Ole Kristian Vingdal

Understanding Team Effectiveness and Agile Practices

A Qualitative Survey Exploring Factors of Effectiveness in Agile Teams

Master's thesis

NTNU

Norwegian University of
Science and Technology

Ole Kristian Vingdal

Understanding Team Effectiveness and Agile Practices

A Qualitative Survey Exploring Factors of Effectiveness in Agile Teams

Master's thesis in Computer Science
Supervisor: Torgeir Dingsøyr
December 2021

Norwegian University of Science and Technology

NTNU
Norwegian University of
Science and Technology

Abstract

Agile practices have already been a staple for teams to develop software effectively for many years. Understanding what effective teams are and how agile practices affect them can help agile teams take the next step and improve their practices. To this end, the thesis explores agile team effectiveness as seen by practitioners to increase the understanding of agile practices and factors of agile team effectiveness. The researcher conducted a qualitative survey interviewing 10 participants experienced with agile teams. This led to 14 factors found to be important for team effectiveness, related to task understanding, team togetherness, and activity effectiveness. Findings on agile practices include the multiple positive effects daily standups can have, as well as the importance of involving domain experts for improved task understanding. The significance of psychological safety for agile teams is also shown, but few agile practices were found to improve it. The researcher concludes that there are relationships between factors for team effectiveness and the effects of agile practices, but that adding practices that focus primarily on improving psychological safety and making the team more goal-oriented are likely to make teams even more effective.

Preface

It has been a challenge for me to write this master's thesis, but it is a challenge that has given me a lot of new knowledge and understanding about both team effectiveness and agile development. This master's thesis would not have been possible if my informants had not lined up. I want to express my gratitude to all the informants who have taken the time and lined up to help me with excellent and instructive answers. I would also like to thank my supervisor Torgeir, who has been a stable and good help throughout the process. Thank you very much for all the help and guidance. Further, I would like to thank my colleagues in Loopfront, with whom I have discussed teamwork. Thank you for letting me test my ideas with you too. Finally, I would also like to thank my friends who have supported me throughout the process. I also want to thank my family for everything they have helped me with throughout my studies and during this master's period. Finally, thanks to my girlfriend, Tiril, for all the help and support during my studies. Without you all, I would not be where I am now. Thank you very much.

Contents

List of Figures

List of Tables

1 Introduction

In the South Seas there is a cargo cult of people. During the war they saw airplanes land with lots of good materials, and they want the same thing to happen now. So they've arranged to imitate things like runways, to put fires along the sides of the runways, to make a wooden hut for a man to sit in, with two wooden pieces on his head like headphones and bars of bamboo sticking out like antennas—he's the controller—and they wait for the airplanes to land. They're doing everything right. The form is perfect. It looks exactly the way it looked before. But it doesn't work. No airplanes land. So I call these things cargo cult science, because they follow all the apparent precepts and forms of scientific investigation, but they're missing something essential, because the planes don't land.

This is a famous quote from Feynman (1974) where he coins the term *Cargo Cult Science*. The story about the cargo cults is a story that is often used in the agile community to highlight the importance of understanding the methods you use in order to make them as effective as possible.

Agile practices have already been widespread in the software industry for many years (Rodríguez et al. 2012). However, simply using agile practices is not agile. Agile is adapting to use the best tool for the task. But to understand what is the best tool for the task, you need to understand what the task is and what the tools you have actually do.

To gain a deeper understanding of how agile teams can be more effective, team theory and empirical knowledge of different process consequences need to be considered in combination. If achieving a higher degree of team effectiveness is a goal of an agile team and agile practices are the course of action, what are the intermediate sub-goals and steps of approach? Can the relationship between the ultimate goal and the immediate action be made more transparent?

In addition to exploring the factors team effectiveness consists of, this thesis will

study agile software development teams and the motivation and (perceived) effects of their practices. With this knowledge, the thesis will attempt to answer the following research questions:

RQ1. What are the key factors for team effectiveness in agile teams?

RQ2. Can factors of team effectiveness be used to explain the use of agile practices?

RQ3. Are there conditions needed for team effectiveness that agile practices do not aid in fulfilling?

The target audience of this thesis are practicioners of agile with introductory knowledge about agile software development. Most terms regarding team effectiveness will be presented in section 2, but if the reader is interested in learning about agile the Agile Manifesto (Agile Manifesto n.d.) is a good place to start.

The thesis will proceed as follows: Firstly, section 2 defines team effectiveness, presents earlier research related to team effectiveness and agile practices, and presents the 5W1H framework for categorizing factors for team effectiveness. Next, section 3 describes and evaluates how the research was conducted, and the results are presented in section 4. Section 5 discusses the results of the study and directions for future research before section 6 concludes the thesis.

2   Background

This section will provide the reader with background regarding the concept of team effectiveness, earlier models to understand team effectiveness, a new framework synthesizing factors from past research, as well as an introduction to effects of agile practices.

2.1   Defining Team Effectiveness

Team performance is the most common way evaluate the capabilities of a team, and is used because it evaluates what is useful to organizations, namely outputs (Mathieu et al. 2008). For instance, measures Forsgren et al. (2018) software delivery performance with four metrics (Figure 1).
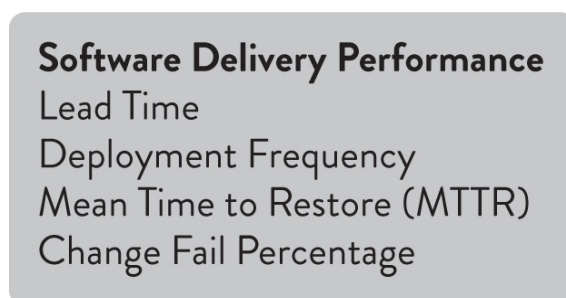
Figure 1: Software Delivery Performance (Forsgren et al. 2018)

In short, *lead time* is time it takes to satisfy a customers request, *deployment frequency* is how often code is being deployed, *mean time to restore* is how fast services are operative again after failure, and *change fail percentage* is how much of deployed code that leads to failure.

Although performance may be easier to assess, Salas et al. (2005) make the case that team effectiveness is a more suitable way to evaluate the team:

> Team performance accounts for the outcomes of the team's actions regardless of how the team may have accomplished the task. Conversely, team effectiveness takes a more holistic perspective in

3

considering not only whether the team performed (e.g., completed the team task) but also how the team interacted (i.e., team processes, teamwork) to achieve the team outcome. This is an important differentiation because many factors external to the team may contribute to the success (or failure) of the team, and therefore in some cases team performance measures may be deficient in understanding the team. (Salas et al. 2005, p. 557)

However, team effectivity is still concerned with success. The first principle of the Agile Manifesto concerns delivering *valuable* software, and the term value has become a key component of successfulness (Agile Manifesto n.d.; Alahyari et al. 2017). Alahyari et al. show that value is not only related to how quickly software can be developed but also to the quality of the software.

Analogously to the point of Salas et al. there needs to be a differentiation between value creation that the team can and cannot control. To illustrate, simply because one team gets a task that has the potential to deliver more value than another, it does not make that team more effective.

Therefore, this thesis defines team effectiveness as *the rate of value creation over time compared to the value potential in their tasks*.

Finally, *team efficiency* is another way of assessing the capabilities of a team that will be used in this thesis. While the focus of team effectiveness is mainly concerned with success, efficiency is used to highlight accomplishing with low resource use, often time.

2.2   Studies on Team Effectiveness

This section gives a summary of studies related to team effectiveness. The collection presented builds on the studies found in the literature review of the specialization project done in the fall of 2020. The research stem from multiple fields of science and should provide the reader with sufficient background information.

2.2.1 Models Related to Team Effectiveness

The goal of this section is to explain the main models used to build the *5W1H framework*, the framework presented in this thesis. The research done for earlier models will be briefly presented, as well as the concepts of the models, findings, and relevance for the 5W1H framework. The following sections are based on the work done in the literature review. As the reader is not expected to have read the literature review, relevant information about it will be presented here.

A commonly used model for teamwork effectiveness is the "Big Five" (Salas et al. 2005). In the psychology field, a comprehensive literature review of models and studies of teamwork was done, proposing five core components and three coordinating mechanisms that summarize the knowledge. The "Big Five" consists of team leadership, mutual performance monitoring, backup behavior, adaptability, and team orientation. These will be described when contributing to the framework proposed in the present article as they relate to various conditions. The three required coordinating mechanisms that support the "Big Five" are shared mental models, closed-looped communication, and mutual trust. Shared mental models were defined as "*An organizing knowledge structure of the relationships among the task the team is engaged in and how the team members will interact*". Further, mutual trust as "*The shared belief that team members will perform their roles and protect the interests of their teammates*". Finally, closed-loop communication as "*The exchange of information between a sender and a receiver irrespective of the medium*". Salas et al. argued that adopting the "Big Five" would increase the performance of teams, but that these coordinating mechanisms are needed in addition to assuring team success. Furthermore, they argued that these dimensions are relevant in all kinds of teamwork, explaining how they can be exhibited in different types of task work. Finally, they put forward how the components of the framework influence each other and theorized on the importance of each factor during the different stages of projects. A graphical representation of the model is shown in figure 2.
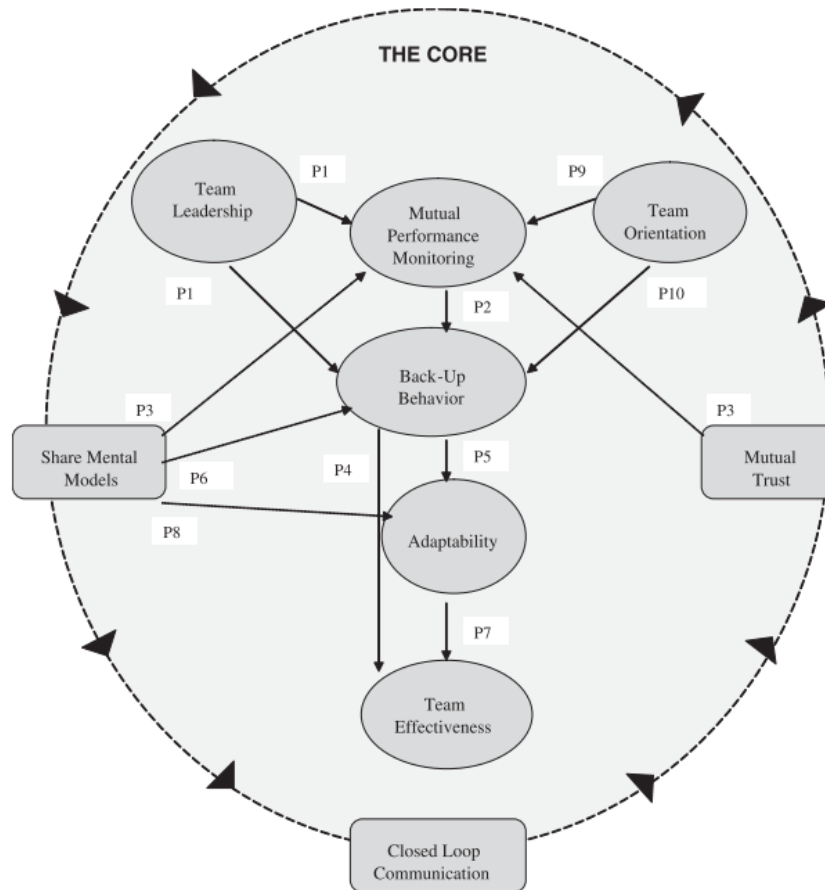
Figure 2:   The Big Five and coordination mechanisms with their high-level relationships as presented by Salas et al. (2005, p. 571).

Another literature review on factors that facilitate interdependent work was done by Okhuysen and Bechky (2009).   In the management field, they reviewed literature on coordination presenting five mechanisms and three integrative conditions for coordination. The five proposed mechanisms that aid in achieving coordination are plans and rules, objects and representations, roles, routines, and proximity. These are derived from the literature review, and are further branched by the different ways they function.   Additionally, they argued that elements enabling coordination goes beyond mechanisms, presenting the three integrating conditions accountability, predictability, and common understanding. In outline, accountability is visible interdependence, responsibilities, and task progress, predictability is the anticipation of interdependent task-related activities, and common understanding is the shared notion of why and how activities are performed. These can be used to better understand what a particular mechanism

accomplishes, and also what it does not. Similar to Salas et al. (2005), Okhuysen and Bechky used the term coordination mechanisms for practices related to teamwork, but the term teamwork was not used. Instead, coordination is the term used throughout the review. Nevertheless, there is correspondence between the definitions of coordination reviewed in the articles and the definition of teamwork applied in this review. Significantly, commonalities are that people work collectively, interdependently, and that a goal, task, or piece of work is achieved.

A model that concerns teamwork quality was also developed in the field of management (Hoegl and Gemuenden 2001). The article specified six teamwork quality facets building a review of relevant literature and exploratory case studies, and measure their significance by 575 interviews related to 145 software development teams. The six facets communication, coordination, balance of member contributions, mutual support, effort, and cohesion capture task-related and social interaction that occur in a team, focusing on the quality of interactions. The study showed that teamwork quality, based on the six facets, notably linked with team performance as evaluated by the members and leaders of the team, as well as external managers. A significant association between teamwork quality and the team members' satisfaction and learning was also found. The review and the study did not, however, focus on the quality of task strategy and task activity, nor on leadership processes. While this perhaps prevents the model from comprehensively capturing the full concept of teamwork effectiveness, it may make it more versatile. For instance, it may be more directly applicable to agile teams, not having to account for typical leadership processes that would need to be handled differently in teams that are self-organizing.

Whereas the original study evaluated the significance of the six facets in traditional software development teams, the study has also been replicated for agile teams (Lindsjørn et al. 2016). Although no significant correlation was found between teamwork quality and team performance as rated by product owners, correlations were found when team performance was rated by the team members and team leaders. Furthermore, teamwork quality, as captured by the six facets, was associated with product quality and almost fully correlated with team

members' satisfaction and learning.

Continuing in the field of information systems, Strode (2016) developed a taxonomy of dependencies in agile software development projects. To simplify, dependencies can be thought of as the inverse of coordination mechanisms, and can hinder interdependent work if unmanaged. They are useful for understanding factors of team effectiveness since they often are the reason to implement coordination mechanisms in the first place. Dependencies are presented as a fundamental element in coordination and considering it as such a central teamwork concept leads the taxonomy to be quite comprehensive. Building on prior knowledge about dependencies in non-agile teams, three agile software development projects were studied to develop the categorization. Included in the taxonomy are three groups of dependencies: knowledge, process, and resource dependencies. Knowledge dependencies are comprised of ones relating to requirements, task allocation, historical, and expertise, while process dependencies have those of activity and business process. Finally, resource dependencies cover entity and technical dependencies. The taxonomy was subsequently considered alongside the coordination mechanisms of agile, by creating an understanding of dependencies to assist the selection of agile practices.

Data from the three cases used in this study also contributed to research on coordination (Strode et al. 2012). Here, models for coordination strategy, coordination effectiveness, and propositions defining the relationship between them were presented. Especially relevant for the present article is coordination effectiveness, the result of a coordination strategy, and how it was modelled. Coordination effectiveness was proposed to have an explicit and an implicit component. Explicit coordination consists of having the "*right thing*", at the "*right time*", at the "*right place*". The implicit components are "*Know why*", "*Know what is going on and when*", "*Know what to do and when*", "*Know who is doing what*", and "*Know who knows what*". These turn out to be closely related to some of the conditions presented in this article and will be expanded on later.

Trust has been considered to be of great importance in teams. For instance, Salas

et al. (2005) and Moe et al. (2010) argued for its inclusion in teamwork models. Additionally, trust has been shown to add confidence and security to relationships, and contribute to open and effective information sharing (Earley 1986; Larzelere and Huston 1980; Yeager 1978; Jarvenpaa et al. 1997). Okhuysen and Bechky stated "*When individuals can trust one another to perform their work in the proper manner, coordination is enhanced*" (Okhuysen and Bechky 2009, p. 481). They suggested that trust could be developed through the familiarity gained by proximity, which is a viable strategy for co-located teams, but more difficult for virtual teams. Hence, a model of trust in global virtual teams was assembled by Jarvenpaa et al. (1997). In this model, the attributes that compose trust in a team member are their perceived ability, benevolence, and integrity. Additionally, the concept of "swift trust" is used seeing as "*Members of such teams do not have the time to develop trust in a gradual and cumulative fashion. Rather, the team members act as if trust is present from the start*" (Jarvenpaa et al. 1997, p. 56). Continuing, they stated "*Whereas trust is typically conceptualized as either an affective or a cognitive construct, swift trust is a form of depersonalized action*" (p. 56). This comes in the form of adopting high-trust strategies to achieve their benefits at the same time as trust is developed. The strategies found were proactive action, output driven communication, optimistic team spirit, dynamic team leadership, task goal clarity, role division, time management, frequent communication, and thorough feedback.

2.2.2   Why Use a New Framework?

While all these models have undeniable qualities and are built on thorough research, this thesis will use a framework assembled by the researcher to try to answer the research questions.

There are a few benefits of using the new framework. Firstly, it is tailored to agile software development teams, addressing factors like self-organizing teams. It addresses team effectiveness while including social factors in more detail. And it also granulates the concept of task understanding, which is found to be an important factor for team effectiveness. For instance, in the three agile software

development projects studied by Strode (2016), 76% of coordination mechanism-dependency pairs found were linked to task-related information being required for the project to progress.

2.3   The 5W1H framework

There are, of course, a number of factors that play a role in making the team as effective as possible. In the study done by Dingsøyr and Lindsjørn (2013) on agile effectiveness, more than 70 factors were discovered. The team effectiveness framework presented in this thesis is not meant to be a complete list of all factors that affect team effectiveness, as such a list would be prone to lacking theoretical support, completeness, or atomicity.

Instead, the framework presented in this thesis attempts to group related factors into distinct categories that can be thought of as sub-goals for understanding team effectiveness. This section will try to briefly explain the framework, as well as its theoretical support.

To synthesize a collection of the meaningful conditions of teamwork effectiveness, a system of mapping needed to be found. An attempt was made to use the already established models and append them with new conditions, but ultimately it was decided to map all the discovered conditions into a new framework. In particular, the taxonomy of Strode 2016 appeared enticing to extend due to its structure. However, its focus solely on coordination was regarded as too narrow for a holistic view of team effectiveness. It also included a dependency deemed outside the scope of this review which made a new model seem more expedient. The exclusion of this dependency will be discussed later in this section.

What inspired the 5W1H framework was the integrating conditions for coordination by Okhuysen and Bechky 2009. Predictability can be thought of as knowledge about what needs to be done when, common understanding as to why and how, and accountability as who is responsible. The last W, the concept of where, emerges from the coordination mechanism of proximity, as well as the need for addressing

virtual teams.

Different models have different foci, some focus on coordination of tasks, some focus on the quality of interrelations, and some focus the collaboration. However, the 5W1H framework attempts to address all of these elements. The model of coordination effectiveness (Strode et al. 2012), which was discovered later in the review process, was found to have some similar ideas to the 5W1H framework. The components of the coordination effectiveness model map effortlessly over to the framework presented in this article. However, important team effectiveness concepts like trust and collaboration between team members are not central. In relation to collaboration, coordination, and communication, Sharp and Robinson stated: *"As yet, a good theoretical model for successful agile development has not been reported, but any such model will need to incorporate the role and influence of the three 'c's described here"* (Sharp and Robinson 2010, p. 80).

Each of the models reviewed have acknowledged communication as largely important, particularly in the beginning of team cycles (Salas et al. 2005). 5W1H takes this into account by the knowledge and the understanding of the team being central parts of all conditions. Similarly, coordination is also present in all conditions. Finally, the collaboration aspect is represented in the How condition.

The 5W1H framework contains multiple important factors for teams, but not all. Salas et al. noted that *"The general consensus, however, is that teams require a complex mixture of variables that include not only organizational support and individual skills but also teamwork"* (Salas et al. 2005, p. 591). Internal individual factors such as skill were considered, for instance in the *"Who"* condition. However, due to the scope of the project, external factors were not prioritized, although some of the models reviewed consider them important for team performance. A concept that was omitted was the business process dependency from the dependency taxonomy for agile software development projects (Strode 2016). This was done because it was outside the control of the team, and therefore, outside the scope of this project. A dependency Strode omitted from her taxonomy were the temporal dependency (Espinosa et al. 2007). This was included

to give more insight into virtual teams, and was omitted by Strode as the studied teams were co-located.

Conditions explain why practices are used and how successful teamwork can be accomplished. Rather than specifying the exact teamwork mechanisms needed for effective teamwork, 5W1H highlights key factors that, if achieved, is theorized to improve the overall effectiveness of the team. The flexibility in the framework lies in the concept that all conditions are able to be fulfilled in a number of ways, some perhaps not being known yet. The 5W1H can not hope to be complete model of how to make a team work, but rather attempt to present the essential elements needed teamwork effectiveness. Table 1 summarizes the framework.

Table 1: The "*5W1H*" framework

Condition	Description
What	The team needs knowledge about what they should work on, both as a team and individually, and understand what is needed to finish the task.
When	The team needs an understanding of the project timeline, progress, and how interdependent tasks are coordinated.
Why	The team needs an overall understanding of the goals and objectives for tasks and projects.
Who	The team needs to have the required expertise, feel responsibility, and to understand team members' strengths.
Where	The team needs a space where they can meet for informal communication, and building togetherness and trust.
How	The team needs to know how tasks are worked on, being able to improve over time, and adjust to input.

2.3.1   The First W - What

The team needs knowledge about what they should work on, both as a team and individually, and understand what is needed to finish the task.

Understanding what constitutes the task at hand is imperative for teamwork. As Okhuysen and Bechky (2009, p. 486) stated, *"Being able to anticipate task related activity allows parties to plan and perform their own work, and is essential for coordinated activity"*. This predictability component of teamwork also closely relates to the concept of shared mental models, as *"Working cooperatively requires that team members coordinate by anticipating and predicting each other's needs through common understandings of the environment and expectations of performance"* (Salas et al. 2005, p. 565).

While not knowing the exact requirements for task does not necessarily mean it will not be completed or completed inefficiently, those risks are present. This is the main idea behind the requirement dependency presented by Strode (2016). Moreover, a defining strategy of high-trust teams is achieving task goal clarity. This means an increased frequency of discussing the goals of the tasks, and no reluctance in trying to remove ambiguity from pieces of work. On the other hand, low-trust teams are more prone to make assumptions of the task goals (Jarvenpaa et al. 1997).

Finally, the coordination effectiveness components of *"Know what is going on and when"*, and *"Know what to do and when"*, also support the concept of having shared mental models of the tasks. *"Know what is going on and when"* is the component regarding the knowledge of the team members of the status of the project, and the anticipation of future tasks. Also straightforwardly, *"Know what to do and when"* regards the individuals being informed about what tasks they should be addressing and how their tasks fit into the team tasks (Strode et al. 2012). These components also intuitively apply to the next condition.

2.3.2   The Second W - When

The team needs an understanding of the project timeline, progress, and how interdependent tasks are coordinated.

Similarly to the "*Know what to do and when*" component previously described, "*Predictability allows people to fit their own tasks into the whole through anticipation of when others will do their work*" (Okhuysen and Bechky 2009, p. 486). The concept to "*Know who is doing what*" (Strode et al. 2012) supports this. Shared mental models also allow teams to coordinate implicitly (Salas et al. 2005), aligning interdependent tasks. This is also the focus of the activity dependency, making sure activities are not prevented by other activities. In software development, interactions of software component also have to be planned, which is the technical dependency (Strode 2016).

A mental model of the tasks at hand and their coordination can be shared in a variety of ways, some using team leader functions. A traditional team leader may be less common in agile teams, but typical team leader functions are considered to be core components of teamwork (Salas et al. 2005). "*Although this article argues that all teams require team leadership to be effective, the manifestation of team leadership may differ across different types of team tasks.*" (Salas et al. 2005, p. 574). Although individual members of the team can not always be expected to coordinate the tasks, performance expectations the team obtains by developing shared mental models can help coordinate tasks swiftly. In addition, high-trust teams "*discussed the assignment schedules, established milestones, monitored the milestones, and kept a close eye on time, reminding other members of impending deadlines*" (Jarvenpaa et al. 1997, p. 55), supporting the importance of time management.

2.3.3   The Third W - Why

The team needs an overall understanding of the goals and objectives for tasks and projects.

The Why condition is a third condition related to the concept of shared mental models. In the coordination effectiveness model, the component of "*Know why*" represents "*each individual working on the project understanding the overall project goal and understanding how a task contributes to that overall goal*" (Strode et al. 2012, p. 1233). Adding to this, Okhuysen and Bechky (2009) argued that understanding the broader context of the coordinated activities, aids in directing the team to common task outcomes. Output driven communication is also something seen in high-trust teams, meaning team members were probable to focus on the results, both in work and in communication with others. Lastly, knowledge about decisions made also has the potential to affect current decision making as well as project process. This is the historical dependency presented by Strode (2016).

Similarly to the first W, *what*, task goal clearity (Jarvenpaa et al. 1997) calls attention to the positive effects of understanding goals at the task level. Related to this, in *Scrum and XP from the Trenches*, Kniberg explains how he tries to find the underlying goals of technically described tasks saying, "The team is normally better suited to figure out *how* to solve something, so the product owner should focus on business goals" (Kniberg 2015, p. 9).

2.3.4   The Fourth W - Who

The team needs to have the required expertise, feel responsibility, and to understand team members' strengths.

For good teamwork quality it is important to utilize skill set of the team members. Individuals must not be constrained from contributing with all relevant knowledge and expertise. This is concept of balance of member contributions (Hoegl and Gemuenden 2001), making both task work and decision-making more effective. This is especially beneficial when team members have expertise in different functional areas. Relating to the concept of backup behavior (Salas et al. 2005) found that flexibility in expertise between team members increase team effectiveness, by assisting or taking over the solving of tasks. This also relate to

the teamwork quality facet of mutual support (Hoegl and Gemuenden 2001).

The notion of variety of contributions being important is also supported in global virtual teams, with cultural diversity being shown to positively influence decision-making (Shachaf 2008).   Having skilled team members is also consequential, as Strode's (2016) expertise dependency highlights the negative effect of lacking technical knowledge.

Further, three different concepts have been used in the models to show the importance of knowing who are working on which tasks. Accountability (Okhuysen and Bechky 2009) emphasises the significance both being responsible for own task and making others responsible for theirs. The task allocation dependency (Strode 2016) supports this by having task owners drive the completion of tasks, and visualising prolonged tasks for the team.

Although agile projects do not always have project spanning roles for team members, Jarvenpaa et al. (1997) showed that in high-trust teams various roles emerged and where co-developed over time. The part roles play in accountability is significant in the model of integrating conditions of coordination (Okhuysen and Bechky 2009), and Salas et al. (2005) argue that mutual trust can be fostered when knowing what team members should be doing. The factor of team members understanding each other is also tightly connected with the next condition.

2.3.5   The Fifth W - Where

The team needs a space where they can meet for informal communication, and building togetherness and trust.

Team members do not work in isolation. They are affected by both their physical and social environment effecting the overall effectiveness of the team.Trust has been shown to be of great importance in teamwork (Salas et al. 2005; Moe et al. 2010; Jarvenpaa et al. 1997). *"Although trust is important in any type of team, trust is pivotal in preventing geographical distance from leading to psychological distance in a global team"* (Jarvenpaa et al. 1997). Limiting psychological distance

is also important in co-located teams, with both Salas et al. (2005) and Hoegl and Gemuenden (2001) arguing for team togetherness through the concepts of team orientation, and cohesion. They agree on the significance it has on motivation, and Salas et al. consider it important for engaging in mutual performance monitoring, accepting feedback, and accepting assistance. Proximity is proposed to lead to stronger relationships through familiarity and to facilitate performance monitoring through visibility (Okhuysen and Bechky 2009). Additionally, Jarvenpaa et al. argue that trust also can be built without proximity: "*(...) trusting action is as much an antecedent of trust as an outcome of it. The relationship between action and trust appears to be highly recursive in a virtual-team context*" (Jarvenpaa et al. 1997, p. 57). Optimistic team spirit and frequent communication are examples of this kind of behaviour. Another drawback of teams not co-located is the temporal dependencies of Espinosa et al. (2007), where working in different time zones can be challenging. However, this can be mediated to some degree by effective time management (Jarvenpaa et al. 1997). Finally, culture was shown to be tied to software delivery performance by Forsgren et al. (2018). This was also connected to psychological safety, with Edmondson (1999) showing how structural and interpersonal characteristics affect learning and performance in teams.

2.3.6   The H - How

The team needs to know how tasks are worked on, being able to improve over time, and adjust to input.

For team effectiveness teams need to know how processes are done and how they can be improved. For individuals in a team, Hoegl and Gemuenden (2001) argues with the concept of effort, that work norms improve teamwork quality by setting standards for work. Developing agreement on work plans, the group members also gain a common understanding about how the project will develop, and "*Highlights conflicts, discrepancies, and other difficulties*" (Okhuysen and Bechky 2009, p. 474).

Team autonomy is important because it empowers decicion making by team members. Team autonomy was shown positively affects response efficiency, which

in turn positively affects software development performance (Lee and Xia 2010). Also having a relationship to software delivery performance is continuous delivery as shown by Forsgren et al. (2018).

The learning as individuals and teams are also important to improve processes. The team members' success, including work satisfaction and learning, was shown to be tightly connected with teamwork quality (Hoegl and Gemuenden 2001; Lindsjørn et al. 2016). But learning and adapting are by themselves imperative for improving teamwork quality, both individually and as a team. For individuals, objects and representations have been found to assist in learning work processes (Okhuysen and Bechky 2009), as well as several types of feedback. Salas et al. (2005) found that both mutual performance monitoring and backup behavior allow for team member feedback that can assist in positive adjustments. High-trust teams also exhibit thorough feedback (Jarvenpaa et al. 1997). On the team level, routines have the potential be improved on every iteration, improving performance over time (Okhuysen and Bechky 2009). This is supported by the adaptability concept of Salas et al., to continuously asses the environment and task to see if the current processes are apt.

This thesis will use the 5W1H framework as a tool to analyze the data collected through the qualitative survey, empirically exploring the views of experienced agile practitioners on effective teams. As previously discussed, the thesis will also explore the effects of agile practices.

2.4   Studies on the Effects of Agile Practices

This final background section will present studies researching agile practices to highlight some of their qualities.

Firstly, Sandstø and Reme-Ness (2021) did litereature reviews on both commonly reported agile practices and their effects. It was found that the most common agile practices were:

- Sprints

- Continous code integration

- Incremental releases

- Value-prioritized requirements

On the effects of agile practices, all of co-located teams, customer involvement, and stand up meetings all had positive effects on knowledge sharing. Retrospectives were found to have multiple positive effects such as both direct and indirect internal communication, providing transparency, and fostering trust.

From the previously mentioned study by Strode (2016), twelve agile practices were found to address more than three dependencies. This remains true if the business process dependency is removed, as the only agile practice found to address this, "iteration or sprint (1 or 2 week)", was found to address three more as well.

These agile practices are:

- Cross-team talk

- Informal face-to-face negotiation with external parties

- Sprints of 1 to 2 weeks

- A wallboard displaying current stories, tasks, and task assignment

- User stories for managing requirements

- A co-located team

- Iteration planning sessions

- Story breakdown sessions

- A product backlog

- A done checklist

- Working software at the end of each sprint

- A single priority team

Przybilla et al. (2018) found that retrospectives have a significant impact on the elaboration of information and the team reflexivity, which is discussions regarding processes, tasks, and goals. They also found positive effects of daily stand-up meetings, but these effects were less pronounced.

Elaboration of information was shown to reduce the negative effects of subgroups and increase performance by reducing conflict and increasing satisfaction. However, team reflexivity was found to have a slight increasing effect on conflict and increasing effect on satisfaction. Figure 3 shows a visual representation of the results.
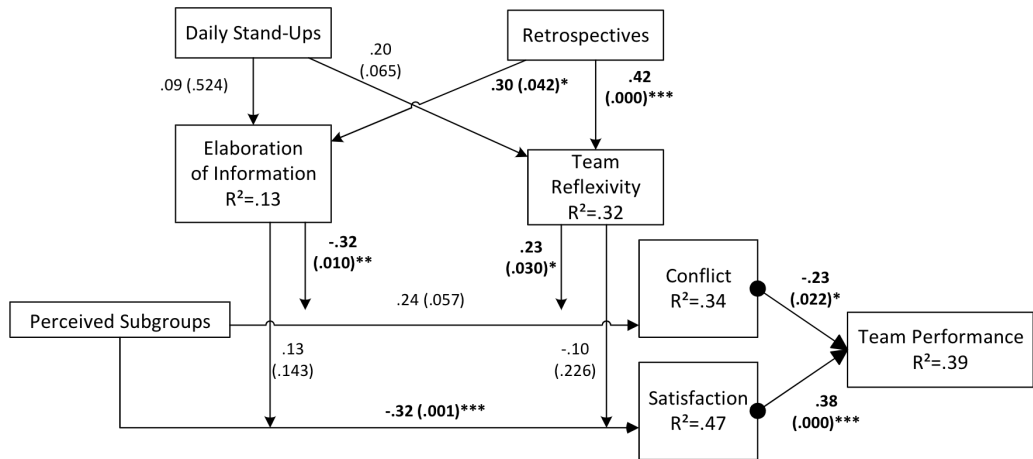
Figure 3: Overview of the results from Przybilla et al. (2018)

Finally, Yu and Petter (2014) used shared mental models theory to understand agile software development practices. They found that daily standups were important for a multitude of reasons. It helped communicating progress, foster togetherness, and motivate team members.

On-site customers where also highlighted as critical for increasing customer satisfaction, because of the mental model of the task that it fostered.

In this thesis, in addition to exploring factor for team effectiveness, it will also be researched how these factors support the use of different agile practices. The

following section will present how the research was conducted in the attempt to answer these questions.

3 Methodology

This section presents how the research was conducted. It goes into detail about the research strategy, presenting the goals and what was done in order to achieve them. Sampling, procedure, and data analysis is explained, and ethical considerations are presented. Finally, the research is evaluated.

3.1 Research Strategy

This thesis builds on research done in the fall of 2020, where an analysis based on grounded theory was done to find key factors for teamwork effectiveness. For this project, the aim was to add to the understanding of effectiveness in agile teams empirically, by researching how experienced agile practitioners view the precursors of team effectiveness. While increasing this insight was a goal in itself, an additional desired result was to advance the understanding of the relationships between team effectiveness and agile practices as an extension of this research. To be more specific, this second objective was to take into account both exploration and explanation of agile practices. Explaining the effects of agile practices on team effectiveness and exploring limitations of agile practices to start a conversation about what can be done to supplement them, to achieve a higher degree of team effectiveness. These goals, in turn, led to the research questions presented in section 1. Finally, the "Possibe Products" categories presented by Oates (2006, p. 21), helped classify two sought-after contributions. Firstly, it can be identified as "An exploration of a topic, area or field", exploring of agile team effectiveness and agile practices. Secondly, as "A new or improved model or perspective", a team effectiveness model based on empirical and theoretical research and a perspective on agile practices as tools with specific effects related to team effectiveness. Using the classification of Robson (2002, cited in Runeson and Höst, 2009), the purpose of this research was to be both exploratory and explanatory.

While Runeson and Höst put forward that explanatory research is related to positivist case studies, it was deemed that the exploratory purpose of this project

where more in line with their description of interpretive case studies. They described it as studies that "attempts to understand phenomena through the participants' interpretation of their context" and assimilated this to Robson's exploratory type of purpose (Runeson and Höst 2009, p. 135). The research in this project also corresponds with the classification of interpretive information systems research presented by Klein and Myers (1999).

As the research intended to explore the variety of perceptions on the topic, a qualitative survey strategy was chosen rather than a case study. To elaborate, a qualitative survey was considered more probable of having a greater diversity of participants, and possibly also thoughts and opinions. The purpose of the research also harmonizes with the definition of qualitative surveys by Jansen (2010), separated from statistical surveys concerned with the frequencies of member characteristics within a population rather than the diversity. Oates (2006) and Runeson and Höst (2009) mainly presented the quantitative survey variant, but mention that there can be other types. The qualitative survey is such an example and is defined as studying the diversity, not distribution, in a population (Jansen 2010).

Finally, with using interviews as the data generation method, which is further described in section 1, an overview of the resulting research process is shown in figure 4.
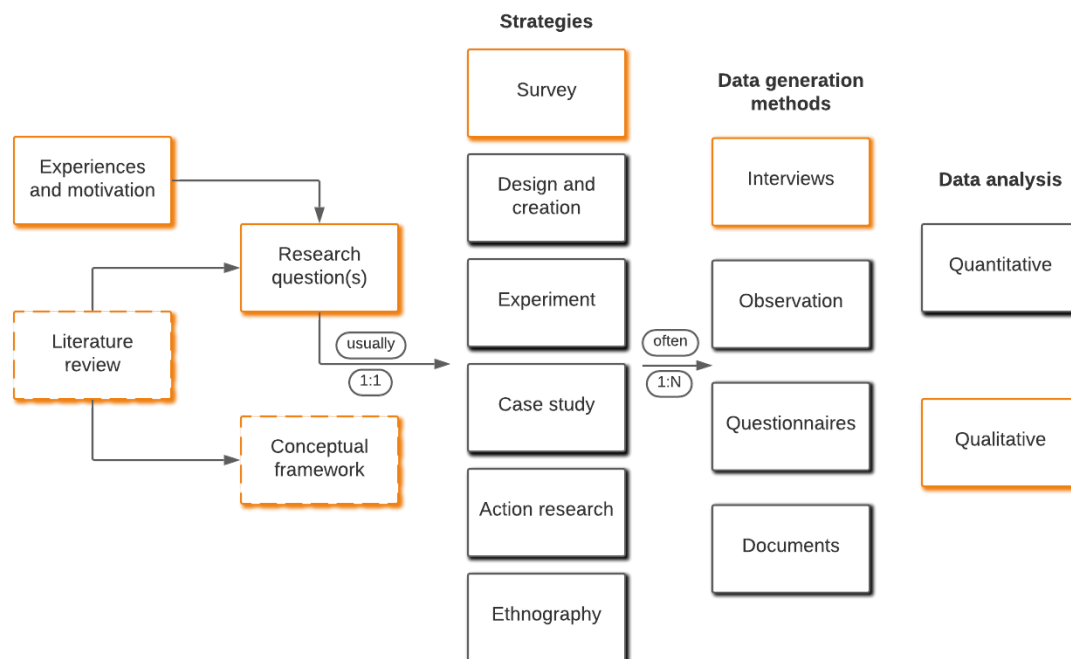
Figure 4: Model of the research process adapted from Oates (2006, p. 33). The process followed for this project is highlighted, with a literature review and a conceptual framework (5W1H) being a part of the specialization project.

3.2   Sampling

To explore the variety of opinions surrounding the antecedents of effective agile development teams, it was a goal to interview a wide range of agile practitioners.

The number of participants was not set before the interviews started. Instead, additional participants were found until most of the opinions surrounding agile team effectiveness in new interviews had already been brought up in previous ones. This was done in an attempt to maximize the diversity at the same time as limiting the total amount data. This sampling method is in line with the suggestion of Jansen (2010) for qualitative interview survey, arguing that saturation in this type of study is an empirical question, while it would be more theoretical in grounded theory.

The participants were selected to cover different backgrounds and team roles within agile software engineering. *Purposive sampling* (hand-picking respondents),

as well as *Snowball sampling* (respondents suggesting new respondents) were also used for sampling. This was suitable for the study because of its exploratory nature (Oates 2006). It was known using snowball sampling could lead to less diversity in respondents. However, since this method was only used to find a small number of the respondents, and new respondents were interviewed until no new topics were found, it was not likely to have a significant effect on the results. Snowball sampling allowed reaching experienced practitioners that would not have been found otherwise. Using the described sampling method, ten participants were ultimately interviewed, with seven separate companies represented. The diversity in backgrounds was something that was strived for. Some of the respondents had relatively few years of professional use of agile development but with significant academic experience, while others had no formal theoretical education but two decades of practical knowledge.

The final list of participants can be found in table 2.

Table 2: Respondents interviewed in the survey

Reference	Current Role	Other Roles	Years of Experience
TL1	Technical Lead	Scrum Master, Software Developer	4
SD1	Software Developer		4
SD2	Software Developer	Scrum Master	10-14
TL2	Technical Lead	Software Developer	20
SD3	Software Developer	Product Owner	16
TL3	Team Lead	Scrum Master	15-20
SD4	Software Developer	Scrum Master	21
SA1	Software Architect	Software Developer	15
TD1	Technical Director	Software Architect	13-14
SA2	Software Architect	Team Lead, Scrum Master	16

3.3   Procedure

Due to the global pandemic, the interviews needed to be held over video conference calls. Therefore, an additional focus was put on earning the trust of the participants and making them feel comfortable, making them open up more easily. This is mentioned by Myers and Newman (2007), who propose seven guidelines for qualitative interviews. Especially related to this was the guideline of minimizing social dissonance, which was appemtped by focusing on giving a good first impression.

The other guidelines were also tried following. For instance, mirroring was used in order to promote participants to discuss the concepts they brought up with their own words. There was also the guideline of having flexibility, using *semi-strucured interviews* that uses incomplete interview scripts. This was done to allow for improvisation, being able to focus on new topics that emerge during the interviews.

All the interviews were done in Norwegian, based an interview guide that can be in its English form in appendix A. The conference calls were recorded, and finally they were transcribed to be ready for analysis.

3.4   Data Analysis

Because the 5W1H framework was created using grounded theory, it was decided a bottom-up analysis would be redundant and that a top-down analysis based on the framework would be more apt. When analyzing, the answers of the respondents generally fit into the six categories of the 5W1H framework. Hence, the results will be presented in themes related to each category. However, a seventh theme is added to showcase the results that are not in accordance with the definition of other groups. The themes are further divided to connect the results with each of the research questions. This is done by looking specifically at how agile practices were described addressing factors presented as important for team effectiveness. Other means of addressing these factors are also added in an attempt to capture conditions that agile practices may not cover on their own. This leads to a structure where

related factors believed to contribute to team effectiveness are presented together, as well as how they can be manifested.

3.5 Ethical Considerations

The main ethical concern for this study was related to the processing of personal data. To ensure that this was handled properly, a request was sent to the NSD, Norwegian Centre for Research Data, to evaluate if the project was compliant with privacy regulations.

The participants were each sent an email with information about participation in the study. This followed the requirements of "Information and consent" (Norwegian Centre for Research Data 2021). There the respondents were informed about their rights regarding the data collected from them.

Consent of participation was given orally at the beginning of the recorded interviews and could be withdrawn at any point for any reason. A minimum of personal information was gathered, and it was only accessible by the researcher. The personal data was also encrypted, and anonymized as soon as it was no longer needed.

3.6 Evaluation

To evaluate the quality of the research, criteria for evaluating interpretive studies where used. Arguments can be made that aspects of the method assimilate positivistic research, viewing the 5W1H framework as a hypothesis and that the research uses empirical data to prove or disprove it. However, the 5W1H framework can also be seen as themes usable as a tool to categorize data to build a theory on key factors of team effectiveness. Moreover, the second and third research questions set additional focus on what is considered characteristically interpretivistic exploration and explanation of relationships (Oates 2006) between team effectiveness and agile practices.

The research was evaluated using the principles for interpretive field research

proposed by Klein and Myers (1999). It was judged that these were extensive enough to not supplement with those described by Oates (2006).

The Fundamental Principle of the Hermeneutic Circle

The fundamental principle of the hermeneutic circle proposes that understanding the meanings of humans can only be done by considering the whole they are a part of.

To achieve this, the results have been purposely presented to give the reader context as far as it was possible, to add more transparency to the grounds of the conclusions. It was also the aim of the format that the reader should be able to draw their own conclusions by understanding the results.

The Principle of Contextualization

The principle of contextualization states that the background of the research setting needs to be reflected so that the audience can understand the investigated situation.

This was attempted by presenting relevant background information in section 2. With all the available studies on team effectiveness and agile, some information is likely to have been overseen, but the discussed theory should be sufficient to give the reader an overview.

The Principle of Interaction Between the Researcher and the Subjects

The principle of interaction between the researcher and the subjects addresses critical reflection in how the interactions between the researcher and the subjects affected the data.

This was mainly a focus before the interviews, where the possible effects the researcher could have on the responses had to be considered. The main result of this focus was presented in section 3.3.

The Principle of Abstraction and Generalization

The principle of abstraction and generalization concerns the need for interpreting the data through the whole it is a part of and the theoretical background.

In the attempt to fulfill this need, the discussion of the data tried to relating the findings of the analysis to the background information. This will be presented in section 5.

The Principle of Dialogical Reasoning

The principle of dialogical reasoning regards the sensitivity to contradictions between the theory the research is built around and the findings.

To allow for data contradicting the theory that the thematic analysis was based on, the thematic analysis also incorporated an additional theme. This theme was to address ideas not fitting into the framework, to promote challenging the original beliefs.

The Principle of Multiple Interpretations

The principle of multiple interpretations addresses the sensitivity to different interpretations of the participants.

A natural response of a study of this exploratory nature is uncovering variances of perceptions. Differences of interpretations were anticipated and also found. Examples of this will be presented in section 5.

The Principle of Suspicion

The principle of suspicion concerns the sensitivity of the research biases from the participants.

While this was not a main concern during the research, there was attention put into not over-emphasizing the impact of single opinions. However, it was never an option to rule out a statement of a participant.

Constructing a Contribution

Meeting evaluation criteria is not sufficient on its own, as noted by both Klein and Myers (1999) and Walsham (2006). Walsham argues that in addition to the principles above, research needs interesting results. To this end, a framework is provided for how to construct a contribution that is compelling. This framework consists of four questions:

1. Who are the audience of the work?

2. To what literature we are aiming to contribute?

3. What does the piece of written work claim to offer that is new to the audience and the literature

4. How should others use the work?

This thesis attempts to address these questions describing the audience in section 1, and the literature aimed to be contributed to in section 2. While this section presents what goals this research tries to accomplish, contributions will be discussed in section 5, together with applications of the research.

32

4 Results

This section highlights the main findings from the qualitative study. It shows the main results of the interviews structured under seven themes. Of these seven, six are based on each of the team effectiveness factor groups of the 5W1H framework, and the final theme addresses results that did not fit into the framework.

4.1 Concerning What

The first theme concerns the team having knowledge of which tasks they are going to do and what is needed to finish those tasks. For this theme, results included the significance of good task specification, introducing domain knowledge into agile teams, and the positives of planning with the waterfall approach.

Importance for team effectiveness

Several respondents presented gaining a common understanding of the tasks as a key to effective teamwork. Communication, in general, was something all of the respondents touched on the importance of, but specifically, communication regarding clarifying tasks was a reoccurring theme. Of the various aspects of task clarification, some expressed as significant included: unambiguous task descriptions, breaking down tasks to more manageable pieces, and having a definition of done to promote completion.

One of the respondents, a technical lead previously involved in several projects with a varying degree of participation from product owners and the business side, was clear about the impact this could make. The respondent emphasized that

> it's important that when you go into sprints, you have to know what to do. Then you have to have a plan, and you should also have, to some extent, a specification of what to do. One of the great things that often makes one succeed or not succeed is involvement from the business side.

> [...] You need to include the ones who will have the solution in the end.
> If you don't have them on board, my opinion is that agile development
> doesn't work. (TL3)

Two main takeaways are evident from this statement and were also representative of several of the other interviews. Firstly, the interviewee believed that teams will benefit from having tasks that are defined well enough to be started on from the beginning of the sprint. Secondly, it is notable that TL3 presents involvement from domain experts and the business side as critical. They explained this as a key not only to have projects succeed more often but also to work agile at all.

Another factor a few of the respondents brought up as impactful for team effectiveness was the decomposing of tasks. When asked about factors of team effectiveness, an experienced software architect also touched on the importance of good task specification, as well as task granularity:

> It has to be broken down. It is an important starting point to divide
> any issue so that you can have an entity delivered that has value. And
> it is important that it has value so that you can put it into production,
> and it can be used for something. And then there is interdisciplinarity. I
> think that it is very important that those who are actually going to use
> the system, some of them are actually also included. (SA2)

Here, the respondent emphasizes the usefulness of splitting tasks into more atomic parts that give value on their own when completed, and deploying them early. Similarly, other interviewees also pointed out the importance of continuous delivery for team effectiveness. They used terms such as "failing early and often", as well as "getting feedback early".

Both of these quotes highlight having a task specification and working closely with domain experts as essential factors for agile team effectiveness. When elaborating on what constitutes a good task specification, some of the repeating motifs were creating a common understanding between the business side and the technical side, and

having small enough tasks with a clear definition of done to encourage continuous delivery. The two following themes will show additional parts presented as vital for a good task specification.

Achieving through agile practices

When asked what the team can do to get feedback from domain experts to increase task understanding, TL3 replied:

> This is not a given, and you can also influence that. It's getting them into the discussions, showing things along the way and involving them early, and not just coming up with finished products. Then they also see the half-products and how it has developed.

Achieving through other means

During the interviews, there were mentions of particularly one practice not considered agile that aid in task specification. On a few occasions, respondents brought up differences in the planning phases of the agile and waterfall approaches. SD2, for instance, explained that they like aspects of the waterfall method because it enforces thorough specification of tasks. SD2 expands on this in section 4.3.

4.2   Concerning When

The second theme is about the team understanding when tasks need to be done and planning for coordination. Results for this theme involve the importance of predictability of tasks and interdependent coordination and the effects of agile practices such as sprint planning and daily standups.

Importance for team effectiveness

In addition to understanding what current tasks consist of, understanding how they fit into the timeline was also identified as valuable. Several of the interviewees mentioned that if the team was unaware of when they should do tasks or how they affect the progress of others, it could break their momentum.

When asked about factors for team effectiveness, an experienced software developer (SD4) was clear about the importance of predictability:

> Predictability in both requirements and platform and what is to be delivered is alpha and omega. I have often felt that we are in the middle of things, and we are trying to use a well-known method that we try to close tasks before we take new ones so that we do not end up with many half-finished or ongoing tasks. [...] It is only a prerequisite for agile to work as intended, i.e., show its good side. With agile development, there must also be predictability. (SD4)

Clearly, the respondent thought of the ability of the team to anticipate the relevant aspects of their tasks as a central part of agile development. They also made a case for limiting the number of tasks worked on at a time. The interviewee also mentioned that predictability in what is to be delivered is consequential. Other respondents noted in a like manner the benefit of the team being able to estimate the size of tasks and what they can deliver at specific points in time.

Reprioritization of tasks during the sprint was also discussed in relation to this. It is described as a danger to predictability but also as an interruption of work processes. More answers related to disturbances will be presented in section 4.7.

Another aspect of having an overview of the timeline was the effect on coordinating interdependent tasks. A software developer explained the following:

> I think it's a little scary when all the developers just care about their little bubble and don't really think anything about thinking outside of it.

> Because it gets a little different when you have a relationship with those
> who sit on the frontend and know that they are sitting and waiting for
> your work, for example, then you'd like to start having a closer dialogue
> with them to make sure it's right the first time they get it. You'll also feel
> more that you should maintain the deadlines because you know people
> are waiting for you. (SD2)

This comment seems to suggest that when you have an understanding of the
subtasks that need to be done in order to complete a larger task, it becomes easier
to coordinate with your team. The respondent also suggests that knowing the
deadlines can motivate you to do so.

Achieving through agile practices

Like SD4 mentioned in their quote about predictability, they saw it as a well-known
method to finish current tasks before starting on new ones. Aside from that, there
are not too many results on how tasks should be prioritized.

However, sprint planning was repeatedly brought up as a solution to increase
timeline understanding. For instance, TL3 explained their use of sprint planning
to increase both timeline understanding and predictability. They split their
planning into two parts, one where the product owner presents the expected tasks
and one where the team plans the tasks in detail.

While the respondents generally thought of sprint planning as a suitable practice
for getting an initial overview of the sprint, they also presented other agile practices
as valuable for coordinating interdependencies. SD2 notes that both sprint reviews
and daily standups help gain insight into what others are doing, allowing for better
coordination. Nevertheless, several of the respondents stressed keeping the daily
standups short.

Achieving through other means

One reason given to keep daily standups short was also given by SD2. They explained how it is important that daily standups do not replace communication the rest of the day:

> I've encountered several companies where the standup goes all the way from 9 to 10 every morning. All the world's problems are solved in the standup, and then nothing happens for the rest of the day. And then it's just another day and then. Then we have often had 24 hours of work where people may have done things a little wrong instead of trying to discuss along the way. I think it's important to have open communication all the way.

This statement highlights that even if the team has daily standup meetings, it is crucial to have more communication. The respondent emphasizes that discussions and updates need to occur at all times, not just during standups.

4.3 Concerning Why

Continuing on the topic of task understanding, the third theme concerns the team having a focus on task and project goals. The results for the theme covers why understanding the reasons behind tasks is significant and how this can be achieved.

Importance for team effectiveness

Similar to the results in the two previous sections, communication and shared mental models were in focus. But for this theme, the importance highlighted lies in allowing team members to deliver more value in each task. A software developer said this about the noteworthiness of understanding the needs of the customers:

> I think many people find it easy to start solving tasks before they have

> really understood the need of the users, to get a sense of progress. At the
> same time, I don't think there's much more time needed to understand
> the problem before you get a big return both in how effectively the team
> works, but also in value you can get out to the end-user. (SD1)

Here, the respondent presents the significance of understanding the needs of the users and the meaning behind the task. They suggest that this benefits both the team and the end-users.

Additionally, participants noted the significance of having an overview of goals on a higher level than tasks. For instance, SD2 was vocal regarding the utility of team members seeing the bigger picture and understanding what is needed to move towards the company's goals. The respondent expressed how this was likely to lead to more value created, due to developers being more likely to complete tasks satisfactorily, with fewer misunderstandings. And on the other hand, if developers are not critical to the task specification, SD2 explained how the developers could make exactly what was asked without it being what was actually needed. Further elaborating that

> it is often because that specification tends to come as part of a slightly
> more generic user story or something similar where several things are
> open to interpretation. And it is your responsibility as a developer not
> only to write the code but also to figure out "of the five different methods
> here, what exactly is the best way to do it?"

Achieving through agile practices

Analogously to the results in section 4.1, involving domain experts was the described way of increasing task understanding. Most of the respondents also mentioned having a product owner close to the development team.

Achieving through other means

Related to team members seeing the bigger picture, SD2 also discussed the waterfall and agile approaches differ in addressing the aspects surrounding the tasks. They explained:

> With a waterfall model and non-agile patterns, you would have other roles there that would have had just that as their job. An architect, for example. His job is to sit and think about all the quirks and things that can happen and define this down so well that any developer in the world could implement it, because it is impossible to make mistakes. Then they haven't followed the specification. As long as they follow the specification, it will be right. Agile development is not that way.

This quote indicates that the interviewee believes that agile development is not naturally organized in handling the goals and implications of tasks. They describe how this is not too much of a case in the waterfall approach since the specification is better defined when it gets to the developer, but with an agile approach, the developer has more responsibility.

4.4   Concerning Who

The fourth theme is about the expertise and contribution of team members. Included in the results is the importance of these factors and how they can be influenced.

Importance for team effectiveness

Multiple of respondents, in particular ones with a leader role, were concerned about having the required expertise on the team. One of them presented the following roles that the team should be able to cover:

> You must at least have some developers, you must have someone who is

> responsible for testing, and you must have someone who is responsible
> for business. It's kind of a minimum requirement. If you don't have it,
> I think it's hard to get to the finish line. (TL3)

Other interviewees were not too concerned with roles as long as the expertise was
available to the team and working together well. More results related to the
availability of expertise are presented in section 4.7. On collaboration, a
respondent touches on team dynamics:

> And then there's team collaboration, which is more important than
> individuals. A team that works well is a team that actually manages to
> work together. [...] I have experienced teams that have become more
> effective by taking out the best developer of the team. And it's kind of
> interesting, but it just shows the dynamics, right? That it was a
> slightly dysfunctional team where you had a super developer who in
> one way destroys the productivity of everyone else, because he was so
> good and thus involved in everything and didn't let the others do their
> job. (TD1)

This observation makes the case that a balance of member contributions is even
more significant than individuals having a large amount of expertise.

Achieving through agile practices

On increasing the team expertise, some of the participants highlighted the benefits
of pair programming:

> Pair programming, I think, increases effectiveness a lot in teams. Sitting
> together to solve a problem, two and two, or three or whatever it should
> be. But the pair programming part we have used a lot. It has been an
> effective way to share knowledge and expertise. It's one of those things
> I think has been really useful. (SA1)

However, the opinions about pair programming were more mixed. Other respondents described it as not the most efficient way of solving tasks in many cases. For instance, SD2 only thought it was useful if the developers were at the same level. They explained that if the difference in experience were too extensive, it would slow the fastest developer down.

Also, estimation of tasks was pointed out that could limit expertise over time. When asked if they thought there were agile practices that could reduce the effectiveness of teams, SD1 explained how agile practices with estimation might lead to less diversity in the tasks of team members. This was due to developers tending to select tasks in areas they have experience with to give shorter and more confident estimates. The respondent believed that this would prevent improvement in other domains. On the other hand, if team members chose tasks in areas they were less comfortable with, they suggested it would widen the expertise of the developers.

Achieving through other means

Besides the comment of TL3 about the team becoming more effective after removing the best developer, how to get team members to feel responsibility and balance of member contributions was not mentioned. However, it was suggested that psychological safety could impact team members contributing with their thoughts and ideas. More results regarding psychological safety will be presented in the next section.

4.5   Concerning Where

This theme is about the role of proximity, culture, and togetherness within the team. Results include informal communication and psychological safety.

Importance for team effectiveness

The part co-location plays was mentioned by some of the respondents with experience with distributed teams. Overall, The main problem with distributed teams seems to be related to communication. SA1 highlighted this:

> I have worked in teams that have been co-located, and I have worked with teams that have been split quite far, both geographically in other Nordic countries and also in India and elsewhere. It is much more difficult when you have a split in the team, especially at a long geographical distance. Or no matter how long it is really, it's enough that the team is split over one floor [laughter]. That gets far enough. There is a lot of informal communication going on, verbal informal communication. It's almost enough that one of the team members doesn't speak Norwegian, so that's enough to make information disappear. We have had several cases where we discuss almost next to a person who does not speak Norwegian and he does not catch up on perhaps important things then. So I think it's really important that a team sits together.

From this statement, it looks like the respondent thinks distributed teams are harder to work with, but that they do not believe that the physical distance is the issue. Instead, it would seem that the lack of informal communication can cause important information not to be transferred.

Another topic a number of the interviewees discussed was the effect psychological safety, or a lack of it, could have on the effectiveness of the team. Several potential problems were mentioned in relation to this. One of the respondents highlighted:

> Having some kind of psychological safety or security that you don't feel like you're being hanged for mistakes. This means that there is a culture of openness. It's very important. [...] We see that some of the teams, perhaps especially from India, with Indians. They have a little more that culture of fear with them, so they're a little afraid to ask, a little

afraid to come up with what they think are stupid questions, and then of course they don't do everything right automatically so that might be a problem. (SD4)

This comment emphasizes the role psychological safety could have in delivering what was wanted. While the culture of openness was tied geographically in this instance, lacking psychological safety was undoubtedly also seen in co-located teams. Another interviewee stated that:

It's also about getting good at sharing, and not only sharing what you have done, but sharing issues you have so that there's openness for discussion in teams. You can quickly become afraid to come up with problems because you appear not skilled or not competent, but that it should become an environment for that to be okay, I think is very important for effectiveness.

It's a bit about personality types, some can get stuck for days if there's a problem and don't dare to ask, right? But if they had asked someone, they might have had that discussion and solved it in an hour or two. (TL3)

This was an additional example of lacking psychological safety but less tied to culture and more tied to personality. It also points out how it could lead to less effectiveness.

Finally, SA1 elaborated on how lacking psychological safety could cause team members not to challenge each other. They presented a case where a team with several junior developers had a determined product owner. The project failed because the product owner had it their way while perhaps not knowing enough because none of the developers dared to disagree. In the end, they had to scrap everything since no one caught the problem before it was too late.

Achieving through agile practices

From the interviews, there was no clear indication that agile practices directly affected informal communication or psychological safety. Nevertheless, on how to increase the team togetherness during the global pandemic, one respondent said:

> Now in these corona times, maybe the standup has got a different function. Now it's no longer about making it so short; now it's really about covering this "coffee-talk" as well so that people get to talk a bit of this and that. And then maybe it's even more important not to be too many. (SA2)

Achieving through other means

Similar to the last quote, social events were brought up by several as a way to improve culture and promote informal communication. SD1 summarized some of the effects of social happenings like this:

> Having something social in your team makes teamwork more efficient because you lower the threshold for asking for help. That provides more skill sharing, which solves current and future problems faster. The culture also gets better, which makes employees more motivated.

There were also other means identified for increasing the psychological safety in the team. When asked about how you could get psychological safety, TL3 suggested

> maybe it's trying to share a little yourself, and maybe paying a little extra attention to those who don't necessarily say anything. Also to have some space and time for competence sharing. We have a kind of "code of the week" where we set aside a few hours to have common projects, or things that don't necessarily actually deliver anything. There we look at new products, look at code, and sometimes people present things they've

done more in depth. Yes, and discussion forums, I hope that creates some increased common competence. And with competence comes safety as well.

This quote touches on seeing team members on a personal level and promoting competence sharing. The respondent highlights two different ways of competence sharing: events where people can work together only focusing on transferring knowledge and forums for discussion.

One respondent was in a team that, despite using agile practices, still had effectiveness issues lacking experience and routines. After discussing how teams can be developed over time with the use of routines and knowledge sharing, the respondent was asked if there is anything a new team can do to be effective from the start:

> Yes, have forums where they can talk and at least not blame each other. That's because it creates one of those "it was his fault or her fault that made us not get to the finish line on this" or that things are wrong in a way. So rather being able to ask each other a lot to help each other, I think that's very important. (TL1)

The response above also addresses having a place where team members can ask questions, but it also puts additional emphasis on creating habits for not blaming each other.

4.6   Concerning How

Important processes of the team are what this second last theme is about. Results for the theme are related to feedback loops, team autonomy, and process improvement.

Importance for team effectiveness

In the same way as continuous delivery was used in section 4.1, most of the respondents mentioned feedback loops in some form related to increasing task understanding. For instance, TL3 explained trying to involve the product owner continually from the start of a task, and the team of SD1 discussed implementing continuous deployment to get feedback early.

When asked about what they thought an effective team was, one of the respondents regarded having some common standards for work as indicative:

> Then there is the platform itself and the development environment and things like that, that you are fairly aligned. Not necessarily that it should be 100% homogeneous, that everyone should absolutely use the same tools all the time, but that there is a good common basis, and so that there is some freedom at the top there. (SD4)

Further, some of the respondents mentioned team autonomy in the form of in-team decision-making as the most effective method of making decisions. SD1 emphasized trying to solve as many problems as possible at the team level because developers often are good problem solvers, and it is faster than waiting for decisions to be made externally. Furthermore, SA2 explains that in order to have effective decision-making, you need enough responsibility within teams to make decisions. Team autonomy also relates to findings for limiting external dependencies in section 4.7.

Finally, TL2 expands on what they think signifies the most effective teams: " The best teams are the ones who somehow manage to take a step back at all times, reflect on how they work and what they should improve on."

Achieving through agile practices

Firstly, TL1 highlights the importance of sticking to the routines of the agile practices: "And then there's having a process, from scratch. Having a scrum

master who is confident in the process. Then the developers can complain as much as they want, but they will experience it in the end that it becomes effective."

The respondents generally agreed that following the agile practices were routines that fostered the effectiveness of the team, and that using all practices from Scrum was effective in teams developing new functionality. TL1 argued that you should try to use all practices from Scrum, and remove the ones that are not working for the team. Some of the respondents, however, believed that Kanban was better suited when maintaining software.

Of the agile practices, retrospectives were highlighted several as critical for improving team effectiveness, but there was some variation in the opinions about the expendability of retrospectives in hectic times. For instance, SD2 believed it was not always necessary, while SD1 suggested that the effects of retrospectives might be the greatest and most needed in hectic times. The participants also noted a few pitfalls that could limit the utility of retrospectives. These included: having them too often, setting difficult-to-measure improvement goals, and focusing on small or unimportant problems.

Finally, the uses of daily standup meetings and sprint reviews as feedback loops were described positively by the respondents, with the only exception being if they were excessively long. Other positive effects of standups were also presented in section 4.2 and 4.5.

Achieving through other means

One last way to refine the processes of the team was explained by SD1. They presented having regular "tech talks" for developers, where technical process improvements could be suggested.

4.7   Results Outside the Framwork

This final theme of the analysis cover findings that did not match any of the other themes. This section will not follow the structure of the previous ones, as it mainly addresses new factors of team effectiveness. Nevertheless, the results found on how to achieve these factors will be presented. The findings are related to external dependencies.

When discussing important factors for team effectiveness, SA1 put forward limiting external technical dependencies. The respondent explains that focus over extended periods is essential for teams that deliver value, so noise and distractions can significantly hinder team effectiveness. They mention that one factor that can cause noise is having dependencies to the outside of the team, especially technical dependencies. An example of this was buying a solution to solve an immediate problem, but that creates a dependency that is not worthwhile in the long run. To solve this they try to build as much as they can within the team.

Another external dependency mentioned was the dependency to people outside the team. An interviewee explained the following:

> When you're dependent on others, you see that in practice as well, that when you're dependent on someone outside of your team, there's going to be a lot of waiting. And that's always the death of efficiency, right? [...] and instead of waiting, you grab something new and start working on something else. Then it starts ticking in feedback, input and stuff, and then suddenly there's a lot of balls in the air at once. And then you don't necessarily handle it as effectively. (SD4)

This suggests that these kinds of dependencies also affect efficiency and effectiveness negatively. Finally, related to this comment was the perspective of another respondent regarding team composition. They believed that a key to team effectiveness was

> a team that has the roles needed to complete something from start to

finish. It involves everything from design to it actually running in production, delivering real value. So all the roles are supposed to be in the team, I think. And there has to be a productive team that needs to be able to sort of clarify all the questions and figure things out, preferably in the team. It's important. (SA1)

This comment could also have been in section 4.4 but it highlights how limiting external dependencies can be achieved. Similarly, building on effective decicion-making discussed in 4.6, SA2 argued that if you have to consult people outside the team often, those people should rather be included in the team.

5 Discussion

In this thesis, the researcher has explored agile team effectiveness as seen by practitioners to increase the understanding of agile practices and factors of agile team effectiveness. More precisely, the research questions studied were:

RQ1. What are the key factors for team effectiveness in agile teams?

RQ2. Can factors of team effectiveness be used to explain the use of agile practices?

RQ3. Are there conditions needed for team effectiveness that agile practices do not aid in fulfilling?

To answer these questions, the researcher conducted a qualitative survey interviewing 10 participants experienced with agile teams. This resulted in findings across seven themes concerning factors suggested as being important for agile team effectiveness, in addition to how these factors could be achieved.

This section will discuss the findings related to the three research questions and present a revision of the 5W1H framework. The results will also be seen in the light of earlier studies. Further, contributions and limitations will be highlighted, and finally, directions for future research will be discussed.

5.1 What Are the Key Factors for Team Effectiveness in Agile Teams?

What

The first clear finding related to What was the importance of involving the business side. One respondent advocated aspects of waterfall development, but as others mentioned, you do not necessarily need to do a full waterfall specification phase because that works against the agile principles of failing early and often, and correcting the course. Involving the business side into agile specification-creation

could remove some of the fails, and planning how smaller parts can be released to fail earlier, are also likely to improve effectiveness. As long as the development team can understand it well enough, it would seem that involvement from a product owner or domain experts can mediate shortcomings in the specification agilely. To summarize, it would seem that important team effectiveness factors related to What have to do with task correctness and task completion. Firstly, specifying tasks well enough to be done efficiently while still giving the team the understanding to start working ensures a starting level of correctness. Secondly, specifying small enough deliverables to quickly complete enables faster delivery and promotes course correction for further correctness.

When

Results related to When indicate that predictability is important for two reasons. Priority understanding is a significant concept since the findings indicate changing the focus can create noise and interruptions for the team. This is in line with the results from Strode et al. (2012) and in addition, it shows how changing priorities can be a challenge.

It also appears that if team members have an overview of dependence on other team members, they will coordinate better and be more motivated. This supports the findings on interdependent coordination from earlier studies (Salas et al. 2005; Okhuysen and Bechky 2009; Strode et al. 2012) but it also highlights the effect on motivation.

Why

The importance of output driven communication was presented by Jarvenpaa et al. (1997), focusing on results being a sign of well functioning teams. The present research supports this while also paying attention to how communication about the goals and how the results lead toward the goals is beneficial.

The importance of trying to solve problems instead of simply completing tasks was

another finding. While practitioner-guides such as Kniberg (2015) stress understanding task goals so the developers can solve the problems in the way they see most fit, previous models for team effectiveness examined in this paper do not focus on this. This could be because agile development is more concerned with delivering value than traditional teamwork and that it relies more on the decision-making of team members.

Who

Two factors related to the Who conditions where found to directly affect team effectiveness. Having and gaining relevant expertise was one of the most brought-up factors and is in line with theory suggesting lack of expertise can be a hindrance (Strode 2016).

Further, while the findings have shown that a balance of member contributions is important, it also suggests that more team members being responsible positively affects effectiveness by increasing motivation and promoting a balance of contributions. The factors balance of member contributions and responsibility is supported by Hoegl and Gemuenden (2001) and Okhuysen and Bechky (2009) respectively.

There was not found support for flexibility in expertise being important, allowing for support on the work of others as proposed by earlier models (Salas et al. 2005; Hoegl and Gemuenden 2001), but this does not mean that this is without importance.

Where

The most common finding of factors of team effectiveness was the significance of psychological safety. The respondents highlighted a number of positive effects from team members feeling that asking for help and challenging each other is low-risk. Findings indicate that this increases the development speed and the likelihood of addressing bad decisions.

Another interesting finding was the perceived importance of informal communication and how it can be a reason why some co-located teams were more effective than distributed ones. It was found to improve task understanding, skill-sharing, and improving culture. This encompasses a significant portion of the factors found to be essential for team effectiveness.

Psychological safety and informal communication seem to be linked together, relating to the team culture. Seen in the light of the findings of Jarvenpaa et al. (1997), with trusting actions being both a precursor and a result of trust, perhaps informal communication and psychological safety behave in a like manner.

How

Having some common basis for how the team members work was seen as a signifier for effective teams, but it was not discussed why this could be the case. A likely explanation is that having a common understanding of how team members work makes it easier to help each other and find which work procedures are the most effective. Okhuysen and Bechky (2009) describe how developing an agreement on work plans can highlight potential limitations of processes so they can be addressed. They use the concept of *routines* that could encompass the practices examined in this thesis, of which several were found to increase team effectiveness.

One of the factors that several of the routines found in the present research have in common, especially related to increasing task understanding, was their function as feedback loops. The significance of continuously improving routines was also found and could be related to feedback loops addressing work processes. This supports the findings of both Salas et al. (2005) and Okhuysen and Bechky (2009).

Finally, the research data indicate that making decisions and solving problems within the team is effective. This is comparable to the findings of Lee and Xia (2010).

Other

Since the 5W1H framework was created to address team effectiveness factors within the team, some findings concerning relations to the outside of the team did not fit into any of the framework themes. The findings of the research outside the framework regarded limiting external dependencies of two types: technical and people dependencies.

Because the scope of this thesis is factors the team can do something about, limiting external dependencies does not qualify as a factor, often being outside the control of the team. Nevertheless, the findings suggest that the team can manage aspects of the external dependencies. Technical dependencies to the outside of the team are likely to be present in many software development teams. Still, the results indicate that if it is viable to remove a technical dependency by creating a solution internally, there are long-term benefits that could outweigh the immediate use of resources. Since this factor is something the team can control, it could be included in the concept of team autonomy, allowing the team more freedom in their processes. Similarly, a respondent mentioned that if information is frequently needed from people outside the team, they should be added to the team. Hence, there could be situations where the team can control this dependency. This factor then relates to the concept of having the relevant expertise in the team.

Revisiting the framework

Other than the factors mentioned above, the researcher could not find any that were proposed to be important for team effectiveness not in the 5W1H framework. This could be due to overlooked factors or the framework being too broad. Having a broad framework is not necessarily positive since factors could fit even if they do not affect team effectiveness. However, a sign of health for the 5W1H framework is that the respondents did not find aspects they would remove when presented the framework at the end of the interviews.

Figure 5 shows the resulting factors for team effectiveness categorized in the 5W1H
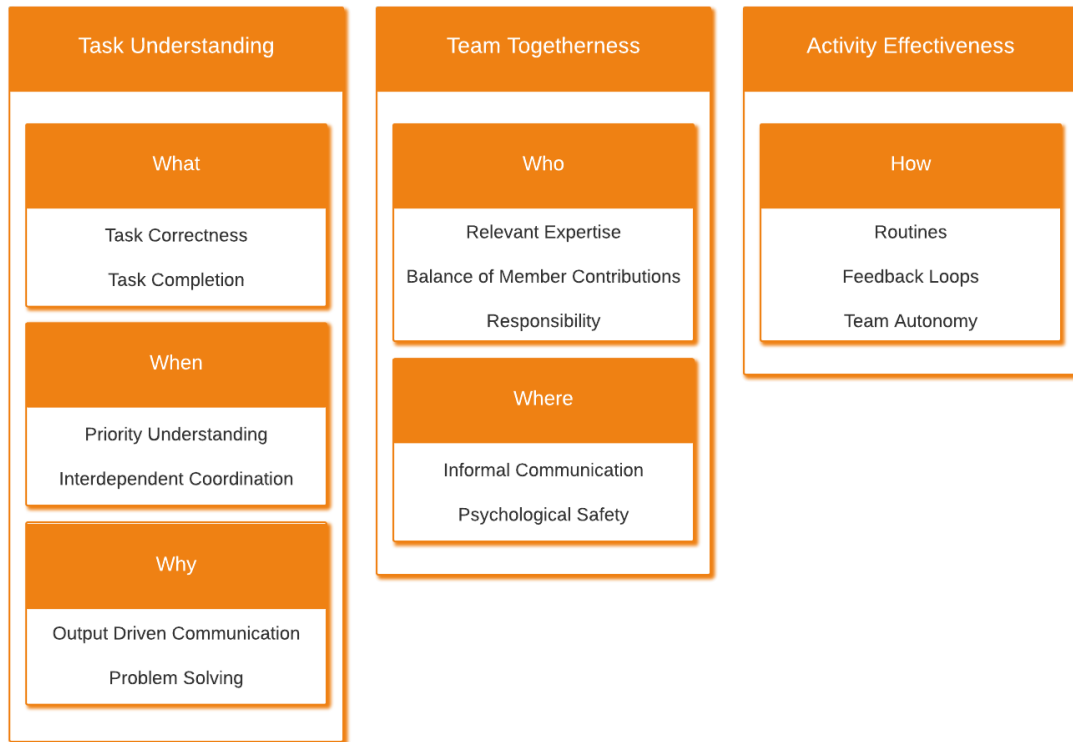
framework.

Figure 5: The 5W1H framework with the found factors

As visualized, the 5W1H framework was suited to incorporate the factors found in the study. It shows how there is an empirical basis for dividing the concept of task understanding into separate parts although they are connected. As a part of this, there is an explicit focus on problem solving, which is not a focus in earlier team effectiveness models. Additionally, the data of the study supports emphasizing more factors for togetherness than trust, contrary to earlier models. Finally, the model has empirical support for including the agile software development-related factors of feedback loops and team autonomy.

5.2   Can Factors of Team Effectiveness Be Used to Explain the Use of Agile Practices?

What

From the results of the interviews, the team effectiveness significance of understanding what to do was noticeable. The data showed that a useful practice to increase task correctness and task completion was the involvement of domain experts. The findings also suggested that the team can influence this themselves by inviting to discussions and reviews during development. Earlier studies show similar results, with the domain experts being the customers. Sandstø and Reme-Ness (2021) found customer involvement to increase knowledge sharing but did not highlight what knowledge is being transferred. However, the effects of on-site customers, found by Yu and Petter (2014), was related to the understanding of tasks.

When

The findings suggest sprint planning aided in creating an understanding of which tasks should be done first, as well as facilitate coordination of interdependent tasks. Further, while sprint reviews were found to gain insight into the work of others, easing coordination, this effect might be smaller due to reviews often occuring late in sprints. This does not mean they are not useful as feedback loops with domain experts, increasing task understanding for future work.

Daily stand up meetings are more frequent were also found to be able to improve interdependent coordination. Related to this Sandstø and Reme-Ness (2021) again found that standups increase knowledge sharing, and Yu and Petter (2014) found that they communicate progress, which is important for coordination.

Why

There were not found any practices directly addressing problem solving or output driven communication, but similarly to What, an understanding about task goals should be improved by involving domain experts and feedback loops. There was not much theory for practices on goal understanding alone, but as mentioned, Kniberg (2015) argued for trying to understand the goals of tasks during task specification. Although not mentioned, sprint reviews could be a relevant practice to promote output driven communication.

Who

How to get a balance of member contributions and team members to feel responsibility was not mentioned explicitly, but the findings imply that assigning team members to tasks or roles and showing trust in them to do their job. Daily standups could be a space where team members accountability is promoted. This is supported by Yu and Petter (2014) showing that standups motivate team members.

While pair programming was found to be an effective way of sharing relevant expertise throughout the team, some respondents were critical to the practice arguing that it is a slow manner of solving tasks. Nevertheless, it would seem that pair programming would increase the effectiveness of the team over time.

Where

None of the agile practices discussed during the interviews were explicitly stated to address informal communication or psychological safety. However, it was pointed out that during the global pandemic, where home offices are frequent, daily standups could play a role as informal communication. This adds to the notion of informal communication and co-location could have similar effects. Yu and Petter (2014) found, in a related manner, that daily standups fostered team togetherness.

How

Retrospectives were found as critical to discussing and improving processes. While team reflexivity might not increase team performance in the short term, like shown by Przybilla et al. (2018), discussing the tasks of the team and processes as well as reflecting on goals and strategies is likely to improve the effectiveness of teams over time.

In addition to this, daily standups, sprint reviews, and continuous delivery were found to work as feedback loops, improving the task understanding of the team.

5.3 Are There Conditions Needed for Team Effectiveness That Agile Practices Do Not Aid in Fulfilling?

What

One respondent mentioned how specifying tasks with a waterfall approach could lead to a more thorough specification. Nevertheless, by applying practices of involving domain experts and introducing feedback loops, developers can likely emulate this task understanding, both being able to start developing faster and catching misunderstandings earlier.

When

The present research found several agile practices address factors related to When. A key takeaway from the findings was that although formal practices facilitate communication, there should be communication regularly occurring during the rest of the day. This seems to be important in all aspects of teamwork where a common understanding is beneficial.

Why

An area where relatively few practice suggestions were found was related to goal-understanding. This study found that the responsibility for problem solving is shifted toward the developers in agile teams compared to those using a waterfall approach. It was suggested that agile developers might need to be more critical to the task specification since generally less time is spent on it than waterfall development.

Output driven communication was also not discussed in relation to agile practices. It is likely that practices like daily standups could incorporate to achieve some of its benefits.

Who

A finding of the study was that estimation of tasks might negatively affect the acquisition of expertise within the team since team members might be inclined to pick tasks they know how to do, to give shorter and more consistent estimates. This suggests that a team trying to increase the overall expertise to boost effectiveness in the long run, could adopt a strategy where team members are encouraged to challenge themselves.

Where

An essential finding of the thesis was that psychological safety was seen as very important and not solved by agile practices. There were a lot of suggestions on how to address this without the use of agile practices, most of them involving some kind of activity where there was room for informal communication. In addition, psychological safety seems to be closely related to culture. Fostering a team culture where it is encouraged to ask for help, challenge each other, and not blame individuals will likely stimulate the many positive effects of psychological safety.

How

Some ways the team can increase the team autonomy by limiting external dependencies have already been discussed, by incorporating needed technology or people into the team. Other than this, there were no findings on how team autonomy can be improved.

Aside from this, tech talks in the team were found to be a method to improve the technical processes and can be seen as a way of improving routines using feedback loops.

5.4   Contributions

The main contribution from this thesis is an exploration of agile team effectiveness and agile practices. It shows how several factors related to task understanding, team togetherness, and activity effectiveness are likely to lead to increased team effectiveness. Task understanding is highlighted as encompassing several kinds of factors that can be independent. Another significant finding for theory and practice is the benefits of teams being more goal-oriented. This study also calls attention to the importance of focusing more on psychological safety since many benefits were found, but few agile practices address this. The perspective of agile practices as tools with specific effects related to team effectiveness is an additional input into the field of software development and could be useful for theory and practice.

Furthermore, the thesis presents a team effectiveness framework based on empirical and theoretical research. The framework could be used similarly to how it is used in this thesis, or it could be used by practitioners to get an overview of which areas their team could improve. It sets itself apart from other models for team effectiveness on a few points: It takes into consideration the typical hierarchy of agile teams, with the responsibility being shifted towards the team for task and process decision-making. It also makes the comprehensive concept of task understanding easier to comprehend by identifying three sub-components. Finally, it addresses team togetherness, and especially psychological safety, which was found to play a significant role in team

effectiveness in this study.

5.5   Limitations

The researcher was very aware of the possibility of confirmation bias. Since factor groups very already set by the literature review, getting tunnel vision on those could be a detriment to the study. In the interviews, the respondents were asked if they disagreed with any of the factor groups found or if there where anything they thought the framework was missing to try to avoid this.

In Norwegian, the words for effectiveness and efficiency are the same, so this had to be considered during the translation of the interviews. In the information sheet given to the interviewees prior to the interviews, the research questions were presented in English, with the word effectiveness used consistently.

5.6   Future Research

Psychological safety and informal communication seem to have several positive effects on agile teamwork. The concept of psychological safety has been popularized in recent years, but yet more understanding is needed, especially related to agile teams. While psychological safety can be hard to quantify, the amount of informal communication should be significantly easier to evaluate, making empirical studies on its effects more feasible. This could perhaps also shed more light on how much of the benefit of having co-located team come from informal communication alone.

Another factor found in the present research with less theoretical support was problem solving. Comparing teams where their tasks are specified either with user challenges to solve or decided solutions to implement could provide more insight into when problem solving is the most needed.

6   Conclusion

This thesis presents the 5W1H framework for understanding agile team effectiveness by gathering relevant factors and examining how they can be achieved. The framework is built on both theoretical and empirical research. It differentiates from earlier models for team effectiveness by being created for agile teams, granularizing the concept of task understanding, and incorporating the role the environment has on team members. This last inclusion leads to examining the impact psychological safety and informal communication can have on team effectiveness and how informal communication might be the key to emulating the positive effects of co-location.

The research found factors for team effectiveness that fit into three overarching categories: task understanding, team togetherness, and activity effectiveness. Notably, the findings indicate several aspects of task understanding are important, and that psychological safety has a number of benefits. The overview of the factors found is shown in figure 5.

Further, the factors found for team effectiveness were used to try to explain the use of agile practices. From the results, it seemed that many agile practices have effects related to the factors. In particular, the involvement of domain experts appeared to have an influence on several of the factors related to task understanding, which makes it advantageous. Daily standups were found as another effective practice and seemed to have the potential to impact a number of factors related to task understanding and team togetherness.

The factors found for team effectiveness were also used to research whether agile practices are sufficient for teams to be effective or if additional practices are needed. Generally, agile practices seemed to promote several aspects of task understanding, but no routine to keep the team goal-oriented was found. It is likely that teams could benefit from focusing more on problem solving and output driven communication.

While psychological safety was found to have several benefits for team effectiveness, agile practices did not seem to affect it sufficiently. Additional

activities with a higher degree of informal communication could complement agile practices in achieving more positive effects. Communication was also found to be important regularly throughout the day, and agile practices should be a supplement to it, not a replacement.

Based on the research done in this thesis, it cannot conclude that the factors of the 5W1H framework apply to all agile software development teams. This is due to the qualitative nature of the study, and it was neither the aim. However, the study gives insight into the factors theory and empirical data suggest to affect team effectiveness, and the relationship between them and common agile practices.

A model like the 5W1H framework will to some degree always be a simplification of reality. It might not capture all nuances but sacrifices details for usability. When humans are involved, they can not be factors in simple algorithms that produce the same result every time. But models can help us understand the world around us, and we can use them as tools to achieve our goals. We might realize that the model is flawed, so we make changes to it or try to make a new one, and in trying that, we are likely to take our understanding of reality one step forward. Not just for ourselves but for everyone. That has been the goal of this thesis.

Bibliography

Agile Manifesto (n.d.). *Manifesto for Agile Software Development.* URL: https://agilemanifesto.org/.

Alahyari, Hiva, Richard Berntsson Svensson and Tony Gorschek (2017). 'A study of value in agile software development organizations'. In: *Journal of Systems and Software* 125, pp. 271–288. ISSN: 01641212. DOI: 10.1016/j.jss.2016.12.007. URL: http://dx.doi.org/10.1016/j.jss.2016.12.007.

Dingsøyr, Torgeir and Yngve Lindsjørn (2013). 'Team performance in agile development teams: Findings from 18 focus groups'. In: *Lecture Notes in Business Information Processing* 149.7465, pp. 46–60. ISSN: 18651348. DOI: 10.1007/978-3-642-38314-4_4.

Earley, P.C. (1986). 'Trust, perceived importance of praise and criticism, and work performance: an examination of feedback in the United States and England'. In: *Journal of Management*, pp. 457–473.

Edmondson, Amy (1999). 'Psychological Safety and Learning Behavior in Work Teams Author (s): Amy Edmondson Published by : Sage Publications , Inc . on behalf of the Johnson Graduate School of Management , Cornell University Stable URL : http://www.jstor.com/stable/2666999 in W'. In: *Administrative Science Quarterly* 44.2, pp. 350–383.

Espinosa, A. J. et al. (2007). 'Team knowledge and coordination in geographically distributed software development'. In: *Journal of Management Information Systems*, pp. 135–169.

Feynman, Richard P. (1974). *Cargo Cult Science.* URL: http://calteches.library.caltech.edu/51/2/CargoCult.pdf.

Forsgren, Nicole, Jez Humble and Gene Kim (2018). *Accelerate: The Science of Lean Software and DevOps: Building and Scaling High Performing Technology Organizations.* It Revolution Press. ISBN: 1942788339; 9781942788331.

Hoegl, Martin and Hans Georg Gemuenden (2001). 'Teamwork Quality and the Success of Innovative Projects: A Theoretical Concept and Empirical Evidence'. In: *Organization Science* 12.4, pp. 435–449. ISSN: 10477039. DOI: 10.1287/orsc.12.4.435.10635.

Jansen, Harrie (2010). 'The logic of qualitative survey research and its position in the field of social research methods'. In: *Forum Qualitative Sozialforschung* 11.2.

Jarvenpaa, Sirkka L., Kathleen Knoll and Dorothy E. Leidner (1997). 'Is anybody out there? Antecedents of trust in global virtual teams'. In: *Journal of Management Information Systems* 14.4, pp. 29–64. ISSN: 07421222. DOI: 10.1080/07421222.1998.11518185.

Klein, Heinz K. and Michael D. Myers (1999). 'A set of principles for conducting and evaluating interpretive field studies in information systems'. In: *MIS Quarterly: Management Information Systems* 23.1, pp. 67–94. ISSN: 02767783. DOI: 10.2307/249410.

Kniberg, Henrik (2015). *Scrum and XP from the Trenches*. 2nd ed. InfoQ. ISBN: 978132922427.

Larzelere, R.E. and T.L. Huston (1980). 'The dyadic trust scale: toward understanding trust in close relationships'. In: *Journal of Marriage and the Family*, pp. 595–604.

Lee, Gwanhoo and Weidong Xia (2010). 'Toward agile: An integrated analysis of quantitative and qualitative field data on software development agility'. In: *MIS Quarterly: Management Information Systems* 34.1, pp. 87–114. ISSN: 02767783. DOI: 10.2307/20721416.

Lindsjørn, Yngve et al. (Dec. 2016). 'Teamwork quality and project success in software development: A survey of agile development teams'. In: *Journal of Systems and Software* 122, pp. 274–286. ISSN: 01641212. DOI: 10.1016/j.jss.2016.09.028.

Mathieu, John et al. (2008). 'Team effectiveness 1997-2007: A review of recent advancements and a glimpse into the future'. In: *Journal of Management* 34.3, pp. 410–476. ISSN: 01492063. DOI: 10.1177/0149206308316061.

Moe, Nils Brede, Torgeir Dingsøyr and Tore Dybå (2010). 'A teamwork model for understanding an agile team: A case study of a Scrum project'. In: *Information and Software Technology* 52.5, pp. 480–491. ISSN: 09505849. DOI: 10.1016/j.infsof.2009.11.004. URL: http://dx.doi.org/10.1016/j.infsof.2009.11.004.

Myers, Michael D. and Michael Newman (2007). 'The qualitative interview in IS research: Examining the craft'. In: *Information and Organization* 17.1, pp. 2–26. ISSN: 14717727. DOI: 10.1016/j.infoandorg.2006.11.001.

Norwegian Centre for Research Data (2021). *Information and consent.* URL: https://www.nsd.no/en/data-protection-services/notification-form-for-personal-data/information-and-consent/ (visited on 23rd Oct. 2021).

Oates, Briony J. (2006). *Researching Information Systems and Computing.* SAGE Publications.

Okhuysen, Gerardo A. and Beth A. Bechky (2009). 'Coordination in Organizations: An Integrative Perspective'. In: *The Academy of Management Annals* 3.1, pp. 463–502. ISSN: 1941-6520. DOI: 10.1080/19416520903047533.

Przybilla, Leonard, Manuel Wiesche and Helmut Krcmar (2018). 'The influence of agile practices on performance in software engineering teams: A subgroup perspective'. In: *SIGMIS-CPR 2018 - Proceedings of the 2018 ACM SIGMIS Conference on Computers and People Research*, pp. 33–40. ISBN: 9781450357685. DOI: 10.1145/3209626.3209703.

Robson, Colin (2002). *Real World Research. A Resource for Social Scientists and Practitioner-Researchers.* 2nd ed. Wiley. ISBN: 9780631213055.

Rodríguez, Pilar et al. (2012). 'Survey on agile and lean usage in finnish software industry'. In: *International Symposium on Empirical Software Engineering and Measurement*, pp. 139–148. DOI: 10.1145/2372251.2372275.

Runeson, Per and Martin Höst (2009). 'Guidelines for conducting and reporting case study research in software engineering'. In: *Empirical Software Engineering* 14.2, pp. 131–164. ISSN: 13823256. DOI: 10.1007/s10664-008-9102-8.

Salas, Eduardo, Dana E. Sims and C. Shawn Burke (2005). 'Is there A "big five" in teamwork?' In: *Small Group Research* 36.5, pp. 555–599. ISSN: 10464964. DOI: 10.1177/1046496405277134.

Sandstø, Rebecca and Cornelia Reme-Ness (2021). 'Agile Practices and Impacts on Project Success'. In: *Journal of Engineering, Project, and Production Management* 11.3, pp. 255–262. DOI: 10.2478/jeppm-2021-0024.

Shachaf, Pnina (2008). 'Cultural diversity and information and communication technology impacts on global virtual teams: An exploratory study'. In:

Information and Management 45.2, pp. 131–142. ISSN: 03787206. DOI: 10.1016/j.im.2007.12.003.

Sharp, Helen and Hugh Robinson (2010). 'Three 'C's of Agile Practice: Collaboration, Co-ordination and Communication'. In: *Agile Software Development: Current Research and Future Directions*. Ed. by Torgeir Dingsøyr, Tore Dybå and Nils Brede Moe. Berlin Heidelberg: Springer-Verlag, pp. 61–85. ISBN: 9783642125744. DOI: 10.1007/978-3-642-12575-1_4.

Strode, Diane E. (2016). 'A dependency taxonomy for agile software development projects'. In: *Information Systems Frontiers* 18.1, pp. 23–46. ISSN: 15729419. DOI: 10.1007/s10796-015-9574-1. URL: http://dx.doi.org/10.1007/s10796-015-9574-1.

Strode, Diane E. et al. (2012). 'Coordination in co-located agile software development projects'. In: *Journal of Systems and Software* 85.6, pp. 1222–1238. ISSN: 01641212. DOI: 10.1016/j.jss.2012.02.017. URL: http://dx.doi.org/10.1016/j.jss.2012.02.017.

Walsham, Geoff (2006). 'Doing interpretive research'. In: *European Journal of Information Systems* 15.3, pp. 320–330. ISSN: 14769344. DOI: 10.1057/palgrave.ejis.3000589.

Yeager, SJ (1978). 'Measurement of independent variables which affect communication: a replication of Roberts and O'Reilly'. In: *Psychological Reports*, pp. 1320–1324.

Yu, X. and S. Petter (2014). 'Understanding agile software development practices using shared mental models theory'. English. In: *Information and Software Technology* 56.8. Cited By :52, pp. 911–921. URL: www.scopus.com.

Appendix

A   Interview Guide

Introductory questions

- How much experience do you have with agile software development?

  - What is your role in/associated with a software development teams, and how much experience do you have in this one?

- Can you describe what a normal workday is like for you?

- What is your relationship to agile development?

Main questions

- What do you think is an effective (software development) team?

  - And vice versa?

- What kind of circumstances, actions, and knowledge do you think can lead to increased team effectiveness?

  - Why?

- If you've been part of a team where team effectiveness wasn't that good, what do you think it was because of?

  - And vice versa?

- What agile practices (affecting teamwork) do you use?

  - What are their goals of using this practice?

  - What were the perceived effects of using this practice at the start of the project?

  - How have the perceived effects of using this practice changed during the project?

- What other practices (affecting teamwork), which are not considered agile, do you use to increase team effectiveness?

    - Why?

Perspective questions

- Can agile sometimes come at the expense of team effectiveness?

    - Are there areas where agile methods do not extend?
    - What aspects of agile development can prevent team effectiveness?

- What do you think is important in a agile method?

- What agile practices do you think are the most effective?

    - Why?

- What practices do you have bad experiences with, which good ones?