are challenges in the field of balancing control and agility (Cobb, 2011) and focusing on leadership and collaboration (Gandomani et al., 2013). My study confirms that; indeed, this is challenging for managers which have to take part in this change. Furthermore, I provide additional insights related to the PM role such as involving planning and having clear requirements. When experiencing transition, managers are expected to increase the transparency in the teams and facilitate change to reach agility and be more adaptive to the constantly changing environment. The respondents in this research point this concept to be one of the most challenging ([See Table 6](#)).

Organizational agility is an issue with great importance according to the findings from the literature. This challenge deals with changing the whole organizational culture and internal relationships (Cohn & Ford, 2003). Market capitalizing and operational adjustment are concepts that companies must face during a transition (Lu & Ramamurthy, 2011) and they often are perceived as challenging. Companies that work in traditional settings are efficient and plan- driven and the respondents from my study confirm, that reaching agility is one of the most challenging factors. The results show additionally, that a preferred practice is to change the processes first, followed by the mindset and that this transformation should start upside- down. The respondents stress the importance of persuading the middle management and that this part of the transition is the most relevant element. The transition to agile methodologies should be done by the whole company, meaning that it should not be mixed and used whenever it is convenient. I conclude that when an organization wants to reach agility, it should be done in every department and not in separate projects that use for example waterfall methodology and Scrum or have “waterfall- ish” elements in the Scrum project. Middle management is the hardest to convince and suffers the most from the transition.

Another challenge identified in the literature is changing the mindset (Drury et al., 2012). The findings state that changing the mindset is one of the hardest steps to a transition and my study confirms this claim. This change affects the company on both organizational level (Ngo-Ye & Ahsan, 2005; Sambamurthy et al., 2003) and team level (Begel & Nagappan, 2007). The researchers suggest that it is necessary to start the transition by changing the mindset first, but the results in the thesis state that this is not necessarily mandatory. The beginning of the change can start by changing the processes first as well. The results in my study show, that this is especially hard for managers, because it is their task to start the change and persuade the whole organization. They must provide training for people with non- agile development background and assure support whenever the company faces a problem with the transition. It is also their task to grow the transparency within the development team and eliminate the trust issues between them. Another factor that they must deal with is minimizing the chaos in the beginning of the transition.

Research suggests, that decision making is a challenge during a transition to agile methodologies (Drury et al., 2011; Lindstrom & Jeffries, 2004), but my study results are in contrast with these findings. The companies face challenges in decision making on strategic (Chandler, 1997) and tactical level (Drury et al., 2011). The respondents agree that decision

making is a collaborative process which involves the whole team. These values are in contrast with the waterfall methodology workflow, because decisions are made in advance and the team is facing much more freedom and responsibilities with agile development. Decision making is a collaborative effort connected to many other concepts like communication, transparency and planning. The results in my thesis suggest that collaboration should be highly encouraged, and that self-organization helps to nurture decisions within the team and increases more responsible actions. Whenever there are ambiguities and issues, they should be discussed in advance to prevent possible failures.

Documentation is claimed to be one of the main challenges during a transition from researchers in the agile development field (Knippers, 2011; Jones and Bonsignour, 2011). The literature suggests that lack of documentation is a major challenge for non- agile methodologies companies (Larman & Vodde, 2010) and that incompetent maintenance should be taken more seriously from software development companies (Jones, 2000; Boehm, 1987). My thesis provides the opposite results. Documentation is not considered as challenging and managers and developers find new and innovative ways to deal with this obstacle without turning to “old” waterfall methodology habits but also preserving the most important details of a project. During a transition, it is more important to communicate the change, instead of dealing with documentation.

Choosing tools is considered as a challenge, according to some studies (Nerur et al., 2005; Sircar & Nerur, 2001), and that choosing tools during a transition requires organizational changes as well (Nerur et al., 2005). My thesis found that this concept is one of the least challenging and not a threat during a transition. The results show, that respondents prefer exploring the tools on their own and having their own experience. This rarely requires new skillsets and managers suggest doing a pilot test before using a new tool. This way developers can estimate if the suggested tool fits their needs and capabilities and if not - they are free to self-explore new software.

In the following table, we can see a clear visualisation of the challenges identified in the literature, compared to the results from the interviews. We can see that according to the results, only PM role, change in mindset and Organizational agility are considered to be most challenging. The rest of the concepts are not seen as challenging for the respondents.

| Challenge | Existing Literature | Results |
|------------------------|---------------------|---------|
| Communication | ✓ | X |
| PM role | ✓ | ✓ |
| Change in mindset | ✓ | ✓ |
| Organizational agility | ✓ | ✓ |
| Decision making | ✓ | X |
| Documentation | ✓ | X |
| Tools | ✓ | X |

Table 7: Summary of challenges from literature compared to results (✓- IS a challenge; X- NOT a challenge)

5.2 Implications for practice

In the last question for both groups, the respondents are asked what in their opinion is most challenging and which obstacles future managers should think of when a transition is deployed. The respondents gave clear answers from practical perspective.

It is suggested that future managers should be able to create an appropriate agile development environment when they make a transition and to teach them to change and adapt. Coaching is a big part of any transition and using professional help can help convince the whole organization of the benefits of this huge change- especially middle management which is the toughest obstacle. It is also suggested that iteration should be there at all time and that there should be an innovative way to plan features. Nevertheless, no developers have the same skills, so it would be good for managers to find a way to estimate the skills of every developer in the team. There might be cases when some of them are much more behind others and this can slow down the project and create tension between the members. Managers should be able to make teams comfortable and increase decision making as a common activity. Transparency and flexibility should be kept always in order to fit the team needs.

Finally, when an organization wants to transit to agile methodologies, everyone in it should be aware of both benefits and challenges.

5.2.1 Methods discussion

A structured interview and review of documents were chosen to get a deeper understanding of the problem of my study. What is beneficial about this approach is that I can gather more valuable data and a deeper understanding of the problem at hand in order to reach a generalizable conclusion which is applicable and consistent.

The case study provides a general understanding of the research topic. The interview questions are designed based on an interview guide which can help provide replication for future research. The questions are aiming to understand the respondents general attitude to agile development before he/she started working with this methodology. The respondents consisted of two groups - developers and managers (Scrum masters and agile development coaches) and most of them have relevant experience with agile development and are major representatives in their field of work. The interviewees do not necessarily have the same opinion and experience and I witnessed a division on some questions, but in general all of them agree on the benefits of agility. They provided deeper insights into the topic by showing challenges that I did not identify in the literature. We can see now clearly which challenges from the literature are really challenging for both groups and which ones can be added as companion.

The documents provided by some of the interviewees became a help to additionally understand how the whole transition process affects the team members and managers. By analysing the documentation, we could see more clearly how their meetings are conducted and which challenges they meet.

In addition, only one semi-structured interview was conducted since it was more convenient to both parties. This form of interview allows to be more agile and flexible in the sense that I can ask more questions connected to the responses. This gave even better and unexpected results which provide a great contribution to my thesis. The questions are specifically designed to the respondent background and experience.

The interview questions were based on the literature review and interview guide. After receiving the answers, the content analysis is done from which seven themes are created and 14 corresponding sub-themes. They provided additional information and I can contribute to research even more and fill in the knowledge gap identified in the literature.

In all interviews, the chosen language was English even though this was not the mother tongue for neither the interviewer nor the interviewees. Nevertheless, the questions were

designed according to their proficiency and if there are inconsistencies, additional communication is needed to clear it out. This way I can extract all the needed information and the ideas we perceived in a clear matter.

6. Conclusion

The purpose of my thesis is to investigate the challenges that managers and developers face during a transition to agile development and create a holistic overview of those challenges. The literature suggests that there are seven challenges to be faced during a transition: communication, PM role, decision making, organizational agility, change in mindset, documentation and tools. The results from the conducted interviews shows that only three of those concepts are challenging for both groups: Change in mindset, PM role and organizational agility. In addition to those, there are more challenges that the respondents think are important: transparency, planning and middle management.

Limitations

In the beginning of the study, I worked hard on the idea to do a case study based on a Swedish company, but due to lack of consistent communication, the study changed its course. New interviewees are gathered, but this required more time which lead to a later delivery of my study.

A limitation of the study is that some of the respondents did not go through a transition from a previous methodology and they met the agile methodologies in the company as it is. Because of this, there is a missing information (blank answers) on the questions related to the transition to agile development. This means that I could only analyse the few who did go through a transition.

I managed to turn these limitations into positive characteristics for my study. The interviews are coming from people from different countries and various company cultures and sizes. This shows a greater variety of the transitioning conditions which gives us a broader perspective of the transition phenomenon. This opens doors for future researchers and gives ideas for them which can be used to contribute to knowledge and provide more empirical studies.

Suggestions for future research

Agile development is growing stronger every day and there are many points to be researched. From the knowledge gathered in my study, I derived several points that require attention for future research. First, it would be beneficial to provide more empirical evidence to trace the agile methodologies influence before and after the transition. Second, researchers should investigate each phase of an agile methodologies transition and provide a detailed list of every challenge that can be identified. Third, find the right balance between planning and documentation. Fourth, provide a study that traces how managers can predict the approximate time to develop a feature and calculate estimated price of a project. Freelance developers need an agile methodology that helps them organize themselves and more information in this field is needed. And finally, researchers should consider conducting a quantitative study and provide more challenges in depth.

References

- Agarwal, R. & V. Sambamurthy, V. (2001). Principles and Models for Organizing the IT Function. *MIS Quarterly Executive*, 1(1):1-16.
- Ahmad, M. O., Markkula, J., & Oivo, M. (2013). Kanban in software development: A systematic literature review. *39th Euromicro Conference on Software Engineering and Advanced Applications*, (September), 9–16. <https://doi.org/10.1109/SEAA.2013.28>
- Anderson D. (2003). *Agile management for software engineering, applying the theory and constraints for business results*. Prentice Hall, Upper Saddle River, NJ.
- Beck, K. (2000). *Extreme Programming Explained: Embrace Change*. Reading, Mass.: Addison-Wesley.
- Begel, A., & Nagappan, N. (2007). Usage and Perceptions of Agile Software Development in an Industrial Context: An Exploratory Study. In *First International Symposium on Empirical Software Engineering and Measurement (ESEM 2007)* (pp. 255–264). IEEE. <https://doi.org/10.1109/ESEM.2007.12>.
- Boehm, B. (1981). *Software Engineering Economics*, Prentice Hall.
- Boehm, B. W. (1987) Improving software productivity. *IEEE Computer*, 20, 9, 43-57.
- Boehm, B., & Turner, R. (2005). Management challenges to implementing agile processes in traditional development organizations. *IEEE Software*, 22(5), 30–39. <https://doi.org/10.1109/MS.2005.129>.
- Boehm, B., Turner, D. (2003). Using risk to balance agile and plan-driven methods. *IEEE Comput* 36(6):57–66.
- Cervone, H. F. (2011). Understanding agile project management methods using scrum. *International Digital Library Perspectives*, 27(1), 18–22. <https://doi.org/10.1108/10650751111106528>.
- Ceschi, M., Sillitti, A., Succi, G., De Panfilis, S. (2005). Project management in plan-based and agile companies. *Software, IEEE*, 22 (3):21-27.
- Chandler, D. (1997). *Strategy and Structure in Resources, firms, and strategies: A reader in the resource-based perspective*, N. J. Foss, Ed., ed Oxford: Oxford University Press, 1997, pp. 40-51.
- Cho, J. J. (2010). An Exploratory Study on Issues and Challenges of Agile Software Development with Scrum. *Issues in Information Systems*, 9(2), 599.
- Cobb, C. G. (2011). *Making sense of agile project management: balancing control and agility*. Wiley. Retrieved from: <https://books.google.de/books?hl=en&lr=&id=NtixCAsvT0C&oi=fnd&pg=PT7&dq=>

balance+control+and+agility&ots=_Hxuh5--
Et&sig=cDs4uu3Lv2w3VxU1BEs4X6O53_Y&redir_esc=y#v=onepage&q=balance
control and agility&f=false

Cohn, M., & Ford, D. (2003). Introducing an agile process to an organization. *Computer*, 36(6), 74–78. <https://doi.org/10.1109/MC.2003.1204378>

Conboy, K., Coyle, S., & Wang, X. (2010). *People over Process: Key Challenges in Agile Development*, (99), 48–57. Retrieved from
[https://aran.library.nuigalway.ie/bitstream/handle/10379/1419/IEEE Software July 2010.pdf?sequence=1](https://aran.library.nuigalway.ie/bitstream/handle/10379/1419/IEEE_Software_July_2010.pdf?sequence=1)

Dahlbom, B. & Mathiassen, L. (1993). *Computers in context: The philosophy and practice of systems design*. USA: Wiley-Blackwell.

Damian, D., Eberlein, A., Shaw, ML., Gaines, BR. (2000). Using different communication media in requirements negotiation. *IEEE Softw* 17(3):28–36.

David, J., Anderson (2010). *Kanban: Successful Evolutionary Change for Your Technology Business. Sequim*. WA: Blue Hole Press, 2010.

Dingsøyr, T., Nerur, S., Balijepally, V., & Moe, N. B. (2012). A decade of agile methodologies: Towards explaining agile software development. *Journal of Systems and Software*, 85(6), 1213–1221. <https://doi.org/10.1016/j.jss.2012.02.033>.

Dove, R. (2001). *Response Ability: The Language, Structure, and Culture of the Agile Enterprise*, New York: John Wiley & Sons.

Drury, M., Conboy, K., & Power, K. (2011). Decision making in agile development: A focus group study of decisions & obstacles. *Proceedings - 2011 Agile Conference, Agile 2011*, 39–47. <https://doi.org/10.1109/AGILE.2011.27>.

Drury, M., Conboy, K., & Power, K. (2012). Obstacles to decision making in Agile software development teams. *Journal of Systems and Software*, 85(6), 1239–1254. <https://doi.org/10.1016/j.jss.2012.01.058>.

Dybå, T. and Dingsøyr, T. (2008). Empirical studies of agile software development: A systematic review. *Information and Software Technology*, 50(9-10), pp.833-859.

Fitzgerald, B. et al., (2006). Customising Agile Methods to Software Practices. *European Journal of Information Systems*, 15, pp. 197-210.

Fitzgerald, B. (1996). Formalized systems development methodologies: A critical perspective. *Information Systems Journal*, 6 (1): 3-34.

Galal-Edeen, G.H., Riad, A.M. & Seyam, M.S. (2007). Agility versus discipline: Is reconciliation possible? *International Conference on Computer Engineering & Systems, ICCES'07*, 331-337.

Gandomani, T. J., Zulzalil, H., Azim, A., & Ghani, A. (2014). How Human Aspects Impress Agile Software Development Transition and Adoption. *International Journal of*

Software Engineering and Its Applications, 8(1), 129–148.
<https://doi.org/10.14257/ijseia.2014.8.1.12>.

Gandomani, T. J., Zulzalil, H., Ghani, A. A. A., Sultan, A. B. M., & Nafchi, M. Z. (2013). Obstacles in moving to agile software development methods; At a Glance. *Journal of Computer Science*, 9(5), 620–625. <https://doi.org/10.3844/jcssp.2013.620.625>.

Gbrich, C. (2007). *Qualitative Data Analysis: An Introduction* (1st edn). London: Sage Publications.

Graneheim, U. H., & Lundman, B. (2004). Qualitative content analysis in nursing research: Concepts, procedures and measures to achieve trustworthiness. *Nurse Education Today*, 24(2), 105–112. <https://doi.org/10.1016/j.nedt.2003.10.001>.

Heeager, L. (2012). Introducing Agile Practices in a Documentation-Driven Software Development Practice: A Case Study. *Journal of Information Technology Case and Application Research*, 14(1), 3–24. <https://doi.org/10.1080/15228053.2012.10845693>.

Higsmith, J. A. (2002). *Agile software development ecosystems*. Addison-Wesley. Retrieved from: https://books.google.de/books?hl=en&lr=&id=uE4FGFOHs2EC&oi=fnd&pg=PP17&dq=Higsmith+2002&ots=ONTLqy8q0x&sig=7tiLuXVzzf_7fSXML4BN5iQoDdM&redir_esc=y#v=onepage&q=Higsmith+2002&f=false.

Hoboken, NJ. McAvoy, J., Butler, T. (2009). The role of project management in ineffective decision making within agile software development projects. *European Journal of Information Systems*, 18 (4):372-383.

Holmström, H., Fitzgerald, B., Agerfalk, P.J., Conchuir, E.O. (2006). Agile practices reduce distance in global software development. *Information Systems Management*, 23(3):7–18.

Home - Agile Coaching Institute. (2016). Retrieved September 18, 2017, from: <http://agilecoachinginstitute.com/#ourthinking>.

Iivari, J., & Iivari, N. (2011). The relationship between organizational culture and the deployment of agile methods. *Information and Software Technology*, 53(5), 509–520. <https://doi.org/10.1016/j.infsof.2010.10.008>.

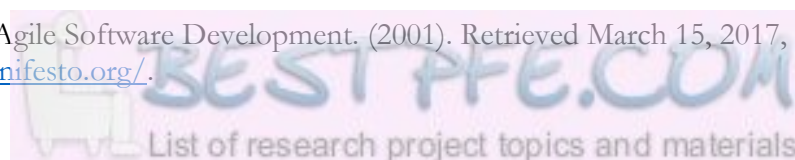
Jalali, S., & Wohlin, C. (2010). Agile Practices in Global Software Engineering – A Systematic Map. Retrieved 26 August 2017 from <http://www.diva-portal.org/smash/get/diva2:835341/FULLTEXT01.pdf>.

Jones, C. (2000). *Software Assessments, Benchmarks, and Best Practices*. Addison-Wesley Professional, Upper Saddle River, NJ.

Jones, C. and Bonsignour, O. (2011). *The Economics of Software Quality*. Addison-Wesley Professional, Upper Saddle River, NJ.

Kennaley, M. (2010). *SDLC 3.0: Beyond a Tacit Understanding of Agile*, Fourth Medium Press.

- Khalil, C., & Khalil, S. (2016). A Governance Framework for Adopting Agile Methodologies. *International Journal of E-Education, E-Business, E-Management and E-Learning*, 6(2), 111–119. <https://doi.org/10.17706/ijeeec.2016.6.2.111-119>.
- Kniberg, H. (2009). *Kanban vs Scrum – how to make the most of both*, Retrieved on 29 August 2017, via: www.crisp.se/file-uploads/Kanbanvs-Scrum.pdf.
- Knippers, D. (2011). Agile Software Development and Maintainability. *Agile Software Development and Maintainability*. Retrieved from <http://referaat.cs.utwente.nl/conference/15/paper/7269/agile-software-development-and-maintainability.pdf>.
- Koskela, J. (2003). Software configuration management in agile methods, VTT publication 5 14, ESPOO, pp I-54.
- Kumar, R. (2011). *Research Methodology* (Third). SAGE Publications Inc.
- Laanti, M., Salo, O., & Abrahamsson, P. (2011). Agile methods rapidly replacing traditional methods at Nokia: A survey of opinions on agile transformation. *Information and Software Technology*, 53(3), 276–290. <https://doi.org/10.1016/j.infsof.2010.11.010>.
- Larman, C. and Vodde, B. (2010). *Practices for Scaling Lean Agile Development: Large, Multisite, and Offshore Product Development with Large-Scale Scrum*. Addison-Wesley Professional, Upper Saddle River, NJ.
- Larman, C., & Basili, V. R. (2003). Iterative and incremental developments. a brief history. *Computer*, 36(6), 47–56. <https://doi.org/10.1109/MC.2003.1204375>.
- Leffingwell, D., & Muirhead, D. (2004). *Tactical Management of Agile Development: Achieving Competitive Advantage* (No. 62204). Colorado. Retrieved from <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.495.4432&rep=rep1&type=pdf>.
- Levy, Y., & Ellis, T. J. (2006). A systems approach to conduct an effective literature review in support of information systems research. *Informing Science*, 9, 181–211. <https://doi.org/10.1049/cp.2009.0961>
- Lindstrom, L., Jeffries, R. (2004). Extreme Programming and Agile Software Development Methodologies. *Information Systems Management*, 21, 41-52.
- Lu, Y., & Ramamurthy, K. R. (2011). Understanding the link between Information Technology capability and organizational agility: An empirical examination 1, 35(4), 931–954.
- Mahanti, A. (2004). Challenges in enterprise adoption of agile methods - A survey. *Journal of Computing and Information Technology*, 14(3), 197-206.
- Malone, T., Crowston, K. (1994). The interdisciplinary study of coordination. *ACM Comput Surv* 26(1):87–119.
- Manifesto for Agile Software Development. (2001). Retrieved March 15, 2017, from <http://agilemanifesto.org/>.



- Nerur, S., Mahapatra, R., & Mangalaraj, G. (2005). Challenges of Migrating to Agile Methodologies. *Communications of the ACM*, 48(2), 72–78.
<https://doi.org/10.1145/1060710.1060712>.
- Ngo-Ye, L., & Ahsan, M. (2005). Enterprise it application systems agility and organizational agility. *11th Americas Conference on Information Systems (AMCIS 2005)*, (August 2005), 2146–2151. Retrieved from <http://aisel.aisnet.org/amcis2005/159/>.
- Ozkaya, I., Nord R.L., Bellomo, S., Brayer, H. (2013). Beyond Scrum + XP: Agile Architecture Practice. Disciplined Agile Delivery in the Enterprise. *The Journal of Information Technology Management (Cutter IT Journal)*. 2013, 26, (6).
- Paasivaara, M., Lassenius, G. (2003). Collaboration in inter-organizational software development. *Softw Process Improv Pract* 8(4):183–199.
- Parnas, D. (2006). Agile methods and GSD: The wrong solution to an old but real problem. *Communication of the ACM*, 49(10):29- 12.
- Pikkarainen, M. & Mantyniemi, A. (2006). An approach for using CMMI in agile software development assessments: experiences from three case studies. *SPICE 2006 conference*, 4-5.
- Pikkarainen, M., Haikara, J., Salo, O., Abrahamsson, P., & Still, J. (2008). The impact of agile practices on communication in software development. *Empirical Software Engineering*, 13(3), 303–337. <https://doi.org/10.1007/s10664-008-9065-9>.
- Polit, D.F., Hungler, B.P., (1999). *Nursing Research. Principles and Methods*, sixth ed. J.B. Lippincott Company, Philadelphia, New York, Baltimore.
- Russo, J.E., Schoemaker, P., (1989). *Decision Traps: The Ten Barriers to Brilliant Decision-Making and How to Overcome Them*. Simon & Schuster, NY, USA.
- Sambamurthy, V., Bharadwaj, A., and Grover, V. (2003). Shaping Agility through Digital Options: Reconceptualizing the Role of Information Technology in Contemporary Firms, *MIS Quarterly* (27:2), pp. 237-263.
- Saunders, M., Thornhill, A., & Lewis, P. (2012). *Research methods for business students* (Sixth). Pearson.
- Saunders, M., Lewis, P., & Thornhill, A. (2009). *Research Methods for Business Students*. *Research methods for business students* (Fifth). Prentice Hall.
- Schmidt, J.B., Montoya-Weiss, M.M., Massey, A.P., (2001). New product development decision-making effectiveness: comparing individuals, face-to-face teams, and virtual teams. *Decision Sciences* 3232, 575–600.
- Schultze, U., & Avital, M. (2011). Designing interviews to generate rich data for information systems research. *Information and Organization*, 21(1), 1–16.
<https://doi.org/10.1016/j.infoandorg.2010.11.001>.
- Schwaber, K. (2004). *Agile project management with Scrum*. Redmond, WA: Microsoft Press.

Schwaber, K., Laganza, G., D'Silva, D. (2007). *The Truth about Agile Processes: Frank Answers to Frequently Asked Questions*. Forrester Report.

Sircar, S., Nerur, S.P., and Mahapatra, R. (2001). Revolution or Evolution? A Comparison of Object-Oriented and Structured Systems Development Methods. *MIS Quarterly* 25, 4 (Dec. 2001), 457–471.

Stazinger, J. W., Jackson, R. B., & Burd, S. D. (2005). *Object-oriented analysis & design with unified process*. Boston: Thomson Course-Technology.

Stelzer D., Mellis W. (1998). Success factors of organizational change in software process improvement. *Software Process Improvement and Capability* 4(4):227–250.

Sutherland, J., Viktorov, A., Blount, J., & Puntikov, N. (2007). Distributed scrum: Agile project management with outsourced development teams. *Proceedings of the Annual Hawaii International Conference on System Sciences*, 1–10. <https://doi.org/10.1109/HICSS.2007.18.0>

Svensson, H., Höst, M. (2005). Introducing an agile process in a software maintenance and evolution organization, in: *Proceedings of 9th European Conference of Maintenance and Reengineering*. (CSMR 2005), pp. 256–264.

Taylor, K. (2015). Adopting Agile software development: the project manager experience. *ITP* 29,4 670: <https://doi.org/10.1108/ITP-02-2014-0031>.

Turner, R. (2003). People factors in software management: lessons from comparing agile and plan-driven methods. *J Def Softw Eng* 22(4):4–8.

Unisys, (2004). *When Agility Outweighs Efficiency: BPO Viewpoint*. Retrieved April 4th 2017 from: <http://www.outsourcingrequests.com/center/jsp/requests/document/index.jsp?documentId=2407>

Vaismoradi, M., Turunen, H., Bondas, T. (2013). *Content analysis and thematic analysis: Implications for conducting a qualitative descriptive study*, 398–405. <https://doi.org/10.1111/nhs.12048>.

Volberda, H. W. (1996). Toward the Flexible Form: How to Remain Vital in Hypercompetitive Environments, *Organization Science* (7:4), pp. 359-374.

Volberda, H. W. (1997). Building Flexible Organizations for Fast- Moving Markets, *Long Range Planning* (30:2), pp. 169-183.

Webster, J., & Watson, R. T. (2002). Analyzing the past to prepare for the future : Writing a literature review Reproduced with permission of the copyright owner . Further reproduction prohibited without permission . *MIS Quarterly*, 26(2), xiii–xxiii. <https://doi.org/10.1.1.104.6570>

Wheeler, B.C., Valacich, J.S., (1996). Facilitation, GSS, and training as sources of process restrictiveness and guidance for structured group decision making: an empirical assessment. *Information Systems Research* 7, 429–450.

Williams, L., & Cockburn, A. (2003). Agile software development: It's about feedback and change. *IEEE Computer*, 36(6), 39-43.

Yin, R. K. (2003). *Case study research. design and methods.pdf*. London: Sage Publications.

Zannier, C., Maurer, F. (2006). Foundations of Agile Decision Making from Agile Mentors and Developers in Extreme Programming and Agile Processes. *Software Engineering*, vol. 4044, P.

Zannier, C., Maurer, F. (2007). Comparing Decision Making in Agile and Non-agile Software Organizations, in Agile Processes. *Software Engineering and Extreme Programming*, 4536, G. Concas, et al., Eds., ed: Springer Berlin / Heidelberg, 2007, pp. 1- 8.

Figures

Figure 1: Communication participants in an agile development project



Figure 2: Concept model of holistic overview of transitional challenge

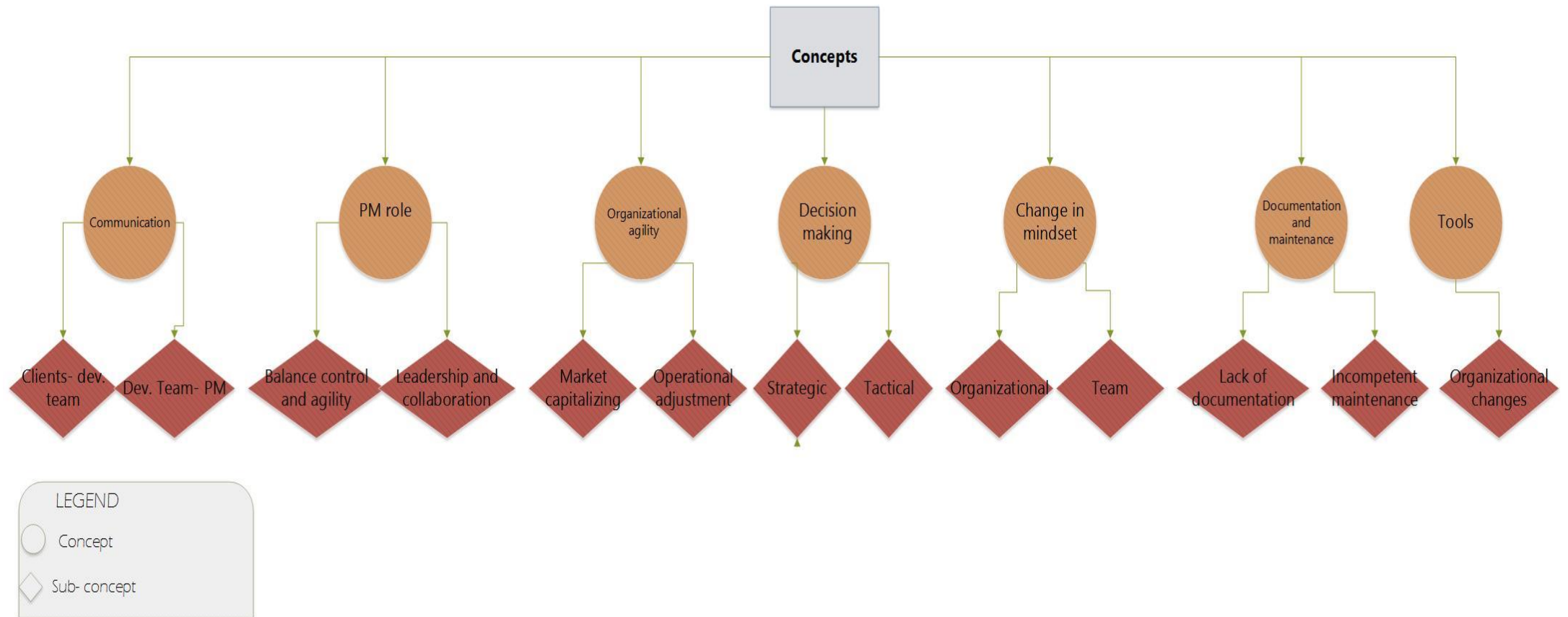
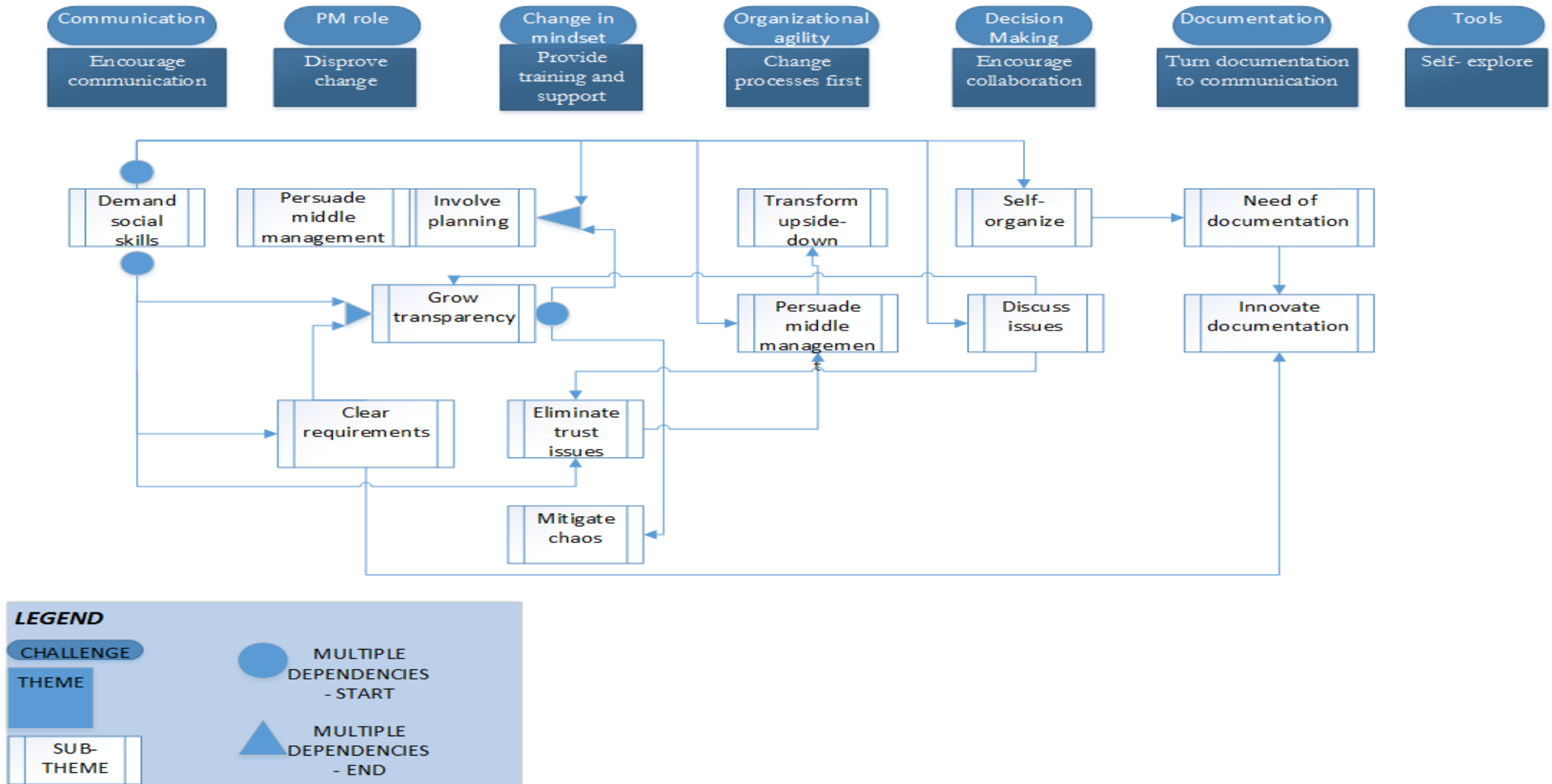


Figure 3: Concept model of themes and sub- theme dependencies



Tables

Table 1: Summary of sub- concepts

| Concept | Sub- concept | |
|-------------------------------|-----------------------------|------------------------------|
| <u>Communication</u> | clients- dev. Team | dev. Team- PM |
| <u>PM role</u> | balance control and agility | leadership and collaboration |
| <u>Organizational agility</u> | market capitalizing | operational adjustment |
| <u>Decision making</u> | strategic | tactical |
| <u>Change in mindset</u> | organizational | team |
| <u>Documentation</u> | lack of documentation | incompetent maintenance |
| <u>Tools</u> | organizational changes | X |

Table 2: Summary of interviewee positions and data collection

| Interviewee | Role in organization | Data collection |
|-------------|---------------------------------------|---------------------------------|
| A | Scrum master | Structured interview |
| B | Software developer | Structured interview |
| C | Web developer | Structured interview |
| D | Agile development coach/ Scrum master | Structured interview |
| E | Agile development coach/ Scrum master | Semi-structured interview |
| F | Project manager | Structured interview |
| G | Software engineer | Structured interview |
| H | Computer science student | Structured interview |
| I | Software engineer | Structured interview |
| J | Software developer | Structured interview |
| K | computer science engineer | Structured interview |
| L | Software developer/ Scrum master | Structured interview |
| M | Software engineer | Structured interview, Documents |
| N | Scrum Master | Structured interview, Documents |
| O | Scrum Master | Structured interview, Documents |

Table 3: Developer challenges

| Interviewee | Communication | PM role | Change in mindset | Organizational agility | Decision making | Documentation | Tools |
|-------------|---------------|---------|-------------------|------------------------|-----------------|---------------|-------|
| 1- J | ✓ | X | ✓ | X | X | X | X |
| 2- I | X | ✓ | ✓ | ✓ | X | ✓ | X |
| 3- C | X | X | X | X | X | ✓ | X |
| 4- B | X | ✓ | X | X | X | X | X |
| 5- H | X | X | X | X | ✓ | X | X |
| 6- G | X | X | ✓ | X | X | ✓ | X |
| 7- K | X | ✓ | X | X | X | X | X |
| 8- M | ✓ | ✓ | ✓ | X | X | X | X |
| Total: | 2 | 4 | 4 | 1 | 1 | 3 | 0 |

Table 4: Manager challenges

| Interviewee | Communication | PM role | Change in mindset | Organizational agility | Decision making | Documentation | Tools |
|-------------|---------------|---------|-------------------|------------------------|-----------------|---------------|-------|
| 1- F | X | X | X | X | X | ✓ | ✓ |
| 2- D | X | ✓ | ✓ | ✓ | X | X | X |
| 3- A | ✓ | ✓ | ✓ | ✓ | X | X | ✓ |
| 4- E | ✓ | ✓ | ✓ | ✓ | ✓ | X | X |
| 5- L | ✓ | ✓ | ✓ | X | X | ✓ | X |
| 6- N | ✓ | ✓ | ✓ | ✓ | X | ✓ | X |
| 7- O | X | X | X | ✓ | X | X | X |
| Total: | 4 | 5 | 5 | 5 | 1 | 3 | 2 |

Table 5: Summary of themes and sub- themes

| Challenges | Theme | Sub- theme |
|------------------------|--|-----------------------------------|
| Communication | <i>encourage communication</i> | <i>demand social skills</i> |
| PM role | <i>facilitate change</i> | <i>involve planning</i> |
| | | <i>grow transparency</i> |
| | | <i>clear requirements</i> |
| Change in mindset | <i>provide training and support</i> | <i>grow transparency</i> |
| | | <i>clear requirements</i> |
| | | <i>eliminate trust issues</i> |
| | | <i>mitigate chaos</i> |
| Organizational agility | <i>change processes first</i> | <i>transform upside- down</i> |
| | | <i>persuade middle management</i> |
| Decision making | <i>encourage collaboration</i> | <i>self- organize</i> |
| | | <i>discuss issues</i> |
| Documentation | <i>turn documentation to communication</i> | <i>need of documentation</i> |
| | | <i>innovate documentation</i> |
| Tools | <i>self- explore</i> | X |

Table 6: Extent of challenges

| Challenge degree | Developer challenges | Management challenges |
|--------------------|--|--|
| Least challenging | Tools, Organizational Agility, Decision making | Decision making, Tools |
| Middle challenging | Communication, Documentation | Communication, Documentation |
| Most challenging | PM role, Change in mindset | PM role, Change in mindset, Organizational agility |

Table 7: Summary of challenges from literature compared to results (✓- IS a challenge; X- NOT a challenge)

| Challenge | Existing Literature | Results |
|------------------------|---------------------|---------|
| Communication | ✓ | X |
| PM role | ✓ | ✓ |
| Change in mindset | ✓ | ✓ |
| Organizational agility | ✓ | ✓ |
| Decision making | ✓ | X |
| Documentation | ✓ | X |
| Tools | ✓ | X |

Appendices

Appendix A: Interview guide for managers

- Your background?
- Your position in the company?
- Agile development in your company
 - Your involvement?
 - How was the process initiated?
 - Who took part in it?
 - Number of teams?
 - Size of teams?
 - Which methodologies they use?
- The challenges
 - Communication
 - Formal or informal communication?
 - Importance of social skills
 - Importance of communication
 - Communication between teams and customers?
 - Between team and PM/
 - Between PM and customers?
 - Any communication problems?
 - PM role
 - Role of PM in agile methodologies?
 - What changes are met from transition to scrum master?
 - Role of scrum master?
 - Difference in responsibilities?
 - Challenges from traditional transition?
 - Less planning contribution?
 - Difference between scrum master and PM?
 - Change in mindset
 - Company perception about agile development
 - Previous experience with waterfall methodology in the teams?
 - Implemented changes?
 - Company reaction?
 - Needed special skillsets?
 - Organizational agility
 - Previous methodologies?
 - Organizational changes?
 - Affected departments?
 - Special techniques?
 - Decision making
 - Reduced power of PM?

- Shifting roles?
- Change in team responsibilities?

Documentation and maintenance

- Documentation handling?
- Affect on final projects?
- What is working well and not so well?
- Combination of agile- waterfall methodology benefit?

Tools and technologies

- Used tools?
 - Variety of tools?
 - New skillsets?
 - Affect on workflow?
 - Which were the barriers?
 - Choice of right tools?
- Biggest challenge for you?
 - Suggestions for future managers?
 - Recommendations on who to interview?

Appendix B- Interview Questions for managers

1. Could you introduce yourself?
2. Why did your company decide to implement agile development?
3. Could you describe the transition process in your company?
4. How were you involved in the adoption of agile development?

The challenges

Communication

5. Were there communication issues between you and the teams and how did you overcome them?
6. How important are social skills?

PM role

7. Which issues do managers face when they transit from waterfall methodology to agile development?

Change in mindset

8. Were there people that have worked with traditional methodologies before? How did they meet the agile methodologies transition?
9. How did you integrate the new mindset?

Organizational agility

10. How were organizational changes implemented?
11. How were the other departments affected while the changes were happening?

Decision making

12. How were responsibilities different when you shifted from traditional to agile methodologies?
13. How did roles shift inside the team?
14. Who made decisions inside the team?

Documentation and maintenance

15. How do agile development teams in your company deal with documentation?
16. How does a combination of agile and traditional methodologies benefit a project?

Tools and technologies

17. How does change of tools affect the workflow?
18. Which challenges were met in the introduction of new agile development tools?
19. How do you choose the right tools?

Closing

20. Which was the biggest challenge for you?
21. Do you think there are more obstacles that future managers and scrum masters should think of?
22. Who would you recommend for an interview?

Appendix C: Interview guide for development team

Your background?

- Your position in your company?
- Agile development in your company
 - Your involvement?
 - Your part in it?
 - Team environment?
 - Which methodologies they use?
 - Prior experience with agile development?
- The challenges

Communication

- Formal or informal communication?
- Importance of social skills
- Importance of communication
- Communication between teams and customers?
- Between team and PM/
- Any communication problems?

PM role

- Vision of PM in agile development?
- Difference in responsibilities?
- Support of collaboration?
- Leadership changes?
- Changes from within the team?

Change in mindset

- Team perception about agile development?
- Previous experience with waterfall methodology in the teams?
- Implemented changes?
- Team reaction?
- Needed special skillsets?

Organizational agility

- Previous methodologies?
- Organizational changes?
- Affected departments?
- Special techniques?
- Overall reaction of colleagues?

Decision making

- Reduced power of PM?
- Leaders from within?
- Shifting roles?
- Change in team responsibilities?

- Diminishing the PM?

Documentation and maintenance

- Documentation handling?
- Affect on final projects?
- What is working well and not so well?
- Combination of agile- waterfall methodology benefit?
- How do you document?
- Is commenting enough?
- How do you introduce the work to new colleagues?

Tools and technologies

- Used tools?
 - Variety of tools?
 - New skillsets?
 - Affect on workflow?
 - Which were the barriers?
 - Choice of right tools?
- Biggest challenge for you?
 - Suggestions for future
 - Recommendations on who to interview?

Appendix D- Interview Questions for development team

1. Could you introduce yourself?
2. What is your opinion about agile development?
3. Could you describe the transition process in your team?
4. How were you involved in the adoption of agile methodologies in your company?
5. Are there teams in your company that mix methodologies? (If yes) How do they collaborate with the other teams working with agile development?
6. How did the teams react to the idea of agile development?

The challenges

Communication

7. How did communication change with your team?
8. Do you facilitate formal or informal communication within the team? Why?
9. Why are social skills important in your team?

PM role

10. What is the role of the scrum master compared to the project manager?
11. Have you encountered difference in responsibilities between traditional methodologies and agile ones? (If yes) Could you describe them?
12. Which issues do developers face when they transit from waterfall methodology to agile methodologies?
13. Did you have to go through a transition? (If yes) Could you explain the process?

Change in mindset

14. What was the general understanding of agile development in the team before it was implemented?
15. How did developers meet the agile methodologies transition?
16. What changes were implemented to integrate the new mindset within the team?
17. Which skillsets are needed to understand the mindset?

Organizational agility

18. Which organizational changes affected the development within your team?
19. How were the other teams affected while the changes were happening?

Decision making

20. Which roles shifted inside the team?
21. Who made decisions inside the team?

Documentation and maintenance

22. How do agile development teams in your company deal with documentation?
23. How does less documentation affect the final version of the project?
24. How does a combination of agile and traditional methodologies benefit a project?

Tools and technologies

25. Which tools for agile development are used in your team?
26. How did the variety of tools contribute to the work of the development team?

27. Does change of tools require new skillsets within the team?
28. How do you choose the right tools?

Closing

29. Which was the biggest challenge for you?
30. Do you think there are more obstacles that future developers within agile development should think of?
31. Who would you recommend for an interview?

Appendix E- Interviews summary

The interviews are conducted with developers and managers that work within agile development. The structure of the interviews starts with a general information about the interviewee and his/ hers experience with agile methodologies, followed by more detailed questions about every challenge identified in the literature and ending with personal reflections of the methodology.

Interviewee A

Interviewee A is a scrum master at a Swedish company. His first years he had the role as a Java developer in various projects and development teams working as a consultant. Today he works within an agile development context as a developer, scrum master or as an agile methodologies coach for 8 years.

The interview started by a short introduction of his work in the company and the situation of the agile development there. He knows the amount of advantages that agile development can bring to a company and he is “*spreading the agile mindset*” in a Swedish company. Even though there are many teams who work with agile methodologies like Scrum and Kanban, but “*We have not really decided to implement Agile*” yet and they still work with waterfall methodology for heavier projects. More details about the interview can be seen in Appendix A.

A major communication challenge for interviewee A is defining “*maturity level*” of his team, meaning that all teammates should be on the same level to make the project work. What is different here is that person A prefers his team communicating with facts and data instead of informal communication.

A great challenge that managers must face in the transition from traditional to agile methodologies is the “*fear of change*”. He argues that many roles are shifting and the responsibilities will be different from what they are used to. “*Losing control*” is also something that can be frightening to every company and it is essential that every decision should benefit the company even if that requires the changing roles inside the organization.

A final point is that the company did not implement agile methodologies completely and would like to study the opportunities through this thesis, which will help them learn the challenges beforehand the complete transition.

Interviewee B

Interviewee B is currently living in Sweden, doing his masters in Software Product Engineering and working part time as Software Developer.

He has great experience in Software engineering back in his home country, and he encountered agile development in Sweden. The interview started with his short introduction and straight to the questions about his opinion and previous experience with agile development. According to him, one of the best qualities in agile development is the “incorporation of requirements and stakeholders”. When the methodology was implemented in the company, he still did not work there which means that he cannot give a real insight of the adoption process. The teams that he works with are rather small- 2 teams with 4 members each.

As it is said before, this participant has not witnessed the transition to agile methodologies so he cannot compare the previous state of the communication processes with the current one. His company uses various communication channels for formal communication like JIRA and Slack for informal. He argues that communication is important because it is important that the stakeholders of the project give feedback and “*this cannot be done without incorporating social skills*”. When the roles change from traditional to agile methodologies, the major challenge is the difficulty to “incorporate change” according to participant B. For the transition a major challenge for the managers from the point of view of a developer, is the level of advancement of the developers and how to make them work together if they have a different level of knowledge. The participant has not witnessed the transition process so he could not give relevant information about changing the mindset of the company and the team. The participant has not witnessed the transition process so he could not give relevant information about the previous state of the organizational agility of the company.

According to interviewee B, the agile development environment requires collaboration which means that “*decisions are made through consultation*”.

In the company where participant B works, documentation is not a priority as long as it is a small amount but effective. He argues that this way the developers save time and gives the project freedom to add more features or remove unwanted ones. The company uses various tools that manage the project and trace the progress in order to bring reliability. These are the only tools he used since he started working there, so no previous comparison is available to show us the impact of switching to different tools.

Interviewee C

Interviewee C is a web developer from Bulgaria. He is studying IT Project Management in Sweden and is working from distance full time. His company is working with agile methodologies but he did not witness the agile methodology transition as well. As a young and modern person, he thinks that agile development is the best decision for his company because it is “fast, productive and allows quick correction of errors”. The way of work in his company includes modules that have to be finished before moving on to the next module, which reminds of a more “*waterfall- ish*” approach but in general the only methodology they use is the agile one. The reason to choose it is that the developers have the “opportunity to fix errors and create better quality software”. The company is rather big consisting of 8 development teams with 6 developers in each team. The communication environment in the company of Interviewee C is highly informal and open which facilitates a better environment for sharing of problems and less stressful working space. It is also important for them to have an informal communication because that way there are no issues and helps them to “be on the same page” so that the goals are clear for everyone. The company relies on documentation of the code with every module. Documentation is an important piece in this company because it answers most of the stakeholder questions in advance and brings them

on board with the project at its current state. Only one tool is used in this company to keep the track of the project which helps to see the tasks and progress.

Interviewee D

Interviewee D is a scrum master and an agile development coach. He went through a transition of agile development in a traditional setting so his background is one of the few who have experience with waterfall methodology. He uses agile development mainly for competitive advantage and started the transition process through a lot of training. Customer collaboration is of great importance and according to him change in mindset of the organization is “the biggest trap”. He provides more freedom to his teams by increasing the decision making and improve the shared responsibilities. The way he deals with documentation is through user stories which is an innovative way of working. He acknowledges that transparency can be an issue for his team when they have traditional background which is an issue outside of the challenges identified in my study. To sum up, the biggest challenges for this interviewee are: PM role, change in mindset, organizational agility and transparency.

Interviewee E

The next interviewee is one of the first accredited agile methodologies coach in Finland who has 10 years experience in Nokia and now is working in a Finish as a main “agileist”. She started the organizational change in Nokia in stages- inside out and not the whole organization at once by providing training and support for the beginning phase. Then according to her the processes changed following the mindset change. Documentation was growing parallel to software and what she made unique is that she turned documentation into communication. She let the teams decide on which tools to use to maintain transparency and visibility. The biggest challenge according to her is convincing the middle management because they feel threatened to lose their positions and also from the new development flow (where comes the PM role challenge): “When you convince top management, you can convince the teams too, but middle management is the biggest challenge. Get top management on board to make the implementation successful”.

Interviewee F

The following interviewee is a project manager at a company in Jonkoping, Sweden and a teacher in web development at JU. From his experience convincing the customer is the biggest challenge since not all companies are aware of the benefits. His company has not implemented agile development in 100% because they rely strongly on customer preferences, so if they do not want it, the team chooses another approach. In the team communication is strongly suggested to communicate all changes and their main goal is to satisfy the customer through effective communication. He communicates to everyone specifically and mostly formally to benefit eventual pitfalls in the future. The responsibilities in the teams are increased when they use scrum and all processes are monitored to keep the customer updated (again transparency). Documentation is required to have when they make maintenance of the product in the future. Adopting new tools requires time and patience and they also must adapt fast to changes. According to him the biggest challenges are: documentation, tools and adapting to change.



Interviewee G

The next interviewee is software engineer in Bulgaria. According to him agile development was welcomed really fast and adopted without any issues. His experience with waterfall methodology is not big, so his comparison was not possible. Documentation is also important for his team because each developer updates the documents according to the implemented story, so documentation is a vital part in his team. They use a variety of tools which saves them time and allows them to “apply the best practices”. According to him flexibility is something that can be lost easily among the developers, so keeping the agile mindset is something fragile that can be lost but it is needed to fit the “team needs”.

Interviewee H

The next interviewee is a computer science student in the 6th semester. He uses scrum in his projects with university so his team had no previous experience with other methodologies. Challenging for him is that the team is resistant to the idea and it is hard to convince them. They value flexibility and communication. It is hard for them that planning is impossible due to the “incremental process” but can also react quickly to requirement changes. Decision making is an issue for his team and convincing the customers to use agile development due to lack of knowledge about its benefits.

Interviewee I

Interviewee I is a student in Uppsala, Sweden doing a master thesis in software engineering and also an internship in the same field. She often implied that agile development is used in small teams and that they use highly iterative approach. Communication is first formal and then informal flexibility and no documentation are highly valued. What changed for her is that her work increased with more but smaller tasks and that the tasks are highly monitored (again transparency). Decision making is highly encouraged and planning is needed at some point. According to her challenging is to plan, the PM role, changing the mindset, organizational agility, and documentation

Interviewee J

Is a software developer in Bulgaria and is the senior PHP developer in a 7 person scrum team. It is challenging for him to prepare the management which is an often-met challenge. Transparency is something completely new for him and maybe the most challenging for him. This also brought the fact that communicating challenges is way easier and documentation changes according to project growth (agility in documentation). The tools are chosen according to the project and the features are planned in the beginning. What is most challenging in the agile adoption is communication, change in mindset, transparency and planning.

Interviewee K

The following respondent is a computer science engineer, who works in France. He considers agile development to be a methodology for young and open- minded people and that it also matches the needs of the customers and their constantly changing mind. In his company, developers use user stories which they split into tasks to create the product backlog and reduce documentation. The communication flow within his team active and it includes the customer on daily basis. It is important for him that the communication is always going and that “having a shy member is not a plus in the agile development team”. The greatest issue he encountered in the agile development transition is the PMs cannot give 100%

accurate requirements and that the adaption is necessary. Documentation is reduced to a minimum to increase flexibility. Not having clear specifications made it harder for him to choose the right tool but he uses agility to test new tools- by iteration. The hardest task for him is the lack of clear requirements and being unable to calculate the possible outcome of the project.

Interviewee L

This interviewee is a software developer and a scrum master in The Netherlands. The company decided to transit to agile development because their clients had often changing requirements and having waterfall methodology did not make sense. The team is introduced to agile methodologies slow and steadily with the help of coaches. They helped even the clients who were not familiar with agile development at this point. The development team has taken a lot of classes in scrum and the agile coach helped them through tough situations. The first agile development project was successful under his supervision as a new scrum master. With the help of the agile development coach, they facilitated more open communication not only within the team, but also with the client. In his opinion *“Communication issues at any level, could jeopardize the success of the project.”* When the transition started, the missing hierarchy was the biggest problem, which includes mostly the management. Some members who had waterfall methodology experience adapted fast, other feared the more responsibilities and did not take ownership of their work. Organizational agility was reached by starting with educational meetings and combining the teams by similar interests. *“The whole company was included in the transition... everybody was included in the team.”*. Responsibilities shifted from the PM to the PO (Project Owner) and the only issues are connected to the former PMs and their loss of power. The whole team made decisions together and the team itself consisted of 3-5 people. Everyone got used to self-organizing and the team members had knowledge about everyone skills and good sides. The handling of the documentation is done by a Service desk which operated with small client requests. Nevertheless, *“planning was a nightmare”* especially when clients requested big changes. Tools did not require big changes in the workflow and the new tools are not challenging. The biggest challenge for him is to make the environment comfortable and that teammates have much more responsibilities so *“their decisions are as important as anybody within the team”*.

Interviewee M

The respondent is a software engineer in Gothenburg, Sweden. He thinks agile development is necessary because it lowers the costs and wastes less time. The transition in his company started after everyone agreed that they needed it according to the constantly changing requirements from their clients. *“The team took small steps over time to lead to self-organization”* through frequent meetings. This is how the team grows on its own and not because the managers require it. The last step is to apply discipline in the team to ensure consistency and ultimately successful implementation of agile development. This is characterized by being on time for meetings, keeping up promises and be informed of the next steps of the software development. The team started to have more frequent meetings and e-mails and everyone kept transparency to a maximum with the help of informal communication. One of the most crucial problems when transitioning to agile development is the resistance of team members to the change and adopt the different agile methodologies values. The team saw it as chaotic without the possibility to plan upfront and lacking a leadership figure. The team had to get used to iteration and implement new changes. The team made decisions together and discussed them in every meeting. This said, the documentation is reduced to a minimum and

replaced with face- to- face communication and done whenever necessary. The new tools are agile and easy to adapt and they tested them with every new sprint. *“The biggest challenge for me was communicating and making sure all team members were informed of my progress on the assigned tasks. That led to somewhat inefficient meetings, especially if members missed a meeting”*

Interviewee N

Respondent N is an app developer and currently a scrum master in a big company situated in Bulgaria. The general understanding was positive when the company intended to transit to agile development. The organization had strong process- based culture and used waterfall methodology in their projects. The transition started with the help of agile development coaches which helped them through the process. In the beginning, they started the change with frequent meetings and placing a supervisor in the place. Communication was challenging in the start mostly because no one was used to the transparency that followed. Slowly, things changed and everyone got accustomed to the new way of work. The management met this transition with resistance, because some managers from the lower levels *“had to go through more classes due to unwillingness to adjust”*. Eventually who whole organization had to work as one and involve everyone. Decision making started to become a part of the team and collaboration increased. The team members are glad that they must deal with less documentation and *“choosing tools was way easier than before”*. The biggest challenge is changing the roles inside the company and dealing with transparency.

Interviewee O

Interviewee O is a scrum master in a software development company in the Netherlands. His experience with agile development dates to 7 years ago, when waterfall methodology had much stronger influence and the company is process oriented. Shifting to agile methodologies is met with high uncertainty and resistance since not many companies are aware at that time and reaching for such rapid change can be catastrophic if not done right. So, the company started to notice that the software development world is changing fast and adapting fast is necessary. This is the goal- to be competitive and give freedom to their clients and developers. Organizational changes started upside- down with inviting many professionals to provide training. His first scrum project was a complete success, because communicating this change made everyone cohesive and united to achieve this freedom to be flexible. Communication is indeed necessary and with it increased decision making and *“working was much more fun and pleasant”*. The environment was positive about the change, but the few who still had bureaucratic way of work, made the organizational reach agility much slower than it should have been. Therefore, *“documentation was reduced to a minimum and choosing tools”* was *“a pleasant task for the developers because I gave them freedom to test what they like”*. The biggest challenge for the company is facing the new changes in connection to organizational agility.

The following documents are available upon request:

- Documents
- Content analysis
- Literature review concept matrix
- Audio recording
- Interview transcripts
- Personal notes