UNIVERSITY OF OULU

Paradox of the Daily Stand-up Meetings in Agile Software Development Context

Abstract

Agile software development has become the norm in the industry, and daily stand-up meeting is the most adopted Agile practice despite the used Agile methodology. Since mid-2000s, the daily stand-up practice has been reported by academics to deliver various kinds of benefits. However, in recent years, numerous studies have emerged, claiming the attitudes of practitioners towards the daily stand-up meeting being quite a bit on the negative side – many practitioners even feeling the practice is just a waste of time.

This thesis is a literature review of existing studies on Agile software development teams, teams' daily stand-up meeting practices, and perceptions towards the daily stand-up meeting. The objective of this thesis is to dive into rationale behind those negative attitudes towards the daily stand-up meeting. By understanding how teams conduct their daily stand-up meetings, and how practitioners perceive both positive and negative aspects of the daily meeting practice, this thesis seeks to understand what makes the practice so popular and disliked at the same time.

Daily stand-up meeting is one of the easiest Agile practices to take in use. Therefore, it is often also one of the first Agile practices that teams start using when switching to an Agile methodology. Benefits of the daily stand-up meeting are notable and comes in various shapes and sizes. However, the original idea and purpose of the daily stand-up meeting is not well known among practitioners. Way too often, practitioners drift from inspection and adaptation towards status reporting, which is the most common daily stand-up meeting anti-pattern. Due to not fully grasping the original purpose of the daily stand-up meeting, practitioners start to experience negative attitudes towards the practice. Those negative attitudes towards the meeting impose undesired impact on meeting activities, which in turn further amplifies those negative attitudes.

Customizing the daily stand-up meeting to suit team's requirements is encouraged. Whatever works for the team can be done if the original purpose of the daily stand-up meeting is not forgotten. Teams need to be highly cautious not to turn daily stand-up meetings into status reporting meetings as that is a sure way to cause negative attitudes among participants.

Abbreviations

ASD   Agile Software Development

DSM   Daily stand-up meeting

HR   Human Resources

JIT   Just-in-Time

NDM   Naturalistic decision making

TPS   Toyota Production System

XP   Extreme Programming

Contents

1.   Introduction

This section starts by providing a brief introduction to reasoning behind the popularity of the most adopted Agile practice: the daily stand-up meeting (DSM). It will be followed by an elaboration on the motivation for this topic. Finally, the research problem and used research method are presented.

1.1 Background

While the number of different Agile software development methods is quite notable, there is also plenty of variation in techniques and practices different Agile methods embrace (Abrahamsson, Salo, Ronkainen, Warsta, 2002). Some Agile methods, such as Scrum, are more prescriptive than others, like for instance Kanban (Stray, 2014). Due to the lack of prescriptiveness, not all Agile methods provide a comprehensive guideline on procedures to be followed. However, certain Agile practices have been widely adopted among practitioners, irrespective of the used Agile software development methodology. Throughout the years, daily stand-up has been by far the most adopted practice among Agile teams (Digital.ai, 2022).

Given the huge popularity of daily stand-up meeting practice, there must be a multitude of good reasons for why teams are so widely embracing this practice. As per the founding fathers of Scrum methodology, daily stand-up meetings have numerous advantages, such as positive impact on team's communication, improved identification of impediments, faster decision-making, and reduced need for other meetings (Schwaber & Sutherland, 2020). Indeed, according to various studies, communication has clearly improved after taking Agile daily stand-up practice in use (Karlström & Runeson, 2006; Paasivaara, Durasiewicz & Lassenius, 2008; Pikkarainen, Haikara, Salo, Abrahamsson, & Still, 2008). Also, Paasivaara et al. (2008) found that not only the quality and frequency of communication increased, but daily stand-up meetings triggered some further one-on-one conversations that were perceived highly beneficial. Furthermore, benefits of daily stand-up meetings are not limited to only collocated teams. Similar improvements have been seen in context of distributed teams, too. Dorairaj, Noble & Malik (2012) observed how the improved interaction increased the level of unity in a team. In the similar vein, Pikkarainen et al. (2008) reported higher level of trust among developers from different sites. Also, thanks to daily stand-up meetings, teams' problem-solving capability has been reported to improve (Pikkarainen et al., 2008). In fact, many studies seem to echo the benefits Scrum methodology claims to deliver through daily stand-up meetings. In his book "Navigating Hybrid Scrum Environments: Understanding the Essentials, Avoiding the Pitfalls", Frederik Fowler (2019) goes even further and claims that when daily stand-up meetings are used properly, they enable success for every sprint.

1.2 Motivation

On a personal level, daily stand-up is a concept that I have gotten very familiar with in all shapes and sizes. With almost 20 years of work experience in the software industry, I have participated in numerous daily meetings in all possible team setups: collocated, remote, and hybrid. Also, I have been participating in daily stand-up meetings in all possible roles: software engineer, Scrum Master, and Product Owner. I am a proponent of the daily stand-up meeting practice. I do see multiple benefits with the practice – many of which have been reported by the past research and gray literature. However, I do

acknowledge there are certain challenges with the practice. In fact, I have personally experienced and witnessed various kinds of challenges with daily stand-up meetings. Especially in larger enterprise context, those challenges seem to be more common and pronounced. I have noticed the same challenges to be present in very different kinds of team setups across various industries. In many teams, individuals have been complaining about those challenges. This means the challenges with daily stand-up meetings have not gone unnoticed – not at least by the practitioners. Given the amount of time individuals spend in daily stand-up meetings, we are talking about a notable cost factor on an organizational level. For that reason, I have been slightly surprised about the lack of interest in improving the daily stand-up practice in companies – it feels like practitioners are slightly reluctant to challenge the status quo. This is what sparked the interest in me and got me to study this topic in more depth.

From academic perspective, the motivation for this thesis topic stems from the observation that there seems to be no literature reviews done on Agile daily stand-up meeting practice. Given how prominent practice daily stand-up meetings are in context of Agile software teams, and since the topic has been studied for the past two decades by academics, I was slightly surprised about the lack of literature review on this practice. Therefore, to fill the gap, and to learn more about the Agile daily stand-up meeting practice from the practitioner's viewpoint, this topic was chosen.

1.3 Research problem and method

Software industry has been advocating Agile principles for two decades. When introduced, Agile was perceived, especially from practitioners' viewpoint, a clear improvement over waterfall process model that used to dominate the industry before Agile started to raise its head. Agile heavily emphasizes the importance of communication and collaboration. One way how it manifests on a practical level is through the practice of daily stand-up meetings, embraced by certain Agile methodologies. While daily stand-up meetings are not the only way Agile practices encourages communication and collaboration, the practice of frequent meetings is certainly one of the most prominent embodiments of that. Indeed, practitioners moving from waterfall to Agile often find daily stand-up meetings very helpful (Asnawi, Gravell & Wills, 2012; Overhage & Schlauderer, 2012; Whitworth & Biddle, 2007).

However, when skimming through studies from the past 10 years, one cannot avoid noticing a clear shift in practitioners' opinion towards daily stand-up meetings. Interestingly, and quite surprisingly too, many of the recent studies are reporting practitioners of daily stand-up meetings finding the practice not worth the trouble or at least have highly mixed feelings about it (Morandini, Coleti, Oliveira Jr and Corrêa, 2020; Singh & Strobel, 2022; Stray, Moe and Bergersen, 2017). Given the big promise of daily stand-up meetings, and the proven positive effects it has on team level, especially on communication, it feels a bit strange practitioners would not find the practice worthwhile anymore.

Paradoxically, despite of the somewhat negative outlook on daily stand-up meetings by practitioners, vast majority of practitioners are reporting regularly participating in daily stand-up meetings (Digital.ai, 2022). In today's fast-paced, time-to-market and winner-takes-it-all driven world of software business, it seems strange that organizations are cumulatively spending so much time on a practice that is not perceived to deliver expected value.

To gain a holistic overview on how practitioners perceive daily stand-up meeting practice, the aim of this thesis is to focus on both positive and negative sides of daily stand-up meetings in Agile software development context. Therefore, the following research question will be addressed in this thesis from the viewpoint of practitioners:

Given the amount of reported dissatisfaction towards daily stand-up meetings, why is the practice so widely adopted by Agile software development teams?

The chosen research method for this thesis is literature review. To conduct the literature review, various databases of scientific, peer-reviewed articles were queried for relevant papers. The goal was to get a broad set of articles that talks about Agile daily stand-up meetings. Acknowledging there is some variation on what people and different Agile methodologies call the daily stand-up meeting practice, the search string had to take that variation into account. For example, in Scrum methodology, the daily stand-up meeting is called "daily Scrum" (Schwaber & Sutherland, 2020). Nonetheless, common to Agile methodologies is that some sort of frequent, ideally a daily, meeting practice is recommended (Strode, 2006).

While daily stand-up meeting is a common practice also outside of the software engineering domain, this thesis only concentrates on software-focused organizations. Therefore, research on non-software teams, such as HR, Marketing and Sales, are excluded.

First, articles that deal with Agile daily stand-up meetings were queried from Scopus. This resulted to 120 articles. Then, another, much broader query on challenges, related to Agile practices, was executed. The thought here was that there could be articles on challenges, related to Agile practices, but with no special focus on daily stand-up meetings only. As expected, this time the query resulted to significantly broader set of articles: 7593 documents were found. Given the first set of articles was somewhat small, it was possible to go through all articles manually. However, the second batch of documents contained way too many articles to be gone through manually. Therefore, a new search was made, limiting the search only to title. The new search resulted to 286 documents. Following queries were made:

1) TITLE-ABS-KEY ("agile" AND "daily" AND ("stand-up" OR "stand up" OR "standup" OR "scrum" OR "meeting")) AND (LIMIT-TO (SUBJAREA , "COMP") OR LIMIT-TO (SUBJAREA , "BUSI"))
2) TITLE ("agile" AND ("challenge*" OR "problem*" OR "obstacle*" OR "barrier*")) AND (LIMIT-TO (SUBJAREA , "COMP") OR LIMIT-TO (SUBJAREA , "BUSI"))

In addition to Scopus, some further searches were made using Google Scholar, IEEE Explore, and ACM Digital Library. This was done to ensure no relevant articles went unnoticed. Used search strings were derived from earlier Scopus searches. Out of the initial set of 400+ articles, close to 70 articles passed the initial screening process. The screened papers were then read, analyzed, and annotated. During this process, those papers that did not directly contribute to this thesis' topic were dropped. Eventually, 34 papers were identified as being highly relevant.

2. Agile software development

This section provides an overview on how Agile movement got started, and what are the principles and core believe behind Agile Software Development (ASD). Also, a look into the prevalence of different Agile methodologies is given, along with a brief description of a few most common ones. Last, daily stand-up meeting practice is examined from the standpoint of those methodologies.

2.1 Agile principles and history

Agile principles were introduced in 2001 by a group of practitioners who felt the way software was developed at that time was not quite optimal. Agile principles were underlining the importance of individuals and interactions, working software, close collaboration with customers, and responsiveness to change. (Beck et al., 2001) This was a clear shift of mindset when comparing to highly process-driven and rigid approach that used to dominate software industry those days. However, it is not the work practices that were new and revolutionary with Agile methods. Instead, it was the realization of how important people are from the viewpoint of successful project execution, coupled with the strong focus on effectiveness and adaptability to ever-changing operational environment (Highsmith & Cockburn, 2001).

In early 2000s, various Agile methodologies started to receive attention, thanks to the big buzz around the Agile Manifesto. After some time, a few of those methods started to gain notable traction as more and more organizations were adopting Agile practices. Digital.ai, the company behind the annual "State of Agile" report, recently reported (Digital.ai, 2022) that the number of organizations adopting Agile practices has been steadily increasing over the past 15 years. The report also revealed Agile practices have become popular also within non-IT groups too, such as Finance, Human Resource, and Marketing. According to the report, Scrum methodology is by far the most popular one with a 66% share. This is followed by ScrumBan, Scrum/XP Hybrid, Kanban, Iterative, Extreme Programming, and Lean Startup – all with a single digit share.

2.2 Scrum

Scrum was developed in 1990s by Ken Schwaber and Jeff Sutherland to help to "generate value through adaptive solutions for complex problems". Scrum methodology has its roots in empiricism and lean thinking. It is an iterative and incremental method that focuses on predictability and control of risks. The Scrum process has four formal events: sprint planning, daily Scrum, sprint review, and retrospective. All these four events occur within a container event, called *Sprint*. With these events, Scrum provides an opportunity to continuously inspect and adapt the execution. (Schwaber & Sutherland, 2020.)

Sprint is a fixed length structured process, typically one month or less, with a clear start and an end. Sprints are the heartbeat of Scrum, providing consistency and predictability to development process. Each sprint starts with a sprint planning session, defining scope and the goal for that sprint. As an outcome of a sprint, value has been created and delivered. During a sprint, development team meets each day to inspect progress towards the agreed sprint goal. This provides a mechanism that allows the team to adapt and adjust if needed. In Scrum methodology, development team's daily meeting is called *Daily Scrum*. Daily Scrum meetings are short, 15 minutes, and have a strong focus on the

progress toward the sprint goal. In the end of each sprint, development team presents the results of that sprint to key stakeholders. This provides an avenue to examine the outcome and to determine future adaptations regarding the product goal. Finally, after the sprint review, the very last event of each sprint is sprint retrospective. For the retrospective, team gathers to plan ways to improve effectiveness and to increase quality. The goal is to continuously improve the way how team operates by identifying work related matters that hinder the process. Also, during the retrospective, team discusses what went well to ensure those good practices will be carried on to the future sprints, too. (Schwaber & Sutherland, 2020.)

2.3 Extreme Programming

Extreme Programming (XP) is one of the early Agile methodologies that started to gain wider attention in the software development community. In early 2000s, XP was one of the most discussed topics and adopted Agile method among software professionals (Newkirk, 2002). Extreme Programming was introduced by Kent Beck in 1999. In his article, "Embracing change with Extreme Programming", Beck (1999) describes Extreme Programming as a new methodology that "turns the conventional software process sideways". Extreme Programming was a response to the stiff waterfall process model that used to dominate the field of software development methodologies at that time.

The radical idea behind Extreme Programming was its approach to project management. While in waterfall software model requirements are defined up-front in the beginning of the project, XP implements Agile principles by delivering software in small increments. Each incremental delivery is treated as its own mini project. This enables adaptability to ever-changing requirements. While XP is not strictly prescriptive on how to do things, it does define 12 practices to be followed. These practices, when implemented together, highly promotes the four core values of XP: 1) communication, 2) simplicity, 3) feedback, and 4) courage (Newkirk, 2002).

Extreme Programming does not mention daily stand-up meetings as one of its major or corollary practices (Beck, 2000). However, Don Wells, one of the early adopters of, and influencers behind the XP methodology, has documented daily stand-up meetings to be one of the "*rules of Extreme Programming*" (Wells, 1999). Also, in his book "*Extreme programming explained: embrace change*", Beck (2000) gives a few examples on when daily stand-up meetings could be used during an Extreme Programming process. Among academics, there seems to be a clear consensus that daily stand-up is an important practice in Extreme Programming (Stray et al., 2016).

2.4 Kanban

Over the years, Lean approach has started to gain more and more attention in the field of software engineering. With its roots deep in the Japan's automotive industry, it was originally called Toyota Production System (TPS). One of the best-known practices of TPS is Kanban. Kanban plays a key role in Lean. In Toyota's House of Lean, Kanban is one of the two pillars. (Al-Baik & Miller, 2015) Kanban is how Just-in-Time (JIT) is achieved in Lean – in fact, Kanban's whole purpose is to enable JIT (Ohno & Bodek, 2019).

In context of Agile software methodologies, Kanban is relatively new entrant. In a 2013 published study (Ahmad, Markkula & Oivo, 2013), it was viewed as one of the most

recent additions to the Agile and Lean software development research area. However, the method has been increasingly gaining traction in the industry. In 2022, Kanban has a solid footprint in the list of most adopted Agile methodologies (Digital.ai, 2022).

Kanban method recommends establishing a cadence of regular meetings. Those meetings include, but are not limited to, daily stand-up meetings, planning, meetings, and review meetings (Corona & Pani, 2013). However, Kanban does not define how one is supposed to conduct those activities. Based on multiple systematic literature reviews (Ahmad et al., 2013; Al-Baik & Miller, 2015), daily stand-up meeting seems to be a commonly used practice by software teams implementing Kanban method. This is no wonder as Ahmad et al. (2013) found daily stand-up meetings to play a key role from the viewpoint of information flow.

2.5 Custom and hybrid models

Agile methodologies are often customized to fit organization's needs. Some researchers hypothesize that majority of Scrum practitioners would not apply Scrum by the letter (Theocharis, Kuhrmann, Münch & Diebold, 2015). Others go even further by saying that every company vary Scrum in some way (Diebold, Ostberg, Wagner & Zendler, 2015). Multiple reasons for customization exist, but a strong consensus among academics seems to be that the main motivation stems from various kinds of challenges, such as management buy-in, or misalignment between management and developers (Kuhrmann et al., 2017). Also, a long history with non-Agile methods, along with a stiff organizational structure, increases the likelihood of variation (Diebold et al., 2015). The latest *State of Agile* survey (Digital.ai, 2022) makes it visible that many organizations customize their Agile process by picking and choosing practices from different Agile methodologies. As per the survey, the list of the most used Agile methodologies contains hybrid models, such as "Scrum/XP hybrid" and "ScrumBan".

By combining practices from different methodologies, organizations can better align the expectations and needs of management and developers (Kuhrmann et al., 2017). Which Agile methodology and a mix of practices organization chooses to use, depends on the context at hand. According to Theocharis et al. (2015), all studies show a clear trend of the most dominating Agile methodologies being Scrum, Kanban, and XP. Even though Scrum methodology is the only one prescribing daily Scrum meetings, daily stand-up meeting is by far the most adopted practice among professionals (Digital.ai, 2020). This is a good example of how popular practices are adopted across different methodologies.

2.6 Daily Stand-up Meetings in Agile Teams

Daily stand-up meeting is one of the most used Agile practices (Digital.ai, 2022; Kuhrmann et al., 2017). Given the benefits of daily stand-up meetings, such as increased social accountability and support, and improved visibility within project, (Whitworth & Biddle, 2007) it is no wonder the practice has been so widely adopted. Other perspectives exist, too (Whitworth & Biddle, 2007). Some academics attribute some of the practice's popularity to the fact that it is the recommended way to start implementing Scrum, and thus, the most likely practice to be picked up by Agile teams (Stray, 2014).

On high level, the purpose of the daily stand-up meeting is to provide visibility and increase awareness within a team. Like Agile methodologies, the daily stand-up meeting practice slightly varies from organization to another (Diebold et al., 2015). Typically,

however, the meeting somewhat follows Scrum methodology's Daily Scrum practice. The Daily Scrum is a daily stand-up meeting of 15 minutes, with a goal of inspecting team's progress towards the Sprint Goal, while ensuring everyone knows what to do next (Schwaber & Sutherland, 2020). Daily Scrum serves two key purposes: 1) ensuring everyone in development team understands each other's work's status, and 2) finding out if anyone in team needs help, and if so, identify the best way to provide that help (Fowler, 2019).

During a Daily Scrum meeting, each participant answers three questions: 1) what I have completed, since the previous meeting, 2) what I'm going to work next, and 3) are there any impediments hindering my progress (Rising & Janoff, 2000). However, in the most recent, 2020 published revision of The Scrum Guide, the three questions for the Daily Scrum have been removed. Reason behind the removal is the authors' desire to make Scrum less prescriptive by making the Scrum framework simpler and better focused. (West, 2020) Duration of the Daily Scrum is kept short on purpose. As the development is put on hold during the meeting, the Daily Scrum is supposed to be highly focused and concise to cause minimal disturbance to team's progress. For this reason, any detailed discussions must take place after the meeting. (Fowler, 2019.)

3. Agile Daily Stand-up Meetings

Based on existing literature, this section digs deeper in how practitioners are conducting daily stand-up meetings, and how the meeting practice is perceived by practitioners. First, characteristics of daily stand-up meetings are presented to gain understanding on how exactly practitioners conduct daily stand-up meetings? Then, an overview of perceived positive aspects of daily stand-up meetings is provided. And finally, we will investigate what kinds of challenges practitioners are reporting about daily stand-up meetings.

3.1 Characteristics

One of the most active researchers in field of Agile daily stand-up meetings, Viktoria Stray, focused on characteristics of daily stand-up meetings in her doctoral dissertation (Stray, 2014). Stray's thesis was based on four case studies as well as one grounded theory study that was conducted in those four case companies. The characterization was based on 83 observed daily stand-up meetings in the four companies. An interesting takeaway of Stray's thesis was the categorization of discussions that took place during those 83 observed daily stand-up meetings. The meetings were transcribed and then all statements were grouped into eight categories. This categorization of statements provided a highly interesting insight into how time is spent during the meetings. The eight categories, including their proportions, were (Stray, 2014):

1. Problem-focused communication (35%)
2. The three Scrum questions (24%)
3. Project management (11%)
4. Clarification (8%)
5. Coordination of tasks (7%)
6. Meeting management (6%)
7. Digression (5%)
8. Criterion (4%).

In a more recently published article, Stray, Moe and Sjoberg (2020) updated the categorization by dropping "clarification" and "criterion" and thus, reducing the number of categories to six. Also, case studies on Agile daily stand-up meetings had been continued since the publication of the thesis, increasing the total number of observed and analyzed daily stand-up meetings from 83 to 102. With these updates, the categorization and distribution of discussion topics became (Stray et al., 2020):

1. The three Scrum questions (34%)
2. Problem-focused communication (31%)
3. Project management (15%)
4. Coordination of tasks (9%)
5. Digression (7%)
6. Meeting management (4%).

Within the category "The three Scrum questions", the distribution was (Stray et al., 2020):

1. Have done (66%)
2. Will do (23%)
3. Impediments (11%)

This categorization, along with the proportions, paints a clear picture on what kinds of topics practitioners typically discuss during daily stand-up meetings. Also, it makes it very visible that teams do have various kinds of conversations during those meetings. Clearly, daily stand-up meetings are not solely focusing on the three Scrum questions. Instead, teams utilize the opportunity to discuss all kinds of team related topics. Within the category "the three Scrum questions", the "have done" heavily dominates the conversation. (Stray et al., 2020)

On high level, many software teams seem to commonly follow the daily stand-up meeting practice by Scrum (Mortada, Ayas & Hebig, 2020; Pauly, Michalik & Basten, 2015; Stray, 2014). Scrum-inspired daily stand-up meetings are not uncommon among teams using other software development methodologies than Scrum (Ahmad et al., 2013; Al-Baik & Miller, 2015; Corona & Pani, 2013; Stray, Moe & Dingsøyr, 2011). Teams try to keep the meeting short, ideally in 15 minutes, by strictly focusing on the three Scrum questions: 1) what you have done since the previous meeting, 2) what you will do next, and 3) are there any impediments. However, how exactly daily stand-up meetings are conducted, seem to vary between organizations, and even between teams within same organization (Stray, 2014). Many studies have found daily stand-up meetings to run overtime (Diebold et al., 2015; Mortada, Ayas & Hebig, 2020; Singh & Strobel, 2022). This is especially the case when participants are sitting during the meeting (Stray, Sjøberg, & Dybå, 2016). Also, some studies are reporting teams finding the daily cadence of meetings to be a bit too much and hence, reducing the cadence to every other day or so (Diebold et al., 2015; Pauly, Michalik & Basten, 2015). Team setups, too, vary from team to team. Some teams are fully collocated, some are remote, and some operate in a hybrid mode by having both remote and collocated members part of the team. Especially, in case of remote and hybrid team setups, it is common to utilize video conferencing technology together with digital task boards (Dorairaj et al., 2012; Lous, Tell, Michelsen, Dittrich & Ebdrup, 2018; Paasivaara et al., 2008).

3.2 Benefits

Given the huge popularity of daily stand-up meetings, there must be a multitude of positive effects that comes with the practice. The Scrum authors, Schwaber and Sutherland (2020), claim that the practice improves communications, helps identifying impediments, and promotes fast decision-making. The claims of the Scrum authors do have plenty of backing up by various academics.

One of the key findings of their research, Pikkarainen et al. (2008) found daily stand-up meetings to improve communication both within a development team and between development team and project leader. In the same vein, McHugh, Conboy and Lang (2012) reported about improved communication being one of the benefits of daily stand-up meetings. This has been also confirmed by Pauly et al. (2015) and Sandstø & Reme-Ness (2021), too. In addition to enhanced communications, McHugh et al. (2012) observed daily stand-up meetings to increase transparency, accountability, and knowledge sharing and feedback. Also, Sandstø and Reme-Ness (2021) found daily stand-up meetings to have a highly positive impact on knowledge sharing. Similarly, in a 2007 published paper, Whitworth and Biddle had reported daily stand-up meetings to increase the level of social accountability and support in a project.

Trust is one of the key ingredients of any well performing team. Trust plays an important role in the level of team's independence and the need for project management's support. (Hasnain, Hall & Shepperd, 2013) Various studies have shown daily stand-up meetings

to help building trust among team members. Hasnain et al. (2013) came into conclusion that thanks to increased communication by daily stand-up meetings, Agile teams enjoy higher level of trust among team members. Similarly, McHugh et al. (2012) witnessed daily stand-up meetings to help establishing team cohesion and building trust, especially with distributed teams. Another 2012 study by Dorairaj et al. also reported about similar findings regarding how daily stand-up meetings deepen the trust among members of distributed teams.

Being able to identify and lift impediments is highly important from team's performance standpoint. As the Scrum authors claim, daily stand-up meetings help in identifying impediments. Pauly et al. (2015) confirmed daily stand-up meetings to be a good venue to raise awareness about potential challenges and thus, lower the impact of problems to team's work. Also, Morandini et al. (2020) came into conclusion that daily stand-up meetings can be considered highly important from the viewpoint of solving task related uncertainties. After analyzing dozens of daily stand-up meetings, Stray, Moe and Aurum (2012) found that the single biggest discussion category was indeed problem-focused communication. According to Stray et al (2012), there were three prominent transitions that steered the discussion to problem-focused communication: 1) asking for a clarification, 2) reporting an obstacle, and 3) discussing about how to coordinate tasks. In a 2020 published paper, Stray, Moe and Sjoberg came into conclusion that daily stand-up meetings are required for effective decision making in Agile teams. However, as noted by Fowler (2019), Daily Scrum is not the right venue to discuss about possible solutions to those exposed problems; problem solving should take place after the meeting with smaller audience.

Stray et al. (2012) studied decision making in daily stand-up meetings. As per their analysis, majority of decisions in context of Agile software teams could be carried out in matter of seconds or minutes. That is great news since strictly time-boxed daily stand-up meeting does not lend itself to deep discussions on possible problems. In their study, Stray et al. (2012) viewed the decision-making process through a lens of naturalistic decision making (NDM). NDM assumes that experts can utilize their experience to make good decisions in relatively quickly manner, even with limited background information about the problem at hand. However, as the authors note, utilizing NDM during the daily stand-up meetings comes with certain implications. For instance, teams usually consist of individuals with various levels of skills. Given the NDM relies on highly knowledgeable domain experts, junior team members can not contribute much to the process. This puts emphasis on the importance of organization's capability to turn junior developers into seasoned experts.

One positive aspect of daily stand-up meetings is that it helps new team members with onboarding process (Buchan, MacDonell & Yang, 2019). Buchan et al. (2019) found that new team members value daily stand-up meetings as a venue to learn about their new team. It also presents an opportunity for newcomers to ask questions and get regular support from other team members. In the similar vein, junior developers can find daily stand-up meetings highly beneficial since the meeting provides an opportunity to discuss about possible challenges and to learn about the team's progress. In their research, Stray et al. (2017) discovered junior developers to be more positive toward daily stand-up meetings than their more senior team members. Authors thought this could be the case as the senior developers may already know very well what is going on in the team and thus, perceive the daily stand-up meetings less beneficial. Furthermore, the problems by juniors tend to be quicker to be solved than those of more senior developers.

Teamwork and productive collaboration are highly important part of modern software development projects. Since most software developers work in teams, their work and behavior are impacted and steered by team norms. Team norms are shared expectations towards how to behave and interact within a team. Stray, Fægri, & Moe (2016) found that daily stand-up meetings may help promoting group norms by reinforcing acceptable behaviors. This will help teams to stick to norms and thus, ensure a unified view on desired behavior is shared and put into practice.

Various other positive effects of applying daily stand-up meetings have been reported. Pauly et al. (2015) observed how, thanks to improved communication and transparency, daily stand-up meetings provided a convenient mechanism to remind team members about various kinds of topical matters. The authors also found daily stand-up meetings to be a great source of new ideas as well. Studies have also reported about improved team spirit and cohesion (Pauly et al., 2015; Strode, Dingsøyr & Lindsjorn, 2022).

3.3 Challenges and obstacles

Daily stand-up meetings are no immune to critique. Given how widely used the practice is, and how prominent part of daily work the recurring get-together is for developers, it comes with no surprise that various kinds of challenges can be seen with the practice. Stray et al. (2013) conducted a study to gain knowledge on what kinds of obstacles Agile development teams encounter with their daily stand-up meetings. Even though the sample size of their study was quite small, they were able to extract various challenges that have been later supported by other studies as well. Based on their analysis, in total of 13 obstacles were identified, which were then grouped into four dimensions: 1) temporal, 2) physical, 3) procedural, and 4) attendee.

Various time-related obstacles were identified by Stray et al. (2013). It was quite common that meetings did not start on time. This was typically because participants did not join the meeting on time, which resulted to the Scrum Master or a team member to go out to find the missing participants. Singh and Strobel (2022) and Mortada et al. (2020) recognized unavailability of members as one of the daily stand-up meeting challenges. Missing a daily stand-up meeting can be problematic for various reasons. None of the software teams that were observed by Stray et al. (2012) documented the decisions they had made during daily stand-up meetings. This was not only viewed problematic from the traceability and re-evaluation viewpoint, but also from the standpoint of absent team members. Since decisions were not documented, information about made decisions did not reach those team members who missed the meeting. Also, by not documenting made decisions, re-evaluating those decisions later becomes much harder. Stray et al. (2013) witnessed that quite often meetings did not start or end on time. Furthermore, meetings tend to often run longer than the recommended 15 minutes (Diebold et al., 2015; Mortada et al., 2020; Singh & Strobel, 2022; Stray et al., 2013). Studies also report about the daily cadence being problematic: some perceive the meetings as an interruption (Singh & Strobel, 2022; Stray et al., 2013), some are involved with more than one team and struggle juggling between multiple daily stand-up meetings (Pauly et al., 2015), and some feel it is not feasible to host the meeting daily for a large team (Diebold et al., 2015). Also, some teams struggle to find a suitable time slot for their daily meetings (Stray et al., 2013), or simply have not booked a fixed time slot for the meetings (Mortada et al., 2020). Singh and Strobel (2022) also identified task coordination challenges, stemming from the geographically distributed team setup.

The second dimension of the categorization by Stray et al. (2013) was related to physical aspects of daily stand-up meetings. In their research, all observed teams reported high room temperature and poor air quality to negatively impact the meeting. Also used conferencing equipment were target for criticism. This observation, however, was mainly impacting distributed teams. Similar findings about technology related challenges have been reported by Singh and Strobel (2022), too. Stray et al. (2013) also identified problems regarding space arrangements. They witnessed that in some teams, the Scrum Master was sitting during the meeting while developers had to stand. This was perceived problematic by participants, especially in case the meeting ran severely overtime. Pauly et al. (2015) observed another space arrangement related challenge: due to the daily stand-up meeting room's location, some participants perceived it very inconvenient to get to the meeting room as it took plenty of time to get to the room from where their desk was located. Lastly, in some teams, use of a non-native language has imposed some challenges: for non-native speakers, using a secondary language can make it difficult to express opinions and thus, negatively impacting discussions and the meeting in general (Stray et al., 2013). In similar vein, some daily stand-up participants find it challenging to follow the conversation due to highly complex nature of the topics that are being discussed – this seems to be more of an issue in teams where individuals have very different kinds of expertise or level of knowledge and experience. Some participants have even expressed fear of talking in front of more senior peers during the daily stand-up meetings (Singh & Strobel, 2022).

In the procedural category, there was one particular challenge that seems to be commonly shared by many practitioners. Various studies have pointed out that teams tend to misuse daily stand-up meetings for status reporting (Moe, Dingsøyr & Dybå, 2010; Nyrud & Stray, 2017; Singh & Strobel, 2022; Stray, Moe & Dybå, 2012; Stray et al., 2013). This is especially the case if during the meeting the Scrum Master acts as a manager rather than a meeting facilitator. When daily stand-up meeting turns into a status reporting meeting, participants can lose interest in the meeting due to feeling like the shared information is not relevant to them. Another problem that falls into procedural category is the imbalance of information distribution. Like status reporting problem, issues with information distribution tend to occur due to Scrum Masters taking more role in the meeting than they should. When Scrum Master leads the discussion, some participants might receive more attention than others. This happens when Scrum Master gives more priority to those participants who are working on tasks that the Scrum Master views of higher value or importance. (Stray et al., 2013) Singh and Strobel (2022) raised micro-management as one of the observed challenges. Micro-management was perceived to cause stress for the developers and negatively impacting creativity. This was especially seen to be the case when managers inquired or even shamed the attendees for answers. Also, when participants felt pressure to report status during daily stand-up meetings, the quality of information shared by developers got negatively impacted and hence, causing other participants to view the shared information irrelevant for them.

Attendee related challenges was the fourth category, identified by Stray et al. (2013). As the number of attendees increases, so does the overall time spent in the meeting. Also, when the number of team members goes up, so does the likelihood of team members working on very different kinds of tasks. This can lead into situation where team members perceive the information, shared by others, irrelevant to them (Stray et al., 2013). Various other studies, too, have found too many participants to have a negative effect on daily stand-up meetings (Diebold et al., 2015; Morandini, Coleti, Oliveira & Corrêa, 2021; Singh & Strobel, 2022; Stray et al., 2017). Another challenge, observed by Stray et al. (2013), is negative meeting attitudes by meeting participants. While the authors were not able to observe the consequences of negative meeting attitudes, they were referring to

existing literature on how negative meeting attitudes affect the perception of meeting's effectiveness and team's potency. Also, the authors pointed out that negative attitudes may cause unwanted impact on one's job satisfaction. In a later grounded theory study, Stray et al. (2016) identified various aspects that have a negative impact on meeting attitudes. Indeed, many of the previously mentioned challenges in this chapter are causing attitudes towards daily stand-up meetings to deteriorate.

4.   Findings and discussion

There is no doubt that the daily stand-up meeting practice enjoys a huge popularity and is highly adopted practice among software organizations (Digital.ai, 2022; Singh and Strobel, 2022; Stray et al., 2017). Even though being an Agile practice, it is commonly used also by non-Agile software teams (Stray et al., 2017). In fact, the daily meeting practice has spread from software teams to other functions; it has been adopted by non-software teams, such as Human Resources (HR) and Marketing, too (Digital.ai, 2022).

Teams usually seem to follow the Daily Scrum format with their daily stand-up meetings by focusing on the three Scrum questions and targeting for a short, 15-minute meeting (Stray et al., 2016). However, various studies have reported that it is quite common for teams to deviate from what The Scrum Guide mentions about the Daily Scrum (Diebold et al., 2015; Pauly et al., 2015). Deviating from The Scrum Guide seem to mainly happen due to practical reasons, such as being able to accommodate different needs of team members, or to make the practice to better fit for team's way of working.

Customizing the meeting to fit to the team's work practices and operational context is not necessarily a bad thing. Quite the opposite. In their article, "Daily Stand-Up Meetings: Start Breaking the Rules!", Stray et al. (2020) highly encouraged practitioners to find out what works the best for their team and adjust the daily stand-up meetings accordingly. The authors also provided various suggestions on what could be changed and taken into consideration. In their grounded theory study, Stray et al. (2016) presented a recommendation to completely omit the first Scrum question, "what did I do since the previous meeting?", to reduce the likelihood of turning the meeting into a status reporting meeting. Interestingly, Stray et al. (2016) were not the only ones seeing the problem: the Scrum authors Schwaber and Sutherland removed the three Scrum questions altogether in their most recent, 2020 published revision of The Scrum Guide (West, 2020). The latest revision of The Scrum Guide now instructs developers to "*select whatever structure and techniques they want, as long as their Daily Scrum focuses on progress toward the Sprint Goal and produces an actionable plan for the next day of work.*" (Schwaber & Sutherland, 2020).

The benefits of conducting daily stand-up meetings are plentiful and come with highly positive impact. Fluent communication is crucial part of teamwork. Various studies reported improvements in both team's internal and external-facing communications (McHugh et al., 2012; Pauly et al., 2015; Pikkarainen et al., 2008; Sandstø & Reme-Ness, 2021; Stray et al., 2016). Also, transparency (McHugh et al., 2012; Sandstø & Reme-Ness, 2021), accountability (McHugh et al., 2012; Whitworth & Biddle, 2007), and knowledge sharing (McHugh et al., 2012) were seen to improve, thanks to daily stand-up meetings. Possibly due to these positive effects, onboarding new team members has been reported to improve, too (Buchan et al., 2019). From the behavioral standpoint, daily stand-up meetings can provide great benefits to teams. Rooting and enforcing desired behavior (Stray et al., 2016), promoting better team spirit (Pauly et al., 2015) and cohesion (McHugh et al., 2012; Strode et al., 2022), and increasing the level of trust within the team (Dorairaj et al., 2012; Hasnain et al., 2012; McHugh et al. 2012) were reported to be positively affected by daily stand-up meetings. From the execution perspective, teams need be able to react to possible challenges early to ensure commitments can be kept. Having a venue to raise and discuss about problems and uncertainties has been found to be highly appreciated benefit of daily stand-up meetings (Morandini et al., 2020; Pauly et al., 2015; Stray et al., 2012).

While numerous positive effects were associated with daily stand-up meetings, teams did experience some challenges with the practice as well. Literature reported about various kinds of challenges, though not nearly all were specific to daily stand-up meetings. For instance, things such as meetings starting or ending late, running overtime, or high room temperature and poor ventilation causing discomfort among participants would be annoyances for participants of any type of meeting. However, these kinds of challenges were found to increase negative attitudes towards the daily stand-up meetings and hence, having a negative impact on various meeting activities (Stray et al., 2016). Temporal challenges were widely represented in the literature. One of the most reported challenges was that the meeting often ran overtime (Diebold et al., 2015; Mortada et al., 2020; Singh & Strobel, 2022; Stray et al., 2013). Studies were also reporting practitioners finding the daily cadence of meetings to be too frequent. Some viewed the meeting as an interruption to their work (Singh & Strobel, 2022; Stray et al., 2013), and some belonged to multiple teams and had therefore troubles attending to all those daily stand-ups (Pauly et al., 2015). Also, bigger teams felt the daily cadence not being feasible since the meeting took much more time than the recommended 15 minutes (Diebold et al., 2015). According to The Scrum Guide (Schwaber & Sutherland., 2020), *"The purpose of the Daily Scrum is to inspect progress toward the Sprint Goal and adapt the Sprint Backlog as necessary, adjusting the upcoming planned work."*. However, literature reveals that too often Scrum Masters shift the focus of the daily stand-up meeting from inspection and adaptation to status reporting (Moe et al., 2010; Nyrud & Stray, 2017; Singh & Strobel, 2022; Stray et al., 2012; Stray et al., 2013). This is a common anti-pattern and should be avoided. In fact, using daily stand-up meetings for status reporting seems to be one of the biggest mistakes one can make with the daily meeting practice, since by doing so, negative attitudes towards daily stand-up meetings will evidently increase (Stray et al., 2016). Furthermore, when daily stand-up meetings were used for status reporting, participants felt the shared information was not relevant to them and they lost interest towards the meetings (Stray et al., 2013).

Based on these findings from prior research and literature, a clear picture starts to emerge on what makes the daily stand-up meeting practice so widely adopted by software organizations, despite some negative attitudes towards the practice. Even though many recent studies have reported dissatisfaction among practitioners towards the practice, the positive aspects clearly overweight the negative side. Given the notable benefits, teams do not want to drop the practice, even though some challenges are experienced as well. Especially since many of the reported challenges could be classified rather annoyances than notable problems that would impact teams' performance or capability to deliver. Also, quite a few of the mentioned challenges are not strictly specific to daily stand-up meeting practice – many of the challenges are most likely seen in other meetings, too. Based on the literature, it also became very clear that daily stand-up meetings are widely mis-used purposefully or due to lack of understanding on what the purpose of the meeting really is. And last, from the wide adoption standpoint, one cannot ignore the fact that the daily stand-up meeting is one of the first steps towards becoming Agile, making the practice highly exposed to practitioners.

5.   Conclusion

The intent of this thesis was to gain understanding on how practitioners are conducting daily stand-up meetings, and how the extremely widely adopted daily meeting practice is perceived by professionals. By conducting a literature review, the goal was to learn especially about the positive and negative aspects of daily stand-up meetings to gain understanding on what makes the daily stand-up meeting practice so popular among software organization. The research question of this thesis was "*Given the amount of reported dissatisfaction towards daily stand-up meetings, why is the practice so widely adopted by Agile software development teams?*". Being able to answer the question, an understanding about how teams are conducting daily stand-up meetings in practice had to be acquired; what are the characteristics of the daily stand-up meeting in Agile teams? Then, a solid understanding about both the benefits and the challenges had to be formed.

When analyzing both positive and negative aspects of the daily stand-up meeting practice, one thing became quite apparent: the positive aspects tend to be outcomes of the meeting practice, whereas many of the negative aspects seem to stem from external factors and are surfaced during the meeting. For instance, positive aspects, such as improvements in communication, team cohesion, trust, and accountability can be witnessed due to the daily stand-up meeting practice. These positive aspects influence how team operates and performs also outside of the daily stand-up meeting context. However, many of the negative aspects, such as participants skipping the meeting, meetings starting or ending late, or managers re-purposing the meeting for their own needs, are results of external factors. Those external factors obviously introduce challenges that are negatively impacting the perception of daily stand-up meetings. Of course, many of those challenges do understandably cause negative attitudes towards the daily stand-up meeting practice but they are not limited to daily stand-up meetings per se. That is, those same things would be challenges with any type of meeting. This does not mean that all those negative aspects wouldn't affect the team outside of the daily stand-up meeting context, though many of them have somewhat limited impact – especially when contrasting with the positive aspects.

Quite a few of the challenges, presented in the literature, seem to stem from culture. Some from organizational culture, others from societal culture. For example, if it is common in an organization that meetings run overtime, or participants arrive late or skip meetings without a notice, one can most certainly expect those to happen with daily stand-up meetings, too. In similar fashion, team dynamics often differ between egalitarian and hierarchical societies. Juniors feeling afraid to speak in front of more senior colleagues is something I have witnessed often in countries with hierarchical society, but to a significantly lesser extent in countries with egalitarian society. These challenges are not because of the daily stand-up meeting, they exist for other reasons and just become visible with the practice.

One of the most prominent problems with daily stand-up meeting is the fact that many teams use it for reporting status. Previous studies have identified several reasons for why that happens. Two common reasons are that 1) managers turn the meeting into status reporting by accident or on purpose, and 2) practitioners do not fully understand the essence of daily stand-up meetings. When daily stand-up meeting turns into a status reporting meeting, some of the original benefits are lost while plenty of new challenges will emerge. This seems to be so common problem that even Scrum authors reacted to it and completely removed the three Scrum questions from The Scrum Guide in 2020.

Methods and guidelines are not set in stone. For instance, The Scrum Guide has been refined and updated a few times after its introduction in 2010. On the one hand, it is great that updates are made when a need arises. On the other hand, it requires practitioners to be aware of the updates, and especially to understand the reasoning behind done changes. I, for one, was not aware of the removal of the three Scrum questions in 2020. This was slightly surprising since being both a certified Scrum Master and a certified SAFe Agilist, I feel I have quite decent grasp on Agile methodologies, and especially on Scrum. After discussing with several people, asking if they were aware of the removal, it became apparent I was not the only one who had completely missed this information. In fact, none of the people I talked with had heard about the change.

This literature review comes with some limitations and is far from being systematic or repeatable. Even though a serious effort has been put into finding as many articles as possible that talk about daily stand-up meetings, some relevant articles have most likely been missed. Also, reader should be aware that the author of this thesis has been exposed to Agile methodologies, and especially Scrum, for almost the past two decades. Having been in thousands of daily stand-up meetings, I cannot claim to be unbiased. On top of that, for several years, I have been working for Digital.ai (former CollabNet / VersionOne), which is one of the biggest commercial entities in this field. Their studies, namely the *State of Agile* report, has been referred to in numerous studies that were part of my literature review.

When going through previous studies, one cannot help but notice how the research during the past 10 years in this domain has been conducted by quite limited number of researchers. Also, the number of case studies, and especially the number of case companies, is still quite small, even though a great effort and contributions has been made by a few highly active researchers. However, during the past 10 years the daily stand-up meeting practice has been studied from various perspectives and through numerous lenses. That said, it would be highly interesting to see a systematic literature review done on this topic.

References

Abrahamsson, P., Salo, O., Ronkainen, J. & Warsta, J. (2002) Agile software development methods: Review and analysis, VTT publication 478, Espoo, Finland, 107p.

Ahmad, M. O., Markkula, J., & Oivo, M. (2013, September). Kanban in software development: A systematic literature review. In *2013 39th Euromicro conference on software engineering and advanced applications* (pp. 9-16). IEEE.

Al-Baik, O., & Miller, J. (2015). The kanban approach, between agility and leanness: a systematic review. *Empirical Software Engineering*, *20*(6), 1861-1897.

Asnawi, A. L., Gravell, A. M., & Wills, G. B. (2012). Emergence of agile methods: Perceptions from software practitioners in Malaysia. In *2012 Agile India* (pp. 30-39). IEEE.

Beck, K. (1999). Embracing change with extreme programming. *Computer*, *32*(10), 70-77.

Beck, K. (2000). *Extreme programming explained: embrace change*. addison-wesley professional.

Beck, K., Beedle, M., van Bennekum, A., Cockburn, A., Cunningham, W., Fowler, M., Grenning, J., Highsmith, J., Hunt, A., Jeffries, R., Kern, J., Marick, B., Martin, R. C., Mellor, S., Schwaber, K., Sutherland, J. & Thomas, D. (2001). Manifesto for Agile Software Development. Retrieved May 13, 2022. from https://agilemanifesto.org/

Buchan, J., MacDonell, S. G., & Yang, J. (2019). Effective team onboarding in Agile software development: Techniques and goals. *2019 ACM/IEEE International Symposium on Empirical Software Engineering and Measurement (ESEM)*, 1–11.

Corona, E., & Pani, F. E. (2013). A review of lean-kanban approaches in the software development. *WSEAS transactions on information science and applications*, *10*(1), 1-13.

Diebold, P., Ostberg, J. P., Wagner, S., & Zendler, U. (2015, May). What do practitioners vary in using scrum?. In *International Conference on Agile Software Development* (pp. 40-51). Springer, Cham.

Digital.ai, 2022. 15th Annual State Of Agile Report. https://digital.ai/resource-center/analyst-reports/state-of-agile-report (accessed March 2022)

Dorairaj, S., Noble, J., & Malik, P. (2012, May). Understanding team dynamics in distributed agile software development. In *International conference on agile software development* (pp. 47-61). Springer, Berlin, Heidelberg.

Fowler, F. M. (2019). Navigating Hybrid Scrum Environments. Apress Berkeley, CA.

Griffin, L. (2021). Implementing lean principles in scrum to adapt to remote work in a Covid-19 impacted software team. In *International Conference on Lean and Agile Software Development* (pp. 177-184). Springer, Cham.

Hasnain, E., Hall, T., & Shepperd, M. (2013). Using experimental games to understand communication and trust in agile software teams. In *2013 6th International Workshop on Cooperative and Human Aspects of Software Engineering (CHASE)* (pp. 117-120). IEEE.

Highsmith, J., & Cockburn, A. (2001). Agile software development: The business of innovation. *Computer*, *34*(9), 120-127.

Karlström, D., & Runeson, P. (2006). Integrating agile software development into stage-gate managed product development. *Empirical Software Engineering*, *11*(2), 203-225.

Kuhrmann, M., Diebold, P., Münch, J., Tell, P., Garousi, V., Felderer, M., Trektere, K., McCaffery, F., Linssen, O., Hanser, E., & Prause, C. R. (2017). Hybrid software and system development in practice: Waterfall, scrum, and beyond. *Proceedings of the 2017 International Conference on Software and System Process*, 30–39.

Lous, P., Tell, P., Michelsen, C. B., Dittrich, Y., & Ebdrup, A. (2018). From Scrum to Agile: a journey to tackle the challenges of distributed development in an Agile team. In *Proceedings of the 2018 International Conference on Software and System Process* (pp. 11-20).

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

Moe, N. B., Dingsøyr, T., & Dybå, T. (2010). A teamwork model for understanding an agile team: A case study of a Scrum project. *Information and software technology*, *52*(5), 480-491.

Morandini, M., Coleti, T. A., Oliveira Jr, E., & Corrêa, P. L. P. (2020). Considerations about the efficiency and sufficiency of the utilization of the Scrum methodology: A survey for analyzing results for development teams. *Computer Science Review*, *39*, 100314.

Mortada, M., Ayas, H. M., & Hebig, R. (2020). Why do software teams deviate from scrum? reasons and implications. In *Proceedings of the International Conference on Software and System Processes* (pp. 71-80).

Newkirk, J. (2002, May). Introduction to agile processes and extreme programming. In *Proceedings of the 24th International Conference on Software Engineering. ICSE 2002* (pp. 695-696). IEEE.

Nyrud, H., & Stray, V. (2017). Inter-team coordination mechanisms in large-scale agile. In *Proceedings of the XP2017 scientific workshops* (pp. 1-6).

Ohno, T., & Bodek, N. (2019). *Toyota production system: beyond large-scale production*. Productivity press.

Overhage, S., & Schlauderer, S. (2012). Investigating the long-term acceptance of agile methodologies: An empirical study of developer perceptions in scrum projects.

In *2012 45th Hawaii International Conference on System Sciences* (pp. 5452-5461). IEEE.

Paasivaara, M., Durasiewicz, S., & Lassenius, C. (2008). Using scrum in a globally distributed project: a case study. *Software Process: Improvement and Practice*, *13*(6), 527-544.

Pauly, D., Michalik, B., & Basten, D. (2015). Do daily Scrums have to take place each day? A case study of customized Scrum principles at an e-commerce company. In *2015 48th Hawaii International Conference on System Sciences* (pp. 5074-5083). IEEE.

Pikkarainen, M., Haikara, J., Salo, O., Abrahamsson, P., & Still, J. (2008). The impact of agile practices on communication in software development. *Empirical Software Engineering*, *13*(3), 303-337.

Rising, L., & Janoff, N. S. (2000). The Scrum software development process for small teams. IEEE software, 17(4), 26-32.

Sandstø, R., & Reme-Ness, C. (2021). Agile practices and impacts on project success. *Journal of Engineering, Project, and Production Management*, *11*(3), 255-262.

Schwaber, K., & Sutherland, J. (2020). The scrum guide. *Scrum Alliance*.

Singh, K., & Strobel, J. (2022). Exploring lived experiences of agile developers with daily stand-up meetings: a phenomenological study. *Behaviour & Information Technology*, 1-21.

Stray, V. (2014). *An Empirical Investigation of the Daily Stand-Up Meeting in Agile Software Development Projects*.

Stray, V. G., Lindsjørn, Y., & Sjøberg, D. I. (2013). Obstacles to efficient daily meetings in agile development projects: A case study. In *2013 ACM/IEEE International Symposium on Empirical Software Engineering and Measurement* (pp. 95-102). IEEE.

Stray, V. G., Moe, N. B., & Aurum, A. (2012). Investigating daily team meetings in agile software projects. In *2012 38th Euromicro Conference on Software Engineering and Advanced Applications* (pp. 274-281). IEEE.

Stray, V. G., Moe, N. B., & Dingsøyr, T. (2011). Challenges to teamwork: a multiple case study of two agile teams. In *International conference on agile software development* (pp. 146-161). Springer, Berlin, Heidelberg.

Stray, V. G., Moe, N. B., & Dybå, T. (2012). Escalation of commitment: a longitudinal case study of daily meetings. In *International Conference on Agile Software Development* (pp. 153-167). Springer, Berlin, Heidelberg.

Stray, V., Fægri, T. E., & Moe, N. B. (2016). Exploring norms in agile software teams. In *International Conference on Product-Focused Software Process Improvement* (pp. 458-467). Springer, Cham.

Stray, V., Moe, N. B., & Bergersen, G. R. (2017). Are daily stand-up meetings valuable? A survey of developers in software teams. In *International Conference on Agile Software Development* (pp. 274-281). Springer, Cham.

Stray, V., Moe, N. B., & Sjoberg, D. I. (2020). Daily stand-up meetings: start breaking the rules. *IEEE Software*, *37*(3), 70-77.

Stray, V., Sjøberg, D. I., & Dybå, T. (2016). The daily stand-up meeting: A grounded theory study. *Journal of Systems and Software*, *114*, 101-124.

Strode, D. E. (2006). Agile methods: a comparative analysis. In *Proceedings of the 19th annual conference of the national advisory committee on computing qualifications, NACCQ* (Vol. 6, pp. 257-264).

Theocharis, G., Kuhrmann, M., Münch, J., & Diebold, P. (2015). Is Water-Scrum-Fall Reality? On the Use of Agile and Traditional Development Practices. In P. Abrahamsson, L. Corral, M. Oivo, & B. Russo (Eds.), *Product-Focused Software Process Improvement* (pp. 149–166). Springer International Publishing.

Wells, D. (1999). Daily Stand Up Meeting. http://www.extremeprogramming.org/rules/standupmeeting.html (accessed May 2022).

West, D. (2020). Scrum Guide 2020 Update - What has been removed. https://www.scrum.org/resources/blog/scrum-guide-2020-update-what-has-been-removed (accessed June 2022).

Whitworth, E., & Biddle, R. (2007). The social nature of agile teams. In *Agile 2007 (AGILE 2007)* (pp. 26-36). IEEE.

Strode, D., Dingsøyr, T., & Lindsjorn, Y. (2022). A teamwork effectiveness model for agile software development. *Empirical Software Engineering*, *27*(2), 56.