



FACULTY OF TECHNOLOGY

IMPLEMENTING AGILE PRACTICES IN A LARGE GAME DEVELOPMENT COMPANY

Emilia Hekkala

INDUSTRIAL ENGINEERING AND MANAGEMENT

Master's thesis

May 2022

ABSTRACT

Implementing Agile practices in a large game development company

Emilia Hekkala

University of Oulu, Industrial Engineering and Management

Master's thesis 2022, 115 pp. + 1 appendix

Supervisors at the university: Jere Lehtinen, Jaakko Kujala

Large game development companies working with complex projects need to find ways to work iteratively and adjust to changing requirements in order to succeed. To achieve this, many of them turn to Agile frameworks and practices. In this thesis, the purpose is to analyze how these large game development companies implement Agile working practices and what are the related benefits and challenges. The roles of human skills in the implementation process are also examined. This is done by conducting a single-case study of the Agile implementation process in a case company.

The findings of the study indicate that the Agile implementation process in the case company follows a general change process structure, which consists of seven core activities: recognition and start, diagnosis, planning, implementation and review, sustaining change, learning, and managing people issues. During the implementation process, the most significant challenges the case company faced were connected to implementing Agile ways of leading development, such as Product Ownership and Products Backlog. In addition, the relationship of human skills and Agile implementation appears to be cyclical; while well-developed human skills such as support, communication, and conflict solving skills make the implementation process easier, properly implemented Agile practices may also improve human skills.

This study contributes to Agile implementation research by expanding the knowledge of Agile implementation in large game development organizations and elaborating the connection between the implementation process and change management theory. The findings of the study can also help those implementing Agile in game companies to predict the structure of the process and anticipate the potential challenges of it.

Keywords: agile, agile implementation, game development, human skills

TIIVISTELMÄ

Implementing Agile practices in a large game development company

Emilia Hekkala

Oulun yliopisto, Tuotantotalous

Diplomityö 2022, 115 s. + 1 liite

Ohjaajat yliopistolla: Jere Lehtinen, Jaakko Kujala

Suurten ja monimutkaisten projektien parissa työskentelevien pelialan yritysten täytyy löytää tapoja työskennellä iteratiivisesti ja sopeutua muutoksiin menestyäkseen. Tästä syystä monet niistä kääntyvät ketterien menetelmien puoleen. Tässä diplomityössä tarkoitus on analysoida sitä, miten nämä yritykset ottavat käyttöön ketteriä menetelmiä ja mitä hyötyjä ja haasteita prosessiin liittyy. Myös inhimillisten taitojen rooleja käyttöönottoprosessissa tutkitaan. Tutkimuksessa perehdytään ketterien menetelmien käyttöönottoon kohdeyrityksessä yksittäistapaustutkimuksena.

Tutkimuksen tulokset osoittavat, että kohdeyrityksessä ketterien menetelmien käyttöönottoprosessi mukailee yleisen muutosprosessin rakennetta, joka koostuu seitsemästä toiminnoista: muutostarpeen tunnistamisesta ja prosessin aloittamisesta, diagnosoinnista, suunnittelusta, käyttöönotosta ja arvioinnista, muutoksen ylläpitämisestä, oppimisesta ja ihmisiin liittyvien asioiden johtamisesta. Keskeisimmät kohdeyrityksen käyttöönottoprosessin aikana kohtaamat haasteet liittyivät ketterien menetelmien mukaisten johtamistapojen käyttöönottoon kuten tuoteomistajuuteen ja tuotteen tehtävälisöihin. Inhimillisten taitojen ja ketterien menetelmien käyttöönoton suhde puolestaan vaikuttaa sykliseltä; samalla kun hyvät inhimilliset taidot kuten tuki-, kommunikaatio-, ja konfliktien selvittelytaidot tekevät käyttöönottoprosessista helpomman, voivat ketterät menetelmät puolestaan kehittää inhimillisiä taitoja.

Tämä tutkimus edistää ketterien menetelmien käyttöönottoon liittyvää tutkimusta kasvattamalla ymmärrystä menetelmien käyttöönotosta suurissa pelialan yrityksissä ja lisäämällä tietoa käyttöönottoprosessin ja muutoksenhallinnan teorian välisestä yhteydestä. Tutkimuksen tulokset voivat myös auttaa ketterien menetelmien käyttöönottoa pelialan yrityksissä johtavia henkilöitä ennakoimaan käyttöönottoprosessin kulkua ja siihen liittyviä haasteita.

Avainsanat: ketterät menetelmät, agile, agilen käyttöönotto, pelinkehitys, pehmeät taidot

FOREWORD

This Master's thesis was written between November 2021 and May 2022, and it was done for a case company. The aim was to study the way the company implements Agile working practices and the related benefits, challenges, and human skills. Due to time limitations, the study was finished before the Agile implementation process in the company was completed. Because of this, it is important for everyone reading this thesis to understand that this is a snapshot of a certain moment in time in the case company and does not describe the final results achieved there.

I want to thank the case company and their representatives for the thesis position and for the interesting topic and case to study. Many thanks for every interviewee, you made it possible to conduct this study by providing the data needed and sharing your views and opinions with me. Also, special thanks to my supervisor in the case company for all the support and encouragement you gave. Your trust was important for me throughout the process.

Furthermore, I want to thank my supervisors Jere Lehtinen and Jaakko Kujala at the University of Oulu. You gave me insightful feedback and valuable advice anytime I needed it and inspired me to do my very best here. It was a pleasure to work with you and learn from you.

Finally, I want to thank my partner and my family and friends for all the support you gave me during this time.

Oulu, 10.5.2022

Emilia Hekkala

Emilia Hekkala

TABLE OF CONTENTS

ABSTRACT

FOREWORD

TABLE OF CONTENTS

1 Introduction	6
1.1 Research phenomenon and questions.....	6
1.2 Empirical context and methods	7
1.3 Thesis structure	8
2 Literature Review	10
2.1 Agile.....	10
2.2 Agile in practice – the Scrum framework	18
2.3 Scaling Agile for large organizations.....	20
2.4 Agile and game development.....	30
2.5 Human skills.....	35
2.6 Synthesis of the literature review	39
3 Research Process	46
3.1 Research design.....	46
3.2 Case context	46
3.3 Data collection	49
3.4 Data analysis	50
4 Research Findings	52
4.1 Reasons for implementing Agile in the case company	53
4.2 Practices used when implementing Agile in the case company.....	55
4.3 Benefits gained through implementing Agile in the case company.....	60
4.4 Challenges faced when implementing Agile in the case company	64
4.5 Human skills in implementing Agile in the case company.....	73
5 Discussion	79
5.1 Agile implementation practices.....	79
5.2 Agile implementation and human skills.....	86
5.3 Product Ownership and Product Backlog	89
5.4 Supplementary topics	92
6 Conclusions and Recommendations	99
6.1 Key findings	99
6.2 Theoretical contribution	100
6.3 Managerial implications.....	102

6.4 Research limitations	103
6.5 Future research	106
REFERENCES.....	108

APPENDICES:

Appendix 1. Agile implementation process in large game development organizations, following the structure of the change process defined by John Hayes (2018)

1 INTRODUCTION

1.1 Research phenomenon and questions

AAA game development, meaning developing video games with high development and marketing budgets, is a demanding field to operate in. There are several external and internal challenges to face: First of all, the competition is intense. To survive in the midst of it, companies need to commit to complex projects that combine technical excellence with outstanding creativity and large amounts of content. This means having huge development teams filled with people with diverse skillsets working for years to create extraordinary games. However, at the same time the environment for game development keeps constantly changing. New technology is developed and players rapidly gain and lose interest on different video game trends and genres. Because of this, the large teams working with complex projects – they need to be agile, too, in order to survive and succeed.

To become more agile, game development companies need to find ways to work iteratively and readjust to changing requirements. To achieve this, many of them turn to Agile frameworks such as Scrum. It is a framework that provides people, teams and organizations methods and tools that help to find adaptive solutions in complex environments (Schwaber and Sutherland 2020). Implementing Scrum is a big and challenging organizational change because it affects many integral parts of the organization, such as team structure, meeting structure and planning processes. To make it successful, organizations need to heavily invest in it and understand what the effect of human skills like empathy and communication skills for the process is.

The difficulty with Scrum is that it was created for teams with ten or fewer people (Schwaber and Sutherland 2020), so in its original form it is not suitable for bigger organizations, and they need to find other ways to implement agility in their operations. There is research done about implementing various Agile frameworks in large organizations, but in the context of the game industry such research is scarce. There the focus of research has primarily been on small organizations, and the role of human skills in the Agile implementation process has also been overlooked. These are significant deficiencies considering the increasing size of game development teams and organizations and the complexity and magnitude of the game industry – the estimated

value for global game market is over 175 billion dollars in 2021 (Wijman 2021). If the impacts of implementing and executing Agile in big game companies were more deeply understood, the implications could be remarkable as that would allow choosing the best methods for working and further developing them and thus increasing the efficiency of the industry.

Gathering insight of the role of human skills in implementing Agile frameworks in large game development organizations would be valuable as well, since previous research has proved that they are essential factors for successfully practicing Agile (Beck et al. 2001; McHugh et al. 2012; Thorgren and Caiman 2019). Understanding the significance of these skills in the industry could guide the actions of the leaders and thus improve the processes and frameworks and make them more successful, which in turn would help the industry to develop.

Based on the background described above, the aim of this research is to develop a better understanding of how large organizations in the game industry implement Agile practices and how such practices suit these organizations. The roles of human skills in the implementation process are also studied and described. Following two research questions summarize the focus of this study:

1. How do large game development organizations implement Agile working practices, and what are the related benefits and challenges?
2. What kinds of roles do human skills play when implementing Agile in large game development organizations?

1.2 Empirical context and methods

The research questions are approached through a single-case study design. The research process began with a literature study to gain understanding of the phenomena connected to the questions. Then, a single case study was conducted. The aim of the study was to elaborate on the existing theoretical understanding of Agile implementation practices focusing on large game development organizations.

The case company of the study is a large game development organization that develops AAA games and currently employs around one hundred people. In 2021, the company

started to implement Agile practices – first they implemented the Scrum-framework in autumn 2021 and then the Large-Scale Scrum (LeSS) -framework in spring 2022. To collect the data needed to study the Agile implementation process there, 11 interviews were conducted and the organization's relevant documentation was collected.

Interviews were transcribed and all data was analyzed following the qualitative content analysis -method. It is typically used to interpret meaning from text data to develop and extend knowledge (Hsieh and Shannon 2005), which made it a good choice for the study. Simultaneously with analyzing the interview data, the collected documentation was used for triangulation, meaning that the interview data was reconciled with the documentation to see if they align and to improve the accuracy of the study (Jick 1979).

1.3 Thesis structure

The thesis is structured as follows: In this chapter, the research phenomenon and research questions were introduced, as well as the empirical context and methods. In chapter 2, the theory of Agile implementation is described. The chapter provides foundation for the empirical analysis and also gives the reader of this document the information needed to understand the discussion in later chapters. In the beginning of the chapter, the basic ideas of Agile are defined, followed by a description of how Agile can be done in practice, first in general and then specifically in large organizations. In these sections two Agile frameworks, Scrum and Large-Scale Scrum (LeSS), are also introduced. After this, the relationships of Agile and game development and Agile and human skills are explained. In the end of the chapter, the topics are synthesized and connected to the research questions.

In chapter 3, the focus moves from theory to the case and the research process. The research design is introduced and the case context described, as well as the ways data was collected and analyzed. That includes describing the interviews and how they were organized and the data analysis method chosen. In chapter 4, the focus is on the research findings. The chapter is structured with the research questions in mind, which means the reasons for implementing Agile practices in the case company are described first, followed by the practices used and the benefits and challenges faced during the implementation. In the end of the chapter, the roles of human skills in the implementation process are also introduced.

In chapter 5, the research findings are further discussed and compared to the previous research of implementing Agile. In the beginning of the chapter, the answers to the research questions are formed, and in the later parts of it additional interesting findings are discussed. Finally in chapter 6 the key results of the study are described, including the theoretical contribution and managerial implications of those. The limitations of the study are also addressed and future research ideas introduced.

2 LITERATURE REVIEW

Theoretical basis for the research consists of several areas. First, it is important to understand the basic concepts of Agile and why so many organisations pursue it (section 2.1). Understanding the Scrum -framework (section 2.2) is essential, too, for two reasons: it is the most common way to implement Agile in organizations, and numerous large-scale Agile frameworks are built upon it. After that, the focus moves on to scaling Agile for large organizations (section 2.3). In this section, scaled Agile methods are first discussed on a general level, and then the Large-Scale Scrum (LeSS) -framework is studied more in detail, as it is the framework implemented in the case organization of the research. After that, the focus is on the game industry and the way its unique features affect the implementation of Agile frameworks (section 2.4), followed by a description of human skills and their role in Agile and in the game industry (section 2.5). Finally, the key issues of the Literature Review are synthesized and the connection between Agile implementation processes and change process structures is discussed (section 2.6).

2.1 Agile

“Agile” is a collective term used to describe different methodologies that support the values and principles of the Agile Manifesto, (Doug 2015) and in essence it is all about enabling inspection and adaptation in environments with high amounts of variability (Measey and Radtack 2015, pp. 33–36). Agile is based on values and principles, not on specific processes, and therefore it works in many kinds of situations and environments (Hohl et al. 2018).

Agile as a concept was born in 2001, when 17 software development experts who had recognized a need for lightweight processes to guide software development formulated the Agile Manifesto and Agile Principles (Beck et al. 2001; Hohl et al. 2018). At the time, software projects were mostly done using traditional software development methods known as “Waterfall”-methods. In these methods design, development, testing, and delivery were undertaken only serially and the process was rigid and siloed, which often led to projects failing (Sommer 2019). With Agile the idea was to restructure the balance between governance and collaboration and make development processes more adaptable (Beck et al. 2001).

2.1.1 Agile Manifesto

The basis for the Agile approach is the Agile Manifesto (Hazzan and Dubinsky 2014). It describes four values, which are the guidelines for thinking like an Agile team and the foundation for every Agile practice (Doug 2015, chap. 3; Measey and Radtac 2015, pp. 4–10). The Manifesto is introduced in Figure 1:

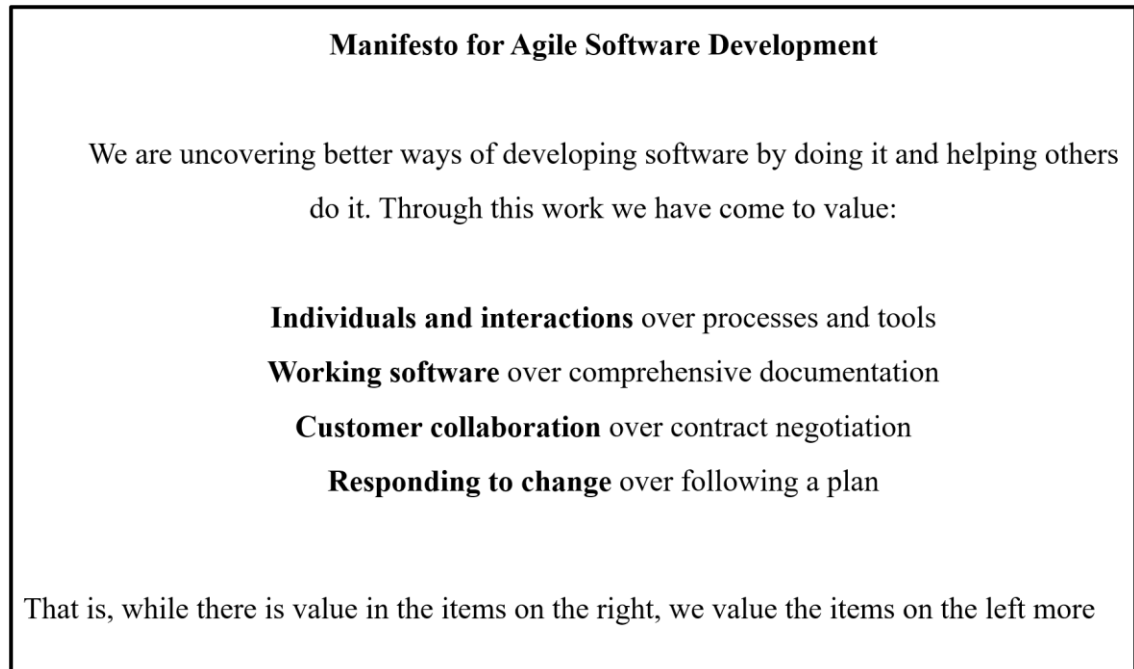


Figure 1. Agile Manifesto, according to Beck et al. (2001)

The Agile Manifesto is structured so that the values on the left side are more valuable than the ones on the right, even though those have value too. The experts forming the Manifesto felt that the values on the right were given too much emphasis at the time and wanted to turn the focus on the matters on the left side. (Doug 2015, chap. 3)

The wording used in the Agile Manifesto is universal, which makes it still applicable and successful today. (Hohl et al. 2018) However, the universality makes the manifesto rather ambiguous too, and for that reason the values have been more profoundly discussed in several publications. Measey and Radtac (2015) for instance describe that the first of the values, “Individuals and interactions over processes and tools”, tells that enabled and motivated people and their effective interaction as a team are the most important factors when delivering value to the customer, and the processes and tools are there only to support these individuals, not the other way around. In the second value, “Working software over comprehensive documentation”, the message is that in Agile the focus is

on providing working software and thus adding value directly to the customer, according to Measey and Radtac. Only documentation that adds value to the product should be created and it should be made synchronously with the delivery of the working product, not solely at the beginning or end of the project (Measey and Radtac 2015, pp. 4–10). In addition, Hazzan and Dubinsky (2014) claim that starting the actual development as early as possible instead of only creating documentation at the beginning is critical for the quality of the product.

The third value in the Agile Manifesto, “Customer collaboration over contract negotiation”, changes the perception of the customer role in the development process (Hazzan and Dubinsky 2014) as it highlights how important it is to create collaborative relationship between the supplier and customer and how contracts should support this collaboration, not complicate it (Measey and Radtac 2015, pp. 4–10). This approach also makes it easier to react to changes, which is the focus point of the fourth value, “Responding to change over following a plan”. This value builds on the idea that changes happen as the project develops and the chances of success are maximised by remaining flexible (Measey and Radtac 2015, pp. 4–10). Often too much time is spent on planning in advance and not enough time responding to change. In Agile the aim is to spread the planning throughout the entire development process. (Doug 2015, chap. 3; Serrador and Pinto 2015).

2.1.2 Agile Principles

When the 17 software development experts created the Agile Manifesto, they also defined twelve Agile Principles that support and expand it (Beck et al. 2001). The principles are listed in Figure 2:

Principles behind the Agile Manifesto

We follow these principles:

Our highest priority is to satisfy the customer through early and continuous delivery of valuable software.

Welcome changing requirements, even late in development. Agile processes harness change for the customer's competitive advantage.

Deliver working software frequently, from a couple of weeks to a couple of months, with a preference to the shorter timescale.

Business people and developers must work together daily throughout the project.

Build projects around motivated individuals. Give them the environment and support they need, and trust them to get the job done.

The most efficient and effective method of conveying information to and within a development team is face-to-face conversation.

Working software is the primary measure of progress.

Agile processes promote sustainable development. The sponsors, developers, and users should be able to maintain a constant pace indefinitely.

Continuous attention to technical excellence and good design enhances agility.

Simplicity--the art of maximizing the amount of work not done--is essential.

The best architectures, requirements, and designs emerge from self-organizing teams.

At regular intervals, the team reflects on how to become more effective, then tunes and adjusts its behavior accordingly.

Figure 2. Agile Principles, according to Beck et al. (2001)

While the Agile Manifesto is rather abstract, the Agile Principles are more concrete examples of how to be Agile in practice. They embrace several matters, such as customer engagement, collaboration, flexibility and rapid delivery of working products and act as the foundation for all Agile methods and Frameworks (Sommer 2019). For example, the 12th principle, “At regular intervals, the team reflects on how to become more effective, then tunes and adjusts its behavior accordingly.” refers to Retrospectives, meetings that are integral part of almost every Agile framework (Measey and Radtac 2015, pp. 4–10). In these meetings, the aim is to discuss which areas and processes work well for the team and which need to be improved (Schwaber and Sutherland 2020).

The importance of understanding and internalizing the Agile Manifesto and Agile Principles is emphasized in several articles. For instance, Sommer (2019) argues that the

Agile values and principles are the key for success and that creating a truly Agile organization means integrating the principles of Agile in every element of the company. Kiv et al. (2018) also states that understanding Agile Manifesto thoroughly is the most efficient way to understand and adopt Agile.

2.1.3 Benefits and challenges of Agile

What makes Agile a tempting option for organizations are the many benefits it can bring if successfully implemented. LEGO Group, for example, has been successful with it: according to Sommer (2019) they adopted Agile in 2018 and started to see positive effects within a year. The most significant effect was the improvement in employee motivation and satisfaction, and on top of that, project delivery times were reduced from months to weeks and the speed of response to change was considerably improved (Sommer 2019). These results are well in line with other research; numerous articles argue that properly implementing Agile results in higher team productivity, motivation and morale, better quality, faster time to market and lower risk levels than using traditional approaches such as Waterfall methods (Serrador and Pinto 2015; Rigby et al. 2016, 2018). One more benefit that is often emphasized is the Agile organizations' ability to respond to changes quickly and effectively, which is important in constantly changing and highly dynamic environments (Rigby et al. 2016; Sommer 2019).

Even though the list of benefits is long, there are also multiple challenges related to Agile. They often arise from the fact that organizations tend to focus on “doing Agile” instead of “being Agile”, which means putting the emphasis on executing a framework instead of having a truly Agile mindset (Measey and Radtac 2015; Hohl et al. 2018). According to Kiv et al. (2018), the reason for this is that development teams often do not devote enough effort into understanding Agile Manifesto and its values and principles, they just uncritically adopt and follow the rules of a specific Agile framework. This is problematic because implementing Agile is a massive, fundamental change for any organization, and the fundamental principles and attitudes need to change too, to make the implementation successful and long-lasting (Sommer 2019).

Hohl et al. (2018) mention that another factor making implementing Agile challenging is the context sensitivity of Agile methods. They argue that even though the ideas and values of Agile work in many kinds of environments, the same does not apply to all Agile methods, as some of them work better in very specific environments. For this reason,

choosing the best methods for specific situations can be challenging and should be carefully done (Hohl et al. 2018). The simplicity of Agile Frameworks may cause issues too because organizations regularly already have complex processes and systems in place and simplifying and streamlining those may be challenging (Measey and Radtac 2015, pp. 33–36).

Yet another typical challenge when implementing Agile is the changing role of leadership and how that is received in the organization (Sommer 2019). The fifth Agile Principle states that “Build projects around motivated individuals. Give them the environment and support they need and trust them to get the job done” (Beck et al. 2001). This means that the individuals and teams need to be able to make decisions independently and managers need to step aside on team decisions and instead focus on setting the strategic direction (Rigby et al. 2016). Such a fundamental change may be hard to accept or even realize without proper training and often leads to situations where managers unwittingly continue to manage in ways that are against Agile principles and undermine the effectiveness of Agile teams (Sommer 2019).

2.1.4 Implementing Agile

Previous research has outlined many practices that can help overcome the challenges mentioned and obtain the benefits of Agile. First of all, it is important to understand the particular context of the organization implementing Agile and carefully compare and choose the best Agile and non-Agile frameworks and methods for that environment (Measey and Radtac 2015, pp. 33–36; Sommer 2019). Another important element, according to Kiv et al. (2018), is training and coaching members of the organization and helping them truly understand Agile, including the Agile Manifesto and Agile Principles; this way they do not only execute Agile frameworks but learn to be Agile in all circumstances. Training and coaching senior managers is particularly important since they need to acknowledge the changing role of leadership and learn to work with self-managing teams (Kiv et al. 2018). Sommer (2019) also notes that teams should be encouraged to continuously improve their work methods with the tools Agile Frameworks provide.

In some articles, the writers have created lists of steps that help implement Agile successfully. In 2016 Rigby, Sutherland and Takeuchi outlined six crucial steps for this implementation. Those are (1) learning how Agile really works, (2) understanding when

it is applicable, (3) starting small and letting passionate evangelists spread the word, (4) allowing teams to customize their practices once they have mastered them, (5) practicing Agile at the top and (6) destroying corporate barriers standing in the way of Agile behaviours (Rigby et al. 2016). These steps then inspired the LEGO Group, for instance, in their successful Agile implementation. However, according to Sommer (2019) they did not take the steps as they are but made some changes and redefined them into five categories that fit their organization the best. The five categories that LEGO Group used to implement Agile were (1) organizational structure and evolving from existing structure to a product-oriented structure, (2) mandate and shifting ownership and mandate of deliverables from managers to product teams, (3) financial processes and moving from yearly budgeting processes to dynamic, frame-based budgeting that takes into account the strategic aims, (4) performance measures and redefining them to focus on value, product and team, and (5) delivery processes and changing from end-of-project delivery to continuous deliveries (Sommer 2019). These steps are high-level guidelines for internalizing the correct mindset, while Agile Frameworks offer more detailed instructions for implementing and executing Agile in practice.

In addition to the steps that focus specifically on implementing Agile, general change process structures can also be used to describe the implementation process; Sommer (2019) for example notes that implementing Agile successfully requires a change management approach. The reason is that implementing Agile is essentially a massive change for any organization as it consists of several consecutive steps, alters many integral parts of the organization such as team structures and planning processes, and also requires changes in the principles, attitudes, and the culture of the organization (Sommer 2019). The change process structure created by Hayes (2018) is a suitable framework and lens for viewing this change as it is simplistic by design but still covers all the core activities of change, including those that continue throughout the process. In Hayes's structure, there are seven core activities: (1) recognition and start, (2) diagnosis, (3) planning, (4) implementation and review, and (5) sustaining change, which come one after another, and (6) learning and (7) managing the people issues, which take place throughout the process. The flow of these activities is visualized in Figure 3. In practice, the boundaries between them are not always clear-cut but may overlap and blend together, depending on the nature of the change (John Hayes 2018, pp. 22–41).

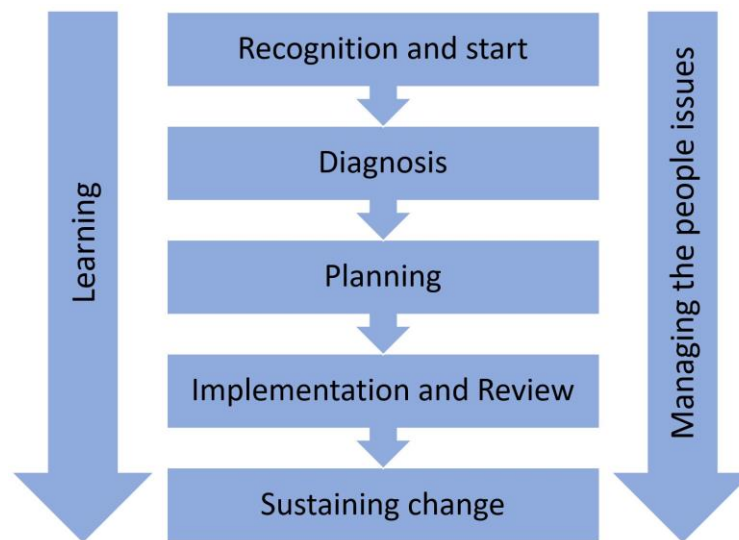


Figure 3. The seven activities of change process according to John Hayes (2018)

The first activity “recognition and start” refers to recognizing the need for change, in this case understanding that there is a need to implement Agile in the organization, and starting the change process. It is followed by diagnosing what needs to be changed. This stage involves both assessing the present state and its problems and opportunities and developing a realistic but desirable vision of a preferred future state. In the planning-phase the focus is on defining how the vision and related goals will be achieved and choosing a change strategy. The level of detail in planning can vary: if the desired end state is very clear, the plans can be quite detailed, but if there are any open-ended questions, planning needs to be an iterative process that evolves over time. When implementing the change and reviewing progress, the focus shifts from planning to action. In this phase, it is important to monitor progress and seek feedback – things may not progress as expected and readjustments may be needed. After the implementation is done, it is time to sustain the change and make sure the new ways of working become the norm. (John Hayes 2018, pp. 22–41, 68–85)

There are also those two activities that need to happen throughout the change process: learning and managing the people issues. Continuously learning from experiences and feedback makes it possible to adjust and improve the actions as needed, and leading and managing the people issues, such as communication, trust, internal politics, motivation and support, helps to keep stakeholders involved and motivated. Both of these are essential to make the implementation process successful. (John Hayes 2018, pp. 22–41, 491–500)

2.2 Agile in practice – the Scrum framework

There are tens of documented Agile methods, approaches and frameworks built on the principles and values of the Agile Manifesto that aim to describe how those should be applied in practice (Sommer 2019). Scrum is the most popular and best-known of those (Rigby et al. 2016; Hohl et al. 2018) and it is applied in many industries, including the game industry (Koutonen and Leppänen 2013). The large-scale Agile frameworks are also typically built upon Scrum (Sommer 2019). For these reasons, having insight of the core ideas of Scrum is a good foundation for both understanding how to apply Agile in practice and how the large-scale Agile frameworks work.

The guiding principle of the Scrum-framework is to empower creative, cross-functional teams and help them find adaptive solutions to complex problems (Rigby et al. 2016). Schwaber and Sutherland (2020) tell that Scrum is built on three pillars – transparency, inspection, and adaptation – that guide the work methods. They describe that transparency focuses on how processes and results need to be visible for everyone performing and receiving the work. This enables inspection, meaning frequently reviewing the progress to detect problems and undesirable variances, which in turn enables adapting the processes whenever the process deviates outside acceptable limits (Schwaber and Sutherland 2020).

Schwaber and Sutherland (2020) describe Scrum Teams as self-managing and cross-functional, which means that the team members have the skills needed for fulfilling the goals of the team. They add that Scrum Teams consist of ten or fewer people, and there are three types of roles in them: Product Owner, Scrum Master and Developers. The Product Owner is the person responsible for maximizing the value of the product and managing the Product Backlog, an ordered list of what is needed to improve the product the team develops, and this person also bridges the gap between business stakeholders, customers and the developers (Schwaber and Sutherland 2020). This role is fairly similar to traditional project managers (Noll et al. 2017). The Scrum Master, on the contrary, is the person accountable for coordinating the daily work of the team and the adherence to the Scrum process (Hron and Obwegeser 2018), and the Developers are the team members committed to creating value for the product (Schwaber and Sutherland 2020).

Scrum emphasizes an iterative and incremental approach for planning and organizing work, which helps to increase predictability and to control risks. An essential part of this approach are Sprints, 1-4 weeks long events that contain the other Scrum events (Sprint Planning, Daily Scrums, Sprint Review, and Sprint Retrospective). During a Sprint, the Scrum Team creates a new Product Increment, which is a concrete steppingstone towards the final product. A new Sprint starts immediately after the conclusion of the previous one, and their length stays fixed. (Schwaber and Sutherland 2020)

The Scrum Events that happen within a Sprint are Sprint Planning, Daily Scrums, Sprint Review, and Sprint Retrospective (Schwaber and Sutherland 2020). This structure is visualised in Figure 4. The events are often facilitated by the Scrum Master, but in most cases the Developers and the Product Owner are the people presenting the actual content, such as updates and feedback (Noll et al. 2017). Schwaber and Sutherland (2020) describe the four events as follows: In the Sprint Planning the Scrum Team defines their Sprint Goal and selects which items listed in the Product Backlog they will do during the Sprint, thus creating a Sprint Backlog. The Daily Scrum is a 15-minute long event that happens every day throughout the Sprint, as the name implies, and there the purpose is to check the progress toward the Sprint Goal and to plan upcoming work. This meeting improves communication and quick decision-making and makes it easier to identify impediments early. In the Sprint Review the team and relevant stakeholders inspect the outcome of the Sprint and determine future adaptations, and in the Sprint Retrospective the Scrum Team discusses how to increase quality and effectiveness in their work (Schwaber and Sutherland 2020). James (2010) adds that even though Backlog Refinement is not specified as a Scrum Event, in practice it is often considered as one. It typically happens once in every Sprint and the goal there is to define and refine the items listed in Product Backlog (James 2010).

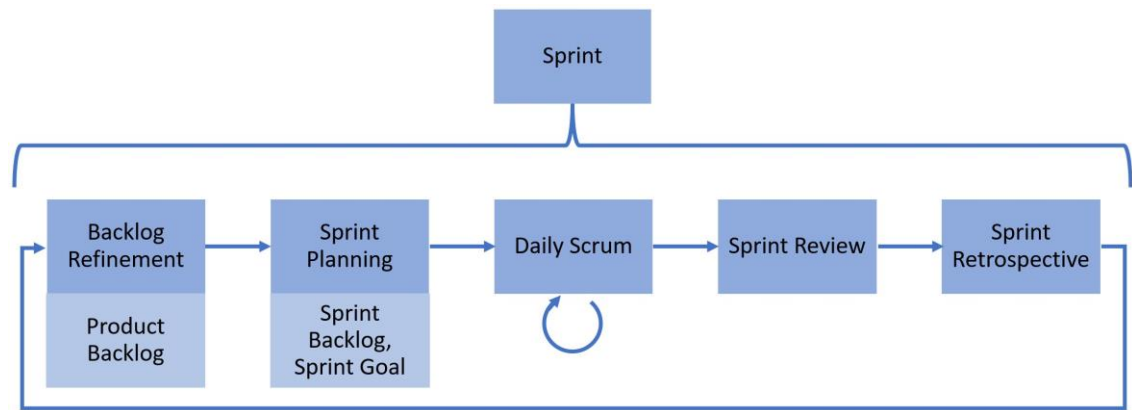


Figure 4. Visual representation of the Scrum process and the way Scrum Meetings are structured inside the Sprint, based on Schwaber and Sutherland (2020).

Schwaber and Sutherland (2020) also argue that the Scrum framework is “purposefully incomplete” i.e., it does not cover all the details of how the work should be organized but leaves space for learning and adapting. Thanks to this, Scrum can be implemented in many kinds of environments (Schwaber and Sutherland 2020), even though the best conditions for adopting Scrum are creative cultures with high levels of collaboration and trust (Rigby et al. 2016). It is very common that organizations do modifications to the Scrum framework when adapting it (Diebold et al. 2015). There are several reasons for doing this, such as geographically or otherwise distributed teams, a particular need to combine Scrum with other methods or use particular tools with it, increased focus on usability, adopting Scrum in large organizations or for large projects, and using Scrum in non-traditional contexts (Hron and Obwegeser 2018). The modifications made to Scrum are often referred to as ScrumBut or Scrum anti-patterns and include for example changing the Scrum roles and their responsibilities and modifying or skipping some Scrum events (Eloranta et al. 2016). The results of the modifications vary; sometimes they are good and justified, sometimes made with not clear understanding of the consequences of the deviation and end up harming the processes (Diebold et al. 2015; Eloranta et al. 2016).

2.3 Scaling Agile for large organizations

As described before, there are many benefits related to becoming Agile, and Scrum is the most popular way of implementing it in practice (Rigby et al. 2016; Sommer 2019). However, it is not optimal for large organizations, since in Scrum the focus is only on one team with less than ten people (Larman and Vodde 2017). Fortunately, there are several

frameworks and other practices defined for the needs of large organizations too, describing how the Agile methods can be implemented across multiple teams while still keeping the focus on one product or portfolio. In the first part of this section, the benefits and challenges of these practices are described, followed by views of how they should be implemented. After that, large-scale Agile frameworks are described on a high level. In the final part, the focus moves on to Large-Scale Scrum (LeSS) -framework. LeSS is a great example of large-scale Agile frameworks, since it is directly built on top of Scrum and relatively simple to adopt and use (Vodde et al. 2014) and understanding it helps to understand how these frameworks work altogether. In addition, LeSS is the framework implemented in the case organization of this study, and for this reason it is necessary to know its ideas and practices.

2.3.1 Benefits and challenges of scaling Agile

Large-scale Agile frameworks can bring many benefits to organizations. They attempt to resolve the issue with large team size and also several other issues of large organizations related to matters like project constraints, customer involvement, interacting with partners and end-users, business case approval, and project benefits realization (Alqudah and Razali 2016). When implementing large-scale Agile frameworks, organizations typically seek to increase their level of agility, improve the collaboration between teams (Uludağ et al. 2021), and to remain competitive in the market (Putta et al. 2021). The benefits organizations have gained reflect these expectations; benefits include better coordination of Agile teams, improved satisfaction, motivation and engagement of employees, enhanced software quality, frequent deliveries and additional value provided to the customer (Uludağ et al. 2021).

Despite the various potential benefits of large-scale Agile frameworks, there are also numerous challenges for organizations to overcome when implementing them. The most significant challenges relate to the size of the organizations; the greater the number of teams and the larger the size of projects, the higher is the need for communication, coordination, and managing dependencies and resources, which makes work methods and processes increasingly complex (Uludağ et al. 2021). Teams that are distributed to multiple sites also add to the complexity (Kalenda et al. 2018), as well as the interfaces between Agile and non-agile teams (Theobald and Schmitt 2020). According to Kalenda et al. (2018), the large size of the projects also poses challenges for Product Owners or Product Owner Teams, as handling all the requirements in the backlog grows into a

massive task and seeing the whole of the project becomes increasingly hard. A bigger group of Product Owners may be able to handle the situation better, but they often have other types of challenges, for example finding consensus may be hard for them (Kalenda et al. 2018).

Another factor raising challenges when implementing large-scale Agile frameworks is that when planning the implementation, some organizations may do the comparing of different frameworks quickly and without in-depth consideration. This can lead to implementing a framework that does not suit the needs and context of the organization (Conboy and Carroll 2019). Too ambitious and strict roll-out timeframe can also cause challenges, as well as putting too much pressure and workload on teams: these factors often lead to postponing and skipping meetings, not using any effort on continuous improvement, and even deteriorated quality of work (Kalenda et al. 2018). The direction of the change process also needs to be considered – a mix of top-down and bottom-up methods have often produced the best results, while one-way processes have made it difficult to succeed (Conboy and Carroll 2019).

Additionally, organizations tend to find balancing the Agile framework with the existing organizational structure challenging, particularly when the change is massive or the organizational structure keeps constantly evolving (Kalenda et al. 2018; Conboy and Carroll 2019; Uludağ et al. 2021). Another factor affecting the success of implementation is the willingness of the teams and individuals to transform (Kalenda et al. 2018). If the willingness is not high, the frameworks provide little guidance on how it can be increased (Uludağ et al. 2021). This is a big challenge, since lack of commitment often leads to severe failures in the implementation process. (Kalenda et al. 2018) Furthermore, Conboy and Carroll (2019) add that there is a lack of empirical case studies of the frameworks, which can cause troubles especially if an organization faces a problem that is not covered in the original framework papers. This challenge is amplified by the fact that the concepts and routines described in the frameworks are often rather abstract, which can lead to inconsistent interpretations in the organization, especially when applying the frameworks outside of their intended context (Conboy and Carroll 2019).

Other typical challenges when implementing a large-scale Agile framework are putting too much emphasis on 100% framework adherence (Conboy and Carroll 2019) and focusing on the rulesets and details of the framework instead of changing people's culture

and mindset (Uludağ et al. 2021). Conboy and Carroll (2019) describe how with an approach like this the principles and values behind the frameworks are not internalized, there is no flexibility, and some value may not be obtained. According to them, maintaining the autonomy of the developers can also be hard. Autonomy typically becomes increasingly difficult at scale, and strictly following a framework may hinder it even more (Conboy and Carroll 2019; Moe et al. 2021).

There are many recommendations how organizations can overcome the aforementioned challenges. These recommendations include for instance developing a common, well-defined vocabulary for the implementation, evaluating the organizational readiness beforehand, carefully identifying what new structural changes a specific framework causes in the organization and weighing-up the benefits and drawbacks of those, and planning for the optimal degree of transformation (Conboy and Carroll 2019). Training personnel, involving those actors that can push the agility further, and informing and engaging people in the implementation process have also been mentioned as ways to master challenges (Kalenda et al. 2018).

2.3.2 Implementing scaled Agile

When implementing Agile in a large organization, several factors are important to make the process effective and successful. According to Sommer (2019), the centrality of Agile principles and values, high employee-motivation, modularized operating architecture, and flexible planning are the key ingredients. Kalenda et al. (2018) complement this list by emphasizing the importance of receptive and motivated company culture, management support, unified values, and prior experience in Agile.

Kalenda et al. (2018) also note that even though there are various approaches for implementing Agile frameworks in practice and some of them may be more widely used than others, there is no consensus on the best set of practices. In various research articles, there is noticeable ambiguity in the advice and guidelines given about the subject matter. For example, in one article it is argued that effectively implementing Agile in a large organization requires integrating it in every aspect of the company (Sommer 2019), while another article claims that not every function needs to be organized into Agile teams, as long as they properly support the functions that operate in Agile ways (Rigby et al. 2018). It is also noted that the pace of the change can vary: the implementation can be done in one single step that requires lots of resources, total leadership commitment, receptive

culture and enough experienced Agile practitioners, or it can be done one small step at a time, with each team matching the implementation steps to its capabilities (Rigby et al. 2018). The variety of the methods and guidelines implies that there is no specific ruleset that works for all. Tailoring the implementation process to the needs and the context of the organization while maintaining the core values and principles of Agile seems to be the safest approach (Kalenda et al. 2018).

2.3.3 Large-scale Agile frameworks

There is a wide range of large-scale Agile frameworks to choose from and implement, and the latest annual State of Agile Report has listed which of them are used the most. Scaled Agile Framework (SAFe) is the most popular, and it has been growing over the years and is now used by 37% of the responders of the survey. It is followed by Scrum of Scrums (9%), Enterprise Scrum (6%) and Spotify Model (5%). After those, there are Agile Portfolio Management (APM), Disciplined Agile (DA), Large-Scale Scrum (LeSS), and Nexus, each of which is used in 3% of the responded organizations. (digital.ai 2021)

All of the frameworks mentioned attempt to scale Agile methods for large organizations, and they typically do this by adding new structures, roles and events on top of Scrum or other pre-existing Agile frameworks to enhance the communication and coordination between several Agile teams (Uludağ et al. 2021). Kalenda et al. (2018) tell that SAFe, for instance, adds three hierarchical levels to the organization: Portfolio level, Program Level, and Team Level. Portfolio Level defines the mission of the organization, while Program Level manages and supports the Team Level when they create the solution (Kalenda et al. 2018). There are many kinds of constructs and roles attached to this structure in SAFe, for example Agile Release Trains (ART), which synchronize multiple Agile teams for joint development of a product increment (Paasivaara 2017). Kalenda et al. (2018) note that Scrum of Scrums, on the other hand, has a simpler approach: it only adds one additional daily meeting called Scrum of Scrums or SoS on top of regular Scrum, where representatives of different Agile teams coordinate and synchronize the actions of the teams. Many other frameworks, LeSS for instance, fall in between these two in terms of complexity and the number of added roles and events (Kalenda et al. 2018).

Teams and organizations have different reasons and expected benefits when choosing the framework to implement, and satisfaction levels also alternate between the frameworks

(Putta et al. 2021), so it is valuable to carefully select the most suitable framework for the organization and its context. However, it is also common for the organizations to start by implementing one large-scale Agile framework, and then complement it with practices and methods from other frameworks (Laanti and Kettunen 2019) - this approach makes it possible for organizations with enough expertise to tailor unique solutions that work in their own context (Kalenda et al. 2018).

In the following section, the focus will move on to the Large-Scale Scrum (LeSS) - framework and its structures and practices, as that is the framework the case organization of this research chose to implement. In order to understand their implementation process, it is necessary to know the ideas and practices of the particular framework.

2.3.4 Large-Scale Scrum – LeSS

The Large-Scale Scrum (LeSS) -framework created by Craig Larman and Bas Vodde in 2007 is not the most popular out of the large-scale Agile frameworks (digital.ai 2021), but it is relatively simple and directly built on top of Scrum (Vodde et al. 2014), and for these reasons it is an appealing option for some organizations. LeSS shares its values and methods with Scrum and complements them with solutions that help to spread them in a large-scale context and it is aimed mainly for organizations where 2-8 teams work on one big project (Larman and Vodde 2017). It is much more lightweight than some other large-scale Agile frameworks and is built on the idea that large organizations do not need overcomplicated processes to succeed (Kalenda et al. 2018). These features make adopting LeSS relatively straightforward, especially for organizations that are already familiar with Scrum (Alqudah and Razali 2016).

The focus of LeSS is on the mindset, values and principles (Kalenda et al. 2018). It is built on ten principles that help to understand the framework and guide the implementation process in any specific context. The principles are (Vodde et al. 2014; Larman and Vodde 2017):

1. Large-Scale Scrum is Scrum – LeSS is not a new framework, only a scaled version of Scrum.
2. Transparency – it should be embraced and increased, and practices like short feedback loops and frequent communication help with that.

3. More with less – decreasing the amount of complexity and embracing simplicity in the organization makes the work more efficient.
4. Whole-product focus – having only one product, Product Backlog, Product Owner and common sprints throughout all the teams helps to keep the focus on the big picture.
5. Being customer-centric – identifying what is valuable and wasteful in the eyes of the customer helps making the right choices.
6. Continuous improvement towards perfection – improvement should be constant and never-ending.
7. Lean thinking – its two pillars, continuous improvement and respecting people, support embracing change.
8. Systems thinking – seeing the organization as a dynamic system and understanding the mental models and local optimizations affecting it helps to improve it and overcome obstacles.
9. Empirical process control – adapting the processes and practices in situationally-appropriate ways, instead of blindly using so-called best practices and ignoring the context, is the key for making changes successful.
10. Queuing theory – managing queues intelligently helps to reduce cycle time.

In LeSS the idea is to stay as simple and Agile as possible, and for that reason there are not many additional roles or processes in it compared to Scrum (Kalenda et al. 2018). There are three key responsibility areas: (1) product creation and delivery, (2) product vision and direction, and (3) organizational capability improvement (Larman and Vodde 2017, pp. 113–132). Vodde et al. (2014) explain that the first area, which includes creating and delivering the product and coordinating the development process, is the responsibility of the cross-functional, self-managing teams. They add that the second area, including tasks like providing vision and direction for the product, managing the Product Backlog, and supporting the strategic direction of the organization is the responsibility of the Product Owner, while the third area, which includes improving the capabilities of the organization, deciding about structures and policies, and making strategic decisions is the responsibility of the traditional managers, such as human resource managers. The Scrum Masters, responsible of coaching the organization and supporting continuous improvement, operate in the middle of the three key responsibility areas and support all of them (Vodde et al. 2014). There are several Teams and Scrum Masters in LeSS but only one product, Product Backlog and Product Owner, and one shared Sprint at a time

(Larman and Vodde 2017). The structure of the responsibility areas is visualized in Figure 5.



Figure 5. The responsibility areas in LeSS and the roles related to them, based on (Vodde et al. 2014)

In LeSS, the events are very similar to Scrum, some of them are just scaled to include multiple teams or their representatives, like Scrum Masters (Kalenda et al. 2018). According to Vodde et al. (2014), Sprint Planning is divided into two parts; Sprint Planning 1 is a meeting attended by the Product Owner and representatives from all the teams and the goal is to select which Product Backlog -items each team will work on, while Sprint Planning 2 refers to teams' internal meetings where the team members plan how to execute the chosen tasks. Vodde et al. also tell that in LeSS Daily Scrum -meetings and Product Backlog Refinement both happen the same way as in Scrum, and they are organized individually for each team. There can also be additional multi-team Product Backlog Refinement -meetings to increase alignment between teams (Vodde et al. 2014). The continuous coordination between the teams can take many forms; it can be a Multi-Team Daily Scrum, for instance (Alqudah and Razali 2016). However, Vodde et al. (2014) explain that de-centralized and self-organized approaches are preferred over formal, typically slow coordination mechanisms, and the members of the teams are encouraged to reach out and “just talk” with people from other teams whenever needed. They add that exchanging team members and having scouts monitoring other teams and

reporting back to their own team are also recommended ways for handling the coordination between multiple teams. In general, LeSS does not define one correct way for managing this issue, but advises organizations to find out what works best for them and to favour decentralized solutions to avoid bottlenecks (Vodde et al. 2014).

In the end of a Sprint there are three meetings in LeSS that are rather familiar from Scrum: Sprint Review, Retrospective and Overall Retrospective. In LeSS, Sprint Review is a meeting attended by everyone from the teams and all relevant stakeholders, and it should optimally happen in a large room where there are multiple areas with team members showing and discussing the items developed during the Sprint (Vodde et al. 2014). The Retrospective happens the same way as in Scrum, focusing on individual teams and how they should improve, according to Larman and Vodde (2017). They also note that in Overall Retrospective, on the other hand, the focus is on how to improve the overall system. This meeting is attended by the Product Owner, Scrum Masters and representatives from each team (Larman and Vodde 2017). The structure and the events of LeSS are visualized in Figure 6.

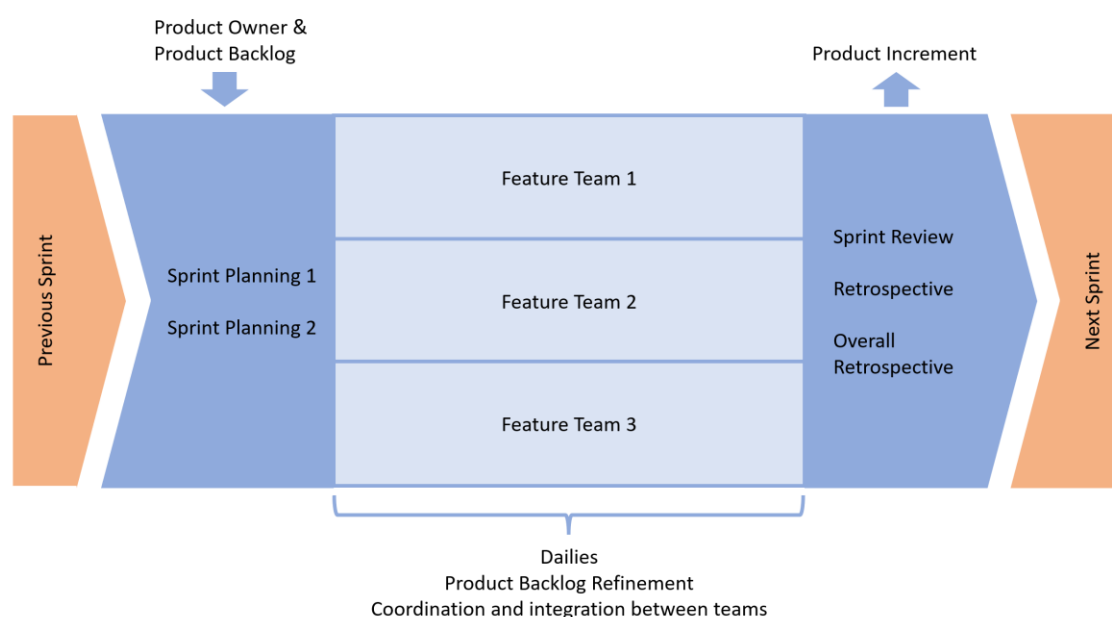


Figure 6. The structure and events of LeSS, based on (Larman and Vodde 2017)

Implementing the LeSS-framework in an organization can be challenging because it requires some fundamental changes: the structure of the organization needs to change and the existing assumptions about how organizations should work need to be challenged (Larman and Vodde 2017, pp. 53–71). There are some recommendations that can help and guide the implementation process. First of all, Vodde et al. (2014) have defined six

steps to follow in the process. The steps are (1) educating everyone to understand Agile and LeSS and why the implementation is done, (2) defining ‘product’, which means defining the scope of the product, the content of the Product Backlog, and who is suitable to be the Product Owner, (3) defining “done”, which means defining a common criteria for what tasks need to be completed so that a feature can be done, (4) having appropriately-structured feature teams that have the skills and resources to achieve their goals, (5) making sure only the Product Owner gives work to the teams to ensure staying focused, and (6) keeping project managers away from the teams, as the Product Owner and the Teams already share the project management responsibilities (Vodde et al. 2014). These steps help teams to work towards a common goal and have a shared understanding and instructions to follow.

In addition to the steps mentioned above, there are also three principles to follow in the LeSS implementation process: (1) “deep and narrow over broad and shallow”, which means that to minimize risks the adoption should first happen in one product group, and enough support should be provided to this group, (2) “top-down and bottom-up”, which recommends using both top-down and bottom-up methods in the implementation and get both employees and managers involved and motivated in the process, and (3) “use volunteering”, which recommends allowing and empowering people willing to take responsibility to do so (Larman and Vodde 2017, pp. 53–71). Additionally, using experienced coaches to guide the adoption process makes the possibility for success remarkably higher (Vodde et al. 2014). Also, in the beginning of the implementation process, it is recommended to follow the standard version of LeSS, since starting with it gives everyone a proper insight of the principles and practices and makes a solid foundation for further improvements and adjustments (Larman and Vodde 2017, pp. 53–71). It is essential to understand that the implementation process in LeSS never really ends but is a continuous effort, following the sixth LeSS principle, “continuous improvement towards perfection” (Vodde et al. 2014).

As mentioned before, LeSS is meant for 2-8 teams, which means the maximum amount of people in it is around 80. When the number of teams is more than that, a single Product Owner can no longer have a comprehensive overview of the product and its Backlog. That is why bigger organizations have their own version of LeSS called LeSS Huge. Its structure is visualized in Figure 7. (Larman and Vodde 2017)

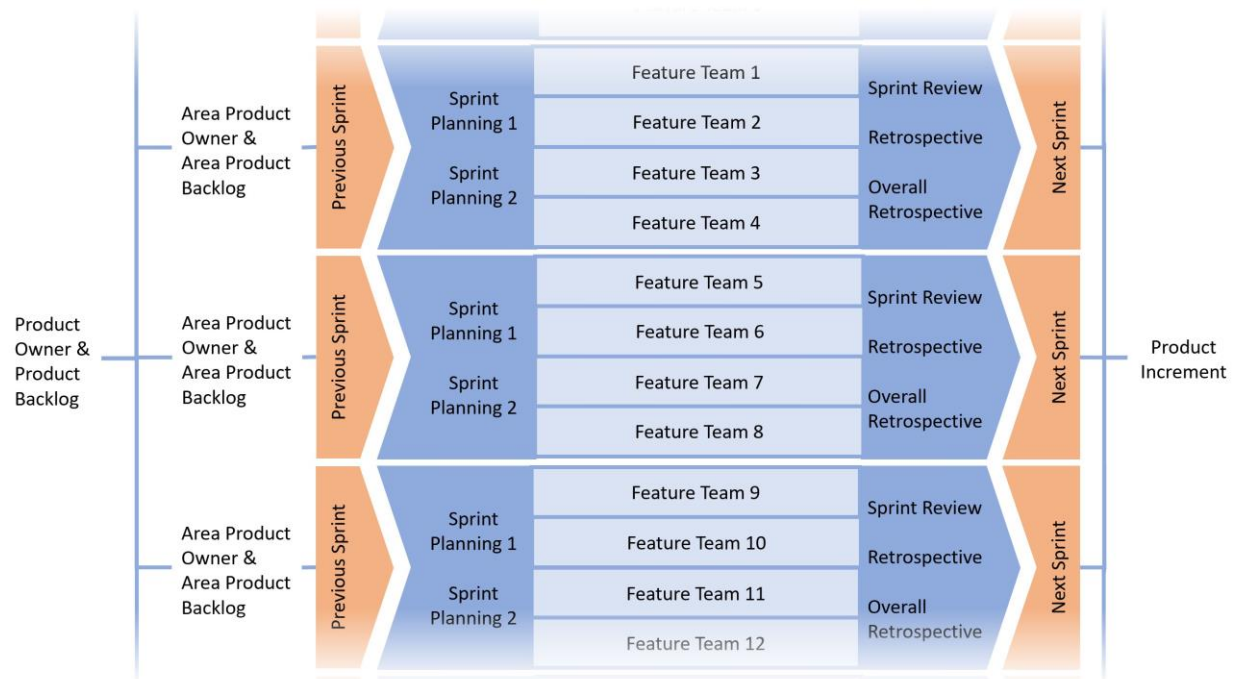


Figure 7. The structure of LeSS Huge, based on (Larman and Vodde 2017)

LeSS Huge is built on the same idea as LeSS; coordinating the teams as effectively and efficiently as possible, which means the number of additional structures has been kept minimal (Alqudah and Razali 2016). Larman and Vodde (2017) tell that in LeSS Huge the work is divided into several Requirement Areas that all have their own Area Product Backlogs and Area Product Owners. They describe that each Requirement area is a big entity that usually consists of four to eight teams, and from a team's perspective working in a Requirement Area is like working in the basic LeSS framework, the area is just smaller than one product. There is one thing that all the requirement areas share though, and that is the Sprint; all areas have a common Sprint that ends in one integrated product increment (Larman and Vodde 2017).

2.4 Agile and game development

Even though there are great quantities of academic publications on Agile software development, Agile game development has received far less attention. This seems like an overlook, considering that video games are a massive creative industry with a projected revenue of over \$175 billion USD in 2021 and on track to surpass 200 billion dollars in 2023 (Wijman 2021). The industry is filled with success stories, but also with complex projects, tough competition and changing conditions causing many projects and even studios to fail (Kanode and Haddad 2009; Hile 2020). There are also huge differences

between game development companies: while some of them have hundreds of employees, most of them have only a few. For example, in Sweden 94 % of the game companies employ less than 50 people (Milton et al. 2021, p. 15) and in Finland the median number of employees in game companies is only 8 and the average 25 (Hiltunen et al. 2020). Based on this, in this paper “large” game development company refers to a company with more than 50 employees.

According to Keith (2010, chap. 7), a typical game development process can be divided into four phases: concept, pre-production, production and post-production. He explains that generating ideas happens in the concept phase, and in the pre-production phase they are explored further; this includes finding and further developing the fun and engaging aspects of gameplay and planning how to build the assets for the game during production. Keith adds that in the production phase, the processes discovered in pre-production are put into use, and the team focuses on creating all of the content for the game. After that, in the post-production phase team polishes the game experience to a shippable quality and submits it (Keith 2010, chap. 7). Nowadays in large game development companies the game development teams are usually multidisciplinary feature teams, meaning that there are people with diverse skills and abilities, for example programming, art, and design skills, in the same team working together and focusing on specific features, like creating characters for the game (O’Hagan and O’Connor 2015). The goal is to have such teams that have all the skills needed for executing their tasks from start to finish.

Traditionally, game development has been based on the waterfall model or its variants, but for the latest decade or so game companies around the world have increasingly adopted Agile development frameworks and practices (Koutonen and Leppänen 2013; Graft 2021; McKenzie et al. 2021). There are no exact numbers of how widely Agile is now used in the game industry, but it has been researched on a more limited degree. The focus in this research has mostly been on Agile in general or on small-scale Agile frameworks like Scrum, while large game organizations have received no specific attention. One survey, where the sample size was 20 post-mortem project analyses made by game companies, defined that in the sample 65% of game companies applied iterative game development practices and 45% of those explicitly used Agile Practices (Politowski et al. 2016). Another survey, made for 20 Finnish game companies, found out that 95% of the respondents deployed Agile methods to some extent, and over 50% of those used specifically Scrum (Koutonen and Leppänen 2013). A third study, made by Kasurinen et

al. (2017), was an online survey and got 33 answers from various countries and continents. In this study, 61% of respondents did not follow any systematic development methodology and only 39% followed one. However, 67% of those who followed one reported using Scrum while 33% used other Agile approaches (Kasurinen et al. 2017). This suggests that if a game company uses any systematic process model, it is typically Agile. The sample sizes in these surveys are rather small and the results vary to some extent, but in general these indicate that a significant proportion of game companies use Agile methods in their development. One systematic literature review studying the software development processes used in game development also supports this view. In the review, 404 papers were analysed and 23 process models extracted of those; 47% of the models were purely Agile and 53% were hybrid processes that combine Agile and Waterfall methods (Osborne O'Hagan et al. 2014).

2.4.1 Reasons for game companies to implement Agile

There are several reasons why so many game companies turn to Agile. One of them is the successful outcomes other companies have had with it. Agile practices have for example improved team communication and responsibility taking and mitigated risks in companies (Ruonala 2016). The practices also help with scope and time management, and the iterative and incremental approach supports inventing, designing and testing features effectively, which can have a positive effect on game quality (Koutonen and Leppänen 2013). In addition, Agile was originally introduced to address challenges in traditional software development that are very similar to the challenges game development companies face today (Beck et al. 2001; Sommer 2019): the typical challenges in the industry relate to people, not technologies, and to themes like management, planning, and communication (Politowski et al. 2021). Solution for problems like these is not developing new tools or technologies but developing the processes and methods of working, which is the focus of the Agile Frameworks (Sommer 2019).

The most prominent challenge for the game industry is the complexity of the game development caused by the fact that video games are both advanced software products and works of art and creativity (Engström et al. 2018). The skills needed in the development are diverse, and as mentioned, the teams typically multidisciplinary. Managing such teams, their pipelines, and communication is complex because there are so many processes and ways of working that need to be understood and factors that need to be taken into account (McKenzie et al. 2021). The interests of different disciplines may

also conflict with each other (Koutonen and Leppänen 2013). Another typical factor increasing the complexity in game development is the need to work with external partners, as it causes a need to continuously communicate and co-operate with them. Many game development companies work with a separate publisher and have to align to their goals, schedules and deliverables (Ruonala 2016), and outsourcing parts of the game with external companies or freelancers is also common, especially in situations where the bandwidth of the company is not enough to finish the game (Game Developer 2009). As stated before, Agile and different Agile frameworks are made to environments with high amounts of complexity (Denning 2015; Schwaber and Sutherland 2020), which indicates them being suitable for game industry.

Another major factor making game development challenging is the rapidly changing nature of the industry. In changing markets, it is hard to anticipate what kinds of games will be successful, and companies need to be able to adapt to changes (Koutonen and Leppänen 2013). The industry is also highly risky since the competition is tough and publishers need to invest up to tens of millions for development projects that may or may not become successes (Koutonen and Leppänen 2013). Development also largely focuses on subjective matters such as making gameplay fun and engaging, which causes a need for constant iterating and may generate a need to do changes even in late parts of the process (Kasurinen et al. 2017). In addition, there are indications that as the quality and scope of video games keeps increasing, so do the expectations of the players (Wilson 2021). It becomes harder and harder to match the expectations, which makes game development projects even more complex and ambitious (Kanode and Haddad 2009; Wilson 2021). What makes Agile useful in an environment like this is how it aims to enable inspection and adaptation in environments with high amounts of variability (Denning 2015; Schwaber and Sutherland 2020).

The third typical challenge of game industry, crunch, is in a sense an unfortunate consequence of the other challenges. Crunch means periods of extreme workload when employees end up making tens or hundreds of hours of uncompensated overtime, and when it continues for long, it has negative effects on the wellbeing and motivation of the teams and individuals and on the quality of the product (Edholm et al. 2017). A need to crunch is very common in the industry: according to a survey organized by International Game Developers Association in 2021, one third of responded game developers said their job involved crunch-time and 22% said they need to work periods of extended work hours

but did not refer to it as crunch. During these periods of long hours, 20% of respondents worked 45-49 hours per week, 29% worked 50-59 hours and 26% worked for more than 60 hours (Weststar et al. 2021). Edholm et al. (2017) tell that typical reasons for crunch are upcoming deadlines combined with poor time management, planning issues and an excessive scope. There are indications that internalizing and correctly implementing Agile practices and especially the fourth value, “Responding to change over following a plan” helps to reduce the extreme workload and crunch and thus decreases the negative effects of it (Edholm et al. 2017). Fundamentally Agile and crunch strongly contradict with each other; as the eighth Agile principle states, “Agile processes promote sustainable development. The sponsors, developers, and users should be able to maintain a constant pace indefinitely” (Beck et al. 2001).

2.4.2 Challenges of game companies when implementing Agile

Even though Agile seems to be the solution for many of the common challenges in the game industry, the results with it are not always successful. Several articles argue the reason is that Agile and related frameworks are often partly misunderstood and misapplied in the organizations, which leads to results not being purely Agile, and that the root cause for this is typically lack of experience or training with Agile practices (Politowski et al. 2016; Ruonala 2016; McKenzie et al. 2021). The misapplication of Agile may lead to development problems and failures, which in turn may give the impression that Agile causes the observed malfunctions, even though the actual problem is in the misuse of it (McKenzie et al. 2021). Koutonen and Leppänen (2013) also recognize that in many cases implementing Agile has not been enough to solve all the difficulties, mentioning for instance crunch and feature creep remaining in several companies.

Another factor making implementing Agile challenging in game companies is the wide range of processes companies use and the big differences between them (McKenzie et al. 2021). A model that would be suitable for every game company does not exist, so the companies need to carefully consider which model suits best their unique context (Osborne O’Hagan et al. 2014). Furthermore, McKenzie et al. (2021) argue that there is no single Agile framework that would by itself be sufficient to meet all the needs of game development, especially when it comes to the multidisciplinary nature of the industry. According to them, the frameworks need to be adapted or used in conjunction with each other to get successful results.

2.5 Human skills

Human skills refer to competencies tied to human interaction. They can also be defined as non-technical skills (Berdanier 2021) or as a “collection of people management skills” (Matteson et al. 2016). Traditionally, these skills have been called “soft skills”, but that term has been criticized lately, as it is seen to undervalue their relevance and significance (Berdanier 2021). For that reason, in this paper the term “human skills” is used in place of “soft skills”. This section focuses on human skills and their relationship with Agile and game industry. In the beginning of the section, the focus is on the big picture: understanding what human skills are, what kind of human skills there are, and how they can be categorized. The importance of human skills in leading and managing people is also covered. After that, the focus moves on to the relationship of human skills and Agile. The role of trust, resilience, and psychological safety is discussed, as these are the most researched human skills in that context. In addition, the role of human skills in implementing Agile is covered. In the last part of the section, the importance of human skills in the game industry is examined. That subject has not been researched much, but several non-scientific articles argue that they are needed to achieve success (Hamilton 2019; Thornton 2020).

The concept of human skills has not been formally defined, which makes it rather ambiguous (Matteson et al. 2016). Many articles have attempted to understand the subject by listing these skills, but it appears that there is no full consensus of a definite list. However, the lists often mention a variety of the following skills: communication skills, customer service competencies, emotional intelligence, ethics, feedback skills, leadership skills, mentoring competencies, problem-solving skills, abilities to resolve conflicts, resilience, self-management, sensitivity to diversity, sociability, and teamwork skills (Parente et al. 2012; Matteson et al. 2016). In research these skills have received uneven amounts of attention. Communication skills, for example, are mentioned in many contexts (McHugh et al. 2012; Wawro 2017; John Hayes 2018, pp. 212–254), while things like mentoring competencies have been considered much more rarely. Yet another approach to identify and define human skills has been to divide them into three categories: (1) interpersonal skills, such as communication and teamwork skills, (2) thinking skills, such as decision-making, and (3) personal skills, such as self-management and sociability (Matteson et al. 2016). Human skills can also be compared against technical skills. Technical skills, referring to abilities needed to perform specific tasks such as

programming, are typically easier to identify, influence and measure, while human skills are more indefinite and abstract (Parente et al. 2012).

Human skills are highly important to managers and leaders. Katz (1974) has created three categories that cover the skills these roles need to be effective, and they are (1) technical skills that are specific to a particular field, (2) human skills, and (3) conceptual skills that require analytic and diagnostic abilities and are sometimes referred as strategic skills. This three-part structure still widely acts as the foundation when categorizing managerial skills, and the relative importance of the skills varies with the level of managerial responsibility, as the conceptual skills become more critical on higher levels (Parente et al. 2012). Parente et al. (2012) studied the relationships between these three categories and discovered that human skills are particularly important, as there are indications that they are needed for improving the technical skills efficiently, and together with the technical skills they enable and empower acquiring conceptual skills. Stevenson and Starkweather (2010) on the other hand defined which human skills are seen to be important for successful project management in information technology projects. These core competences are attitude, leadership, verbal and written skills, the ability to communicate at multiple levels, and the ability to deal with change and ambiguity.

2.5.1 Human skills and Agile

As mentioned before, not all human skills have received the same amount of scientific attention. When it comes to their relationship to Agile, research effort has focused on these aspects: trust, resilience, and psychological safety. Resilience is a human skill itself, while trust and psychological safety can be seen both as enablers for practicing and developing human skills and as results of human skills applied successfully. Trust, for example, has a remarkable role in intra-and inter-team coordination of Agile teams; since the teams are self-managing, other stakeholders need to trust them and their ability to do the right decisions and work efficiently, and the team members also must trust each other (McHugh et al. 2012). Higher level of trust is connected to improved team performance and better relationships and cohesiveness amongst team members (McHugh et al. 2011). However, trust is not only a requirement for using Agile but it is also a consequence of it; these two appear to have a cyclic relationship. Many Agile practices, for example daily meetings and retrospectives increase the amount of human skills such as communication, knowledge sharing, feedback, and accountability practiced in the teams, which in turn raises the level of trust (McHugh et al. 2012).

Another factor emphasized in Agile is responding to change. According to Diegmann and Rosenkranz (2021), to do this well teams and individuals need to have good resilience, which means they need to be able to withstand disruptive factors and tolerate stressful situations. This can improve their efficiency and problem-solving abilities and lead to better results (Diegmann and Rosenkranz 2021).

Psychological safety is also an important element in Agile. Agile approach is built on collaborative relationships among team members, where human skills like communication and teamwork skills are highly valued (Beck et al. 2001). Thorgren and Caiman (2019) describe that for the collaboration to work, there needs to be psychological safety; team members need to feel safe to offer ideas, ask for help, admit mistakes, and provide feedback. According to them, human skills are crucial for creating this environment of psychological safety: when different stakeholders apply human skills, for example leaders welcome comments and feedback and teams take a collective responsibility over their performance, the level of psychological safety increases. Diegmann and Rosenkranz (2021) add that psychological safety is essential for many things in organizations, including successfully implementing and practicing Agile, having active participation, engagement, and information sharing, and being able to learn from failures. These matters eventually lead to better outcomes and improved organizational learning (Diegmann and Rosenkranz 2021).

Human skills can have a significant role in the process of implementing Agile, too. John Hayes (2018, pp. 212–254) for example sees leading and managing people issues as an integral part of implementing change. He particularly stresses the importance of communication, as it plays a key part in helping others recognize the need for change and motivating them to support the implementation. John P. Kotter (1995) shares this view, as when listing the eight steps of organizational transformation he emphasizes a variety of human skills, especially communication. Effective communication through all available channels is needed for instance for establishing a sense of urgency, for communicating the vision throughout the organization, and for institutionalizing new approaches. When it comes to other human skills, Hayes (2018, pp. 233–270) mentions that skills like emotional intelligence and support, sensitivity to diversity, and leadership skills are valuable for leaders when implementing change, while skills like resilience help teams and individuals adjust to it. In addition, offering support for employees and

understanding their views is said to be particularly important in Agile implementation as well as in other major change processes (Koutsikouri et al. 2020).

2.5.2 Human skills and the game industry

When it comes to the role of human skills in the game industry, the scientific research on the area is fairly limited, as there are only a few papers related to it. One study made of student teams developing mobile video games indicates that there is a strong relationship between trust and performance; the higher the level of trust, the better the performance of the team (Cook et al. 2020). Another study analysed the qualities sought on game development job postings and found out that various human skills, such as attitude, communication skills, and interpersonal skills are mentioned frequently (McGill 2008). These studies both signal that human skills play an important part in game development.

The results of the scientific studies are supported by numerous non-scientific articles stating that human skills have a vital role in game industry. Hamilton (2019) argues that game development is a creative, collaborative effort where the most effective teams are typically those filled with solution-oriented people showing empathy and appreciating and building upon each other's ideas, while Thornton (2020) states that human skills are crucial for various roles operating in game development. Providing and receiving feedback, managing other people, resolving conflicts and mediating discussions is a part of the everyday job of many roles, and each of these things requires empathy, diplomacy and tact (Thornton 2020). The tasks of certain roles are even more tightly connected to human skills. According to Shin (2009), producers are often responsible of matters like facilitating the meetings and communicating the plans and schedules of the project, and in these duties they need solid communication skills. Furthermore, the interaction between a producer and their team is ideally built on trust, honesty, and respect, and achieving these things requires highly developed human skills (Shin 2009). Game designers also need to be skilled in this area since they often act as bridges across multiple teams and departments in the development and have a significant role in setting the tone of the project for everyone (Thornton 2020).

The articles describing the connection between human skills and game development often offer advice how game development professionals and students can develop and utilize human skills in their work life. This advice is linked to themes like being sincere and approachable, not taking sides, staying calm, and encouraging others to express their

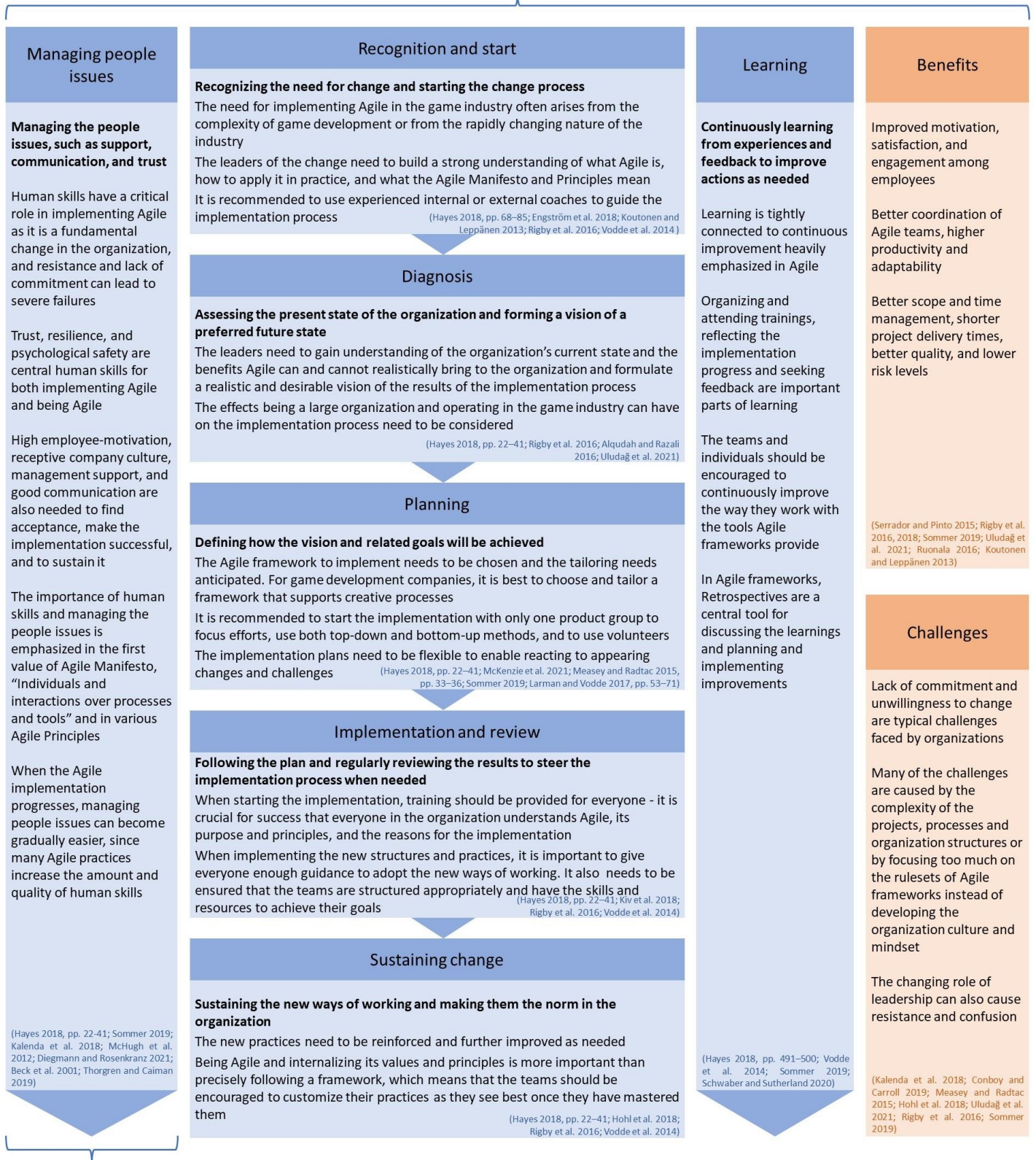
opinions (Thornton 2020). The importance of human skills in keeping the conflicts that happen in the organization healthy and productive is also mentioned. Improving one's communication skills and focusing on clear, concrete specifics can make the difference between productive and toxic conflict (Wawro 2017). However, it is also noted that human skills are tied to our personalities and life experiences and that they are often not as straightforward to teach and learn as technical skills like programming or 3D-modelling (Hamilton 2019).

2.6 Synthesis of the literature review

In this section, the key issues of the literature review are synthesized in light of the research questions. The first research question “How do large game development organizations implement Agile working practices, and what are the related benefits and challenges?” focuses on the process of implementing Agile. As Sommer (2019) notes, implementing Agile and related frameworks is a huge change for any organization, and to make it successful a change management approach is needed. For this reason, the change process structure defined by Hayes (2018) introduced in section 2.1.4 is used as a basis for the synthesis. Hayes's structure has a simple but still comprehensive view on change process as it consists of seven core activities that are recognition and start, diagnosis, planning, implementation and review, sustaining change, learning, and managing the people issues (John Hayes 2018, pp. 22–41). The seventh activity, managing people issues, is tightly connected to the second research question of this study “What kinds of roles do human skills play when implementing Agile in large game development organizations?”, which further increases the suitability of the structure for the synthesis.

The Agile implementation process in large game development organizations, developed based on the previous research and structured following the Hayes's (2018) change process is visualized in Figure 8 and further described in the following paragraphs. The visualization is also included as Appendix 1.

RQ1: How do large game development organizations implement Agile working practices, and what are the related benefits and challenges?



RQ2: What kinds of roles do human skills play when implementing Agile in large game development organizations?

Figure 8. Agile implementation process in large game development organizations, following the structure of the change process defined by John Hayes (2018)

When implementing Agile in a large game development organization, the first step is recognizing the need for change and starting the change process (John Hayes 2018, pp. 68–85). In the game industry, the need for change often arises from the complexity of game development (Engström et al. 2018) or from the rapidly changing nature of the industry (Koutonen and Leppänen 2013) that make it hard to succeed or even survive (Kanode and Haddad 2009; Hile 2020). Agile was created for environments with high amount of complexity and variability, which makes it a good option for handling these challenges (Denning 2015; Schwaber and Sutherland 2020). When starting the change process, it is crucial for the leaders of the change to build a strong understanding of what Agile truly is, how it can be applied in practice, and what the Agile Manifesto and Agile Principles mean (Rigby et al. 2016). The Manifesto and principles are the foundation for every Agile practice and framework (Doug 2015, chap. 3; Hohl et al. 2018), and the importance of internalizing them is emphasized in several papers (Kiv et al. 2018; Sommer 2019). Having prior experience in implementing and practicing Agile is very helpful (Kalenda et al. 2018), and for this reason using experienced internal or external coaches to guide the implementation process is heavily recommended (Vodde et al. 2014).

The second phase in the change process is diagnosing, which means assessing the present state of the organization, including its problems and opportunities, and forming a vision of a preferred future state (John Hayes 2018, pp. 22–41). There are many ways to gather the data for the diagnosis, such as observations, questionnaires, and informal and formal interviews, and then the analysis can be done using various qualitative and quantitative methods (John Hayes 2018, pp. 144–160). To create a realistic but desirable vision, it is important to understand what benefits Agile can realistically bring to the organization, what it cannot, and how well the anticipated payoffs justify the expense and effort of the implementation process (Rigby et al. 2016). In addition to understanding Agile, the leaders also need to build understanding about how being a large organization and operating in the game industry affect the implementation process and the final results. In large organizations, for example, Agile frameworks have the potential to resolve issues with large team size and improve the communication, coordination and adaptability inside and between teams (Alqudah and Razali 2016), but at the same time the implementation process is typically remarkably harder in these organizations due to their size and complexity (Uludağ et al. 2021). In game development companies Agile implementation is often not enough to solve all their typical problems, which means that issues like crunch

and feature creep may remain if no further actions are taken (Koutonen and Leppänen 2013).

In the planning-phase the focus is on defining how the vision and related goals will be achieved and choosing the change strategy (John Hayes 2018, pp. 22–41). In this phase, leaders of the change must choose which Agile framework or other approach to implement based on the context and needs of the organization (Measey and Radtack 2015, pp. 33–36; Sommer 2019). It is necessary to pay careful attention to this step because unlike the values of Agile that work in many kinds of environments, Agile methods and frameworks are more context specific (Hohl et al. 2018). In the game industry, the choice is often not straightforward as there is no single Agile framework that can by itself meet all the needs of game development (McKenzie et al. 2021), so tailoring is needed to make sure the new ways of working support the creative processes. However, it is recommended to start the implementation with as standard version of a framework as possible because that makes a solid foundation for further improvements and adjustments (Larman and Vodde 2017, pp. 53–71). There are also some additional recommendations to consider when planning the implementation of Agile: it is, for example, recommended to start the implementation with only one product group to focus efforts and minimize risks, to use both top-down and bottom-up methods to get all the levels of the organization involved in the implementation process, and to use volunteers whenever possible to make the implementation successful (Larman and Vodde 2017, pp. 53–71). In addition, Sommer (2019) stresses the importance of keeping the implementation plans flexible, so it is easy to react to emerging changes and challenges.

When implementing the change and reviewing progress, the focus shifts from planning to executing the plan. Training forms the foundation for starting this phase; it is crucial for success that all members of the organization understand Agile, its purpose and principles, and the reasons for the implementation (Kiv et al. 2018). Some of the concepts described in Agile frameworks are rather abstract, which can lead to inconsistent interpretations and to difficulties in the implementation process (Conboy and Carroll 2019), and training helps to avoid this. Furthermore, the implementation process needs to be coordinated carefully; regularly reviewing the results and seeking feedback helps to steer the implementation process when needed and to destroy barriers to Agile behaviour as they appear (Rigby et al. 2016; John Hayes 2018, pp. 22–41). Also, it is important to ensure the teams receive enough guidance during the implementation phase and that they

are structured appropriately and have the skills and resources to achieve their goals (Vodde et al. 2014; John Hayes 2018, pp. 433–462), since agility is largely based on the idea that teams are self-managing and cross-functional (Beck et al. 2001; Schwaber and Sutherland 2020).

In the implementation and reviewing -phase, organizations tend to face many kinds of challenges that threaten the success of the process, such as unwillingness to transform the existing ways of working and lack of commitment (Kalenda et al. 2018; Conboy and Carroll 2019). The typical reason behind challenges like these is focusing too much on the rulesets and details of Agile frameworks instead of changing the organization culture and mindset (Measey and Radtack 2015; Hohl et al. 2018; Uludağ et al. 2021). The changing role of leadership also causes challenges in many organizations; in Agile managers need to give space for self-managing teams to make their own decisions, which can be hard to accept, especially without proper training (Rigby et al. 2016; Sommer 2019). Moreover, large organizations often have their own additional challenges; the greater the number of the teams, the higher is the need for communication and coordination and the more complex are the dependencies and work processes in the organization (Uludağ et al. 2021). Balancing the new framework with existing, complex structures is also hard in large organizations (Kalenda et al. 2018; Conboy and Carroll 2019).

After the implementation is done, the change still needs to be sustained and the new ways of working made the norm (John Hayes 2018, pp. 22–41). Some argue that the implementation process is never really finished, as the new practices need to be reinforced from time to time (Vodde et al. 2014) and the teams encouraged to customize their practices once they have mastered them (Rigby et al. 2016). It is more important to find the ways of working that suit the organization and help them to be as Agile as possible than to precisely execute a framework (Hohl et al. 2018), so modifying the practices is highly recommended.

Implementing Agile has benefited many organizations, including the large organizations operating in the game industry, in various ways. Typical benefits are improved motivation among employees, higher productivity and adaptability, shorter project delivery times, better quality, and lower risk levels (Serrador and Pinto 2015; Rigby et al. 2016, 2018; Sommer 2019). The benefits large organizations in particular have gained include better

coordination of Agile teams, improved employee satisfaction and engagement, and additional value provided to the customer (Uludağ et al. 2021) In addition to these, game development organizations have reported improved team communication and responsibility taking (Ruonala 2016), better scope and time management, and higher quality in developed games, thanks to the iterative approach of Agile (Koutonen and Leppänen 2013). However, not all Agile implementations are successful; problems in previous steps, lack of commitment, and too little training and experience in Agile are typical reasons for failures (Politowski et al. 2016; Ruonala 2016; McKenzie et al. 2021).

Learning is an activity that needs to take place throughout the implementation process, from the start to finish and even beyond. Continuously learning from experiences and feedback makes it possible to adjust and improve actions as needed (John Hayes 2018, pp. 491–500) and is strongly related to continuous improvement emphasized in Agile; the sixth principle of LeSS-framework, for example, states that improvement should be constant and never-ending (Vodde et al. 2014). Constant learning and improvement are important for those leading the change, but also for every individual touched by it. Sommer (2019) for example argues that Agile teams should be encouraged to continuously improve the way they work with the tools Agile frameworks provide. There are many built-in solutions for this in various Agile frameworks, such as Retrospectives, where the aim is to find ways to increase the quality and effectiveness of work (Schwaber and Sutherland 2020).

Managing the people issues, such as communication, trust, and support, is another activity that needs to happen throughout the change process (John Hayes 2018, pp. 22–41). The second research question “What kinds of roles do human skills play when implementing Agile in large game development organizations?” is closely tied to this activity. Based on the existing research, it appears that human skills have a critical role in the implementation process. Implementing Agile is a fundamental change that requires everyone in the organization to change their beliefs and attitudes to make the implementation successful and long-lasting (Sommer 2019), and resistance and lack of commitment can lead to severe failures in the implementation process (Kalenda et al. 2018). This means managing the people issues and making sure people are supported and motivated is critical for success. High employee-motivation, receptive company culture, good communication, and management support in particular are mentioned as crucial factors in making the implementation successful (Kalenda et al. 2018; Sommer 2019),

while trust, resilience, and psychological safety appear to have an important role in practicing Agile (McHugh et al. 2012; Diegmann and Rosenkranz 2021).

In addition, the significant role human skills have in implementing and practicing Agile is emphasized in the first value of Agile Manifesto “Individuals and interactions over processes and tools” (Beck et al. 2001) which, according to Measey and Radtack (2015), underlines the importance of having motivated individuals in the organization that can effectively interact with each other. The fifth and sixth Agile principles “Build projects around motivated individuals. Give them the environment and support they need and trust them to get the job done” and “The most efficient and effective method of conveying information to and within a development team is face-to-face conversation” reinforce this view as they highlight various human skills and related themes, such as trust, support, communication, and information-sharing (Beck et al. 2001). The role of human skills in Agile also appears to be cyclic: while advanced human skills make it easier to be Agile, many Agile practices such as Sprint Plannings and Retrospectives also increase the amount and quality of human skills, such as communication, knowledge sharing, accountability and feedback practiced in the organization (McHugh et al. 2012; Thorgren and Caiman 2019). This indicates that when the process of implementing Agile progresses, managing the people issues may become gradually easier.

3 RESEARCH PROCESS

In this paper, the aim is to investigate the way large game development organizations implement Agile and the effect human skills have on that implementation process. This section discusses the empirical research conducted, including the research design (section 3.1), case context (section 3.2), and methodologies used in collecting and analyzing the data for the study (sections 3.3 and 3.4).

3.1 Research design

The empirical research is conducted as a single-case study. Yin (2014, chap. 1) describes case study as an in-depth empirical study where a contemporary phenomenon is examined within its real-world context, and he also emphasizes that the approach works well when the aim is to study research questions that focus on qualitative things i.e., “how” or “why” something happens. As the objective of this study is to build understanding of how Agile is implemented in large game development companies, case study -approach was a natural choice.

Furthermore, this study aims to elaborate theoretical understanding of implementing Agile practices in large game development organizations based on the theory elaboration approach described by Ketokivi and Choi (2014). The existing theory of the topic was introduced in chapter 2 and synthesized in section 2.6, where the current understanding of Agile implementation in large game development organizations was linked to the change process defined by Hayes (2018). In this paper, the intent is to elaborate this through empirical research to see how well the existing theory applies to a large game development organization and what is missing from it. As mentioned in chapter 2, not all areas of Agile implementation have been comprehensively studied before, and especially the viewpoint of large game development organizations has been relatively absent in the previous research.

3.2 Case context

The case of this study is a large game development company located in Europe that currently employs around one hundred people, and more specifically the way the Agile implementation is done there. Over the last few years, the case company has significantly

grown, and the growth is still ongoing. They develop AAA games, meaning games with high development and marketing budgets, and their latest game was released in the first half of 2021. After the release, most of the development teams focused on creating additional content to that game, until in spring 2022 the content was finished and published. Then the teams shifted their focus on the next big project.

In the case company, the first step towards Agile was arguably taken in the end of 2020 when they conducted a survey called “Towards Better Production” (TBP) in the organization. At the time, the development of the latest game of the company was still ongoing. The results of the survey indicated that the current ways of working in the organization were not sustainable: developing the game had had several difficulties, and many were stressed and felt unhappy with the way work was done. Something needed to change. In the first half of 2021, more detailed diagnosis was conducted and planning the change and preparing for it was started. Then in autumn 2021 when developing the additional content for the latest game was started, the organization began following the Scrum framework. In spring 2022, when the teams moved on to the next project, it was turned into the Large-Scale Scrum (LeSS) -framework. After that, the company has focused on practicing the new ways of working and modifying and improving them to better suit their needs. The timeline of the Agile implementation in the case company is described more in detail in Figure 9.

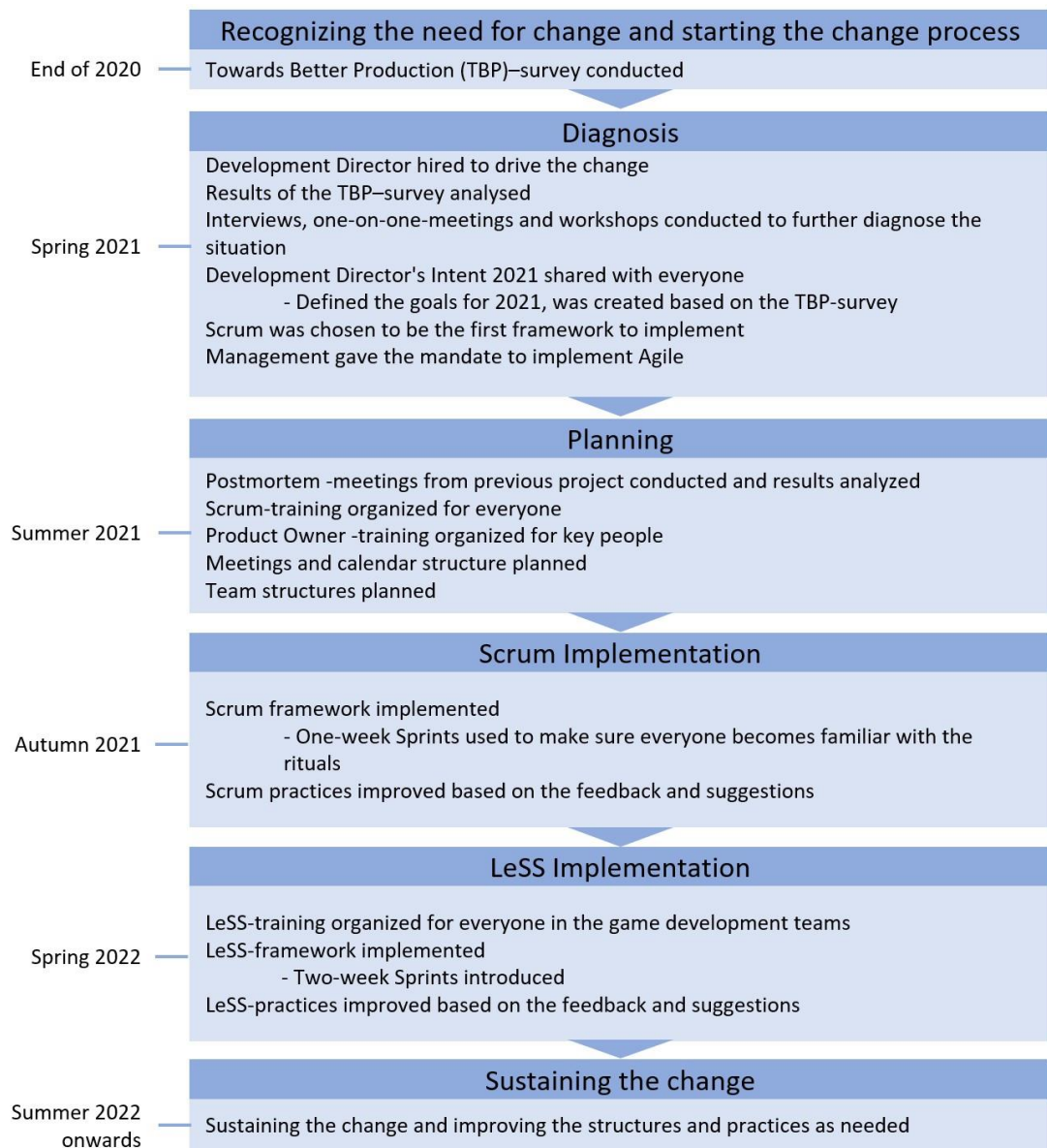


Figure 9: Timeline of the Agile implementation in the case company

The case is considered to be a theoretically suitable context for this study for several reasons. The Agile implementation in the case company is carried out in a rather concise timeframe, which makes it possible to both review the implementation process as a whole and to examine its phases individually. In addition, the fact that Agile practices are implemented throughout the organization makes it possible to study the implementation in various levels and sub-contexts if needed. Also, the researcher is well familiar with the case organization, its employees, and its situation prior to the Agile implementation, and this familiarity and understanding is a good foundation for profound research.

3.3 Data collection

The main method for collecting the data for the research were interviews: ten people were interviewed, one of them twice, and every interview lasted approximately 60 minutes. In the beginning a few knowledgeable interviewees were selected and during the interviews snowball sampling methods were used to identify other informed participants (Biernacki and Waldorf 1981). The interviewees chosen worked in different levels and positions in the organization, varying from the CEO of the organization to people that focus on very practical, hands-on game development work. They also represented several different disciplines, including for example Production, HR, Design, and Art. As the aim of the interviews was to form a comprehensive view of the case, interviewing people from versatile positions and disciplines was considered a reasoned choice.

The interviews were semi-structured, and the questions asked were open-ended. The interviews were recorded and transcribed for further analysis. Every interview followed a common structure: First, the focus was on the interviewee's background and their current role and responsibilities in the organization. The next focus point was the Agile implementation process and its different phases, including the reasons why the process was started, what were the practices implemented, and how the interviewee sees the future steps of the implementation process. The effects implementing Agile has on creative work and organizing it and the role of human skills in the implementation process were also discussed. In the interviews, the aim was to encourage the interviewee to provide their own narrative of the topic at hand, and for this reason the questions were open-ended and they were asked only to guide focus and gather details when needed. Interview approach like this is typical and appropriate for a qualitative case study (Yin 2014).

To support the interview data, organization's existing documentation of the Agile implementation process was collected. Combining these two data collection methods is typical and suitable for qualitative, case-based research (Yin 2014), and supports utilizing triangulation (Jick 1979). The researcher had access to both public and internal documents, and in the end over 20 documents were identified for further analysis and data triangulation. The documents collected mainly included announcements and progress records. The data was used to form a valid understanding of the implementation process, to verify the timing of key actions, and to see how the implementation process was communicated throughout the organization. The interview recordings and transcriptions

and the documentation collected were stored in a cloud service behind a password and a two-factor authentication to ensure data security.

3.4 Data analysis

The data analysis method used in the research is qualitative content analysis. The method is used to interpret meaning from the text data, and it is considered to be a flexible and pragmatic way for developing and extending knowledge (Hsieh and Shannon 2005). During the analysis, the categories for the data were derived directly from the data itself and also from the relevant theory, which means that both conventional and directed approaches of content analysis were used. This was done because even though implementing Agile has been researched quite widely, implementing it in large game development organizations has not received much attention. In this situation, it appeared appropriate to utilize both the existing theory but also to stay open for new learnings and perspectives rising from the data (Hsieh and Shannon 2005).

Directed content analysis approach presents some challenges to the naturalistic paradigm, which are focusing only on data that is supportive to the pre-existing theory, giving cues to participants to answer in a certain way in the interviews, and overemphasizing the existing theory and thus becoming blind to contextual aspects of the phenomenon (Hsieh and Shannon 2005). In this study, these were acknowledged and proper action was taken; open-ended questions were used to reduce the likelihood of interviewees taking cues, data analysis was done in great detail to ensure that all the relevant factors were recognized no matter if they were supportive or non-supportive to the existing theory, and the fact that the researcher is well familiar with the organization and its history supported recognizing the contextual aspects. In addition, simultaneously with performing the content analysis for the interview data, the collected documentation was used for triangulation – using different kinds of data improves the research accuracy as it provides multiple viewpoints for the studied phenomenon (Jick 1979).

Following the methods of qualitative content analysis (Hsieh and Shannon 2005), the data analysis was started by reading through all the transcribed data to gain an understanding of the whole. Then the data was read word by word and all the relevant parts of it were colour coded and copied into a Google Sheets -table. Initial labels were attached to the copied parts, and notes were made about the researcher's first impressions and thoughts.

As the work continued, more and more text parts were grouped under the same labels, which later became the codes of the data. When all the transcribed data was processed, the codes were grouped into meaningful clusters based on how they were related; some of the high-level clusters rose from the existing theory and the research questions, others from the data itself. These clusters became the themes and later they were still grouped under five overarching categories. Developing this categorisation included drawing multiple mind maps to strengthen the understanding of the relationships between all the codes, themes, and categories. Definitions for those were then further improved and example quotations were identified from the data to prepare to report the findings. The results of the analysis are presented in the next chapter.

4 RESEARCH FINDINGS

In this chapter, the results of the analysis are introduced. They are divided to five overarching categories: (1) Reasons for implementing Agile in the case company, (2) Practices used in the implementation process, (3) Benefits gained during the implementation process, (4) Challenges faced during the implementation process, and (5) Human skills and their relationship to the Agile implementation. In Figure 10, there is a visualization of these categories and the themes connected to them. Each category is thoroughly explained in its own section on the following pages, and in the beginning of each section there is also a visualization of the particular category. In those visualizations there are also codes attached to the themes that provide more detailed information of the subject matter.

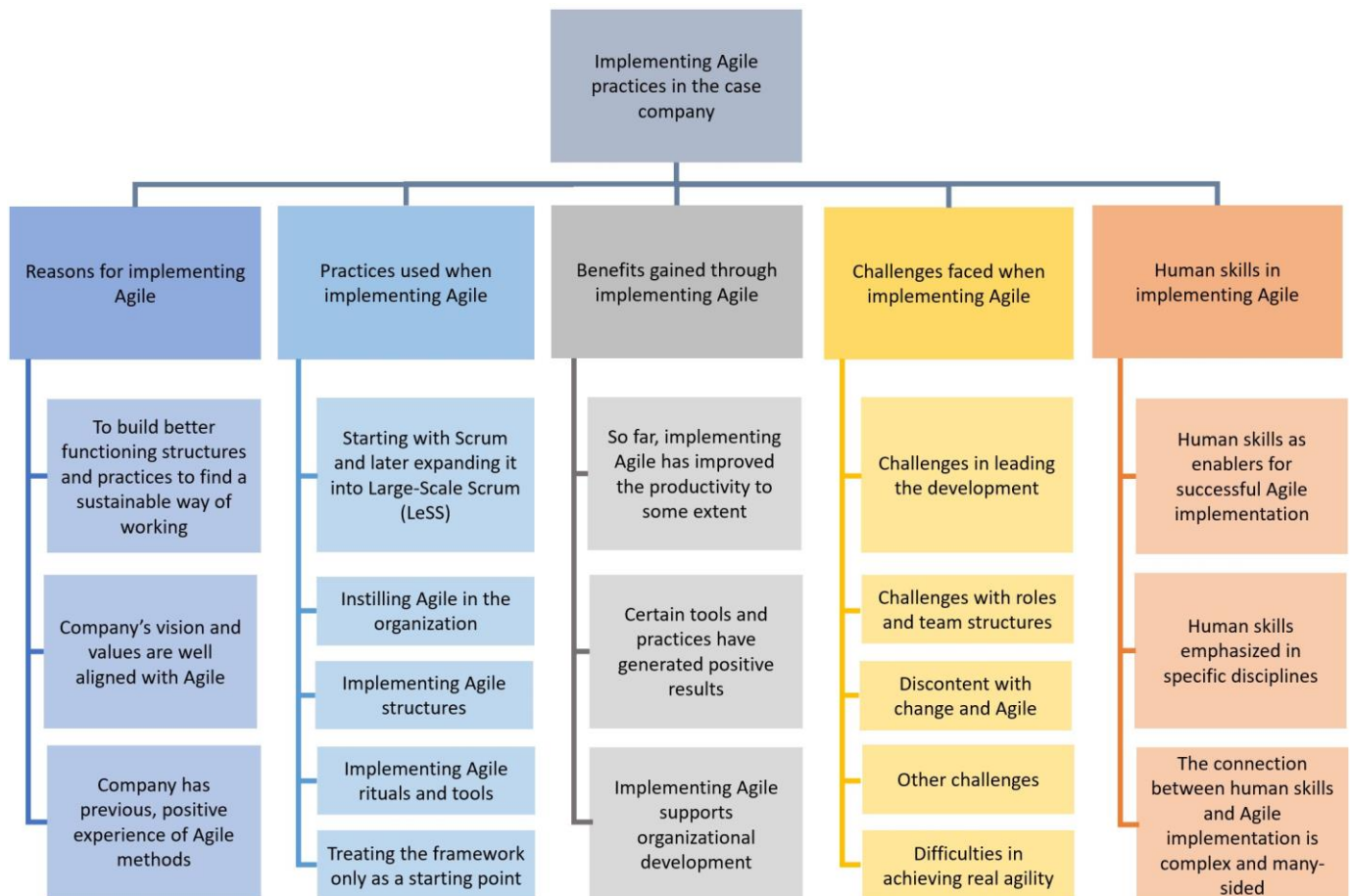


Figure 10: Visualization of the reasons, practices, benefits, challenges, and human skills connected to Agile implementation in the case company

4.1 Reasons for implementing Agile in the case company

“Reasons for implementing Agile” is an overarching category that was derived from the interview data; in many of the interviews, the reasons behind the implementation were brought up and discussed in depth. The reasons and their classification are visualized in Figure 11 and explained in the following paragraphs.

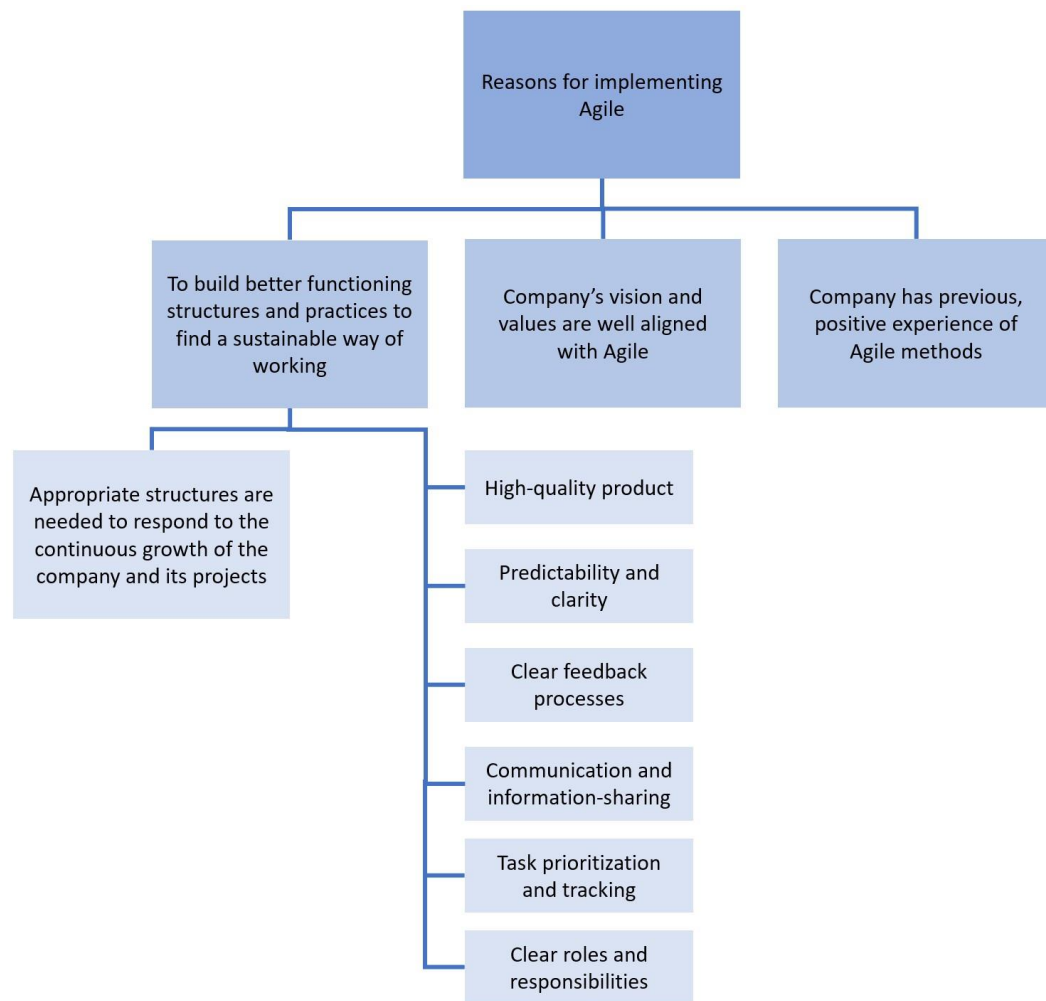


Figure 11: Reasons for implementing Agile in the case company

Based on the interview data, it appears that the most important goal of the implementation is to build more structure and defined processes for the organization. The need behind this seems to arise from the fact that the company and its project sizes and complexity have grown rapidly and significantly over the last years and the growth is still ongoing. As one interviewee stated: *“Now that the size of the team is growing and the project has grown, maybe we have needed common rules and common language to be able to work in a more structured way and to gain better understanding of where we are with the project and as a team.”* Large companies typically need more structure to make the

communication and coordination work well throughout the organization, and this need was visible in the previous game project of the company, too; it was bigger and more complex than any other project the company had done before, and it was missing many of the structures projects this big usually have. These factors made the development process challenging, as stated in the Towards Better Production -survey coordinated in the end of 2020 and in the post-mortems of the teams and disciplines organized in summer 2021.

Challenges in the previous project were brought up in many of the interviews, and the interviewees said that they hope the organization will find improvements and solutions to those by implementing more defined structures and Agile work practices. The matters mentioned include keeping the quality of the product high throughout development, increasing predictability and clarity of the development process, having clear processes for giving and receiving feedback, streamlining communication and information-sharing, making task prioritization and tracking better, and making roles and responsibilities clearer. These actions would make the development process more manageable and decrease the need for reactive work and working overtime when approaching deadlines. In short, it appears that the mission the company wants to achieve by implementing Agile is to find a sustainable way of developing videogames. *"We should find a sustainable way of working, sustainable work culture, and I believe that will be the best way in the end. Becoming organized and disciplined, that is the big thing"*, stated one of the interviewees.

There are also some further reasons why the company has decided to implement Agile practices: in multiple interviews it was stated that Agile and the principles related to it are well in line with the company's vision, missions, and values. It was said, for example, that the forementioned mission to find a sustainable way of working is tightly aligned with Agile. Even though the interviewee did not go into details, this appears to be true: the eighth Agile Principle states that "Agile processes promote sustainable development. The sponsors, developers, and users should be able to maintain a constant pace indefinitely" (Beck et al. 2001). Another mission for the organization that was mentioned multiple times in the interviews is to be a "learning organization", which means valuing learning and continuous improvement and seeing those as an integral part of the work. The Agile ways of working, such as iterative development and reflecting on how to become more effective, are seen as tools to enable this. In addition, the interviewees stated

that the case company has always valued things like nimbly responding to changing requirements and having iterative development processes, which are an important part of Agile, too. It was also noted that giving autonomy for the developers and trusting them as the fifth Agile principle suggests (“Build projects around motivated individuals. Give them the environment and support they need, and trust them to get the job done”, (Beck et al. 2001)) can be a great way to increase the motivation of the creative people who work in the game development.

Yet another reason for the company to implement Agile is that they have already used Scrum methods in some smaller projects years back with positive results. One of the interviewees phrased their experiences the following way: *“Those projects were done using Agile practices, and the satisfaction of the customer and the team was always high. The other projects used other methods and they went the way they went. Then we started to think that maybe we should unify these things throughout the company: it makes sense in this industry to make one small increment at a time and to choose the way that makes it easy to change and adapt in the future, which is the main idea of agility.”*

4.2 Practices used when implementing Agile in the case company

“Practices used when implementing Agile” is the next overarching category derived from the empirical analysis. These practices were discussed in all of the interviews, which further justifies the existence of this category. The practices and their categorization is visualized in Figure 12 and further described in the following pages.

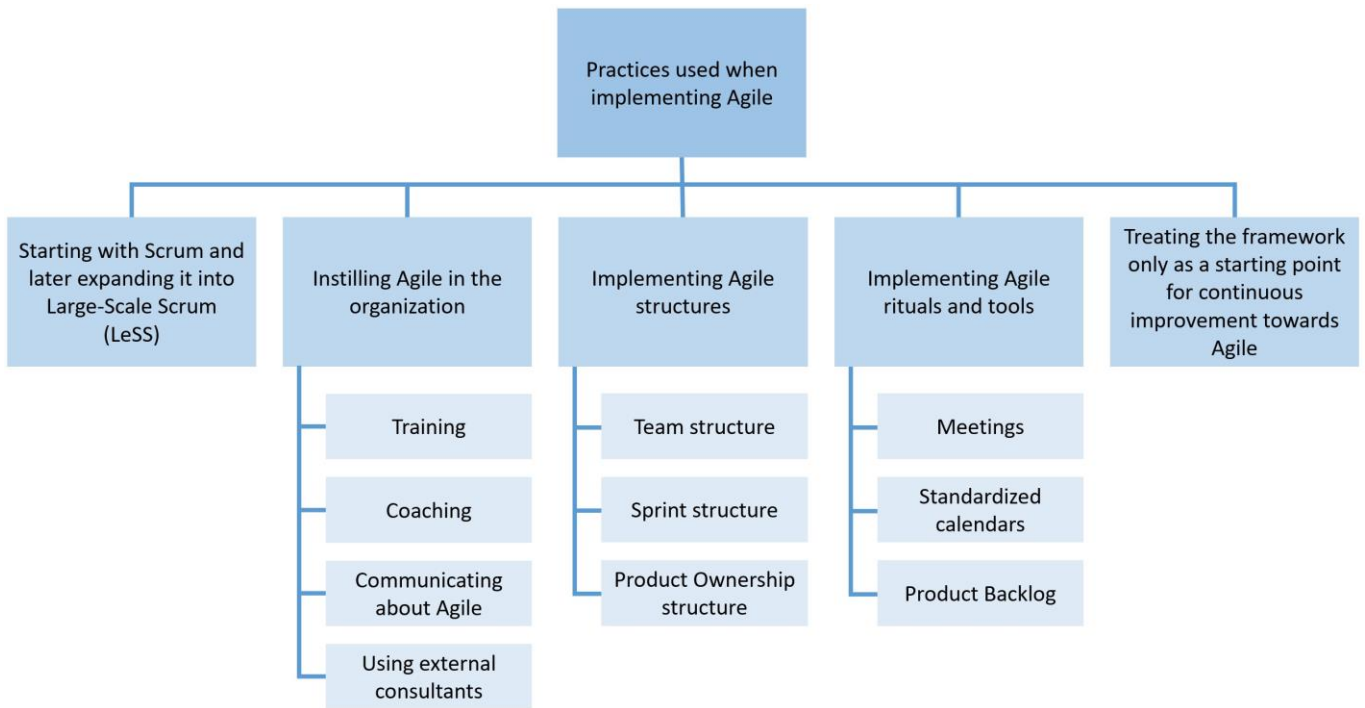


Figure 12: Practices used in the case company when implementing Agile

When studying the Agile implementation process in the case company, one significant thing to notice is that the company started by implementing Scrum and only later expanded it into Large-Scale Scrum (LeSS). Based on the interviews and the documentation collected, the reason for this was that in the beginning of the implementation process the key players in the organization were not yet aware of the LeSS-framework as that is not as widely known as for example Scrum-framework. However, when they started implementing Scrum and modifying it to make it work in a large organization the modifications happened to be well aligned with the LeSS-framework and the LeSS Huge -framework. For example, the idea of Proxy Vision Holders invented in the organization is fundamentally the same as the idea of Area Product Owners looking after their own Requirement Areas in LeSS Huge. When the key players eventually learnt about the LeSS and LeSS Huge - frameworks, they made a decision to implement those as those were initially created for big projects and corporations and could offer more specified solutions for the organization. In addition, at that time the teams were finishing one project and moving on to another, which made it easier to change and modify the ways of working yet again.

The Agile implementation practices used in the case organization can be divided into three groups: instilling Agile in the organization, implementing Agile structures, and implementing Agile rituals and tools. The first group, instilling Agile, consists of actions

like trainings and coaching, communicating about Agile, and working with external consultants. These actions have been particularly important in the beginning of the implementation when there were many uncertainties and people did not yet really know the frameworks and what are the goals of implementing them, but they also have had value in the later phases. In the case company, there have been three big training sessions connected to Agile frameworks organized with external partners, two of them aimed for the whole organization and one for people in lead, director, and producer -positions. In addition, internal training and coaching has been provided and organized when necessary. It has usually focused on more defined and practical things, such as facilitating certain meetings and using the task tracking -tools. Communicating about Agile implementation and its next steps has been done mainly in the weekly company-wide meetings and through producers who have transmitted information about it to their teams. During the LeSS-implementation, a few external consultants have also been there to support the implementation process as needed. In the interviews, most of these actions were described to be rather good and effective, but the trainings divided opinions; it appears that everybody recognizes the value of them, but some say that it could be more efficient to focus on giving more practical advice in the trainings and to offer them primarily for the people who will manage the new system. As one interviewee said: *“It’s not enough for people if they only hear the theory and are then told to put it into practice. The theory should be told first, then show how it can be executed in practice and then rehearse it in practice together with people and help them to get used to those routines.”*

Implementing the needed structures has also been a big part of Agile implementation practices done in the case company. The structures implemented are the Team structure, Sprint structure and Product Ownership structure. In the Scrum and LeSS frameworks, the goal is to have self-managing and cross-functional feature teams that have the skills needed to complete their tasks (Vodde et al. 2014; Schwaber and Sutherland 2020), and the company has tried to find ways to implement this as well as possible; there are now some cross-disciplinary teams, such as the level creation team and the user interface (UI) team, that are able to take certain tasks and carry them out from start to finish mostly by themselves. However, there are also some teams, such as the visual effects (VFX) team, where virtually all team members work in the same discipline and focus on developing that discipline and its processes. The reason behind this is the state of the current project; in the pre-production phase the focus is on making plans and developing processes for the project, and in some cases it is better to do that inside the disciplines. The case

company plans to increase the number of cross-disciplinary teams in the organization in the future, latest once the production phase starts. The communication between the teams in the organization happens in several ways: First of all, there are several LeSS-meetings, such as Sprint Planning and Sprint Review, where either representatives of each team or all of the team members plan and communicate together. Then there are Scrum of Scrums -meetings organized every day where a representative from each team shares with others the things that are useful for them to know, such as updates and changes, concerns, and collaboration needs. Furthermore, the teams and team members are encouraged to have self-organized communication with each other whenever needed.

The Sprint Structure is another structure implemented in the organization. When using Scrum, Sprints in the organization were one-week long to make sure the rituals were repeated often and became a routine. Now that the organization is using LeSS they last for two weeks. This two-week cycle seems to be suitable for planning, executing, and iterating work efficiently – if the Sprints were longer, planning for them would be harder and more work might be wasted, and if they were shorter, staying focused on bigger tasks would be harder and proportionally more time would be spent in meetings.

The Product Ownership structure has been the third structure for the organization to implement. It appears to be the most challenging for them, as it requires the biggest changes in the organization, and it has already gone through various iterations. When Scrum was implemented, Product Ownership was organized in such a way that there was a Project Leadership Team that performed the Product Owner -function, and they were supported by Proxy Vision Holders who channelled the vision of the Project Leadership Team to the development teams and also represented those teams in Product Backlog Refinement -meetings. This approach was not entirely successful, as described in this quote from one of the interviews: *“The teams just worked by themselves and so the project leadership team -- has been basically a reviewing body that has tried to keep an eye on the overall quality.”* Because of the challenges, the Product Ownership -structure was redesigned once the organization shifted to LeSS-framework. Now there is a small Product Owner Group that consists of four people who represent different interests and areas of game development, such as the creative direction, technical objectives, and the business goals. It has been defined that if there are disagreements inside the group, one of the members, the Executive Producer, has the final word to say, which fundamentally makes this person the Product Owner. Product Owner Group manages the Product

Backlog and is supported by Extended Product Owner Group that consists of the Directors of the company.

Implementing the rituals and tools has been yet another big part of Agile implementation practices applied in the case company. The rituals and tools refer to the meetings defined in the Scrum and LeSS-frameworks, to standardised calendars where these meetings have been scheduled for each team, and to the Product Backlog. When implementing the meetings, the results have not been immediately successful, especially with meetings that have a high number of participants who need to communicate and coordinate with each other. However, even those have become better after some time, when improvements based on the earlier experiences have been made and when the participants and organizers have become more familiar with the meeting goals and structures. The timings for all the Scrum and LeSS meetings are defined in the standardised calendars that have been created to uniform the schedules of the teams and to increase the predictability of the work. The Product Backlog, an ordered list of all the tasks that need to be done to create and improve the product (Schwaber and Sutherland 2020), has also been created to guide the development. Like defining the Product Ownership -structures, creating a Product Backlog has also been a challenging task, because the product developed in the case company is massive and its features complex. One of the interviewees described that in the near future the goal is to *“have a Product Backlog that is visualised, that is coherent, that has a standardisation of the size of the items inside it, and that teams are used to the idea of refining those items together.”*

When describing the practices used during the Agile implementation, many interviewees emphasized that the Scrum and LeSS frameworks and the practices defined in those are only the starting point in the process of becoming truly Agile and they should be improved whenever possible based on the needs of the organization and the teams. One of the interviewees described the situation this way: *“The roles and rituals, they are not the thing here. They are just a starting point for the people to learn better things”*, and another said quite the same thing but also added that patience is needed, as the best results are not achieved immediately: *“Here in the beginning it will take time to understand what this change means to us and how we will adopt these things. The end result will look like us, but it will take time to get there.”* In addition, it was also mentioned several times that the frameworks should only be enablers for the actual work, for the game development that happens on people’s minds and desktops. One of the interviewees verbalized it like this:

“Agile and LeSS and all the other buzzwords, their purpose is just to serve the product we are creating and especially the people creating it. For that reason, we need to be open to modifications”, while other one added: *“I think that with any production process, the ultimate goal is for it to become invisible.”* It appears that these views of the frameworks being only a starting point and an enabler for the actual work are widely shared in the organization.

4.3 Benefits gained through implementing Agile in the case company

“Benefits gained through implementing Agile” is another overarching category derived from the empirical analysis. The topic was widely discussed in the interviews, even though the focus was mostly on the Scrum implementation, as the LeSS implementation is still widely in progress and its results cannot yet be genuinely assessed. The benefits and their categorization are visualized in Figure 13 and described in more detail in the following pages.

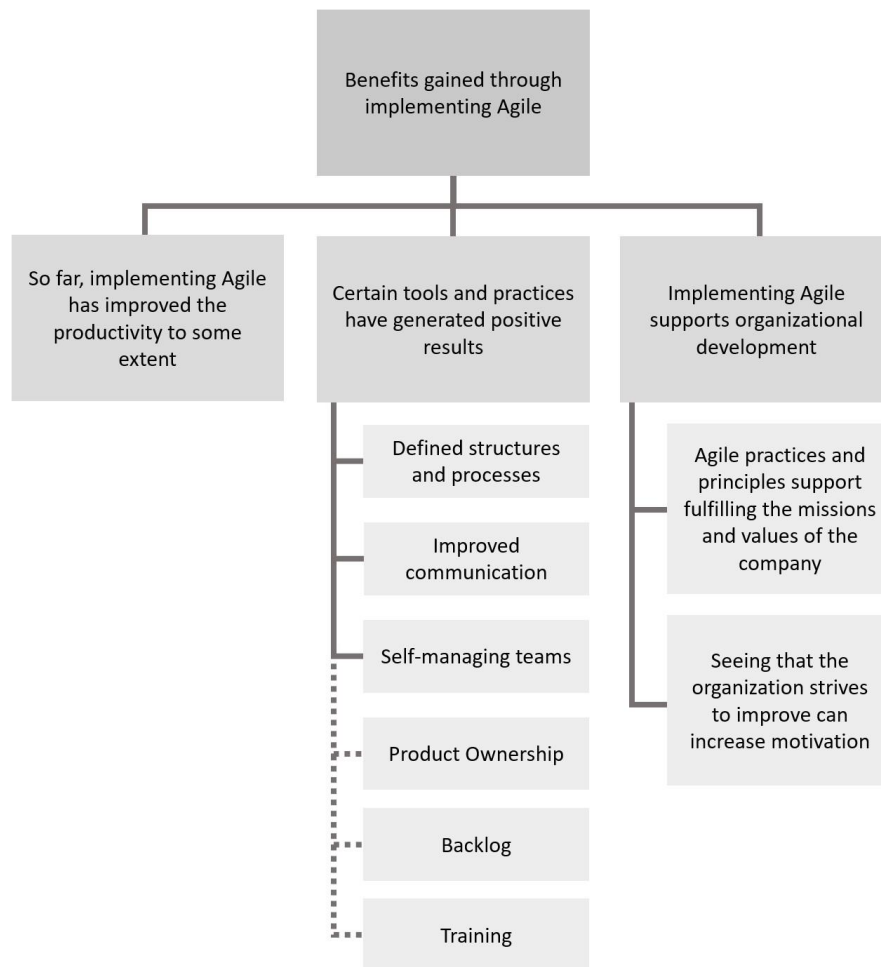


Figure 13: Benefits the case company has gained so far when implementing Agile. The dashed line indicates that there is significant inconsistency in the views towards the practice and some view it as beneficial while others currently find it challenging

The benefits the case company has gained so far during their Agile implementation process can be divided into three groups; the implementation has already improved the productivity of the teams to some extent, certain tools and practices have generated positive results, and implementing Agile seems to support organizational development in some areas.

In the interviews, it appears that the development teams have gained some good results such as increased focus and productivity with Agile and that they have been quite happy with most of the practices the frameworks have brought to them. The reason behind these early gains seems to be the teams' success in the Scrum implementation – many of the interviewees said that they were positively surprised with the speed and success of the development teams when they incorporated the Scrum practices into their workflows. As one person commented: *"I've been so impressed with the way that the teams have just*

adapted to Agile. I think that any issues with Agile adoption in the organization, if we look back and do a post-mortem, they've not come from teams not being able to work in that way." Some point out that the situation where each team has somebody, typically a producer, acting as a Scrum Master is one big reason for the success of the implementation. Scrum Masters have undoubtedly had a crucial role in the implementation as they have been the people guiding the teams through the process, communicating the needs of the teams to other teams and the organization, and facilitating most of the meetings. Based on these comments, it appears that they have performed well.

Another thing mentioned in the interviews was that various concepts, practices and tools introduced in the Agile frameworks have worked well in the case company. Improved communication through successful meetings was pointed out the most often, and there were very positive comments like: *"Sprint Reviews are really good. Regular Retrospectives are even better. And Dailies are the best of the best"* and: *"The usefulness of the Dailies always surprises me, even though I like them. In the previous project our team didn't have those, and now that they're back I just wonder how we were able to manage without them."* The benefits the improved communication has brought to the teams were also mentioned in several interviews. First of all, good communication helps staying up-to-date with what happens and knowing about tasks and needs well in advance, which in turn helps with scheduling and preparing for tasks. In addition, communicating about challenges more openly makes it possible to look for answers together with other teams and individuals and thus find better solutions. It appears that during the Scrum implementation the communication inside and between teams remarkably improved, and people feel that as a result there were less gaps in the communication and information-sharing. It is also mentioned that in the beginning of the implementation the meetings were not that good, but with patience and constant improvement they became better: *"The meetings have been modified, for example to make them better for the artists and to make sure they don't take that much time. I feel that we had just reached a point where no-one really thought a lot about the meetings, they were just done and things went on smoothly and no-one really complained about them."* However, it seems that when the meetings defined in the LeSS-framework were introduced, the happiness with them decreased. It will probably take time and effort to make the new types of meetings work well once again.

Another concept introduced in Agile frameworks that has worked well in the case company are the self-managing teams. Even though the concept is not yet perfected, the teams have been able to manage and control their work more than before and many have been happy with the results. One interviewee describes it like this: *“For example, we have our own internal review in our team. There we have tangible improvement suggestions and we are not married to the idea that we just wait for someone from outside to tell us what to do next,”* and another one explains that when the teams have been allowed to decide their own ways of working, things have gone smoothly: *“I feel that there have been less clashes with the vision-team and what they should focus on. Now we have had clear guidelines that they should focus on the product and what they want to be done, and then the teams decide how to do that. With these guidelines being clear, things have been rather smooth.”* The teams seem to be happy with the way Agile can increase their autonomy and self-confident that they can plan their work and achieve the goals set. Even though it was not directly said, it seems like their views and beliefs are aligned with the fifth and eleventh Agile principles, which are “Build projects around motivated individuals. Give them the environment and support they need and trust them to get the job done” and “The best architectures, requirements, and designs emerge from self-organizing teams” (Beck et al. 2001).

Having more defined structures and processes in the organization is also mentioned as a benefit. These structures have made working and communicating easier, and having a more structured calendar, for example, was said to have made it easier to schedule meetings so that there are no overlaps. The fact that the organization is growing appears to be the biggest reason for valuing the increased structures - one interviewee describes the situation this way: *“Even though we have been kind of proud of having a low-hierarchy organization, it is clear that when so many people join us we need more structure. It does not mean becoming too hierarchical or rigid but bringing in the structure and the safety that make it easier for us to do our daily work.”* When describing the benefits Agile implementation has brought to the case company, some interviewees also mention Product Ownership, Product Backlog and trainings. The relationships inside the Product Owner Group appear to be good which makes their work effective, Product Backlog makes it easier to plan and assign the work, and trainings help everyone to understand Agile better and to share a common language. However, there have also been several challenges linked to these things, which are further described in section 4.4.

Yet another benefit Agile has brought to the case company is that it appears to support the organizational development in a broader sense, too. Agile values and principles seem to be well aligned with the company's values, vision, and missions. This alignment can help the company to accomplish their goals and missions, such as find sustainable ways to develop videogames. The alignment also means that the people in the organization who had already found good, effective ways of working should still be able to work that way in the future. As one interviewee explains it: *"Ultimately, we are simply introducing a small bit of process to the way that the organization wants to work. If we do this properly, the people who are used to working in a very free manner will still feel very free because that's the purpose of Agile. It is to be able to iterate, to inspect and adapt and to make changes based on what you see and then just to quickly go and do something again and then quickly go and try something else. That's exactly the way this organization is working, but all we need to do is introduce a tiny bit of process."* In addition, one interviewee describes that seeing that the organization is striving to become a better workplace has increased their motivation: *"For me, the most remarkable thing is probably that the work morale has gotten higher as I've seen how the company tries to do major changes to improve the well-being at work."* Improved work morale and motivation are beneficial things for the organization, as those can for example improve the wellbeing of people and have a positive impact on their effectivity. They may also make the Agile implementation process itself easier.

As said before, the final benefits of the Agile implementation cannot yet be genuinely assessed because the implementation process is still ongoing. However, the aforementioned initial benefits help to predict the final results.

4.4 Challenges faced when implementing Agile in the case company

"Challenges faced when implementing Agile" is yet another overarching category that was obtained from the empirical analysis. It was also one of the most discussed topics in the interviews, which is reflected in the length of this section. Since the Agile implementation is still ongoing, many of the challenges remain unsolved or solving them is in progress, and it is also highly possible that some challenges have not even emerged

yet. Current challenges and their categorization is visualized in Figure 14 and described in more detail in the following paragraphs.

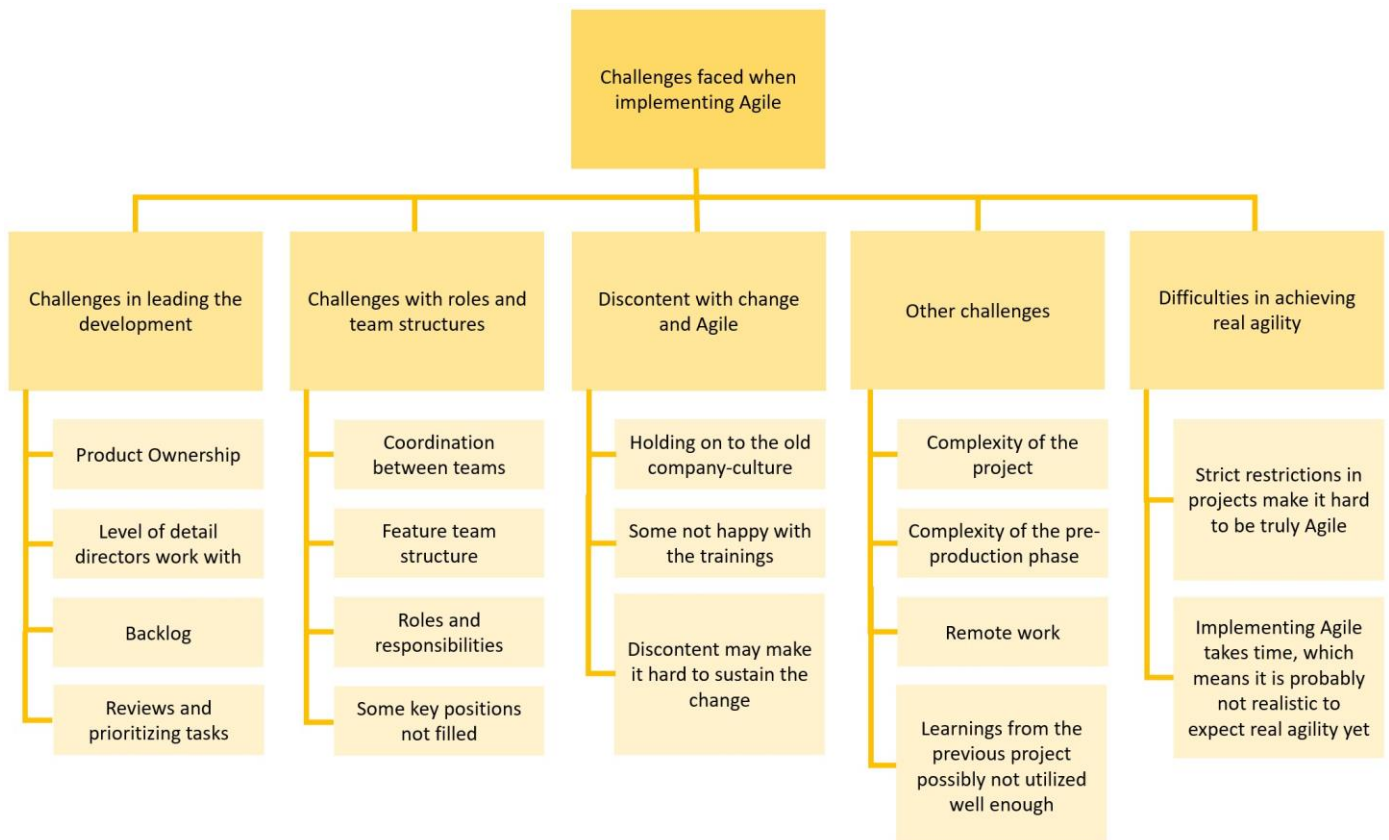


Figure 14: Challenges the case company has faced so far when implementing Agile

Based on the interviews, the case company has faced many kinds of challenges when implementing Agile. These challenges can be divided to several groups; there have been challenges in leading the game development and with the roles and team structures, discontent with the change and Agile, difficulties in achieving real agility, and also some other challenges.

It appears that the most significant group of challenges in the case company has been implementing the tools and structures that are connected to planning and leading the development, such as Product Ownership structures, Product Backlog, and task reviewing and prioritizing systems. Defining the level of detail Product Owners and directors work with has also been hard. The challenges in implementing these structures, especially the Product Ownership, connect to the complexity of the projects – developing AAA-games means that the projects are very big and there are many factors to plan, design, and consider. One of the interviewees describes the responsibilities of the Product Owner the following way: *“The Product Owner is the holder of the product vision, but that's not necessarily the creative vision. That's the conglomeration of the business requirements*

and the business vision, the company's strategic vision; where it wants to be, what kind of games it wants to make, what it wants its reputation to be like, and then the game's creative vision as well. It's the Product Owners' job to channel all those and to use his or her experience in the field to lead the decision-making around what we need to do in terms of the product itself, and basically what features are we going to do and in what order are we going to do them." With a massive project, this is not an easy task, and it can feel impossible to find one person who can do all this. For this reason, the case company first implemented the Product Leadership Team when implementing Scrum, and later modified this structure into Product Owner Group supporting the Product Owner, which has apparently worked rather well so far.

One challenge that is heavily connected to Product Ownership is defining the level of detail Product Owners and other Directors should work with. At first glance, the solution seems rather simple: based on the interviews, many see that the Product Owners and other Directors should provide the high-level goals and then let the development teams break those down into tasks and define how they want to implement them. However, it seems that this is not specific enough, and people see different kinds of risks in it. One interviewee, for example, sees that the guidance can still be too detailed: *"When it comes to the Product Owner -function, my concern is that how do we keep the level of guidance high enough and give enough freedom for the teams. There needs to be enough direction, but at the same time there should be no micro-management"*, while another feels that with too high-level guidance there is a risk of misinterpretations: *"The Product Owners are expected to only give high-level goals, and then the teams break those down into actionable tasks. And I think that can be risky, that is where a lot of misinterpretations can happen and maybe the priorities of things just are not that clear."* It is also mentioned that because teams and individuals have varying needs and preferences, it is challenging to define standardized ways for guiding them: *"I feel that it is incredibly hard to find that balance where you are giving enough direction but not too much. I do not know if there are any real kind of global solutions for that, I think it does come down to how each team and each person tends to work."* In addition, it is mentioned that the Product Owners and other Directors have a lot of expertise and experience and having them to focus only on high-level issues may mean that valuable experience is lost and the teams are left alone to solve challenges that could have been easily solved with the help of a director, for example. Focusing on high-level issues may also lead to a situation where certain details that are important for the big picture get dismissed. However, there are also concerns that

expanding the Product Owners' and directors' focus from high-level issues to lower-level things, too, may cause significant bottlenecks for the development: *"In this organization, the people working on Director-level are used to working with very practical, hands-on things. However, that does not work in a big project if the decision-making of the key people becomes a bottleneck. – On the Director-level, the strategic and operative work is the most important thing, as those thoughts, decisions, and concepts generate the work for everyone else in the project."* All these divergent opinions indicate that finding the working solutions for this challenge may be hard.

Creating and managing the Product Backlog appears to be yet another challenging factor related to planning and leading the development in Agile way. During the Scrum implementation, due to the nature of the project, there was no Product Backlog created together and managed by Product Owner or Product Leadership Team. Instead, the teams managed their own parts of the Product Backlog quite independently. Now that the new project has started and LeSS implementation is ongoing, the case company tries to implement a system with a single Product Backlog managed and prioritized by the Product Owner Group on a high-level. From that Backlog, the teams choose items for their Sprints in the Sprint Planning -meetings together with the Product Owner Group and other teams, and then the items are refined as needed.

The challenging factor with the Product Backlog is that it is a completely new concept for the organization – such a thing has not been used there before, at least not in a project this big and complex. The organization needs to find ways to make it work for them, and that process is still on-going. For this reason, not many concrete examples of the challenges with the Product Backlog were mentioned in the interviews; some just mentioned problems with the software that is used for managing the Product Backlog. However, some challenges in prioritizing and reviewing tasks were mentioned. During the Scrum implementation, there were some bottlenecks in the review processes which slowed things down, some feedback was not reacted to in a timely manner, and sometimes the feedback given contradicted with previous guidelines or feedback. These things need a lot of improvement, and as one interviewee says: *"I think that's the state that we're in, and the state that we have carried with us as a legacy. But we have made changes and we will continue to make more, and now we have another opportunity for us to be very clear about the transition to the new way of working."* The LeSS-framework and rituals may provide some help with these matters in the future, but that has not realized yet.

Implementing the roles and team structures has been another great challenge for the case company during the Agile implementation. One of the problems in this area is that creating cross-functional feature teams is hard when developing big games with complex features, as finishing one feature often requires numerous different experts to work on it. If the task is to create a new type of enemy for the game, for example, you need input from various designers, programmers, and 2D and 3D artists, as well as from UI designers, audio designers and VFX artists. At the same time, Scrum and LeSS-frameworks define that there should be less than ten people in a team, and it is likely that a team this small does not have all the expertise it needs to finish its tasks and features. So, every team needs to find a balance between having enough members to finish their tasks and being small enough to stay agile. In addition, it needs to be ensured that there are enough key people for every team, for example producers taking the role of Scrum Masters. Here is an excerpt of one interviewee describing the situation in their team:

“There were maybe 14 people in our team at maximum, which is quite a lot more than the nine people suggested in Agile. Maybe it could have been split into two teams, but then I wonder if there would have been enough producers for that... In the end, everything worked, but that was still one thing that should have been thought about more.”

As mentioned before, the teams, most often their representatives, communicate with each other several ways: in LeSS-meetings like in Sprint Planning, Product Backlog Refinement, Sprint Review, and Overall Retrospective, in daily Scrum of Scrums - meetings, and through all the formal and informal, self-organized methods they see necessary. But even though there are numerous ways to communicate, the communication and coordination between the teams has still been challenging on several occasions and dependencies and risks have not always been caught early enough. One major reason for this is that it takes time and effort to learn how to facilitate meetings attended by large numbers of people in a way that is efficient and useful for everyone. Especially some LeSS-meetings, like Product Backlog Refinement and Sprint Planning, are considered challenging, as those are attended by tens of people and the goal there is to plan and coordinate complex issues together. Furthermore, some people feel that the number and length of the meetings is currently too much and there is not enough time left to focus on the actual game development. Not everyone shares this view, though; many think that the meetings are the tools for improving communication and coordination, which in turn helps to create focus, improve efficiency, and decrease the amount of work wasted. It is

also noted that the meetings should become shorter and more efficient once they are learnt and developed, which will improve the situation.

In addition, some of the organization's challenges linked to coordination and communication also connect to the fact that there are key positions in the organization that are not filled. The company has been familiar with this challenge for a long time, as one interviewee describes: *"That is one of the big challenges we have, that with resource we've always been very strained. Even with our previous project we were looking for a lot of key positions to be filled and we couldn't do that, so we just had to make do with the resources we had. I think that is going to be one challenge ahead as well."* These key positions not being filled means that there are gaps in the organization, and these gaps make it harder to successfully implement the new practices and processes: *"We have, what, a two-digit number of people we want to hire for the next year. And after we hire those people that's when we will have the complete team to adopt all these practices. Until then there are going to be some gaps that we are going to be struggling to fill, and that will create even more need for exceptions to the rules."* This is also connected to the unclearness in roles and responsibilities in the organization. The fact that some of the roles and responsibilities have not been clearly defined has led to situations where people have too many things they need to focus on and also to situations where no-one is responsible of a certain task or feature and it is not properly taken care of nor finished in time. However, it appears that there is already work ongoing in the organization to find consensus and make these clearer in the near future.

The third group of challenges the case company has faced is connected to the discontent with change and Agile. Change can feel difficult and the results uncertain, which can negatively affect the motivation to engage with it and strengthen the desire to hold on to the old structures and processes. When asked what they think are the biggest challenges in implementing Agile in the case company, one interviewee answered this way: *"Getting everyone on the same page, I think. There's a lot of people, there's always some slight misinterpretations or slightly different expectations on how things should work. We have a very solidified culture around here, this kind of "garage band mentality", and now we need to shift to be like a well-defined orchestra. And I think that is a really big change in thinking for many."* Changing a solidified culture built over years is hard and takes a lot of time. In the case organization, it also appears that the change is often considered negative in situations where people who are used to working in a very free manner are

asked to follow more defined processes in their work, especially when the processes do not yet work appropriately. Implementing and learning the new rituals has also felt hard and frustrating at times, as described here: *"Another thing that I find challenging, which cannot really be avoided, is that in the beginning when practicing the new rituals, it feels uncomfortable for many. And the meetings take a lot of time and they do not feel useful. But the experience has proved us that when we just do these things and actively improve them, it all starts to become better."* Several other interviewees agree with this; in the interviews, it was mentioned many times that the situation is believed to get better once the new routines are learnt and improved and people become accustomed with them. One interviewee put it this way: *"I believe that when people get used to these routines and see their benefits, there will be more positive than negative experiences."*

There has also been discontent with some specific practices used during the Agile implementation, mostly with the trainings. One of the interviewees sees that they are useful for some, but others may experience them negatively: *"There are many kinds of trainings and other things, but I think there are always some problems with those. For many, they can be very boring and the main point may be missed because there are lots of things covered that are not relevant for the person working in the organization and focusing on hands-on tasks. For sure they are much more important for the producers and others like that, as they are the ones organizing the meetings and the systems."* It was suggested that the trainings could more often be targeted for the people working in key roles or otherwise particularly interested in the topic. This way, there would be less negative emotions connected to the trainings, those attending them could get the most out of the experience, and they could later propagate their learnings throughout the organization.

As described, discontent with change, Agile, and the practices used during the change process is a major challenge for the organization. However, the fact that discontent can also make it hard to sustain the change makes this an even more significant challenge for them. Sustaining change is already a difficult task which requires constant reassurance, but it becomes even harder if the processes or practices are not working as supposed or if there is discontent towards them. As one interviewee says: *"It is very easy to abandon the process when people get the slightest hint that it may not be working out."* However, the positive side to this is that many acknowledge that implementing and sustaining the change is a slow and challenging process and embrace the results already achieved: *"I*

think that the culture change is underway but well, the thing about it is that it's constant, right? It's constant and continuous striving for improvement - we can never say that it's all done and everyone's on board. There will always be a need to change it, but I think that we can definitely see that in the last six, seven months teams have embraced a new way of working." When those leading change have realistic expectations for the timeframe and for the results, it is easier to see the setbacks and other challenges as a part of the process and not become paralyzed by them.

In addition to the challenges already described, there are also some other difficulties the company has encountered during the implementation process. The complexity of their projects is one - there is an enormous number of things to plan, execute and consider in big game development projects. This complexity is also the reason behind many of the other challenges the case company has faced; it would be much easier to create and manage the Product Backlog, for example, if the product was less complex, and the Product Ownership and Feature Teams would also be easier to organize and structure. To make the situation even more challenging, the first phases of game development, meaning the concept and pre-production -phases, appear to increase the complexity even further as the plans and goals are only created during these phases and the creativity and iterativeness should not be restricted in any way. One of the interviewees describes pre-production phase this way: *"When you are in pre-production for a creative product you do want to have that exploration and to allow teams to just prototype things that are not going to work out, and then you go back and you know, just allow that creative iterative culture to prevail. I think any kind of production model that is very strict can interfere with that"*, while another comments that it is difficult to implement a new framework during these phases because the need for adjustments is so big: *"The LeSS-framework in its original form will work better when the project has progressed more and we are in production. – In our context, the challenge is that we are implementing a framework that does not directly work with the phase we're in, and we need to kind of break it right away."* Some interviewees also mention that the need to work remotely, caused by Covid-19 in 2020-2022, has also made it harder to implement Agile practices. As one interviewee comments: *"It's difficult to have that connection with people if you're not in the same space with them."* However, in most of the interviews this topic did not arise, which may suggest that many are accustomed to the situation and did not see it as an additional challenge anymore.

One more challenge that was mentioned is that the learnings from the previous project may not have been utilized as well as they could. As one interviewee said: *“It feels like from our previous project there was already quite a lot of knowledge accumulated, whether it was from our mistakes or our successes. And I feel like we have not really been utilizing that knowledge.”* There appears to be a feeling that a holistic, inclusive approach was missing when the previous project was finished and the change was planned: *“The change has been lacking a lot of context sometimes, and I think that does go back to not having all of the key people in these discussions, or not having a clear retro, not having a clear postmortem for things that went right and wrong.”* It is mentioned that gathering the information related to the previous successes, challenges, and learnings into a centralized location known by everyone and inviting every lead, producer, director, and manager to process the lessons together could have been a good approach – if the lessons are only processed separately, the takeaways between different people can vary, and that can lead to misunderstandings or solutions that do not really work: *“If we have different interpretations about what our problems are, then it feels almost random what kind of solutions we are going to be looking for.”* However, it was also mentioned that as the implementation process is still ongoing, it is not yet possible to give definitive feedback on the topic.

Some interviewees also say that there have been difficulties in achieving real agility and that so far it has actually not been accomplished. The main reason for this appears to be the strict, fixed scope of the project that was done during the Scrum-implementation, as there was not much room for agility there: *“It is misleading to say that in that project we were somehow Agile. I don’t think it’s true because there were big restrictions for us with time and money and resources, meaning employees. – In that situation, we couldn’t even have a lot of iterative action inside the team, and we had no time to focus on how to make Agile development work for us.”* Another reason for not becoming truly Agile yet is perhaps that the implementation is still in progress: several interviewees mentioned that it takes time to learn and improve the new ways of working and to internalize the new mental models. Based on this, it seems that it is not realistic to expect the teams to be truly Agile yet – there is still a lot to do before that can be achieved.

4.5 Human skills in implementing Agile in the case company

“Human skills in implementing Agile” is the final overarching category of this analysis. It was derived from the empirical analysis, and it was widely discussed in the interviews. The relationship of human skills and Agile implementation in the case organization is visualized in Figure 15 and further described in the following paragraphs.

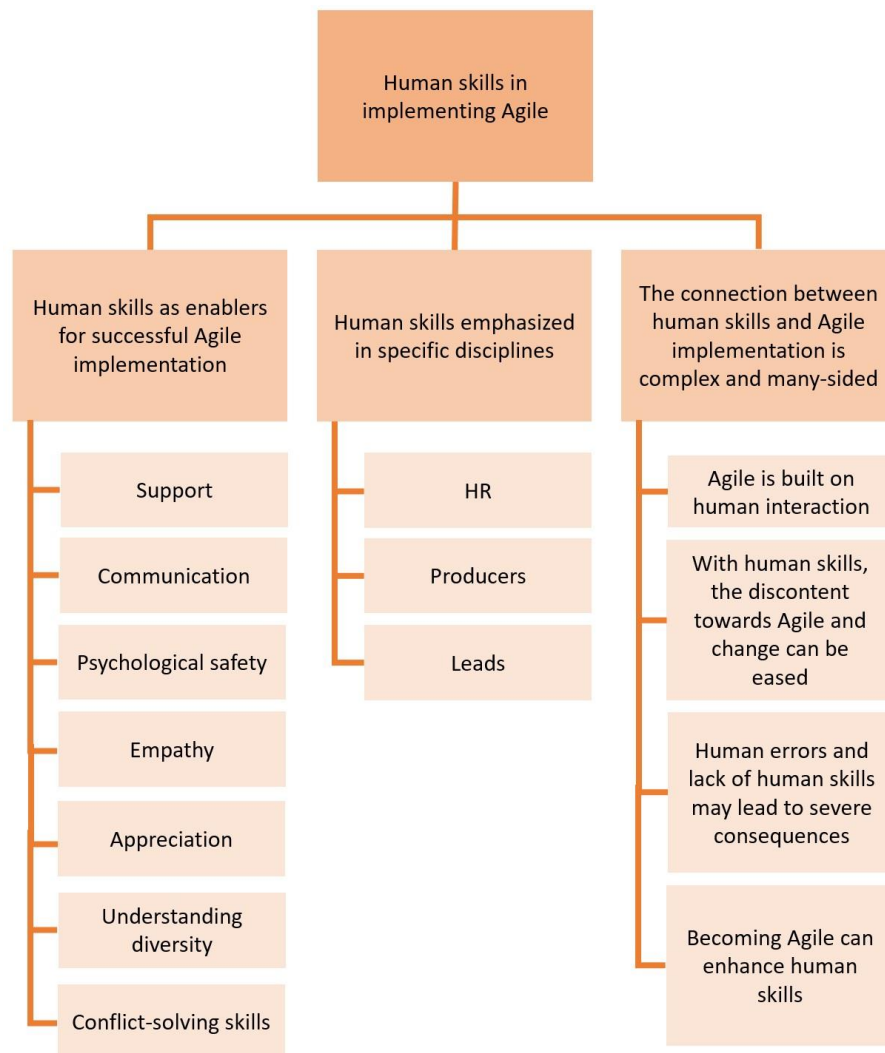


Figure 15: The role of human skills in implementing Agile in the case company

Based on the interviews, it appears that several different human skills have had a role in the Agile implementation in the case company, including psychological safety, support, communication, empathy, appreciation, understanding diversity, and handling conflicts. Out of those, psychological safety seems to be one of the most commonly mentioned – not directly, but by describing the way the organization has allowed and accepted feedback about the Agile implementation. In the interviews, most of the people say that they have felt it safe and accepted to tell their honest opinions about Agile and the

implementation process. Retrospectives are mentioned often, and they seem to be the most suitable situations to share opinions and to offer feedback: *"In every team retrospective, we go through the positive and negative things and improvement ideas. I hope and feel that people have had the courage to say if there has been something, or if there have been some questions, or if they don't like the number of retros. I have listened to those things, and we have tried to solve them. And if they couldn't be solved or changed, we tried to discuss the reasoning behind them."* However, there are also people who have had less positive experiences with psychological safety. One person says that the way their feedback has been received has negatively affected their willingness to join the discussion: *"In the beginning of the implementation I was more eager to give feedback and to take part in the discussion. But after noticing what were the reactions for the critical feedback, I have not wanted to get into that anymore."* Based on the interviews, these things do not happen very often, but that does not mean they can be overlooked. Cases like these can have a damaging effect on trust and psychological safety, which in turn can have severe consequences on the way people in the organization work and interact with each other.

Support is another human skill that has played an important role in the Agile implementation in the case company. To make the implementation successful, the people leading the process need to support and help others whenever needed. As one interviewee said: *"You need people who make these sorts of changes and these sorts of processes appear human. You need to put the face to the role, and you need to have kind and empathetic producers and Scrum Masters who are there to facilitate and to help teams grow and to move to the background when they're not needed, but to be ready to come to the front if the team needs anything."* Several interviewees feel that they have received enough support and help, but there are also some who comment that there has not been enough of those. It is said, for example, that with more complex problems with Agile it is usually rather easy to get sparring, but there are not many experienced people who can offer help and practical advice in these situations. Sudden changes and rapid progression also increase the need for support, and this has not always been recognized, as described by one interviewee: *"At times, the progress has been too streamlined and the need for support has not been realized. Also, people need warnings in advance so that they can know what is going to happen in the long term and when. Some things have happened quite suddenly and that has distressed people, even if there has been no other reasons to be stressed. And that is understandable as the schedules have been like this."* As a result,

it appears that: *“Even more time has to be spent on implementing the change and supporting and helping people.”*

Another human skill, communication, was also widely discussed in the interviews. It appears that in the organization there is a lot of effort invested in making the communication good and providing ways to communicate and give feedback. The meetings defined in the Scrum- and LeSS-frameworks have been implemented in order to improve the communication inside and between teams. In addition to these, there are also other communication methods established in the organization. Some teams and disciplines, for example, organize their own regular calls where the participants can share information and learnings and review each other's work. Then there are also regular 1-on-1 meetings organized throughout the company to make sure everyone has a place where to talk with their lead or other equivalent person about anything that feels important. It is also recognized that people have different preferences and needs for communication, so there has to be different ways to do that: *“Openness and communication, they are very, very important. They're something we have tried to push in the organization for years, to make sure people tell if there are some problems and give feedback, as that is the only way to get things fixed. It is understandable that not everyone feels as content to say these things, especially in front of other people. That's why we have organized employee polls, for example, and ask for written feedback, to provide as many channels for giving feedback as possible.”* However, it appears that there is still room to improve, as in one interview it was commented that there are not enough suitable forums yet to give feedback about the Agile implementation and to discuss the views and opinions related to it: *“I think we do not have all of the right forums for these discussions yet. However, it is not a criticism for the company, because it is not intentional. We just have had so many challenges with shaping the project, people have been unavailable, and we are still starting this thing off, so I am hoping that in the upcoming weeks that will also become a lot smoother.”*

When it comes to communicating about Agile implementation across the company, it appears that it has been done primarily in the Friday Updates organized every week and participated by everyone in the organization. There, the strategy and reasons behind the implementation have been described, as well as the ways the implementation is done. Producers have also communicated about the practices and reasons with their own teams as necessary, and the key people have tried to stay available to discuss the implementation

and Agile with anyone who has questions or concerns related to them. Based on the interviews, people have been rather happy with the level of communication: *“There has been quite a lot of communication about the implementation, which is clearly a positive thing. It tells that the company is committed to Agile and finding the solutions and keys through it. – I also feel that implementing Agile has been justified quite visibly.”*

The importance of empathy and understanding diversity is also mentioned often in the interviews. These skills are linked, as empathy is tightly connected to understanding that people have different, diverse preferences and needs and they are in different kinds of situations. However, empathy is also seen as a wider concept - one interviewee describes its role this way: *“I think leading people often comes down to having empathy and understanding that people are generally trying their best and nobody is intentionally trying to sabotage the project or has those kind of ill intentions. And I think that already sets a really comfortable space to connect with other people.”* Based on the interviews, it appears that when implementing new things, such as Agile, it is particularly important to be emphatic and remember the diversity. As one interviewee says: *“Every time new things are implemented, it needs to be understood that people take the change in different ways. That needs to be taken into account in communication and also in support. There are people who are excited about change, but for others it needs to be explained more profoundly why things are done and how they need to be done. That is not a good or a bad thing, it is just that people need different things. It is very important to acknowledge.”* By acknowledging the diverse needs and preferences of people, it is easier to help everyone stay content with the implementation, which can make it more successful.

Another human skill discussed in the interviews was conflict-handling. In one interview, it was said that conflict as a phenomenon should not be viewed as a bad thing, it is a way to improve and find the best creative and technical solutions through thoughtful discussion and well-founded arguments. But for that to work, there needs to be a culture where the conflict can be professionally handled. In a culture like this, human skills play a big role. One interviewee describes the need for conflict-solving skills this way: *“There have been some pain points, either related to work lost or to a lack of information, and in these cases, the human skills are the way to solve the conflicts. There will always be conflicts, and to solve them one needs to have the ability to understand the edges and to navigate the situation to a resolution that helps us get forward without any permanent schisms or resentment.”* Another interviewee states that for any team, it is important to

have people with good human skills there: *"It is always good to have at least one person with good soft skills in the team to be the connecting tissue and help assist the team to move forward when sometimes they can get a bit stuck. And not stuck due to technical knowledge, but just on the social level, on the soft skills."* In general, it appears that several people in the organization consider conflicts as a necessary and rather neutral part of the development process, and see that human skills are an important part of handling them. Still, as mentioned before, there have been cases where feedback has not been received appropriately or a conflict has not been handled as well as hoped, which implies that there is still room for improvement in this area, too.

Another thing related to human skills that rose up multiple times in the interviews is appreciation; people often stated how they appreciate each other's skills and efforts. These comments were typically related to how some people have done great job in implementing the change and supporting their teams during the implementation. The comments did not lead to longer discussions, but they were quite common, which indicates that there is a lot of goodwill and appreciation in the organization.

Based on the interviews, it is clear that human skills are even more crucial for some particular disciplines and roles in the organization. In Human Resources or HR, the aim is to help and support people, which means human skills are a necessary and integral part of the work in the discipline. For producers and leads, it is also important to have good human skills, as working together with other people and leading and guiding them in various ways is a big part of their job. These people are also often the ones facilitating meetings and human skills play a big role in that as well, as described by one interviewee: *"I feel that it is necessary that the person facilitating the meetings has good human skills. They need to be sensitive for the feelings of others and able to react to things accordingly. – They need to have the right attitude and energy too, so that it feels good for everyone to join the meetings and join the discussion."* Based on the discussions, it seems that many people working in HR or as producers or leads have a high level of human skills. However, the standards they need to meet are high too, and there have been some situations where those have not been met.

When looking at the interview data, it also appears that the connection between human skills and Agile implementation is complex and many-sided. First of all, one interviewee describes that the Agile ways of working revolve around communicating with each other,

coordinating the work together and helping each other, implying that Agile is built on human skills and human interaction. Another interviewee seems to agree with this when phrasing the connection of Agile and human skills this way: *“With Agile, my biggest realization is that ultimately it is all very simple and built on interpersonal skills. Studying and knowing the theory is important so that you have the vocabulary and the tools needed, but in the end, the main thing is to be good with people.”* It is also said that when implementing Agile, learning the practices and processes is not overly difficult as long as there are people who can make the learning process feel okay and make sure everyone is heard and supported; with human skills, the discontent towards Agile and change can be eased.

Another perspective to the connection of human skills and Agile implementation is that implementing the Agile practices can significantly improve communication, teamwork and team building, which can in turn have a positive effect on human connections and human skills. This possibility was mentioned by several interviewees, and some of them also suggested that the Retrospectives can be particularly beneficial in this context, as they are the place where team members meet each other to discuss the challenges faced and improvements needed in the team. The third viewpoint linked to the connection between Agile implementation and human skills is that the remarkable role of human skills also means that failures in this area can have serious consequences for the implementation process. A few interviewees mentioned this possibility and one also further described that when an unpleasant or bad interaction happens and it is connected to Agile in one way or another, it can change the way people affected view Agile itself. It is likely that mistakes and human errors happen during the implementation, but because the consequences can be severe, it is crucial to address and process them appropriately.

To summarize it, human skills are clearly valued in the organization and many think that they play a significant role in the Agile implementation process. However, there are also situations where they have not been adequately applied, which means that there is still room for improvement.

5 DISCUSSION

In this chapter, the findings of the case study are discussed in light of the previous research. The chapter is divided into four sections: First, the focus is on the way the case company has conducted the Agile implementation process and on the benefits and challenges connected to it (section 5.1). After that, the focus moves on to human skills and their role in the Agile implementation process (section 5.2). Then, the biggest challenges the case company faced during the implementation, meaning the challenges with Product Ownership and Product Backlog, are discussed (section 5.3) and some supplementary, relevant topics are presented (section 5.4).

5.1 Agile implementation practices

5.1.1 Recognition, Diagnosis, and Planning -phases of the Agile implementation

In the case company, the Agile implementation process was started by recognizing the need for change. Many view that this step was taken in the end of 2020 when the Towards Better Production -survey was organised. The results of the survey indicated that something needed to change, as there were various difficulties in development processes and many were stressed out and unhappy with the way work was done. The main reasons behind these issues appeared to be increasing complexity of the game projects and the growth of the organization combined with a lack of structure. The existing theory of this area is well aligned with the case. Hayes (2018, pp. 68–85) describes that change process starts with recognizing the need for change and starting the change, while other research articles add that in the game industry, the need often arises from the complexity of game development or from the swiftly changing nature of the industry (Koutonen and Leppänen 2013; Engström et al. 2018). Prior research also suggests that from the beginning, it is crucial that the leaders of the change process have a strong understanding of Agile and how it can be applied in practice (Rigby et al. 2016). For some part this can be achieved by studying Agile, but practical experience is also valuable. For this reason, using experienced internal or external coaches giving advice and guiding the implementation process is recommended (Vodde et al. 2014). In the case company, many of the leaders of the change already had previous understanding and experience of implementing and executing Agile, and they also spent time studying Agile during the process. External coaches have also had a role in the implementation, both conducting the trainings for the

organization and its employees and helping with organizing the day-to-day rituals and practices in the organization during the LeSS-implementation.

After the need for change was recognized, the case company started to diagnose the situation further. The key person doing the diagnosis was the Development Director, who conducted interviews, one-on-one meetings, and workshops to analyze the situation, discuss possible solutions, and to form a vision of the future. Big part of the vision was to find a sustainable way of working by building appropriate structures and practices. At this point, implementing Agile was also profoundly considered, and as its values and principles appeared to fit the organization and the industry and serve as a good starting point for the organizational development, that became key part of the vision. After the results of the diagnosis and the vision were discussed and approved on high-level, they were shared throughout the organization as a Development Director's Intent 2021 - document. These findings resonate with previous literature, which suggests that at this phase organizations should diagnose their present state and its problems and opportunities, form a vision of a preferred future state (John Hayes 2018, pp. 22–41), and consider the effort needed to do the change and the effects the size of the organization or the industry it operates in may have for the process (Rigby et al. 2016; Uludağ et al. 2021).

Next, the case company started to plan the change on a more detailed level. They chose to implement the Scrum-framework, which was later expanded into LeSS, and started to define the team and meeting structures and other such factors. There were several things the organization wanted to address and improve, such as making the roles and responsibilities clearer, improving the communication, information-sharing and feedback processes, and keeping the quality of the product high throughout the development. It was seen that actions like these would make the development process more manageable and decrease the need for reactive work. The above findings are aligned with the theory, where it is said that in planning-phase the focus is on defining how the vision and related goals will be achieved (John Hayes 2018, pp. 22–41) and on choosing the Agile framework to implement and anticipating its tailoring needs (Sommer 2019). However, it is still recommended to start with a standard version of a framework to have a solid foundation for further adjustments (Larman and Vodde 2017, pp. 53–71). This is what the case company has attempted to do, too; to start with the basic versions of Scrum and LeSS and only make changes when the original methods and events have been tried and the improvement needs identified. However, Product Ownership has been an exception

to this. From the beginning, there has been a certain group of people closely cooperating with and supporting the Product Owner, who has the final decision-making power. First, this group was the Project Leadership Team, and later the Product Owner Group. This structure was created because from early on it has been clear that one person cannot handle all the areas needed. Other than that, the planning phase in the case company is aligned with the guidelines described in the earlier research. In addition to what was already said, they have also planned the usage of both top-down and bottom-up methods to get the whole organization involved, as suggested by Larman and Vodde (2017, pp. 53–71), and attempted to keep the plans flexible to make it possible to react to appearing challenges nimbly, as suggested by Sommer (2019).

With the Agile implementation in the case company, one remarkable matter to acknowledge is that it is hard to distinguish clear boundaries between the Recognition-, Diagnosis-, and Planning-phases of the implementation. That is because some actions, like training, have spanned over several phases, and because the schedules of the game projects have affected the timings of the events. Postmortem-meetings where the previous work methods were discussed and reviewed, for example, could be associated with the Diagnosis-phase, but due to the project schedules they had to be organized at a point when the Planning-phase was already widely started. The above findings support notions in previous literature that explain that unclear boundaries between different phases are quite typical and not an issue as long as all the necessary actions get done (John Hayes 2018, pp. 22–41). Another remarkable matter considering the first three phases of the implementation process is that one person, the Development Director, has had a very central role in these phases. They have been the primary person diagnosing the situation and making the initial plans, and while they have asked others to share their opinions, contribute to planning and evaluate the results, they have been the one driving the change. After these three phases, however, the main responsibility of implementing the change was shifted from the Development Director to the game development teams.

5.1.2 Implementation and Sustaining -phases of the Agile implementation

The case company started implementing Agile by providing everyone training about it. That happened already in summer 2021 when some of the planning was still ongoing. At that point, Scrum-training sessions were organized for everyone in the organization so that they would learn the basics of Agile and Scrum and the principles and concepts connected to those. Later there was also LeSS-training offered for everyone in the

development teams and Product Owner -training aimed for certain key people. Over time, several other ways were also used to instil Agile in the organization, such as coaching, using external consultants, and comprehensively communicating about Agile. Organizing all this training and coaching is well in line with what is suggested in literature. There it is said that in Implementation and Review –phase it is important to provide training and other support for everyone to make sure that they understand Agile, its principles, and the reasons for the implementation (Kiv et al. 2018).

In autumn 2021, the implementation-phase started at full power and the case company implemented the new structures, rituals and tools to their systems. This includes concepts like Product Ownership and Product Backlog, Sprint Structure and the related meetings, and a new team structure. With the teams structure, the aim was to form Feature Teams that are self-managing and cross-functional and have less than ten members, as the literature suggests (Vodde et al. 2014). As discussed in sections 4.2 and 4.4, that has been only partially successful due to the size and complexity of the game development tasks. Implementing the Sprint-structure and the Scrum- and LeSS-meetings and other rituals has been done following the guidelines defined in these frameworks, and it has been successful for the most part.

In the case of reviewing the results and seeking feedback, which also are actions that the literature recommends doing regularly (Rigby et al. 2016; John Hayes 2018, pp. 22–41), the case company has mostly trusted that feedback will be shared in team- and overall-Retrospectives and in other discussions the leaders of the change have with the teams and individuals. No specific Key Performance Indicators or KPIs have been defined so far because there is a concern that they could easily become the only focus points for the implementation, which could result in losing the holistic view on the process. With a broad implementation process like this, losing it could lead to misjudgements and failures. This is an interesting view because in existing research, KPIs are often seen as useful and beneficial tools for measuring and developing the functions of the organization (Lindberg et al. 2015; e.g., Krause and Dayanand 2019). Then again, LEGO Group apparently implemented Agile successfully with no definitive implementation plans or KPIs and found that approach effective as it enabled rapidly responding to emerging issues (Sommer 2019). Unfortunately, that particular topic is not widely discussed in Sommer's article, which means that no definitive conclusions can be made. It appears that this should be researched more in the future.

The final phase of the change or implementation process is sustaining the change and making sure the new ways of working become the new norm (John Hayes 2018, pp. 22–41). The key people leading the Agile implementation seem to be aware of this phase and its significance, even though the company has not reached it yet. Many of them also recognize that it can take a long time to truly internalize the new ways of working. That is a promising sign; in the previous research, it has been noticed that too ambitious and strict roll-out timeframe can decrease the success of the implementation, so it is better to give it the time needed (Kalenda et al. 2018).

5.1.3 Managing people issues and learning in Agile implementation

There are two activities, managing people issues and learning, that need to happen throughout the implementation process to make it successful (John Hayes 2018, pp. 22–41). Managing people issues in the case company is described more in detail in the following section 5.2, but to say it briefly, that has been done rather well. Continuous learning, which makes it possible to adjust and improve actions as needed and is important for every level of the organization to practice (John Hayes 2018, pp. 491–500), is also something the case company has put effort into. They continuously encourage teams and individuals to improve the way they work and to experiment with new ways of doing things, for example organizing meetings. The retrospectives are seen as a central tool for discussing and planning these improvements. This approach is well aligned with the literature, where it is recommended that Agile teams should continuously improve their work methods with the tools Agile frameworks provide (Sommer 2019; Schwaber and Sutherland 2020). Lately the case company has also announced that the last Friday of every two-week Sprint is a Learning Day, and all the individuals, teams, and crafts are encouraged to spend that day on learning new useful skills individually and together. This is a new attempt in the organization and the results are yet to be seen. However, it clearly demonstrates how strongly learning is valued there.

5.1.4 Benefits and challenges in Agile implementation

The Agile implementation seems to have already brought some benefits for the case company. Improved communication, for example, has helped many to stay more up-to-date with what happens in the project and to know about tasks and needs more in advance, which has made it easier to prepare for them. More defined processes and structures have made cooperating and communicating easier than before, while the concept of self-

managing teams has enabled teams to have a bit more control over their work than before. Some also say that the organization striving to improve has positively affected their motivation. Overall, it seems that so far the Agile implementation has had a relatively positive effect on the productivity in the organization. However, as the implementation is still ongoing, the full benefits are not yet accomplished.

When comparing the benefits of Agile implementation realized in the case company to those discovered in previous research, it seems that they have quite a lot in common. The benefits mentioned in previous research that are connected to employees and their performance and wellbeing, such as improved motivation and satisfaction among employees, better communication and coordination inside and between teams, and higher productivity and adaptability (Serrador and Pinto 2015; Ruonala 2016; Sommer 2019; Uludağ et al. 2021) seem to all have realized in the case company to some extent. But in contrast, the benefits that are linked to the projects and products, for example better scope and time management, shorter project delivery times, better quality, lower risk levels, and more value provided to the customer (Koutonen and Leppänen 2013; Serrador and Pinto 2015; Rigby et al. 2016, 2018; Ruonala 2016; Sommer 2019; Uludağ et al. 2021) are not visible yet. This is an interesting division, and the data collected from the case company does not give a clear explanation to this. One possible reason could be that changes in people and their attitudes and behaviours may be easier to notice at early stages of the implementation, while changes linked to the projects and products may be noticeable only after the new ways of working have been in use for some time. However, finding the definitive reason for the division would require more long-term monitoring.

In addition to the benefits, the case company has also faced many kinds of challenges during the Agile implementation. Those can be divided to five categories: challenges in leading the development, discontent with change and Agile, challenges with roles and team structures, difficulties in achieving real agility, and other challenges, such as the complexity of the project and the pre-production phase. First of these, challenges in leading the development, refers to challenges with Product Ownership and Product Backlog, reviewing and prioritizing tasks, and defining the level of detail the directors work with. These are arguably the greatest challenges the company has faced so far during the implementation process. What makes them interesting is that in the existing research, their significance is not considered much. There are a few comments that in some companies leaders and managers may find it hard to give space for self-managing teams

to make their own decisions (Rigby et al. 2016; Sommer 2019), but no further discussion of the topic. This is a remarkable distinction between the case and the previous research, and for this reason more attention is paid to it in section 5.3.

As said, there has also been some discontent with change and Agile in the case company, meaning that transforming the existing ways of working has had its challenges, as there are people in the organization that value the old ways of working and are sceptical towards the new ways, especially towards those that have not been exceeded yet. Also, not everyone has been happy with the implementation practices, such as the trainings organized. These findings resonate with the previous research, where it has been noticed that unwillingness to transform the existing ways of working and lack of commitment are rather common challenges when implementing Agile (Kalenda et al. 2018; Conboy and Carroll 2019).

Previous research also recognizes that in large organizations the high number of teams raises the need for coordination and communication and makes the dependencies and work processes in the organization complex (Uludağ et al. 2021) and that balancing the new frameworks with existing, complex structures is also hard there (Kalenda et al. 2018; Conboy and Carroll 2019). This complexity has been noticed in the case company too, and combined with the complexity of their projects, it is a big challenge for them. It also appears to be the reason behind some other challenges they have faced, such as the challenges with roles and team structures and especially with organizing effective communication and coordination between teams and defining the feature team structures. In addition, they have also faced difficulties with achieving real agility in the case company, as the constraints and limited resources in the projects have made it hard to be truly Agile and to have enough time for iteration and improvement. However, in the literature it is emphasized that implementing Agile typically takes quite a lot of time (Kalenda et al. 2018), so possibly it is just too early to expect real agility from the teams and individuals yet.

5.1.5 Effects of the organization size and game industry on Agile implementation

All in all, when assessing the Agile implementation process as a whole, it appears that the case and the structure of the change process defined by Hayes (2018) are rather congruent. All the implementation phases described in the structure are visible in the case, and the activities mentioned have been put into practice. Even though there are also some differences between the case and Hayes' structure, those do not seemingly have strong

effect on the big picture. Ultimately, the results of the case study seem to support the results of the previous research considering the Agile implementation process from the beginning to the end.

When inspecting the effect the size of company can have on the Agile implementation, it appears that large companies often turn to Agile to decrease or better manage the complexity they face in their organizational structures and projects (Alqudah and Razali 2016). This is the goal in the case company too; to find structures and practices that help to manage the growing organization and its complicated projects. However, at the same time this complexity also makes the implementation harder, as there are more issues to plan, manage, and coordinate and more systems and processes to balance (Kalenda et al. 2018). To put it simply, the complexity itself makes it harder to solve the issues caused by the complexity, and as a result, it seems like it is just impossible to find easy ways to do big projects. There are systems and methods that work better in these environments but no solutions that would solve all the problems. This view is shared in the case company too; they acknowledge that in big projects, despite all your efforts, you just have to learn to accept and tolerate a certain amount of complexity and uncertainty.

When thinking about the effect working in game development has on Agile implementation, it appears that in the case company they see that the values and principles of Agile development suit game development well. That is why they believe Agile frameworks are a good starting point for their own organizational development, even though they might need to be heavily modified to meet their specific needs. Previous research seems to have a similar view on this issue: there it is argued that no Agile model or framework can by itself meet all the needs of game development, and any model needs to be adapted and modified to get successful results (McKenzie et al. 2021). But still, the Agile models seem to be good solutions for the industry, as they focus on solving issues with human interactions (Sommer 2019), and the typical challenges in the game development industry are connected to those (Politowski et al. 2021).

5.2 Agile implementation and human skills

As described before, implementing Agile is a massive change for almost any organization, and it appears that human skills have important roles in the process. This applies to the case company, too. There it has been important, for example, for the leaders

of the change and other key people to give support for others whenever that has been needed. Psychological safety and trust have also had a remarkable role, as through them it has been possible to create an environment where people feel safe to share their views and give honest feedback about the implementation process and other things they have on their mind. Resilience appears to be valuable too, both with tolerating the change and Agile but also the complexity of the project work in general. All of these human skills are mentioned in previous research as well: Kalenda et al. (2018) emphasizes the importance of management support in Agile implementation process, while McHugh et al. (2012) and Diegmann and Rosenkranz (2021) say that trust, resilience, and psychological safety have a central role in practicing Agile.

However, there are also several other human skills that have been important for the case company during the Agile implementation process, and they are empathy, appreciation, understanding diversity, and communication. Out of these skills, the first three have been important in building an atmosphere where everyone can feel good and appreciated, while communication has been an integral tool in all work, planning, and information-sharing. These human skills have not been specifically mentioned in previous research connected to Agile implementation, which makes this a remarkable finding. What makes it even more interesting is that many of them have been recognized in change management research and literature, though, for example in John Hayes's (2018) book. It appears that it would be valuable for Agile and Change Management literature to communicate with each other more.

One interesting finding that emerged when discussing the human skills in the interviews was the role of conflicts and conflict solving skills in creative work and Agile implementation. In the interviews, several people said that they view conflicts as a natural and rather neutral part of the development process, and that human skills are an important part of handling them. That makes sense; conflicts appear to be important for creative processes, since often the best solutions for technical and creative problems are found through well-founded arguments and profound discussion – through conflicts, basically. This concept has been recognized in literature, too, and for example DeGraff et al. (2017) have written a book that about it. However, even though it has been rather widely recognized that conflicts are not automatically a negative thing, it is still crucial to remember that they can have severe negative consequences. To avoid that, organizations

need to build a culture where human skills are valued, the level of conflict-solving-skills is high, and conflicts are addressed appropriately and in a timely manner.

Another interesting finding linked to human skills and Agile implementation is that their relationship appears to be cyclic. In the case company, they seem to understand this to some extent: the significance of human skills in the Agile implementation is well acknowledged, and many also believe that the Agile practices will improve the communication and teamwork in the organization, which can in turn improve human connection and human skills. This finding seems to be aligned with the previous literature, where it is mentioned that while good human skills make it easier to implement and be Agile, Agile practices can also strengthen these skills (McHugh et al. 2012; Thorgren and Caiman 2019). As a result, it is likely that progressing with Agile will make it easier to manage the people issues in the long run. When thinking about this phenomenon, it appears that it is actually a great opportunity for the organizations implementing Agile: if they are able to start a cycle where improved human skills are applied to advance the implementation, which in turn improves the human skills even more, they can achieve great results through it.

During the research process, it has also been interesting to realize how important it is that the drivers of the change have excellent human skills and connections. In the case company one example of this is the Development Director, who has been a key player in the Agile implementation, especially in the first phases of the process. This person is an experienced and trusted member of the organization, and many see that the decisions they make are good and reasonable. In the interviews there was an impression that because the decision to implement Agile was voiced by a person with this position and trust, it was easier to accept for many and it is also easier to believe that the results of the implementation will be worth the effort. In contrast, if the decision had been voiced by someone who is not as trusted and not an integral part of the organization, for example by an external consultant, the reception would have almost certainly been more skeptical, even if the reasoning behind the decision had been the same. It appears that human skills and human connection can heavily affect the emotional responses, even though the logical reasoning behind the change stays the same. This phenomenon has been recognized in Change Management -literature too, where for example Hayes (2018, pp. 86–118) has discussed both the importance of the human skills of those driving the change and the typical differences in attitude towards internal and external change leaders in his book. In

Agile-literature these things are not covered as well, which further suggests that it would be valuable if Agile and Change Management -literature communicated more with each other.

All in all, it appears that human skills play many important roles in implementing Agile in large game development organizations and are an integral part of making the implementation successful. As stated in the interviews, learning the practices and processes of Agile frameworks is generally not overly difficult - as long as there are people who can hear and support others during this process.

5.3 Product Ownership and Product Backlog

As mentioned, Product Ownership and Product Backlog have been one of the most challenging concepts for the case company to implement. They were briefly covered in 5.1.4, but in this section they are discussed more in depth, as there are several interesting aspects connected to them. To begin with, it is good to note that the significance of the challenges connected to implementing Product Ownership -structure and Product Backlog is an important finding in this study, as it has been rather overlooked in the literature. Some researchers, like Kalenda et al. (2018) and McKenzie et al. (2021), recognize that the large size of projects can cause difficulties for Product Owners and make the Product Backlogs complex and hard to manage. Still, these descriptions do not capture the intensity of the difficulties encountered in the case company and the effort needed to solve them.

In the case company, the difficulties connected to Product Ownership during the Agile implementation have been quite diverse. For example, during the Scrum implementation process the Product Owner structure became mostly a reviewing body watching over the quality of the product with not much power or focus on planning the work. One major reason for this particular issue was that several key players were needed in another project during that time. The issue is now largely resolved as the projects have changed, but there are still some others left, such as how to share the product vision effectively and successfully throughout the organization and lead the work. The communication about the current Product Owner structure – having one Product Owner who has the final say, and the Product Owner Group working with and supporting that person – throughout the organization has not been fully successful either, as the structure is still unclear for many.

These issues are not covered in detail in previous research, but some have discussed slightly similar topics: McKenzie et al. (2021) for example say that it is important for teams to receive proper direction from the Product Owner and to be able to see the game vision regularly and check if their assumptions about it are correct. According to them, meetings like Sprint Reviews can work for that. With Product Backlog, it has been challenging for the case company to find effective workflows for creating, categorizing, and editing different types of tasks and visualizing the connections between those. In addition, when the current project progresses it will likely become hard to have a comprehensive understanding of the Product Backlog as it grows so big. In the case company, they have tried to follow the LeSS-guidelines defined by Vodde et al. (2014) when organizing meetings and working with the Backlog, but those areas have not been mastered yet.

There are several reasons why Product Ownership and Product Backlog have been such challenging concepts for the case company to implement. One of the reasons is rather contextual: these concepts were new to the organization. A common, regularly updated Product Backlog that would cover all the tasks needed to finish the product had apparently never been used in the company, and Product Ownership being such a wide concept and covering areas like market influences, stakeholder connections, and delivery schedule in addition to the creative vision was also a new thing to them. In contrast, many other Agile practices and structures, like the Sprint structure, were already familiar in the organization and for this reason they were also much easier to implement. Several articles (e.g., Kalenda et al. 2018; McKenzie et al. 2021) resonate with this finding as they mention that previous experience in Agile can significantly help the Agile implementation, while inexperience in certain concepts often leads to misunderstanding or misimplementing them. Another, even more significant reason why the Product Ownership and Product Backlog have become such a major challenges for the case company is probably the complexity of game development. When developing AAA-games, there are countless things the Product Owner needs to understand and plan, and this also makes the Backlog massive and complex. This complexity and its effects to Product Ownership and Product Backlog management has been recognized in previous research too, primarily by McKenzie et al. (2021).

An interesting challenge linked to Product Ownership the case company has been struggling with is defining the level of detail Product Owners and directors should work

with. It has been hard to find sufficient level for it, as the views of the people working in these roles variate a lot, as well as the needs of the teams and tasks. This issue has been mentioned in the literature too, where it is said that during Agile implementation leaders and managers may find it hard to give space for self-managing teams to do the decisions on their own (Rigby et al. 2016; Kalenda et al. 2018; Sommer 2019). However, the articles do not elaborate the problem much, and as a result it seems that the challenge faced in the case company is more multidimensional and intense than what is described in the literature.

Basically, the issue with defining the level of detail Product Owners and directors should work with is about finding the balance between providing enough direction to the teams but also making sure that does not turn into micro-management. In the case company, there are some concerns that without enough guidance, misinterpretations can happen and the priorities may be unclear. Teams and individuals may also have varying needs and preferences for guidance, which leads to a situation where it is challenging to find common ways to guide them. However, these seem to be issues that could for the most part be fixed with good and regular communication and review loops, with no need to micromanage the details. Actually, the Sprint structure and the guidelines of Agile frameworks already try to respond to this challenge by providing several points where to plan the work, communicate about it and review it, such as Sprint Plannings and Sprint Reviews (Vodde et al. 2014; Schwaber and Sutherland 2020). Finding ways to utilize the existing systems effectively could be a good approach to solve the issue.

Another concern that has emerged in the case company is that the directors and Product Owners often have lot of practical expertise that can be valuable when solving practical issues, and it may not be wise to leave that expertise unutilized. However, if these people spend a lot of time solving hands-on challenges, it is likely that they do not have enough time to do their other tasks. If the directors and Product Owners, the people responsible of making the decisions and generating work for everyone else in the project, become the bottlenecks of the production, that can have major negative consequences. The case company has actually undergone this in their previous projects, partly because some key roles were unfilled at the time, and it delayed numerous tasks and slowed down the processes considerably. Based on these experiences, it seems that to ensure the development as a whole progresses as well as possible, it is necessary for the Directors and Product Owners to primarily focus on high-level and strategic work whenever that is

needed, and only participate in more practical work when they actually have time for that. The issue described above is not covered in detail in Agile-literature, but something that is emphasized there is that Product Owners have a crucial role in guiding the work and maximizing the value of the product (Vodde et al. 2014; e.g., Schwaber and Sutherland 2020). That should not be dismissed.

Overall, the challenge with defining the level of detail Product Owners and directors should work with is an interesting issue with no explicit answers. The issue is also deeply connected to human skills, especially trust. If the managers, in this case the directors and Product Owner Group members, do not fully trust the teams, that may lead to micromanagement and suboptimal distribution of time and effort, as described by Artto et al. (2011, pp. 82–123). In contrast, showing trust towards teams and individuals fosters a sense of empowerment in them (Wilson 2009). Sometimes the lack of trust can be justified though, for example in a situation where the teams are not properly resourced. In these situations, it is necessary to increase their resources and to make sure the tasks they need to complete are balanced with those.

As said before, the challenges with the Product Ownership and Product Backlog and especially the importance they have had in the case company are a significant finding in this study. It seems apparent that the unique features of the AAA-game industry, especially the huge size and complex nature of the products developed, make these challenges more pressing and harder to overcome. This is suggested in the previous research too (e.g., Kalenda et al. 2018; McKenzie et al. 2021). However, as the research of Agile implementation in game development industry is rather scarce, the topic has not been studied in depth yet.

5.4 Supplementary topics

5.4.1 Agile framework as a starting point

One interesting finding in the research was that the Agile frameworks are only starting points in the process of becoming Agile, and they need to be modified by the organizations to make them work in their specific contexts. The theme was mentioned in the literature (Kalenda et al. 2018; Sommer 2019) and heavily emphasized in the case company. In the interviews, several people described that the frameworks are only part of the organizational development process and the tools and practices described there

need to be further improved to better serve the organization. There is one practical example that well reflects these views of the case company; as described before, most of the practices defined in the Scrum and LeSS frameworks are already implemented there. However, no-one considers the implementation finished, as the practices have not become routine yet nor been improved and modified enough to really meet the needs of the organization. This reveals how the practices and structures themselves are not the main point of the implementation, but the deeper progress and development in the organization culture. It also resonates with Agile literature, where it is said that the most important part of the Agile implementation is developing the organization culture and mindset towards Agile values and principles, not implementing a specific set of tools and practices (Sommer 2019). Furthermore, it appears that this concept of Agile frameworks being only one step of the organizational development process is closely connected to change management theory and especially sustaining change. As Hayes (2018, pp. 1–42) describes, implementing the change is not enough: the organization culture needs to transform, too, and the new ways of working need to become the norm.

There are also some other perspectives and tools connected to seeing Agile frameworks only as starting points. One perspective emphasized by multiple people in the case company is that the practices and structures of the Agile frameworks should not be judged before they have been tested. Even though a certain practice may seem complicated at first, it is hard to anticipate the value it can bring without testing it a few times. Also, even if the experiment did not work out, it would still provide the organization valuable learnings. Experimenting with different approaches is valuable, as that is a way to gain more practice and knowledge and find new ways to improve. Retrospectives then, they are seen as a great tool for giving feedback of these experiments and other matters, and also for sharing improvement ideas related to any issues the teams and individuals face. Retrospectives are also a way to give everyone a chance to be heard, as they are organized in every team once every Sprint, and the issues discussed there can then be escalated to other arenas as needed. Another thing that makes Retrospectives such an important tool in the case company is that organizing them regularly and successfully makes it possible to catch issues and challenges early on and encourages solving them right away. It is often the best approach to do small improvements often and try to solve the issues before they become too big and complex.

However, even though improving the frameworks and other practices is repeatedly discussed, it appears that perfection can never be reached. As one person mentioned in the interviews, there will always be things that do not work flawlessly and need to be improved. Working with huge and complex projects will never be easy, and that is something that needs to be tolerated and lived with. With limited resources, the most important thing is to make sure that the development progresses as a whole, and the smaller issues and deficiencies can then be prioritized and fixed as needed.

5.4.2 The importance of principles

Another interesting idea learned during the research process was the importance of the values and principles of Agile and different frameworks. In literature, several researchers have emphasized their significance and stated that internalizing the core values and principles of Agile is the best way to implement it (Kalenda et al. 2018; Sommer 2019). In the case company, the Agile values and principles seem to be highly valued, too. Based on the data, it appears that they chose Agile and Scrum- and LeSS-frameworks primarily because those have good and sensible principles that fit the goals and values of the company. It was mentioned that other Agile frameworks could have been rather good starting points for the organization too, as they also have sound principles. Above all, it appears that one of the main goals in the case company is to accomplish a situation where everyone has an intuitive feeling of how to implement the Agile values and principles in their daily work and decision-making. This approach is aligned with the idea described by Hohl et al. (2018) that “being Agile” and having internalized the Agile values and principles is a much better mindset than “doing Agile”, which refers to just focusing on executing a framework.

There are also interesting ideas how the principles that were originally created for different industries can be exercised in a game development company. These came up in one of the interviews, and they are linked to two principles, “Go and See”, and “Stop and Fix”. These are part of Lean Thinking, which in turn is one of the principles of LeSS. “Go and See” means that to thoroughly understand situations and give help, people need to go and see for themselves the real place of work (Vodde et al. 2014). In a game development company, the managers, for example, can practice this by going and talking directly with the developers and seeing how they do the work and what issues they face, and also by joining their meetings to see how those are facilitated and what are the topics handled there. “Stop and Fix” then refers to stopping the system and carefully fixing the problems

whenever they happen to increase the quality and to reduce the amount of technical debt (Vodde et al. 2014). In game development, an example of this is stopping the development work when there is a severe bug in the game or in the game editor and focusing on finding its root cause and fixing it. These ideas demonstrate how the concepts developed for other industries can be implemented in game development, too. The most important thing appears to be to understand the goal behind a certain practice, and only then implementing the applicable elements of it.

5.4.3 Change Management and implementing Agile

The connection between change management and implementing Agile, which has been rather apparent throughout the research process, is yet another interesting topic to discuss. When reading the literature, it seems evident that the ideas and principles of change management can be applied in Agile implementation, and it also appears that the Agile implementation process follows the basic change process structure. In the case company this perspective was not widely addressed though, perhaps because not many people there are familiar with change management theory. Nevertheless, a few people offered some insights about the relationship of change management and Agile implementation. For example, it appears that one of the basic ideas of change management, using both top-down and bottom-up approaches to make the change successful and long-lasting, is understood and utilized in the case company. It was also noted that in the implementation, like in all change processes, one of the first and most challenging steps is to sell the idea of the change and to convince the members of the organization that it is needed. Yet another learning emphasized in the case company that is also familiar from change management theory is that it is important to have patience when implementing change. The different phases of the process can take a lot of time, and as a result the whole process almost always takes more time than originally planned. It is also necessary to make sure that everyone in the organization understands the length of the process and that it takes time, effort, and constant improvement to make it successful.

An important factor connected to change management in the case company has also been the size of the change; the transformation had to be big and wide enough so that it could truly address the issues organization has encountered and change things for good. In one interview, it was stated that many organizational changes fail because there the focus is too much on fixing and optimizing only detached parts of the organization, not the whole; the change needs to be comprehensive enough to be truly successful. Larman et al. (2017,

pp. 53–75) describe a similar phenomenon in their book: according to them, it is much better to do true, system-wide improvements than local optimizations that only improve one perspective of the system and can potentially harm other areas. This issue is also connected to the organization culture, as with small and partial changes there is often no possibility to truly change and improve the culture and community of the company. Based on the interviews, it seems likely that by making a change that is remarkable enough and by doing the right things during it, like providing training and other shared experiences for everyone and giving them possibilities to communicate and coordinate together, a sense of community, mutual trust, and a common outlook towards business problems can be achieved.

In the interviews, there was a comment made that it could be valuable for the case company and their implementation efforts to understand change management theory better and more widely. It was also said that the situation is probably similar in many other game development companies. As a result, it appears that even though the connection between Agile implementation and change management exists, the companies working in the game industry or implementing Agile are not fully aware of it. Recognizing it could provide additional value to them and help them become more successful with the changes they are trying to implement.

5.4.4 Transparency and visibility in implementing Agile

Yet another interesting observation made during the research process is the role transparency and visibility have in Agile implementation process. In section 4.4, it was mentioned that some people in the case company feel that the learnings from the previous project have not been utilized well enough in the current implementation process. However, it appears that some other interviewees have an opposed view to this as they describe that identifying and defining the challenges and successes of the previous project was a significant part of planning the implementation. When considering these differing views, it comes to mind that maybe the issue is not solely on ignoring the learnings of the previous project; maybe lack of transparency is a part of the problem. If the level of transparency and collaboration has been low in the planning phase, the connection between the identified challenges and the chosen and implemented frameworks and practices may be unclear for some. In that case, increasing the level of transparency could decrease the level of skepticism towards the change by making this connection clearer for everyone.

In the interviews, there was also another concern raised that may be related to transparency and visibility in the organization. In section 4.4, some people said that they feel there are currently not enough suitable forums to comment and give feedback about the Agile implementation. But at the same time, many others say that there are numerous meetings and other opportunities organized where to give feedback and discuss. Maybe the opportunities and the people do not always connect? For example, it is possible that people in certain roles have more opportunities to share their views and give feedback, while others have less of them. However, it is also possible that not all people realize the opportunities they have or are prepared to take them once they appear. Whichever is the reason behind this concern, it seems clear that it could be solved by double-checking that everyone in the organization has enough chances to comment and give feedback and that they are all also aware of them. Having enough visibility on the feedback flow is important, as that helps everyone to know well in advance when and where they can give it. In addition, it is valuable for the organization to receive feedback about the implementation process, as that helps to monitor whether the change plan is working and to make the adjustments needed (John Hayes 2018, pp. 22–42).

In the case company, it also appears that the level of transparency has shifted during the Agile implementation process. There is a rather valid reason for this: during the first phases of the implementation, not many people yet understood the concepts of Agile, and that is why diagnosing the situation and planning the first steps was on the shoulders of only a few people. But over the months and through the trainings and the experience gained in the implementation process, understanding of Agile concepts, practices, and values and also other abstract concepts has significantly increased in the organization. As a result, the amount of transparency has been increased too, and progressively more people have been invited to join the discussion about the next steps and future goals of the Agile Implementation and the organization overall. In a relatively short period of time, the training and other experiences have made it possible for a wider group of people to join these conversations and have meaningful input there.

These findings about the level of transparency in the case company, they are mostly speculation with no conclusive evidence. However, it could still be valuable for the case company to double-check the level of transparency in their organization and see if there is something to adjust. On a general level, it appears that transparency has an effect on implementing Agile. Without enough transparency and visibility, people may feel

excluded and become skeptical towards the change – if the reasons behind the decisions are not shared, they cannot be understood and accepted.

6 CONCLUSIONS AND RECOMMENDATIONS

In this chapter, the thesis is concluded. First, the key findings are presented (section 6.1). After that, the theoretical contribution of the research is described (section 6.2), followed by the managerial implications (section 6.3). Then, the results and limitations of the study are assessed (section 6.4), and finally the ideas for future research are discussed (section 6.5).

6.1 Key findings

In this study, the aim was to examine the implementation of Agile practices in the game industry and address the following research questions:

1. How do large game development organizations implement Agile working practices, and what are the related benefits and challenges?
2. What kinds of roles do human skills play when implementing Agile in large game development organizations?

Regarding the first research question, this study showed that large game development organizations appear to follow an Agile implementation process similar to the general change process structure defined by Hayes (2018). The structure consists of seven core activities; recognition and start, diagnosis, planning, implementation and review, and sustaining change, which happen in sequence, and learning and managing the people issues, which need to be performed throughout the process. This connection between the Agile implementation process and the change process structure is one of the key findings of the study and it appears that it has not been theorized nor recognized in previous research.

To form a comprehensive answer to the first research question, the benefits and challenges of the Agile implementation process in large game development organizations also need to be reviewed. In the case study, the benefits could not be definitively characterized, as the study had to be finished before the final benefits emerged in the case company. However, it appears that Agile implementation improves communication and makes the structures and processes of the organization well defined, which can have a positive impact on collaboration and productivity. In the case of challenges, the most

major ones identified in the case study are linked to implementing Agile ways of leading and guiding development, such as Product Ownership, Products Backlog, and defining the level of detail Product Owners and directors work with. The significance of these challenges is another key finding of this study. It appears that the unique features of the AAA-game industry, especially the complexity of the projects, make these challenges so pressing and hard to overcome. There were also other challenges recognized in the case study, and those were connected to things like roles and structures, discontent with change, and difficulties with transforming the existing ways of working. These challenges are congruent with previous research.

Regarding the second research question, the thesis uncovered that human skills have various, significant roles in Agile implementation process. There are several human skills, such as support, communication, psychological safety, empathy, and conflict-solving skills that are needed to make the implementation process successful. Also, it seems that it is particularly important that the leaders and drivers of the implementation process have great human skills and are trusted members of the organization. Additionally, it appears that the roles of human skills when implementing Agile can be cyclical; while good human skills make the Agile implementation process easier, the implemented Agile practices can also improve these skills by increasing the amount of communication, feedback, and knowledge sharing practiced in the organization. In the best case scenario, this connection can lead to a positive cycle where human skills and Agile practices have a continuous positive effect on each other.

6.2 Theoretical contribution

The case of this study, a large game development organization implementing Agile practices, is something no researcher has examined closely before. Overall, Agile implementation in the game development industry has not been widely studied, and especially large game development organizations and their implementation processes have been overlooked. That is a significant deficiency, considering the magnitude of the game industry and the game companies' interest towards Agile. For these reasons, the theoretical contribution of this study can be deemed valuable for the scientific community.

The first key finding of this study, identifying the connection between Agile implementation and change management and specifically the similarities of the Agile implementation process and the change process structure defined by Hayes (2018) is something that, according to my best understanding, has not been done in previous research. However, based on the case studied, the connection seems to exist. This is a valuable finding because it helps to see the Agile implementation as a significant organizational change project, which makes change management practices and tools as relevant for the process as Agile implementation practices.

In this study, it was also witnessed that the values and principles of Agile and different frameworks are important factors in an Agile implementation process. Their importance has already been recognized in earlier research, and now the findings of this study support that understanding and also demonstrate their roles in the implementation process. For instance, when choosing the framework for the organization to implement, it is good to critically analyze the principles of the possible frameworks and compare them to the values of the organization. In addition, for the organization to become truly Agile it is not enough to just implement new practices and structures; everyone also needs to internalize the Agile values and principles and start to intuitively apply them.

Identifying the significance of the challenges a large game development company can face when implementing the Agile ways of leading the development, such as Product Ownership and Product Backlog, is another valuable contribution to the theoretical understanding of Agile implementation. There is some research (e.g., Kalenda et al. 2018; McKenzie et al. 2021) that recognizes how the large size of projects can cause troubles for the Product Owners and make the Product Backlogs hard to manage. This study complements that earlier research by emphasizing the significance of these challenges and also extends it by suggesting that the reason behind them is the complex nature of the products developed in large game development organizations. This complexity makes it challenging to organize the work effectively and easily leads to a situation where the Product Backlog becomes massive and difficult to work with and where it is hard for the Product Owner to have a comprehensive understanding of the product and all its features and the dependencies between them.

Human skills and their role in Agile implementation were another focus area of this study. The study found that many kinds of human skills (e.g., support, communication, and

empathy) are needed to make the implementation process successful and that the connection between human skills and the Agile implementation can be cyclical i.e., they can both improve each other. These findings are valuable additions to Agile research, because even though previous studies have demonstrated that human skills are essential factors for successfully practicing Agile (Beck et al. 2001; McHugh et al. 2012; Thorgren and Caiman 2019) and indicated that the relationship of Agile and human skills can be cyclical (McHugh et al. 2012; Thorgren and Caiman 2019), their role in Agile implementation has not been studied previously in the context of large game development organizations, according to my best knowledge. Now the theory, meaning the central role of human skills in Agile implementation and practice, is better understood in different contexts, which improves its generalizability.

6.3 Managerial implications

This study has several managerial implications that can provide inspiration and value for both the case company of the study and for other companies working in game development or in a similar industry and planning to implement Agile. To begin with, the connection between Agile implementation process and the change process structure defined by Hayes (2018) is one of the key results of this study, and it also has remarkable managerial implications. Those leading Agile implementation or planning to start it should consider familiarizing themselves with the structure, because that can help to predict different phases of the implementation process and to choose the best actions for each phase. In general, it appears that understanding change management theory and change management practices can be useful for any organization implementing Agile.

Another implication for those leading Agile implementation is that it is wise to pay extra attention to the implementation of Product Ownership and Product Backlog, as those can be challenging in organizations that work with complex projects. It is recommended to allocate enough resources for these areas and prepare for a lengthy implementation process. In addition, it is also important to understand the integral role of human skills in implementing Agile and invest enough time and effort on developing and applying these skills. To improve the success of the implementation process, there are many actions related to human skills leaders can take, such as making sure the communication is good in every direction, people feel heard and valued and receive the support needed, and that any conflicts are solved. These actions are connected to change management theory as

well, where stakeholder engagement, managing the people issues, and reducing the resistance to change are seen as central things in implementing change.

Yet another finding of this study that also has its managerial implications is the importance of the values and principles of Agile and related frameworks, which is emphasized both in literature and in the case studied. The principles have a significant role right from the beginning of the Agile implementation process, when choosing the framework to implement: evaluating the principles of different frameworks and analyzing how those fit the goals and values of the organization helps to find the most suitable framework. Another matter tightly connected to Agile principles is the observation that “being Agile” is far better than only “doing Agile”, which refers to just focusing on executing a framework (e.g., Hohl et al. 2018). To achieve this mindset, the people in the organization need to internalize the Agile values and principles and know how to apply them in their day-to-day work and decision-making, and it is important for the leaders and managers to provide them the needed support and training. Also, in the later phases of the Agile implementation and after finishing it, it can still be highly beneficial to regularly remind people of these principles and values and how they can guide the work. Even if leads and managers were familiar with the principles, it is possible that not everyone else remembers to intentionally apply them.

In this case study, there have also been some indications of the importance of transparency and visibility in the Agile implementation process. This issue has some managerial implications, too: Without enough transparency and visibility during the implementation process, people may feel left out and become sceptical towards the change, and misunderstandings can happen. For this reason, it is important to make sure there is enough transparency and visibility in the organization and in the implementation process so that everyone can understand the decisions made and the reasons behind them and also join the discussion when needed. Providing training for everyone may also improve the situation, as that makes it possible for a wider group of people to join the conversations around Agile and give thoughtful input about it.

6.4 Research limitations

When conducting this study, both the time and the resources were rather limited, which had its consequences. The time limitations and deadlines led to a situation where the study

had to be finished while the Agile implementation in the case company was still ongoing. Due to this, it was not possible to study the last phases of the implementation process or analyze the final results of it. However, this issue was already predicted and taken into account when planning the research, and it was decided to focus on the ways the Agile implementation can be carried out and on the benefits, challenges, and human skills connected to it, as these were the topics that could be reliably addressed at that point in time.

Another considerable limitation for the research was that there was only one researcher working with it. That had multiple effects on the research, for example the number of interviews had to be kept quite low. There were efforts to neutralize these limitations, though. The interviewees were from various levels and roles across the organization to ensure the data collected was wide-ranging, and the interview results were also thoroughly analyzed to verify that no valuable information was lost. This approach also helped to overcome the subjectivity and possible biases of the interviewees, as the data collected was rather diverse. In addition, the comments and statements were also triangulated with each other and with the organization's documentation to achieve as high level of objectivity as possible.

An additional advantage that helped to overcome the limited resources of the study was the knowledge the researcher already had about the case company and its work methods, culture, and employees. These factors increased the effectiveness of the research process, as they, for example, made it easier to know who to contact in any situation, and no extra time had to be spent on learning the basics of the organization. The fact that the researcher and the interviewees knew each other beforehand also made the atmosphere in the interviews rather comfortable, which might have increased the quality of the data collected. An additional factor that greatly helped to cope with the limited resources was the support the supervisors of the research and the people from the case company gave whenever needed.

To cope with the time and resource limitations mentioned above, it was also seen reasonable to choose a single-case study -approach for the research. From early on it was realized that the tight schedule and limited resources would have made it very hard to conduct a thorough, high-quality multiple-case study. In addition, it is likely that it would have been difficult to find another case company that was implementing Agile at the same

time. For these reasons, it was decided that it is better to focus on a single case and study that thoroughly than to do a multiple-case study insufficiently.

When analyzing the data for the study, the method chosen was directed content analysis. There are certain challenges linked to this approach, such as giving signals to interviewees to give certain answers in the interviews, paying attention only to the data that supports the existing theory, and becoming blind to the circumstantial aspects of the phenomenon studied (Hsieh and Shannon 2005). To counter these, several actions were taken. In the interviews, only open-ended questions were asked, as that reduces the possibility of interviewees taking cues, and in the data analysis -phase, the data was examined on a very detailed level to make sure no relevant factors were overlooked. Attention was paid to any possible contextual factors, too.

Despite all these measures, conducting a single-case study also sets its own shortcomings for the study, especially regarding its generalizability and scope; when studying only a single case, it is difficult to distinguish all the contextual factors from more general ones. Also, the close relationship the researcher had with the case company was a potential source of subjectivity and misinterpretation. However, during the research process various actions were taken to control these deficiencies and to improve the rigor and credibility of the study. During the research, an effort was always made to follow systematic procedures and to treat all the data and data sources equally. To ensure construct validity, which refers to that the measures used in the study need to reflect the concepts being studied (Yin 2014), it was necessary to use multiple sources of evidence. That was done by conducting the interviews and collecting the documentation of the organization and then triangulating those. It has also been attempted to present the chain of evidence to the reader as clearly and logically as possible, starting from the theory basis and also covering the research methods, results, and the discussion. To ensure the reliability of the study, the procedures of the study were carefully documented, and they are described in section 3. As Yin (2014) states, the reliability of a study is considered good if another researcher would be able to, after conducting the same case study all over again, arrive at the same findings. The aim with documenting and sharing the steps of the process is to make sure that is true for this study, too.

However, despite all of these actions, it still needs to be acknowledged that the results of this study cannot be generalized to cover all the Agile implementations, not even among

large game development organizations. Instead, the results should be seen as signals of issues that can be significant in other Agile implementations, too, and that should be paid attention to. More research is needed before they can be reliably generalized.

6.5 Future research

There are several topics for future research that arise from this study. One particularly interesting would be to study the long-term benefits and other consequences of the Agile implementation in the case organization. As mentioned in the previous section, those could not be studied here, as this study was conducted before the implementation process was finished. If it would be possible to study those in the future, that would also allow evaluating the methods used and analyzing their feasibility in Agile implementation. A study like that would also provide interesting information of sustaining the results of implementing Agile, a topic that has not received much attention in the current research. Alternative approach for future research would be to intentionally apply the change management methods and practices when implementing Agile and see how that affects the success of the implementation.

Another interesting approach for future research would be to widen the perspective and study the Agile implementation processes and results in other game development companies. If there were several case companies, it would be possible to utilize more quantitative and comparative research methods. This would make it easier to identify which factors of the implementation process are more contextual and which are more general, which would in turn make the results more reliable and generalizable. In a study like that, it could also be examined if other game development companies have difficulties with implementing Product Ownership and Product Backlog -structure. Those were the biggest challenges of the case company of this study and it is assumed that those would be difficult for others, too.

There are also other exciting ideas for future research that arose during the research process. One of them are the human skills and their role in implementing change projects. It appears that there is no unified understanding or classification of those done yet. If such a scheme existed, researching human skills in different contexts would become easier and more cohesive and the results could be more comparable, even between individual studies. One more interesting idea would be to study Key Performance Indicators or KPIs

and if they could be useful for Agile implementation. They were not used in this case, and it also appears that they have not been studied in the context of implementing Agile. However, in some industries and organizations they are very popular (Lindberg et al. 2015; Krause and Dayanand 2019). This suggests that it could be valuable to experiment with them in Agile Implementation process and analyze the results.

REFERENCES

Alqudah, M. and Razali, R., 2016. A Review of Scaling Agile Methods in Large Software Development. *International Journal on Advanced Science, Engineering and Information Technology*, 6 (6), 828–837.

Artto, K., Martinsuo, M. and Kujala, J., 2011. *Project Business*. Helsinki, Finland.

Beck, K., Beedle, M., van Bennekum, A., Cockburn, A., Cunningham, W., Fowler, M., Grenning, J., Highsmith, J., Hunt, A., Jeffries, R., Kern, J., Marick, B., Martin, R. C., Mellor, S., Schwaber, K., Sutherland, J. and Thomas, D., 2001. *Manifesto for Agile Software Development* [online]. <https://agilemanifesto.org/>. Available from: <https://agilemanifesto.org/> [Accessed 29 Nov 2021].

Berdanier, C. G. P., 2021. A hard stop to the term “soft skills.” *Journal of Engineering Education*, 1–5.

Biernacki, P. and Waldorf, D., 1981. Snowball Sampling: Problems and Techniques of Chain Referral Sampling. *Sociological Methods & Research*, 10 (2), 141–163.

Conboy, K. and Carroll, N., 2019. Implementing Large-Scale Agile Frameworks: Challenges and Recommendations. *IEEE Software*, 36 (2), 44–50.

Cook, H., Slocum, J. Jr. and Stringer, E., 2020. The Relationship Between Power Distance, Trust, and Performance in Video Game Development Teams. *Press Start*, 6 (1), 97–117.

DeGraff, J. and DeGraff, S., 2017. *The Innovation Code: The Creative Power of Constructive Conflict*. 1st ed. Oakland: Berrett-Koehler Publishers.

Denning, S., 2015. Updating the Agile Manifesto. *Strategy & Leadership*, 43 (5).

Diebold, P., Ostberg, J.-P., Wagner, S. and Zendler, U., 2015. What Do Practitioners Vary in Using Scrum? *In: Agile Processes in Software Engineering and Extreme Programming*. Springer International Publishing Switzerland, 40–51.

Diegmann, P. and Rosenkranz, C., 2021. Investigating the “Socio” in Socio-Technical Development: The Case for Psychological Safety in Agile Information Systems Development. *Project Management Journal*, 52 (1), 11–30.

digital.ai, 2021. *The 15th State of Agile Report* [online]. Available from: <https://digital.ai/resource-center/analyst-reports/state-of-agile-report> [Accessed 16 Dec 2021].

Doug, R., 2015. *Leading Agile Teams*. 1st ed. Pennsylvania: Project Management Institute.

Edholm, H., Lidstrom, M., Steghofer, J.-P. and Burden, H., 2017. Crunch Time: The Reasons and Effects of Unpaid Overtime in the Games Industry. *In: 2017 IEEE/ACM 39th International Conference on Software Engineering: Software Engineering in Practice Track (ICSE-SEIP)*. IEEE, 43–52.

Eloranta, V.-P., Koskimies, K. and Mikkonen, T., 2016. Exploring ScrumBut—An empirical study of Scrum anti-patterns. *Information and Software Technology*, 74.

Engström, H., Berg Marklund, B., Backlund, P. and Toftedahl, M., 2018. Game development from a software and creative product perspective: A quantitative literature review approach. *Entertainment Computing*, 27, 10–22.

Game Developer, 2009. *Survey: Outsourcing In Game Industry Still On Increase* [online]. Game Developer. Available from: <https://www.gamedeveloper.com/pc/survey-outsourcing-in-game-industry-still-on-increase> [Accessed 9 Dec 2021].

Graft, K., 2021. *Dear Investors: CD Projekt shifts to agile dev, focuses on future* [online]. Game Developer. Available from: <https://www.gamedeveloper.com/business/dear-investors-cd-projekt-shifts-to-agile-dev-focuses-on-future> [Accessed 11 Dec 2021].

Hamilton, J., 2019. *How Combining Soft Skills and Technical Expertise will Advance Your Video Game Development Career* [online]. Game Developer. Available from: <https://www.gamedeveloper.com/audio/how-combining-soft-skills-and-technical-expertise-will-advance-your-video-game-development-career> [Accessed 23 Dec 2021].

Hazzan, O. and Dubinsky, Y., 2014. *Agile Anywhere: Essays on Agile Projects and Beyond*. 1st ed. Springer.

Hile, B., 2020. *10 Closed Game Studios That Produced Amazing Games* [online]. Gamerant. Available from: <https://gamerant.com/closed-game-studios-best-games/> [Accessed 7 Dec 2021].

Hiltunen, K., Latva, S., Kaleva, J.-P. and Tyynelä, E., 2020. *The Game Industry of Finland - Report 2020*. Helsinki.

Hohl, P., Klünder, J., van Bennekum, A., Lockard, R., Gifford, J., Münch, J., Stupperich, M. and Schneider, K., 2018. Back to the future: origins and directions of the “Agile Manifesto” – views of the originators. *Journal of Software Engineering Research and Development*, 6 (1), 1–27.

Hron, M. and Obwegeser, N., 2018. Scrum in practice: an overview of Scrum adaptations. *In: Proceedings of the 51st Hawaii International Conference on System Sciences*. 5445–5454.

Hsieh, H.-F. and Shannon, S. E., 2005. Three Approaches to Qualitative Content Analysis. *Qualitative Health Research*, 15 (9), 1277–1288.

James, M., 2010. Six Pages About Scrum. *CollabNet* [online]. Available from: <https://www.agilestrategicsolutions.com/wp-content/uploads/2014/09/Six-Pages-About-Scrum.pdf> [Accessed 14 Dec 2021].

Jick, T. D., 1979. Mixing Qualitative and Quantitative Methods: Triangulation in Action. *Administrative Science Quarterly*, 24 (4), 602–611.

John Hayes, 2018. *The Theory and Practice of Change Management*. 5th ed. Bloomsbury Publishing Plc.

Kalenda, M., Hyna, P. and Rossi, B., 2018. Scaling agile in large organizations: Practices, challenges, and success factors. *Journal of Software: Evolution and Process*, 30 (10).

- Kanode, C. M. and Haddad, H. M., 2009. Software Engineering Challenges in Game Development. *In: 2009 Sixth International Conference on Information Technology: New Generations*. IEEE, 260–265.
- Kasurinen, J., Palacin-Silva, M. and Vanhala, E., 2017. What concerns game developers? A study on game development processes, sustainability and metrics. *In: IEEE/ACM 8th Workshop on Emerging Trends in Software Metrics (WETSoM)*. IEEE, 15–21.
- Katz, R. L., 1974. Skills of an Effective Administrator. *Harvard Business Review*, 52 (5).
- Keith, C., 2010. *Agile game development with Scrum*. Addison-Wesley.
- Ketokivi, M. and Choi, T., 2014. Renaissance of case research as a scientific method. *Journal of Operations Management*, 32 (5), 232–240.
- Kiv, S., Heng, S., Kolp, M. and Wautelet, Y., 2018. Agile Manifesto and Practices Selection for Tailoring Software Development: A Systematic Literature Review. *In: Product-Focused Software Process Improvement*. Wolfsburg: Profes, 12–30.
- Kotter, J. P., 1995. Leading Change: Why Transformation Efforts Fail. *Harvard Business Review*, 59–67.
- Koutonen, J. and Leppänen, M., 2013. How Are Agile Methods and Practices Deployed in Video Game Development? A Survey into Finnish Game Studios. *In: Baumeister, H. and Weber, B., eds. Agile Processes in Software Engineering and Extreme Programming*. Vienna: XP 2013, 135–149.
- Koutsikouri, D., Madsen, S. and Lindström, N. B., 2020. Agile Transformation: How Employees Experience and Cope with Transformative Change. *In: Paasivaara M. and Kruchten P., eds. Agile Processes in Software Engineering and Extreme Programming – Workshops. XP 2020. Lecture Notes in Business Information Processing*. Springer, Cham, 155–163.
- Krause, H.-U. and Dayanand, A., 2019. *Key Performance Indicators for Sustainable Management: A Compendium Based on the “Balanced Scorecard Approach.”* Munchen: De Gruyter Oldenbourg.

Laanti, M. and Kettunen, P., 2019. SAFe Adoptions in Finland: A Survey Research. *In: Agile Processes in Software Engineering and Extreme Programming – Workshops*. 81–87.

Larman, C. and Vodde, B., 2017. *Large-Scale Scrum - More with LeSS*. Indiana: Pearson Education, Inc.

Lindberg, C.-F., Tan, S., Yan, J. and Starfelt, F., 2015. Key Performance Indicators Improve Industrial Performance. *Energy Procedia*, 75, 1785–1790.

Matteson, M. L., Anderson, L. and Boyden, C., 2016. “Soft Skills”: A Phrase in Search of Meaning. *portal: Libraries and the Academy*, 16 (1), 71–88.

McGill, M., 2008. Critical skills for game developers: An Analysis of Skills Sought by Industry. *In: Proceedings of the 2008 Conference on Future Play Research, Play, Share - Future Play '08*. New York: ACM Press, 89–96.

McHugh, O., Conboy, K. and Lang, M., 2011. Using Agile Practices to Build Trust in an Agile Team: A Case Study. *In: Information Systems Development*. New York, NY: Springer New York, 503–516.

McHugh, O., Conboy, K. and Lang, M., 2012. Agile Practices: The Impact on Trust in Software Project Teams. *IEEE Software*, 29 (3), 71–76.

McKenzie, T., Morales-Trujillo, M., Lukosch, S. and Hoermann, S., 2021. Is Agile Not Agile Enough? A Study on How Agile is Applied and Misapplied in the Video Game Development Industry. *In: 2021 IEEE/ACM Joint 15th International Conference on Software and System Processes (ICSSP) and 16th ACM/IEEE International Conference on Global Software Engineering (ICGSE)*. IEEE, 94–105.

Measey, P. and Radtac, 2015. *Agile Foundations : Principles, Practices and Frameworks*. 1st ed. London: BCS Learning & Development Limited.

Milton, A., Tångfelt, G., Isaksson, L. and Johanna Nylander, 2021. *Game Developer Index 2021*. Stockholm.

Moe, N. B., Šmite, D., Paasivaara, M. and Lassenius, C., 2021. Finding the sweet spot for organizational control and team autonomy in large-scale agile software development. *Empirical Software Engineering*, 26 (5), 101.

Noll, J., Razzak, M. A., Bass, J. M. and Beecham, S., 2017. A study of the Scrum Master's role. *In: . Manchester: University of Salford, Manchester.*

O'Hagan, A. O. and O'Connor, R. v., 2015. Towards an Understanding of Game Software Development Processes: A Case Study. *In: . Springer International Publishing Switzerland, 3–16.*

Osborne O'Hagan, A., Coleman, G. and O'Connor, R. v., 2014. Software Development Processes for Games: A Systematic Literature Review. *In: 21th European Conference on Systems, Software and Services Process Improvement (EuroSPI 2014). CCIS, 182–193.*

Paasivaara, M., 2017. Adopting SAFe to Scale Agile in a Globally Distributed Organization. *In: 2017 IEEE 12th International Conference on Global Software Engineering (ICGSE). IEEE, 36–40.*

Parente, D. H., Stephan, J. D. and Brown, R. C., 2012. Facilitating the acquisition of strategic skills. *Management Research Review*, 35 (11), 1004–1028.

Politowski, C., Fontoura, L., Petrillo, F. and Gueheneuc, Y.-G., 2016. Are the Old Days Gone? A Survey on Actual Software Engineering Processes in Video Game Industry. *In: GAS '16: Proceedings of the 5th International Workshop on Games and Software Engineering. Austin: GAS '16: Proceedings of the 5th International Workshop on Games and Software Engineering, 22–28.*

Politowski, C., Petrillo, F., Ullmann, G. C. and Guéhéneuc, Y.-G., 2021. Game industry problems: An extensive analysis of the gray literature. *Information and Software Technology*, 134.

Putta, A., Uludağ, Ö., Hong, S.-L., Paasivaara, M. and Lassenius, C., 2021. Why Do Organizations Adopt Agile Scaling Frameworks? *In: Proceedings of the 15th ACM / IEEE International Symposium on Empirical Software Engineering and Measurement (ESEM). New York, NY, USA: ACM, 1–12.*

Rigby, D. K., Sutherland, J. and Noble, A., 2018. Agile at Scale - How to go from a few teams to hundreds. *Harvard Business Review*.

Rigby, D., Sutherland, J. and Takeuchi, H., 2016. Embracing Agile. *Harvard Business Review*.

Ruonala, H.-R., 2016. Agile Game Development: A Systematic Literature Review. Pro gradu. [online]. University of Helsinki, Helsinki. Available from: <http://urn.fi/URN:NBN:fi-fe2017112251271> [Accessed 6 Dec 2021].

Schwaber, K. and Sutherland, J., 2020. *The Scrum Guide - The Definitive Guide to Scrum: The Rules of the Game*.

Serrador, P. and Pinto, J. K., 2015. Does Agile work? - A quantitative analysis of agile project success. *International Journal of Project Management*, 33 (5), 1040–1051.

Shin, K., 2009. *WTF Do Producers Do All Day?* [online]. Game Developer. Available from: <https://www.gamedeveloper.com/production/wtf-do-producers-do-all-day-> [Accessed 23 Dec 2021].

Sommer, A. F., 2019. Agile Transformation at LEGO Group. *Research-Technology Management*, 62 (5), 20–29.

Stevenson, D. H. and Starkweather, J. A., 2010. PM critical competency index: IT execs prefer soft skills. *International Journal of Project Management*, 28 (7), 663–671.

Theobald, S. and Schmitt, A., 2020. Dependencies of Agile Teams – An Analysis of the Scaled Agile Framework. In: *Agile Processes in Software Engineering and Extreme Programming – Workshops*. Springer, Cham, 219–226.

Thorgren, S. and Caiman, E., 2019. The Role of Psychological Safety in Implementing Agile Methods across Cultures. *Research-Technology Management*, 62 (2), 31–39.

Thornton, S., 2020. *Soft Skills for Game Developers* [online]. Game Developer. Available from: <https://www.gamedeveloper.com/disciplines/soft-skills-for-game-developers> [Accessed 23 Dec 2021].

Uludağ, Ö., Putta, A., Paasivaara, M. and Matthes, F., 2021. Evolution of the Agile Scaling Frameworks. In: *Agile Processes in Software Engineering and Extreme Programming. XP 2021. Lecture Notes in Business Information Processing*. Springer, Cham, 123–139.

Vodde, B., Larman, C., Yin, T., Atlas, A., Quah, B. and Dittlau, F., 2014. *Large-Scale Scrum* [online]. The LeSS Company B.V. Available from: <https://less.works/zh-CN/less/homepage> [Accessed 15 Dec 2021].

Wawro, A., 2017. *A dev's guide to ensuring studio conflict is healthy and productive* [online]. Game Developer. Available from: <https://www.gamedeveloper.com/production/a-dev-s-guide-to-ensuring-studio-conflict-is-healthy-and-productive> [Accessed 23 Dec 2021].

Weststar, J., Kumar, S., Coppins, T., Kwan, E. and Inceefe, E., 2021. *Developer Satisfaction Survey* [online]. Available from: https://igda-website.s3.us-east-2.amazonaws.com/wp-content/uploads/2021/10/18113901/IGDA-DSS-2021_SummaryReport_2021.pdf [Accessed 11 Dec 2021].

Wijman, T., 2021. *Global Games Market to Generate \$175.8 Billion in 2021; Despite a Slight Decline, the Market Is on Track to Surpass \$200 Billion in 2023* [online]. newzoo. Available from: <https://newzoo.com/insights/articles/global-games-market-to-generate-175-8-billion-in-2021-despite-a-slight-decline-the-market-is-on-track-to-surpass-200-billion-in-2023/> [Accessed 7 Dec 2021].

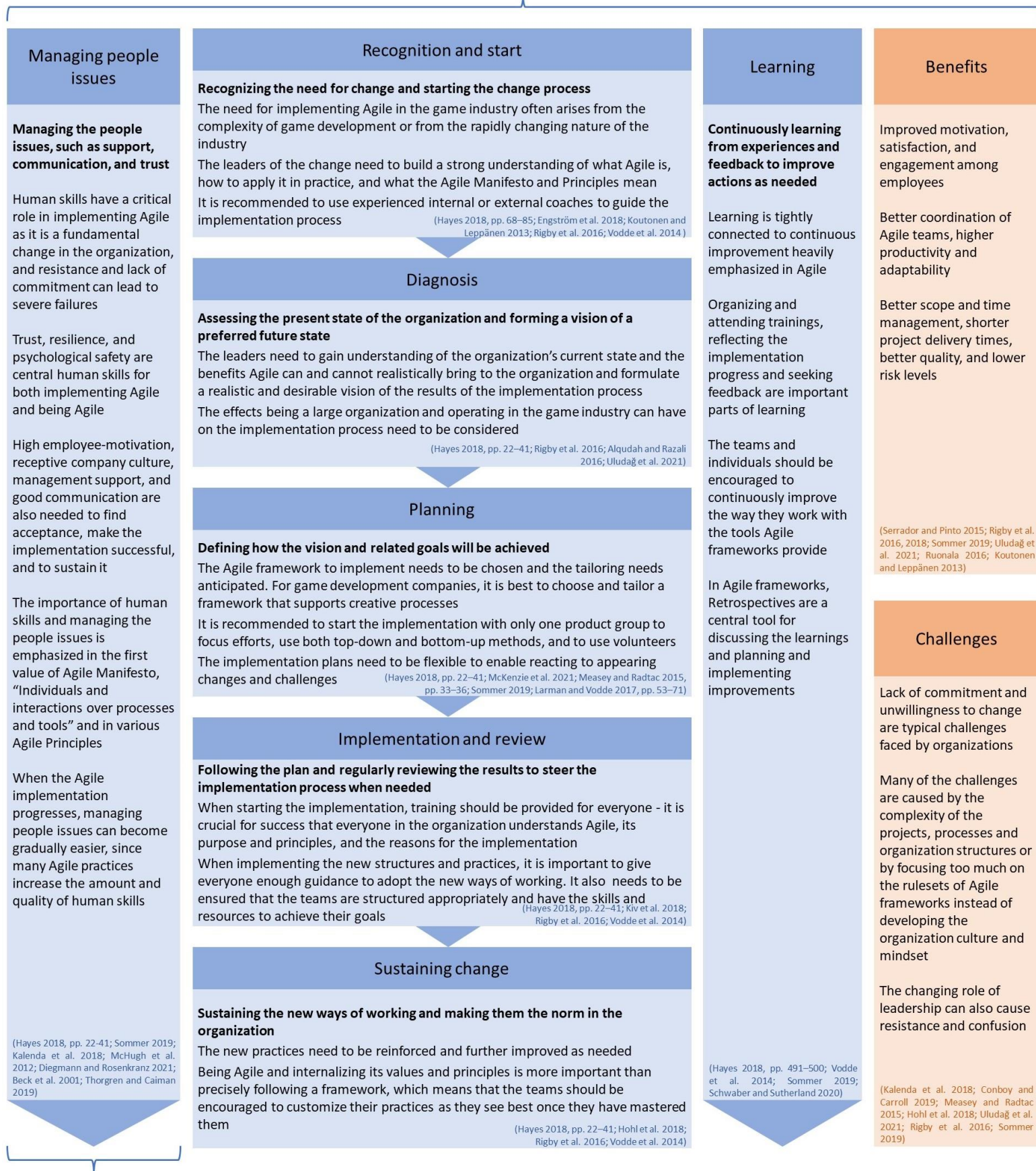
Wilson, C., 2009. Trust: The Critical Factor in Leadership. *The Public manager*, 38 (1).

Wilson, T., 2021. *Video Games and Disappointment: Are Our Expectations Too High?* [online]. Gamecrater. Available from: <https://www.thegamecrater.com/video-games-and-disappointment-are-our-expectations-too-high/> [Accessed 25 Jan 2022].

Yin, R. K., 2014. *Case Study Research - Design and Methods*. 5th ed. Los Angeles: SAGE Publications, Inc.

Appendix 1. Agile implementation process in large game development organizations, following the structure of the change process defined by John Hayes (2018)

RQ1: How do large game development organizations implement Agile working practices, and what are the related benefits and challenges?



RQ2: What kinds of roles do human skills play when implementing Agile in large game development organizations?