REALIZING THE POTENTIAL OF RETROSPECTIVES WITH TEAM REFLEXIVITY

An intervention study to improve reflection and planning in evaluation meetings of Scrum teams

01-10-2018

Cosima Patzak – s1385771

EDUCATIONAL SCIENCE AND TECHNOLOGY

EXAMINATION COMMITTEE:

Dr. Maaike Endedijk, m.d.endedijk@utwente.nl

Marijn Wijga MSc, m.wijga@utwente.nl

UNIVERSITY OF TWENTE.

Acknowledgement

I am proud to present to you my Master's thesis. Finalizing this thesis is the last step to graduating from the Master program "Educational Science and Technology" at the University of Twente. Only through the encouragement of numerous parties, critical reflection and unrelentless hard work has this study found its current format. And for that, I would like to thank a number of people.

First of all, thanks goes to my supervisors from the University of Twente, Marijn Wijga MSc and Dr. Maaike Endedijk. Marijn, your enthousiasm, your detailed feedback and our countless sessions have made working with you a real pleasure. Maaike, your experience and insight into Educational Design have given this study more depth than I had ever imagined. Both of you have helped me to develop myself as a researcher and an Educational Designer. For that, I am very grateful.

This study could not have happened without the time and courage of all of the 32 team members who let me observe their retrospectives. Your openness, to a let an unknown student inspect and improve your retrospectives, has moved me. I want to thank you for all these sessions, during which I have gotten lots of insights into Scrum, software development and the struggle with continuous improvement. I would also like to thank Frank, who helped with gathering the data and always had an open ear. Your support and guidance have helped me to carry out this project and learn a great deal.

Also, I want to thank Prof. Celeste Wilderom, the heart and brains behind the 'Change Leaders' Honours trajetory. You have encouraged me to choose this project to challenge myself and given me the tools to support this process. Also, you made it possible for me to be coached by the hardworking Kirsten who has often helped me create order from chaos.

Finally, I want to thank my family and friends who always were there for me. Franziska, you have reviewed many early drafts and helped me find my voice. Dominik, you have stuck with me through the thick, knowing that I had to see this through. All of you have listened and helped, even when some of it was sometimes difficult to understand.

This thesis is the result of my studies in the Netherlands, which have allowed me to learn so much and get an inspiring fresh look at another culture. As a proud European citizen, I am happy to have gotten the chance to study abroad and will remember this time fondly.

Thank you all for making this possible.

Cosima Patzak

September 2018

Table of Contents

Summary

Nowadays, the tasks of software development (SD) teams have become increasingly complex. In that environment, the quality of teamwork is vital for the effectivity of SD teams. This has led to the use of agile SD methods, which engage collaborative teamwork methods and develop software in shorter iterations. Scrum is the most common agile SD method, and regularly evaluates the development process during a 'retrospective' meeting during which improvement goals (AP's) are set. However, although Scrum teams often report problems, they cannot effectively respond to these issues. The goal of this study is to support Scrum teams from a Dutch SD firm with an intervention to improve the team reflexivity. Team Reflexivity is "the extent to which team members collectively reflect upon the team's objectives, strategies and processes" (West, 1996, p. 559). The first two stages of team reflexivity, reflection and planning, as well as ways to improve team reflexivity were identified as important. The research question was: *How can we design an intervention that supports Scrum teams to enhance team reflexivity during their retrospectives?* This study employed a quasi-experimental control group design with a pre-test and a post-test. Three SD Scrum teams participated, team A and C in the experimental condition and team B in the control group. Per team, five video observations of the retrospectives were made. The intervention was a one-hour training aiming to: give the teams feedback, teach them how to make SMART AP's and practice making SMART AP's. The effect of the intervention was evaluated with a questionnaire and an interview. Also, the behavioural changes of participants were identified by coding reflection episodes, wrap-ups, action points and the quality of the planning, with the coding program Observer XT. Statistical analysis with SPSS revealed that episodes ended with a wrap-up in 11.11% up until 25% of the episodes. Also, 28 AP's with varying degrees of SMART were identified, the most found SMART criteria were 'Relevant' and 'Specific'. The quality of planning increased for all participating teams, probably because of a new Scrum Master with other facilitation methods, which made it difficult to apply SMART correctly, which has led to team A and C applying SMART less strictly. Also, incorporating SMART into practice takes time. Yet, both Scrum Masters agreed that they would continue to apply SMART. This study has found that there is room for improvement in the continuous improvement through retrospectives, as a lot of examples of low quality team reflexivity were identified. Also, the intervention was not enough to solve the team's issues with continuous improvement as there were only small, brief changes. More information is needed to see real change. This study extends the conceptual and practical knowledge of team reflexivity and ways to improve team reflexivity in the real-life work environment of agile SD teams. Limitations of this study are small sample size, short duration of the training and limited generalizability. Also, further studies on qualities of reflection, adaption process of AP's and role of the Scrum Master could provide additional insights.

Keywords: retrospectives, team reflexivity, wrap-up, reflection, SMART goals, intervention

Nowadays, the tasks of software development teams (SD teams) have become increasingly complex.

Their working environment can change rapidly with the introduction of innovative products. In addition, SD teams also must be able to effectively integrate different types of knowledge of several highly specialized team members (Moe, Dingsøyr & Dybå, 2010). Previous studies have shown that when a task is novel, complex and uncertain, the way people work together is crucial (Hoegl, Parboteeah & Gmuenden, 2003). This is particularly true for innovative projects where tasks are often associated with unknown technologies that make it difficult to predict the future and unforeseen developments (Hoda, Noble & Marshall, 2012). Therefore, the quality of teamwork (e.g., regular and effective collaboration, as well as continuous improvement) is vital for the effectivity of SD teams in mastering these increasingly complex future tasks (Hoegl & Gmuenden, 2001; Hoegl et al., 2003; Moe, 2013).

As a consequence, an increasing number of SD teams started to make use of Scrum. Currently, *Scrum* is widely used to develop new software through shorter iterations while more effectively managing the complex environment of SD teams (Chow & Cao, 2008). Essential to these *Scrum teams* is that their development process is regularly evaluated. For instance, at the end of each iteration, the team is asked to hold a 'retrospective' meeting. These retrospective meetings serve to create an opportunity for reflexivity among team members to assess what went well, what needs to be improved and what actions need to be taken to improve the problems (Dybå, & Dingsøyr, 2008). Because these meetings are held regularly, they are an essential part of the continuous improvement of the Scrum teams (Salo & Abrahamsson, 2007). Scrum belongs to the 'agile' SD methodologies. Agile is an umbrella term for specific types of SD methods that are developed to deliver new software faster and to increase customer satisfaction by avoiding unnecessary documentation and actively involving stakeholders and customers in collaborative teamwork (Hoda et al., 2012; Dingsøyr, Nerur, Balijepally & Moe, 2012).

Yet, previous studies on SD teams working with Scrum showed that although teams often report problems during retrospectives, they cannot effectively reflect and respond to these issues (Moe, 2013; Ringstad, Dingsøyr & Moe, 2011; Salo & Abrahamsson, 2007). In some cases, the topics that are discussed during the retrospectives are too complex to be easily solved (Lehtinen, Itkonen & Lassenius, 2017). In other cases, the team does not come up with a good enough alternative because team members prefer to give their opinion, instead of the hard evidence (McAvoy and Butler, 2007; Nerur, Mahapatra & Mangalari, 2005). On other occasions was observed, that some teams did not spend enough time to reflect on improving their work or failed to discuss apparent problems (Stray, Moe & Dingsøyr., 2011).

It appears that there is a need for methods to support the continuous improvement in Scrum teams. At a Dutch SD firm, three Scrum teams also struggled to realize the potential of the retrospective. The goal of this study is to support these teams with an intervention to improve the team reflexivity, by improving the quality of their planned actions, so they are better equipped to solve their problems.

2. Theoretical Framework

2.1 Continuous improvement in self-managing teams

Scrum teams are *self-managing*, which means that these teams are responsible "not only for executing their tasks but also for monitoring and managing their own performance" (Moe, Aurum & Dybå, 2012, p. 844). 'Self-management' has also been described as a strategy for learning and improvement of teamwork as the team members are empowered to make decisions on the operational level (Dybå, Dingsøyr & Moe, 2014; Salo & Abrahamsson, 2007). Self-management can also lead to more satisfied employees, lower turnover and lower absenteeism (Cohen & Bailey, 1997). Moreover, self-management has also been associated with more effective teamwork and innovation (Hoegl & Parboteeah, 2006).

In order to establish fully self-managing Scrum teams, prerequisites need to be met by these teams (Kirkman & Rosen, 1999; Moe, 2013). One of the two most important prerequisites is that the decision making power should be shared in self-managed teams. Joint decision-making can be beneficial because it involves many stakeholders, which in turn can improve the quality of the product (Moe et al., 2012). However, shared decision-making power can also bring with it some difficulties, as it can lead to ineffective decisions in cohesive groups due to deadlock situations, which is often referred to as 'groupthink' in the literature (McAvoy & Butler, 2009; Moe, 2013; Nerur et al., 2005; Ringstad et al., 2011). *Groupthink* is a "psychological drive for consensus at any cost that suppresses disagreement and prevents the appraisal of alternatives" (McAvoy and Butler, 2009, p. 374). Groupthink may occur in very cohesive teams because team members are not willing to examine problem causes, as they want to avoid interpersonal conflict and prefer to conform to other team members, rather than speaking up. As a consequence, a groupthink situation can lead to ineffective decision making (McAvoy & Butler, 2007).

The second most important prerequisite is that Scrum teams also need to be familiar with and follow the Scrum method, in order to be successful (Chow & Cao, 2008). The Scrum method, the most common agile SD method, is in use since the late 1990ies (Chow & Cao, 2008). The term originates from Rugby, where it is used for a team of eight individuals that huddle together to get the ball across the field. Scrum is especially useful for gradually developing software in complex, quickly changing environments (Rising & Janoff, 2000). The teamwork in Scrum is organized with guidelines from the Scrum Guide, with assigned roles and meetings, that each fulfil a specific purpose (Schwaber & Sutherland, 2013). In Scrum, the main working unit is the cross-functional Scrum team. According to Kim (2007), the most important roles in the Scrum team are the *Scrum Master*, *Product Owner* and members of the *Development Team*. The Scrum Master ensures that all the members of the Scrum team understand the principles of Scrum and act accordingly (Dingsøyr & Lindsjørn, 2013). The Product Owner is responsible for the quality of the final product. The development team is a self-organizing, cross-functional team, whose members develop the product (Schwaber & Sutherland, 2013).

Scrum teams work with fixed, time-boxed events to structure their activities (Dybå & Dingsøyr, 2008). The software is developed in short development cycles, the sprints, which last around three to four weeks. Also, the team frequently meets in various kinds of meetings. Each meeting serves a specific purpose, such as planning, monitoring of execution or refining the planning. In the Scrum method, the most important meeting for continuous improvement is the retrospective. It is essential for Scrum teams to engage in retrospective meetings to stay learning and improve continuously (Lehtinen et al., 2017; Moe, 2013). That is because the retrospectives offer an opportunity for the Scrum team to inspect itself by identifying and discussing obstacles as well as feelings and to create a plan for improvements. This plan is usually formulated as a list of action points (AP's) that are executed during the next sprint. Also, analysing previous AP's and diagnosing underlying issues are an essential part of retrospectives (Andriyani, Hoda & Amor, 2017). Because these meetings are held regularly, they are an essential part of the continuous improvement of SD teams, also referred to as software process improvement (Salo & Abrahamsson, 2007).

It seems that if the prerequisites for fully functioning self-management, shared decision-making and adherence to the Scrum method, are not met, problems arise in the meetings, such as groupthink and participation bias. Also, studies have reported that team members can become frustrated with the retrospectives because they do not see results, which has often led to them skipping the retrospectives (Moe, 2013; Salo & Abrahamsson, 2007; Stray et al., 2011). In such cases, it is only logical that the teams are not able to realize the potential of the retrospective and improve their teamwork. However, literature is missing on methods to support the teamwork processes in agile teams. That is why, in the next section, this study will investigate underlying team processes that can explain how the improvement process takes place in agile teams.

2.2 Team reflexivity

A fitting way to explain how continuous improvement in agile SD teams takes place is the theoretical concept of *team reflexivity*, which is defined as "the extent to which group members overtly reflect upon, and communicate about the group's objectives, strategies (e.g. decision-making) and processes (e.g. communication) and adapt them to current or anticipated circumstances" (West, 2000, p. 296). Team reflexivity consists of three stages that together form an iterative process of improving team performance, which fits the retrospective practice well (Vlietland, Solingen & Vliet, 2016; Yu & Petter, 2014).

The three stages of team reflexivity are reflection, planning and action (West, 2000). During *reflection*, the team as an entity overtly explore work-related issues (Schippers, Hartog & Koopman, 2007). According to West (2000) reflection includes a range of behaviors such as "questioning, planning, exploratory learning, analysis, diversive exploration, making use of knowledge explicitly, planfulness, learning at a meta-level, reviewing past events with self-awareness, and coming to terms over time with a new awareness" (p.4). In *planning*, "goals are presented and ways to achieve these

goals are planned" (Widmer, Schippers & West, 2009, p. 3). Planning can function as a bridge between reflection and action because these plans to achieve the goals are consequently implemented in the action phase (Widmer et al., 2009). The third stage is *action*, during which the team shows goal-directed behaviors to achieve "the desired changes in team objectives, strategies, processes, organizations or environments identified by the team during the stage of reflection" (Widmer et al., 2009, p. 3). If the actions lead to new information about the issue, this can, in turn, lead to more reflection, planning and action. In that case, team reflexivity is an iterative and ongoing process (West, 2000).

Team reflexivity is a fitting way to explain how continuous improvement in agile SD teams takes place because it also engages an iterative feedback loop, such as Scrum, and is beneficial to deal with their complex tasks. Specifically, research indicates that engaging in team reflexivity is important for innovative teams, as they need to deal with a lot of uncertainty in their work (Hoegl & Parbooteah, 2006; Müller, Herbig & Petrovic, 2009; Schippers, West & Dawson, 2015; Tjosvold, Tang & West (2004). Reflexive teams are particularly well suited to deal with this environment, since the team "is more likely to be questioning and tackling challenges produced by the continuously changing environment of innovative projects" (Hoegl & Parboteeah, 2006, p. 118). Team reflexivity is also essential to improve team performance and learning (Albrechtsen & Hoyden, 2010; Gurtner, Tschan, Semmer & Nägele, 2007; Matthew & Sternberg, 2009; Widmer et al., 2009). Team reflexivity facilitates team learning as it "refers to the team's action of reflecting on the current reality and on how to adapt to the current and future reality to achieve the team goals" (Raes, Boon, Kyndt & Dochy, 2015, p. 478). This makes sense since, in order to learn effectively, teams need to experience that their learning has consistently led to them reaching their goals. Otherwise, the teams lose the motivation to engage in learning at all, as observed in the case of several agile teams (Moe, 2013; Ringstad et al., 2011).

2.3 Improving team reflexivity

Many of the problems SD teams face in retrospectives are complex and involve careful coordination and communication (Lehtinen et al., 2017). Good team reflexivity seems to be essential to cope with this. For instance, Ringstad et al. (2011) found that planning is often inefficient in agile teams and used planning to improve teamwork quality. Also, Moe and Dingsøyr (2008) found that Scrum teams often deviated from the Scrum practice and failed to plan in the long term and handle problems. Therefore, the next sections give an overview of relevant ways to improve team reflexivity.

2.3.1 Giving feedback

Team reflexivity can be improved by giving *feedback* (Müller et al., 2009). Presenting and discussing strengths and weaknesses of the teams can initiate reflection, since this knowledge allows the team to utilize their strengths and work on their weaknesses and therefore allows for adaptation of their strategies (Matthew & Sternberg, 2009). To ensure that the feedback is effective, the feedback should be of a high quality. To be of a high quality, the feedback needs to be accurate and specific as well as given in a non-threatening way. What is more, high-quality feedback can best lead to performance

change, when it is given before a team activity in which the team plans improvement goals and acts upon the goals (Gabelica, Van den Bossche, Maeyer, Segers & Gijselaers, 2014; Schippers et al., 2007). The feedback should also be given at the team-level, as it is most effective when it is directed at only one level, individual or team, and not both (DeShon et al., 2004; Kleingeld, Mierlo & Arends, 2011). Moreover, having cooperative goals can contribute to team reflexivity, more than with individual or competitive goals (Tjosvold et al., 2004). Therefore, the team should best be considered as a unit when giving the feedback.

2.3.2 Improving reflection

The team reflexivity can also be improved by enhancing a separate stage of team reflexivity, in this case, reflection. Studies on reflection have made a distinction between implicit knowledge and explicating that knowledge during reflection activities (Mathew & Sternberg, 2009; Müller et al., 2009). To elaborate, according to Müller et al. (2009), implicit knowledge is either individual knowledge, that is not available to the team because individuals do not share the knowledge, or that task relevant aspects are not communicated in the team. This study builds on Müller et al.'s (2009) differentiation between implicit knowledge and explicating that knowledge. Moreover, this study proposes that teams in retrospectives show either low quality reflection or high quality reflection, which differ through if the team explicates implicit knowledge or not.

To improve the quality of reflection, Lehtinen et al's (2017) have found that it is important that the team actively reflects on the knowledge that otherwise would remain implicit, especially in relation to goals that are ill-defined and lack a protocol. Since many software process improvement problems are complex and involve careful coordination and communication, high quality reflection seems to be essential (Moe et al., 2013). Reflection can be elicited during an intervention activity and several studies have found that it can be especially beneficial for innovative teams (Albrechtsen & Hovden, 2010; Gurtner et al., 2009; Mathew & Sternberg, 2009; Müller et al., 2009). For instance, Müller et al. (2009) found that teams developed more innovative products after either explicating implicit individual or team knowledge. Additionally, Mathew and Sternberg (2009) found that reflection improves practical problem solving skills, especially when limited information about the situation is available. These findings indicate that engaging in reflection could be advantageous for solving the numerous complex problems that Scrum teams face in retrospectives.

Further, studies show that, in many cases, reflection during retrospectives can be improved (Andriyani, Hoda & Amor, 2017; Lehtinen et al., 2017; Moe, 2013; Stray et al., 2011). It can be done by first, identifying the underlying problems and second, voicing contradictory or unpopular opinions to avoid groupthink. For instance, Stray et al. (2011) have found that often teams do not hold retrospectives or fail to discuss apparent problems. In such cases, it is clear that the teams did not use the opportunity to reflect or that the quality of their reflection was not high enough to address the real problems. Yet, if the teams do not reflect on the background information of the problem, Schippers et

al. (2015) pose that in such cases, it is most likely that their solution is merely a superficial and ineffective 'easy fix'. Then, in all probability, the problems will arise again during the next sprint, and the team will become unmotivated to hold retrospectives because they cannot see the results of their efforts (Stray et al., 2011). What is more, reflection activities could help to avoid groupthink. Studies have shown that in cohesive groups, it is especially important to voice contradictory or unpopular opinions that go against the group consensus. During reflection activities, it could be easier to play the 'devil's advocate', and to identify the actual problem and its influencing factors (McAvoy & Butler, 2009; Müller et al., 2009).

Another possible way to improve reflection is for the team to end discussion with a conclusion. Formulating conclusions to reflections can help to affect action because an agreement is an indication for the creation of mutually shared cognition in the team, also referred to as a shared mental model (Raes et al., 2015; Van den Bossche et al., 2006). By having a shared mental model, teams can realize their goals easier, since it creates a framework that helps teams understand what they need and why for solving the problem (Dybå et al., 2014). Only with this shared understanding can the team effectively apply their teamwork skills and learn from their mistakes (Ringstad et al., 2011; Salas, Sims & Burke, 2005; Stray et al., 2011).

2.3.3 Improving planning with SMART

The planning stage can be improved by improving the quality of the goal. The quality of goals is so important because clear goals, as well as performance feedback, motivate employees (Lawlor & Hornyak, 2013; Locke, 1968; Volet & Mansfield, 2006). Moreover, having specific, difficult goals leads to higher group performance than nonspecific goals (Kleingeld, Mierlo & Arends, 2011). This finding has led to the conceptualization of the five SMART criteria for goal-setting (Locke & Latham, 1990). The five *SMART criteria* allow for a critical review of the goal in regard to 1) how *specific* the goal is (what to reach, who is involved and why to reach it), 2) if reaching the goal is *measurable* (if the goal is formulated so that progress can be detected), 3) if the goal is *attainable* (given the team's abilities, resources and previous experiences), 4) if the goal is *relevant* to the team and involved parties and 5) if the goal is *timely* (there are steps which can be done at a specific point in time and the execution of the goal has a beginning and end).

Studies have shown that SMART is an effective tool to improve goal-setting in various types of contexts. The acronym has been used successfully in businesses and for training interventions in SMART goal-setting for health clinicians, patients with chronic diseases and student teams (Lawlor & Hornyak, 2013; Marsland & Bowman, 2010; Monaghan, Channell, McDowell & Sharma, 2004; Swanson, 2016). These studies have shown that individuals and teams that are able to formulate their goals SMART, were more motivated and achieved better results by basing their decision making on solid arguments (Lawlor & Hornyak, 2013). Marsland and Bowman's (2010) study has shown that

SMART goal formulation can lead to the formulation of more concrete and feasible AP's, although some delay of effects and follow-up support was required to make significant changes.

2.4 Current study

At a Dutch SD firm, three Scrum teams struggled to realize the potential of the retrospective. The goal of this study is to support these teams with an intervention. Previous studies have found that in Scrum teams, the potential of the retrospective is often not fully realized because, although teams managed to identify the problems, they did not act upon them (Moe, 2013; Ringstad et al., 2011; Salo & Abrahamsson, 2006; Stray et al., 2011). It appears that there is a need for methods to support the continuous improvement in Scrum teams. This study has identified team reflexivity as a fitting way to explain the continuous improvement process of Scrum teams during retrospectives (Albrechtsen & Hoyden, 2010; Andriyani et al., 2017; Gurtner et al., 2007; Hoegl & Parboteeah, 2006; Schippers et al., 2015; Tjosvold et al., 2004; Widmer et al., 2009). Furthermore, this study has identified ways that can improve team reflexivity such as giving feedback, the quality of reflection, conclusions of discussions and quality of planning with SMART criteria (Gabelica et al., 2014; Matthew & Sternberg, 2009; Müller et al., 2009; Raes et al., 2015; Salas et al., 2005; Swanson, 2016). Having a clear and feasible starting point for improvement could enable the teams to act upon the problems, before the routine work of the sprint takes over again and the improvement efforts are pushed into the background. Therefore, the goal of this study is to support these teams with an intervention to improve the team reflexivity, by improving the quality of their planned actions, so they are better equipped to solve their problems. In order to fulfil this goal, this study aims to answer the following research question:

- How can we design an intervention that supports Scrum teams to enhance team reflexivity during their retrospectives?

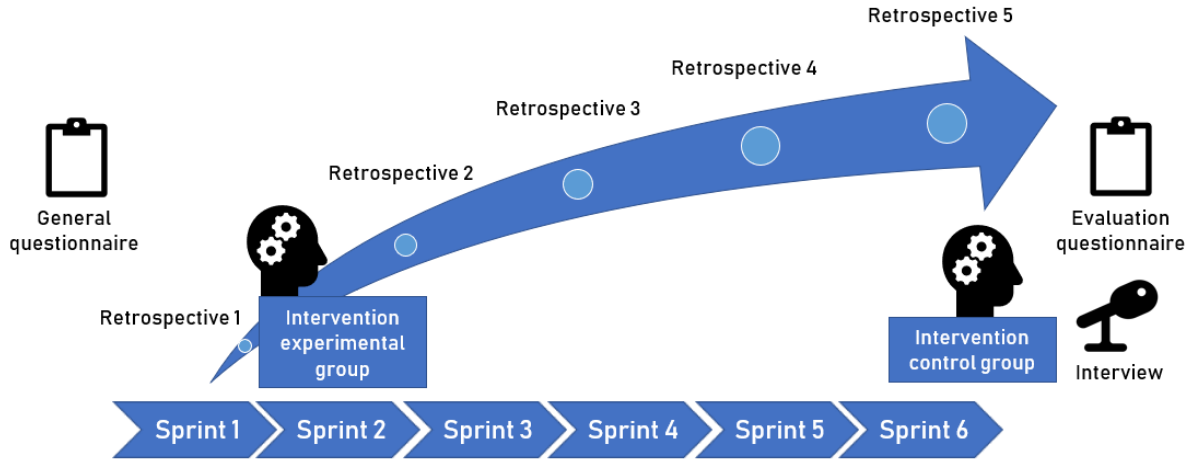
3. Method

3.1 Design

This study is an intervention study with a quasi-experimental control group design and a pre-test and a post-test. It set out to teach three Scrum teams to support their continuous improvement, with an intervention to improve team reflexivity, so they are better equipped to solve their problems. The goal of the study was to be achieved by following these steps:

Data was collected to test the effect of the intervention. Five video observations of the retrospective were made per team, two questionnaires were distributed and two interviews were conducted (see Figure 1). Three teams participated, two in the experimental condition and one in the control group.

Figure 1. Overview over sprints, observation moments, treatments and other measurements.



3.2 Participants

Respondents were collected at a Dutch software company. In the three SD teams (Team A, B, and C) 32 team members participated in total (at the beginning of the study: M_{age} = 41.29, age range: 28 - 53[1]). Whereas, the teams A and C participated in the experimental condition, team B was in the control condition[2]. On average, participants had been a member of the team for 16.04 months (ranging from 0.25 to 108 months). Participants were for a large part male (92.63%) and 9.38% female. The common level of education was higher vocational training (50%) and 46.9% completed academic education programs[3]. The roles of participants in their team were mostly developers (37.5%) and Product Owners

[1] Four participants did not include information about their age.
[2] One person was a member of all participating teams
[3] One person did not answer the question about the level of education.

(28.1%). Remaining roles were testers (9.4%), Scrum Masters (6.3%), Scrum Coaches (3.1%) and Other[4] (15.6%). The team meetings were conducted in Dutch.

3.3 Description of teams

Team B and C had a designated Scrum Master. Team A distributed the Scrum Master responsibilities between team members and switched responsibilities every two sprints. Team A was the oldest and largest team that participated in this study (see for an overview Table 1). However, this could have been due to the many changing Product Owners (6 Product Owners) that are involved in this team, given the nature of their tasks. Their tasks included managing large architectural landscape with numerous software applications. Due to that, the tasks of this team were often unpredictable and were frequent coordination and upkeep of the software landscape important. Team B was the most recently formed team. This team had one dedicated Product Owner who was present during all of the retrospectives. Team C was the smallest, especially since their Product Owners were often not present during retrospectives.

Table 1

Relevant demographics of participants per team

	Team A	Team B	Team C
Members team	15	11	8
Runtime project	2 – 3 years	0.5 years	1 – 2 years
Mean length of membership of team members	2.21 years	4.4 months	11.57 months
Average age team members	40.55 years	43.4 years	39.43 years

3.4 Intervention

The main goal of the intervention was to teach the teams to use the full potential of the retrospective, in order to solve their problems by formulating AP's that can affect change. This problem was put into the focus of the intervention, creating an opportunity to coach the teams on their specific reasons that hindered them to plan their improvement. The goal of the intervention was reached, by realizing three training steps during a one-hour training. The training steps were:

a) give the teams feedback on how they run their retrospectives,

b) teach them how to make SMART AP's, and

c) practice making SMART AP's.

In the following, the training is described by specifying the three steps and the materials and activities that were used during the steps. The first step was, to give feedback on the way the teams run their

[4] Participants included in this category were for example Business Analyst, Functional Developer, Technical lead and Software Delivery Manager.

retrospectives and illustrate it with a video sequence of the team itself. The first step was chosen to give feedback during the training in order to facilitate reflection about the way the teams runs their retrospective. The researcher chose feedback to illustrate the strong and weak points of each team. The feedback was based on literature about typical issues in retrospectives. The researcher's feedback was adjusted, per team, to the different kind of challenges that the teams experienced. The researcher further illustrated this feedback by showing to the teams a short video sequence about the things that the team could change (see logbooks). The team members were then asked to watch the sequence and report what they saw afterwards. This activity was chosen to further facilitate reflection on the way the teams run their retrospectives.

The second step was to teach the teams SMART, in order to improve their team planning. The SMART-roadmap (see Appendix X) was used to teach how to improve the quality of a goal. To teach SMART, the two-page SMART roadmap was presented to the team members. Additionally, every step in the roadmap and its importance was explained. The premise was that, given the tools to critically challenge AP's with SMART, the teams would be better equipped to determine feasibility of their goals and to concentrate on fewer but more concrete goals.

The third step was to practice making SMART AP's, by applying the SMART roadmap to a familiar, real-life topic. The team picked one topic from APs they formulated during their previous retrospective. Under the supervision of the researcher, they discussed the topic and made a SMART AP. The researcher encouraged all team members to participate in roughly the same amount. This team activity was done so the team could practice and experiment with the newly learned theory. Also, the researcher could create an environment for intelligent failure, by ensuring that the method was used as intended (Cannon & Edmondson, 2005). This way, if the team used SMART incorrectly, the mistake was detected early on and the team received feedback.

3.5 Measures

3.5.1 Video observations

The video observations were made with 360 degree cameras. A 360 degree camera can record sound and all visual input. This way, every activity during the meeting was recorded relatively non-intrusively and processes were measured objectively. The goal of gathering video observations was to determine behavioural changes of the participants on the third level of evaluation, the behaviour (Kirkpatrick, 1996).

3.5.2 Interview

A semi-structured interview was conducted, with each Scrum Master of the teams in the experimental condition. The goal of the interview was to first conduct an evaluation on Kirkpatrick first level of evaluation, the reaction level (Kirkpatrick, 1996). Second, the goal was to examine possible reasons for why the intervention has or has not achieved the expected effect. Third, the goal was to assess

(anticipated) long-term effects of the intervention. The interview was transcribed and episodes of relevant utterances were identified.

3.5.4 Questionnaires

The participants filled in a general questionnaire about relevant demographics such as age, gender, length of membership in team, runtime of the project and role in the team. After the training, the teams in the experimental condition also filled in an anonymous questionnaire to evaluate the intervention anonymously (see Appendix C). Completing the second questionnaire was voluntary.

3.5.4 Logbooks

Several logbooks were kept to document and reflect on the process. Most importantly, routines and typical issues in retrospectives were documented, as well as observations about the strong and weak points of each team.

3.6 Procedure

First, the elements of team reflexivity that needed improvement were studied. For that, a literature review was conducted. The researcher then applied to the ethics committee of the university for ethical approval of the study (including design, participants, sampling procedures, treatment, instruction of participants and informed consent). The application was approved.

Participants were recruited in collaboration with contacts at the Dutch software firm who had been involved in previous studies. The contacts encouraged employees of the company to attend a presentation of the intervention study. The first supervisor and the researcher gave the presentation and explained the background, research aim and procedure of the study. The attendees could ask questions and look at the 360 degree cameras. Also, privacy considerations were addressed as well as the question of how much time participants would need to invest for participating in the study. Furthermore, the attending teams were told that a Scrum team could only participate in the study when every single team member agreed to participate. The participating team members were given the chance to ask remaining questions about the study. Finally, all participants signed their informed consent forms and filled in a general questionnaire.

A Pilot test was performed with team A to practice recording with the 360 degree cameras so that it would be as unobtrusive as possible. After that, the official data collection started. All gathered data were saved on several encrypted external hard drives and the names of the participants were anonymized. Meanwhile, the choice was made to focus on the planning stage of team reflexivity since a satisfactory amount of information was found about planning with SMART. After that, the training materials were developed by the researcher. The training was set up to be as time-efficient as well as effective as possible, which is why only a one-hour training session was conducted per team. The intervention was designed around the elements that are feasible to change in the given timeframe. The action stage of team reflexivity was excluded, because it is not part of the retrospective, since the goals

are executed during the following iteration cycle and not during the retrospective. The researcher arranged a room at a fitting time spot for the training. The intervention was executed shortly before the third observed retrospective, for teams in the experimental condition.

Finally, the researcher developed an interview scheme for a semi-structured interview with the Scrum Master and an anonymous questionnaire for the rest of the team to evaluate the treatment (Appendix B and C). It was chosen to only include participants from the treatment condition since only they received the treatment and could evaluate it. The interview was conducted shortly before the last video recording. Before beginning the interview, the researcher asked for permission to record the interview and explained that the name of the interviewee would be anonymized. After the interview, an anonymous questionnaire was distributed to all team members via email. Completing the questionnaire was voluntary.

3.7 Evaluation of the effect of the intervention

The effect of the intervention was evaluated with a summative evaluation (Dick, 2002). The behavioural changes of the participants were determined analysing the 360 degree video recordings with qualitative and quantitative data analysis. The reactions of participants to the intervention were assessed with an interview of the Scrum Masters and an anonymous questionnaire.

3.7.1 Data analysis

Three video recordings per team were analysed with the coding scheme that can be found in the next section. The video recordings were analysed by following several steps. First, the data was selected in which the teams were reflecting on a topic. Some examples of low quality and high quality reflection were identified during that process, but this was just a way to select the data for the next steps. The second step was to code if the reflection was concluded with a wrap-up. Third, the moments in which the teams were by formulating AP's were identified. Finally, the quality of the planning was assessed by coding how many SMART criteria could be assigned to an AP. The video coding program The Observer XT was used for coding the video data. The researcher had previously been trained in the use of Bron and Endedijk's (2016) coding scheme for episodes and wrap-ups by one of the researchers herself.

The expected effect of the intervention is, that the quality of AP's of teams in the treatment condition increases after the intervention and the quality of AP's of the team in the treatment condition remains the same over time. To test the expected effect, a statistical data analysis of the codes was done with IBM SPSS Statistics 25. First, frequencies and sums of episodes, wrap-ups, AP's and SMART criteria were calculated per team and observation. Also, the percentage of wrap-ups per episode was calculated and a graph was made. Then, the degree of SMART (DoS) was determined, by first summarizing how often a SMART criterion was given per AP. Then, the means of the DoS for all AP's formulated during the retrospective was calculated. A graph was made that shows the mean DoS per

team over the observations. In order to test the expected effect, differences of scores were calculated, by creating three new variables for the changes in DoS scores. The differences of scores from three observation moments were compared over the treatment conditions, similar to Marsland and Bowden's (2010) study. The goal was to test for a) short-term changes (comparing Pretest and Posttest 1 score), b) effects over time (comparing Pretest with Posttest 2 score) and c) the declining of effects of the intervention over time (comparing Posttest 1 and Posttest 2 score).

3.7.2 Codebook

This codebook was used to code the video observations of the retrospectives.

- First, reflection episodes were identified. An episode is a sequence of utterances that are about the same subject, social talk is excluded (Molenaar, 2011; Wijga & Endedijk, 2017). Reflection was operationalized as an episode since the main purpose of the retrospective is to reflect upon what went well, what went bad and what can be improved. Reflection during retrospectives is defined as a team engaging in reflective behaviours, such as sharing and discussing information and insights about what went well, what went bad and what can be improved. Examples for reflective behaviours are "questioning, planning, exploratory learning, analysis, diversive exploration, making use of knowledge explicitly, planfulness, learning at a meta-level, reviewing past events with self-awareness, and coming to terms over time with a new awareness" (West, 2000, p. 4).
- In this step, the existence and kind of a conclusion of a discussion, called wrap-up, was identified (Raes et al., 2015; Bron & Endedijk, 2016). Typically, the wrap-up is formulated at the end of the episode since, per definition, it is the conclusion of a discussion. Yet, it is also possible that a wrap-up about one topic is being made during the next episode. In that case, the wrap-up was coded during the next episode and received a comment indicating that the wrap-up refers to the earlier episode. A difference was made between 1) no wrap-up, and b) wrap-up. The wrap-up was later specified in a) a postponed wrap-up, b) a cognitive wrap-up or c) an action wrap-up. If a wrap-up existed, the utterances constituting the wrap-up were assigned the corresponding wrap-up code. Also, the person doing the uttering was coded. For this step, the descriptions of the wrap-ups from Bron and Endedijk's (2017) coding scheme were used (see Table 2).

Table 2

The different types of wrap-ups and their description

Type of wrap-up	Description
No wrap-up	When none of the team members ends the topic with some kind of agreement or conclusion.
Postponed wrap-up	When there is an agreement to postpone a decision about the topic of discussion is expressed.
Cognitive wrap-up	When an agreement or decision about the topic has been reached, and is explicitly summarized or repeated as a conclusion of the discussion about the topic.
Action wrap-up	When there is a conclusion of the discussed topic and there is an explicit intention for action expressed.

- In this step, the borders of each AP were identified. In the Scrum method, planning takes place by formulating AP's about future improvements, which is why the AP's were coded (Schwaber & Sutherland, 2013). Decisive for determining the AP was the topic of the episode, as formulated by the team, as well as the decision of the team to make an AP out of the topic and to act upon it. The decision to make an AP out of the topic should have already been identified as an episode by the first coding step, which served as basis for identification of the AP. This was included because, sometimes, team members formulate an AP, but it is not taken up because it would involve too much work. If there was no repetition of the AP at the end of the retrospective, the coding of the AP as an AP was removed. Another issue was that sometimes, during retrospectives, an AP had several subparts, or sub-to do's, to achieve a big main goal. The talking about the big main goal AP was coded as one AP, in case the fulfilling of all the criteria applied to all of the subparts. That means, that agreements were made which applied to all sub-goals, such as when a point in time for solving all the agreements was made. If the agreements regarding the big goal did not apply to all sub-goals, the AP's were coded separately.
- During this final step, the quality of planning was assessed, by determining the degree of SMART (DoS). The DoS was determined by judging if an AP fulfilled all of the five SMART criteria that were necessary for the AP to be considered SMART (see Table 3). The AP was judged on if it fulfilled a criteria by first writing down the topic of the AP. Then, the utterances of the team members about the AP were observed closely. When an utterance fulfilled a criterium, this utterance received the corresponding code. A similar method, named SMART goal evaluation method, has been used in the clinical health sector, to judge the quality of written goals (Bowman, Mogensen, Marsland & Lannin, 2015).

Table 3

The five SMART criteria and their description

SMART criteria	Description
Specific	A concrete description of the goal, answering at least three of the five questions: What do we want to reach? Who is involved in reaching the goal? When will we begin actions to reach the goal? Where do we need to be to reach the goal? Why do we want to reach this goal?
Measurable	It is measurable if the goal has been reached or not. For that, the goal has to be formulated, so that progress can be detected.
Acceptable	Reaching the goal is realistic for the team, given the team's skills, resources and previous experiences with similar goals.
Relevant	The goal is relevant to the team (and possibly others) since it is related to other important and relevant goals.
Timely	The execution of the goal has a clear beginning and end.

- If the topic of the AP has uncertain or controversial elements, this could make it impossible for the team to make it completely SMART. To be precise, if it is not possible to determine if the AP is or is not SMART because of external reasons, this could have prevented the team to apply their SMART skills and would skew the data. Therefore, AP's about such topics will receive the code "uncertain". For example, determining the timeliness of a goal is difficult when attaining the goal is dependent on other factors such as the clients wishes or funding. Uncertainness of the AP is determined by if team members disagree about e.g. the attainability of the goal or grave concerns about other aspects of the AP which are not in the power of the team to change.

4. Results

This study set out to support three Scrum teams with an intervention to improve team reflexivity so they are better equipped to solve their problems. The goal of this study was to be reached by following three steps. The first step was to study elements of retrospectives that needed improvement. The literature revealed that team reflexivity is a fitting way to explain how the improvement process takes place in agile teams. The second step was to design an intervention. The intervention aimed to elicit reflection on the way the teams run their retrospectives and to teach the teams to improve the quality of their AP's with SMART. Finally, the third step was to test the effect of the intervention. The effect of the intervention was evaluated, in regard to a) the reactions of participants to the intervention, such as usability or anticipated long-term effects, b) if the intervention was successful in achieving the expected effect, and c) possible reasons for why the intervention might not have achieved the expected effect. The expected effect of the intervention was that it would improve the quality of the AP's from the teams participating in the experimental condition. Inferences were drawn from the analysis of video recordings from the retrospectives of the teams. To that end, nine hours and seventeen minutes of 360 degree video recordings were analysed. In the beginning, the data was analysed by selecting data through identifying episodes in which the teams were reflecting on a topic. Some examples of low quality and high quality reflection episodes were identified during that process, as a way to select the data for further analysis. Then, it was examined if the reflection episode ended with a wrap-up and what percentage of episodes ended with a wrap-up. Also, the AP's of the teams were identified. Finally, the quality of the AP's was assessed by coding how many SMART criteria could be assigned to an AP. To illustrate the coding process, examples were identified of a cognitive wrap-up and AP's with both a low and a high quality.

4.1 Reflection

During the first step of the data analysis, the data was selected in which the teams were reflecting on a topic. A total of 253 reflection episodes were identified (see Table 6). Some examples of low and high quality reflection were identified by differentiating between if the team explicated implicit knowledge or not. This excerpt was chosen from the first retrospective of team C that was observed during this study. This excerpt illustrates how low quality reflection can look like in retrospectives (see Table 4).

First, a discussion about the overarching issue of underestimating the complexity of some stories is initiated. The team reflects shortly on how they noticed the issue during the past sprint, on first ideas for how to solve it and how it might be important in the future. Yet, these reflections are quite superficial and short as the team members hop from one aspect of the issue to another aspect of the issue without exploring the various aspects in-depth. To elaborate, first, the team talks about how some stories were not analysed well and that they need to take more time for that. Yet, they do not elaborate, for example mention how exactly they are going to use the time they are going to take for analysing, what activities to do and why. Then, they swiftly talk about involving Tom for functional things and how that went right during the brainstorm. Yet, then, they switch to the next aspect and talk about the

functional changes of xyz. To conclude, no discussion develops about the various aspects but rather, they simply agree with each other and continue with the next aspect.

Table 4

Example of low quality reflection

Subject name	Utterance	Utterance number
Tilman	And then I also had underestimating the complexity of some stories. That is for the xyz[5] stories.	1
Scrum Master	Yes.	2
Tilman	Apparently, those were not analysed well.	3
Jaime	Yes, that was like that in the whole document, with the yzx too. Then you are scanning it a little bit and you think, yeah, the impact is about this and that, and in the end, other things still join.	4
Tilman	Yes.	5
Scrum Master	I think that we need to take some more time for that.	6
Tilman	Yes. More time and maybe we can also involve Tom in it, for example for functional things.	7
Jaime	Yes.	8
Scrum Master	That is true.	9
Jaime	Yes, like yesterday indeed, I think that was right during the brainstorm.	10
Scrum Master	Yes, that it is, sure. *thinks* It is going to become interesting because he has not been there the whole time.	11
Jaime	No.	12
Scrum Master	With the coupling, there are going to be new things all around, that is going to be interesting.	13
Tilman	For example, the xyz brings some functional changes with it and so on. *pauses* Shall I already do the green?	14

The utterance that directly follow the example in Table 4, however, show how differently reflection can be. The end of the first example is that Tilman asks if he should continue with another issue, from the "green" post-it notes. Yet, in the next example, which can be found in Table 5, the Scrum Master (Steve) does not agree with Tilman to continue with a whole new issue. Instead, he engages in a higher quality of reflection. Steve continues to talk more about the first issue, the problem of underestimating the complexity. The higher quality reflection can be observed by Steve showing the following reflective behaviours: First, he connects the problem to implicit team knowledge, namely the feedback that the Scrum Coach gave them during their last retrospective. Second, he questions the previous behaviour of the team with self-awareness, as he says that they tend to prefer to do the "building, building and building. That is what you like to do so to say". Finally, Steve explores alternatives to their previous behaviour and comes to the conclusion that the team needs to reserve more time for analysing the stories and not only building the stories. These utterances shows that he reflected on the knowledge differently than before, by engaging in reflective behaviours of a higher quality. Steve

[5] Names of stories or other names of software tools were given a codename to assure anonymity of participants.

made a conscious decision to dive deeper into the topic. If he had not insisted on further reviewing the past events, the retrospective would probably have continued with other topics and the high quality of reflection would not have occurred.

Table 5

Example of high quality reflection

Scrum Master	Yes, no, in fact, I wanted to continue to talk about this. You said about yzx that we do not take enough time for it and I think also think that is the case. What has triggered me, so to say, is, Fred[6] also had it about that: for us, the pressure has been too high, or we are busy with too many things, and these kinds of things are not paid enough attention to. So I think that we are keeping ourselves indeed too busy by bringing the user stories still into the sprint. That is why I think that we need to be careful in order to not be too quick to add another story. We need to make sure that we have time left to think of these kinds of things and not only for building, building and building. That is what you like to do so to say. But I think that if we take notice to analyse in time, that building will also go more smoothly.	15
Jaime	Yes.	16
Scrum Master	I think that it is a lesson for us too, to see if we manage to not work on that much. So that we can have the things around in order, too.	17

Table 6

A summary of the number of episodes per team, number of different kinds of wrap-ups, total of wrap-ups and percentage of wrap-ups per episode

	Team A	Team B	Team C
Retro 1			
Episodes	36	44	16
Action wrap-up	0	3	1
Cognitive wrap-up	4	3	1
Postponed wrap-up	0	0	1
Total wrap-ups	4	6	3
Percentage of wrap-ups per episode	11.11%	13.64%	18.75%
Retro 3			
Episodes	20	35	26
Action wrap-up	2	0	0
Cognitive wrap-up	2	3	4
Postponed wrap-up	1	4	1
Total wrap-ups	5	7	5
Percentage of wrap-ups per episode	25%	20%	19.23%
Retro 5			
Episodes	15	39	22
Action wrap-up	2	3	1
Cognitive wrap-up	0	4	2
Postponed wrap-up	1	1	1
Total wrap-ups	3	8	4
Percentage of wrap-ups per episode	20%	20.51%	18.19%

4.2 Wrap-ups

An indication of effective reflection is the existence of a conclusion, also called a wrap-up. During the coding, a total of 43 wrap-ups were identified (see table 6). The amount of wrap-ups was compared with the amount of episodes by calculating a percentage of wrap-ups per episodes. This revealed a relatively low percentage of wrap-ups per episode, ranging from 11.11% up to 25% (see Figure 2). The data shows varying changes of the percentage of wrap-ups per episode as the changes in percentages are different

[6] Fred is one of the Scrum Coaches at the software development company and was asked to be present during the previous retrospective to give feedback.

for each team. For team A, the percentage of wrap-ups per episode strongly increases, at first, then it declines again. For team B, the percentage also goes up, but then it remains relatively stable. Meanwhile, for team C, the percentage remains roughly the same over the course of the study.

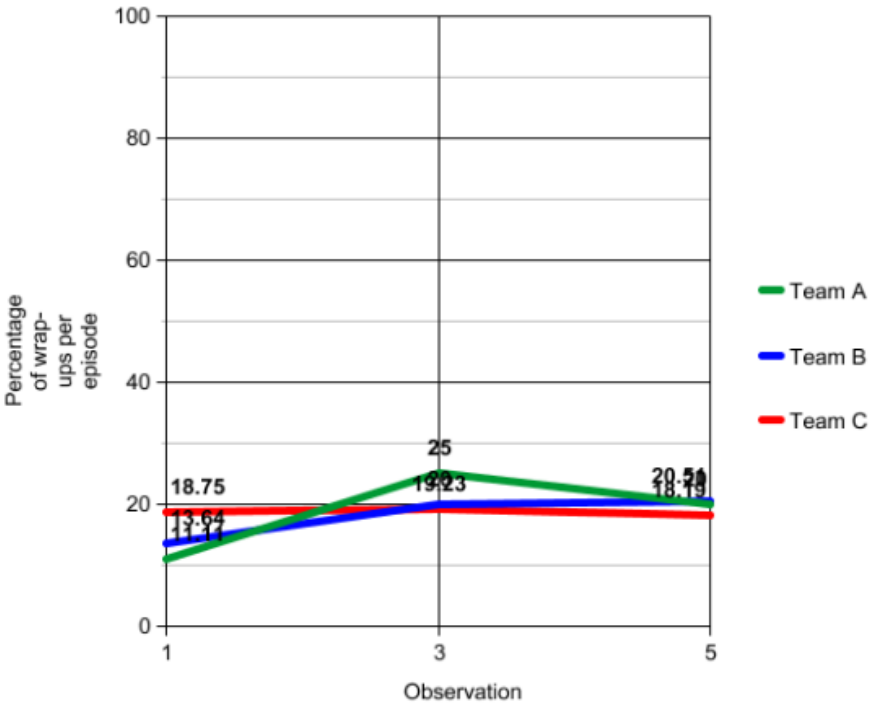


Figure 2. The percentage of wrap-ups per episode per team and observation moment.

The following example from team C illustrates how a cognitive wrap-up can look like (Table 7). These utterances begin where the utterances from Table 6 end. The team continues to engage in the topic. Finally, the team agrees that this topic is important and the knowledge is jointly summarized in a cognitive wrap-up, from the second utterance on, until the end. The agreement of the team members makes it clear, that the knowledge is being accepted by the team members when Jaime agrees and even joins in with formulating the wrap-up.

Table 7

Example of a cognitive wrap-up

Subject name	Utterance	Utterance number
Tilman	Yes. Does that mean that we will also take on fewer stories? *The discussion goes on about if that means taking on less or more stories and other examples of the topic*	1
Scrum Master	But I see that as a danger to take on more user stories because the whole process around it also has to keep on going.	2
Jaime	Otherwise, it will become sloppy.	3
Scrum Master	Yes. And we need to watch out for that.	4
Jaime	Ok.	5

4.3 The effect of the intervention

4.3.1 Planning and SMART criteria

Planning was identified as the most important element of retrospectives that needed improvement. Therefore, the intervention focused on improving the quality of planning. In retrospectives, planning takes the form of an AP. In total, 28 AP's were identified. The quality of planning was determined by how many of the SMART criteria can be assigned to each AP (see Table 8). It is notable, that the most often found SMART criteria were 'Relevant' and 'Specific', while 'Timely' rarely was assigned.

Table 8

The frequencies of SMART criteria per team and observation

	AP's per Retro	Specific	Measurable	Acceptable	Relevant	Timely
Team A						
Retro 1	5	4	1	5	4	2
Retro 3	2	2	1	2	2	2
Retro 5	3	3	3	3	3	3
Team B						
Retro 1	4	4	3	3	4	1
Retro 3	1	1	1	0	1	1
Retro 5	3	2	2	1	3	0
Team C						
Retro 1	5	0	1	0	1	0
Retro 3	1	1	1	1	1	1
Retro 5	4	4	4	3	4	1
Totals	28	21	17	18	23	11

4.3.2 Examples of planning with AP's

Several examples of AP's, as observed during this study, can be found in this section. These examples highlight the varying quality of planning, by mentioning the corresponding SMART criteria in the code column. Table 9 shows the formulation of two AP's with a low quality, by team C. These AP's were formulated during the first retrospective that was observed. The 'AP' Code in the right column indicates the beginning of a new AP. In these examples, it is notable that the Scrum Master is the only one busy formulating the AP, while talking quietly, with the back to the team, while the rest of the team says nothing. Then, he is interrupted by another person that begins talking about another, related topic and the Scrum Master stops formulating the AP. The old topic is also not taken up at another moment. The team proceeds to talk about the new topic and the AP of the previous topic only can be given the SMART criteria Relevant.

Table 9

Example of two AP's with a low quality

Subject name	Utterance	Code
Scrum Master	So, actually *goes to the board and writes what he quietly says* we need to watch out as a team that we have time for the analysis and involve Tom. *writes*	AP, Relevant
Tilman	Actually, I find the allocation of roles also unclear. That is why, I cannot remember the name, Jan?	
Jaime	Jim	
Scrum Master	Jim, ye	
Tilman	Jim, yes. He first does the analysis for the financial process and that is what he is discussing with Tom. I find it a bit vague where the problem lies.	
Scrum Master	Yes, that is true. Tom is, of course, the communicator here he goes between us and him and what he talks about with Jim that communication runs a bit hurried which is why it does not arrive at us.	
Jaime	Yes, that is because Tom is not on our team.	
Scrum Master	No, but he is still in our refinement session. Yes, I think that I need to go to him about that. He is sitting there almost always.	
Jaime	Yes, that is true.	
Scrum Master	Yes, and when they do have these meetings, they might come up with more stories. I hope that this is not the case.	
Jaime	Yes.	
Scrum Master	Then that is the way it is.	
Jaime	Ok.	
Scrum Master	Then we have here *writes*	AP
Jaime	Less story points maybe?	Measurable
Scrum Master	Yes. *Silence for 50 seconds or inaudible* Looping back to maybe. I still need to write all of that down. An architect could do this differently then.	AP
Jaime	Yes.	
Scrum Master	Maybe we can do another action point, to be sure, for the retrospective. The right communication of a retrospective in any case. Jon will be here next time in any case.	
Jaime	Mm-hm.	
Scrum Master	We agreed on that one on Monday.	
Jaime	Yes.	

This example is typical of some of the struggles of team C which were mentioned in section 4.1. These findings also apply here to how team C formulates their AP's. They stay on a superficial level and quickly go from one topic to the next one. The Scrum Master seems to try to improve the AP for some time, however, his formulation process is broken off by a Tilman, when he introduces another topic. Because of that, this example of planning results in several AP's with a low quality.

The second example was chosen because, during this meeting, the team manage to formulate a high-quality AP (see Table 10). This example is from the third retrospective, which was conducted shortly after the training and under the guidance of a new Scrum Master (Anthony). The following utterances begin after the Scrum Master had asked for the team to present possible AP's, even when

they are not completely good yet. Especially the utterances after utterance six are interesting here, as the Scrum Master asks what it means to 'be attentive'. He asks the team to explore their understanding of attentive, to explicate their implicit knowledge of what it means to be attentive. That is advantageous because this explicit knowledge then can be utilized to make a very concrete AP that in this case fulfils all the SMART criteria. For example, by explicating that attentive means to not accept when new items come into the sprint, it became clear that this means, that after the sprint planning is done, no new items are allowed to be added. By explicating this knowledge, the AP becomes measurable because it is easy to determine if items were added after the planning meeting or not.

Table 10

Example of an AP with a high quality

Subject name	Utterance	Code	Utterance
Tilman	I wrote down three to four things.	AP	1
Scrum Master	Come to class.		2
Tilman	I have one about the sprint and the fact that the items come later. An also that the hotfixes are a bit vague.		3
Scrum Master	Yes, yes.		4
Tilman	Yes, so what do we have to do. I wrote here that we need to be attentive with the content of the sprint, at the begin of the sprint.	Relevant	5
Scrum Master	So, what does attentive mean? I mean, when you are attentive, I cannot do a lot with that, when you are attentive. What does it mean to be attentive?		6
Tilman	Yes, that we need to strictly hold strictly to the content of the sprint. And that we do not accept when new items come into the sprint.		7
Scrum Master	Yes, ok. So, after the sprint planning it is locked.		8
Tilman	Yes.		9
Scrum Master	Ok.		10
Tilman	Yes and that we do the sprint planning		11
Scrum Master	*interrupts Tilman* I will just write down one improvement for now. *writes and says what he writes* What: sprint is locked after the planning. We have had the planning and then it is just locked. Ok, we can observe that one, true or false, right.	Measurable	12
Tilman	Yes. Who		13
Scrum Master	Yes, who will do something for that? Yes.		14
Tilman	Everybody, but especially the Scrum Master. We can also ask the colleagues from Jira to lock it up *laughs*		15
Scrum Master	Yeah, it can also just be an agreement. I belief that you can do it with the technology but that is something you can just agree on.		16
Tilman	Yes, otherwise it is going to be a bit		17

Scrum Master	With your Product Owner.		18
Tilman	Yea, otherwise you just talk with your Product Owner.	Acceptable	19
Scrum Master	Just with the Product Owner, yes. But when that is the case, you immediately put pressure on the mister about everything like what is not ready will not be taken on.	Specific,	20
Tilman	Yes.		21
Scrum Master	That means that he has to go to work.		22
Tilman	Yes, exactly.		23

4.3.3 Testing the expected effect of the intervention

This section sets out to test the expected effect, which is that the quality of AP's of teams in the treatment condition increases after the intervention and the quality of AP's of the team in the treatment condition remains the same over time. For that, the DoS was calculated (see Table 11).

Table 11

The degree of SMART (DoS) per AP, team and observation and mean DoS per observation

	DoS of AP1*	DoS of AP2	DoS of AP 3	DoS of AP 4	DoS of AP 5	Mean DoS per observation
Team A						
Retro 1	3	4	3**	3	3**	3.2
Retro 3	5	4**				4.5
Retro 5	4	5	5			4.67
Team B						
Retro 1	5	4	4	2		3.75
Retro 3	4					4
Retro 5	3	3	2			2.6
Team C						
Retro 1	1	1	0	0	0	0.4
Retro 3	5					5
Retro 5	4	4	5	3		4

* Degree of SMART can range from 0 – 5, depending on how many criteria were fulfilled
** Marked with Uncertain code

Then, the differences of scores were compared over treatment conditions (see Table 12). First, the Pretest and Posttest 1 scores were compared to give an indication of short-term effects, by subtracting the mean DoS score of the Pretest from the Posttest 1 score. Then, the Pretest scores were compared with Posttest 2 scores, in order to assess the effects over time. Finally, it was being investigated if there was a change in scores between the second and third observation, by comparing Posttest 1 and Posttest 2 scores. This was done to examine if the effects of the intervention declined over time.

Table 12

Differences in mean DoS scores when comparing over observations

	Difference Posttest 1 minus Pretest	Difference Posttest 2 minus Pretest	Difference Posttest 2 minus Posttest 1
Team A	1.3	1.47	0.17
Team B	0.25	-1.15	-1
Team C	4.6	3.6	-1

There was an increase in mean DoS scores for both teams in the treatment condition (team A and C) when checking for short-term effects and effects over time. Team C shows the strongest increase in mean DoS scores, as they went from a score of 0.4 to a score of 5. However, the score for team C declined again after the second observation, indicating that the effect of the intervention has decreased over time. The mean DoS score of team A also increased during this study. Interestingly, the scores from team A showed a continuous improvement of scores, from a score of 3.2 over a score of 4.5 up to a score of 4.67. When looking at the control group (team B), there was an unexpected slight increase in mean DoS when comparing the first and the second observation, from a score of 3.75 to a score of 4. That could have been due to a Hawthorne effect of the study and the introduction of a new Scrum Master, that occurred at roughly the same moment as the intervention. The Scrum Master began facilitating the retrospectives in a different way than the teams were used to which could affect the results. Also, the Scrum Master used the premise of formulating AP's at the end of an retrospective which were 'observable improvements'. The concept of observable improvements is largely concurrent with some of the SMART criteria which could explain the increase in DoS scores of team B. The Scrum Master had previously witnessed the intervention with the teams in the experimental condition and could have used the knowledge from the intervention in the control group. However, the mean DoS score of team B decreased again strongly in the third observation. The mean scores in DoS over teams can also be found in Figure 3.

Finally, it can be concluded that the intervention has partly achieved the expected effect. All teams in the experimental condition showed an increase in mean DoS scores, when comparing scores from before and after the intervention. However, the team in the control condition also showed an increase in mean DoS scores for that time which might be due to confounding variables such as the introduction of a new Scrum Master. Moreover, the effects of the intervention quickly decreased again for team C.

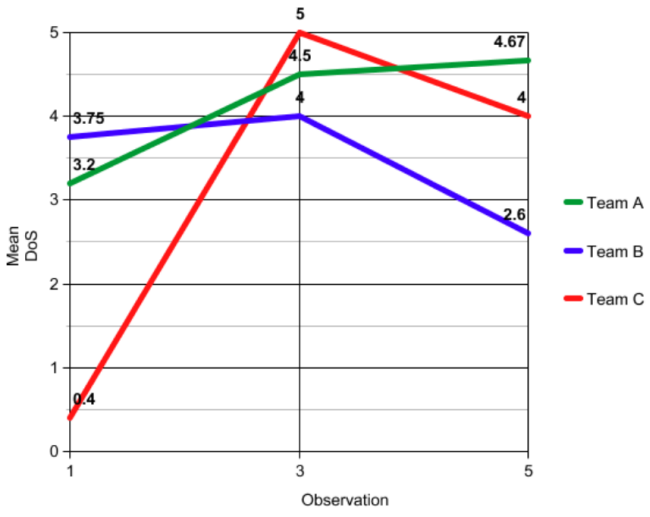


Figure 3. Mean DoS scores per team and observation moment.

4.4 Interview

The purpose of the interview was to assess the reactions of participants to the intervention. The questions centred around the perceived usability of the training, possible reasons for why the intervention has not achieved the expected effect and anticipated long-term effects. The interview was conducted with two Scrum Masters from the teams in the experimental condition (Piet from team A and Steve from team C). First off, both interviewees stated that they thought that the training was useful. Steve stated:

> Yes, I found the training useful. I also found it nice to look at the retrospectives in that way, so to say. And, the things we struggle with were also really established. Yes, in my eyes it made sense and was given in a pleasant way. (…) I have been doing the retrospectives for a long time, also at my previous team. Yet, I noticed that at a certain moment in time, a certain automatism is coming into it (…) I found it in any case (…) I enjoyed that I had a sounding board or received feedback on the way in which it goes. Yeah, you just take that with yourself as baggage, you do something with it. So, I hope that the automatism, I expect that the automatisms adjust to that. And in the beginning, I was handling this very conscientiously. And then you notice that you indeed look at it differently because you are also looking at it during the sessions. And out of that, other attention points also emerge.

For example,

> During previous sprints, a story was quite big and I think that, in the past, we should have said more earlier until we can agree to something. Now, we have it mapped and concrete – these are the acceptance criteria, these are the answers and with this set, we are going to do it. (…) And we saw indeed in the sprint that everybody felt more comfortable in the way the sprint ran. The expectations are clearer now so to say.

For team C, this was also the case. When asked for an example of when SMART was useful, Steve connected the feedback points and how SMART can help them to deal with their weaknesses and said:

> The moment will show [when it is useful]. Well, you have participated in our retros and, sometimes, we get hung up in, well, discussions is a big word, we get hung up a bit in the back and forth about what we are going to do with it. And we do realize this too when we do that. And I think that, in order to hit the nail on the head, that somebody says something to do it like this or that [SMART] we then can make a point out of it. If not, then we have to go to the next topic, because otherwise, we will still be there [with the same topic] half an hour later.

The interview also investigated the possible reasons for why the intervention has not achieved the expected effect. First, Piet experienced difficulties in applying SMART when a new Scrum Master began facilitating their retrospective in another way. He phrased it like that:

> Every [Scrum Master] has its own way of doing things. Especially after your intervention, we of course had the retro of Uwe and he tried very hard to apply it. And after that, (…), somebody else became Scrum Master and he did it in his own way too. I think that it is that we need to keep in the back of our minds in trying to come to actual concrete actions and that was a bit the big ill.

Team A coped with this problem by applying a more abstract or loose view of SMART by only focusing on feasible AP's:

> We also tried (…) to only name points of which we knew that are ourselves responsible for them and ourselves can change them. That is a bit more concrete because, of course, we also had some retrospectives during which we said 'that does not work and that does not work' and then these are things that are outside of the scope of the team. This makes it difficult to formulate actions about them. (…) I think that it is something that you just need to do, something that needs to belong in a retrospective. (…) Because I do not have the impression that we are really going to follow the loose steps of the SMART roadmap during a retrospective in order to come to an AP. We maybe should have done that for just two or three retrospectives for it to sink in. But that has not happened.

With time, Steve from team C also applied SMART less rigorous than in the beginning:

> Yeah, at that point in time, I have to say that I use it [SMART] more in the back of my mind than really the roadmap in the sense that I have the criteria actually lying in front of me. It is more that I myself know it.

What is more, according to Piet, it is possible that the team has not yet managed to fully incorporate SMART into their retrospective practice since that also takes time: "Yes, I think in the broad sense. (…) I think that the message was received but I think that it still has to sink in for a bit. It needs to take shape over time." The last possible reason is that team C focused their attention more on improving other meetings than the retrospective, namely, the refinement. Yet, since this meeting was not part of this study, these changes cannot be accounted for. After being asked after changes due to the intervention, Steve said: "Everybody is critical to it [SMART] in the refinement and always asks more concrete questions. I think that we as a team indeed have become better in it."

Finally, the interview assessed the anticipated long-term use of SMART, as perceived by the Scrum Masters. Both Scrum Masters agreed that they would continue to apply what they learned from the training in the future. After being asked if they would continue with the SMART criteria, Piet said:

> I do think that somebody will sometimes comment on them (…) Surely, one team member more than the other, but I think that somebody will comment when are again at a point at which we think what do we have to do about it, how are we going to approach it. I think that somebody will say like, hey, what if we look at it 'SMART'. Yes, I guess that this will happen then.

Steve also thought that they would continue to use SMART:

> I think that we are just going to stay using it in order to see to keep ourselves critical and alert. So I think that it is good and that we will keep on applying it. I cannot guarantee for the whole team that everybody will keep doing it but from my point of view, I am an advocate.

4.5 Summary

The results have given examples of elements of team reflexivity that need improvement and tested the effect of the intervention. This chapter illustrates what reflection and planning and possible ways to improve team reflexivity look like during retrospectives of SD teams working with Scrum. Also, the results show that there is potential for improving team reflexivity in retrospectives of the participating teams. A low percentage of wrap-ups per episode was found, ranging from 11.11% to 25%, which indicates that most episodes did not end in a conclusion. Over time, teams showed a varying change in the percentage of wrap-ups per episode, as team A first strongly increased but then declined again, team B showed an increase and then stagnated, and team C showed almost no change. Also, the results indicate that the intervention had a mixed effect on the quality of planning. There was an increase in quality of planning for all participating teams, when comparing scores from before and after the intervention, regardless of the assigned condition. The strongest increase was for team C. However, the effects declined again after the second observation, indicating that the effect of the intervention has decreased over time. Team A showed a continuous increase in their quality of planning. The control group also had a slight increase in the quality of their planning which decreased strongly again after the second observation. The interview revealed that the team found the intervention useful and has learned how to use SMART. However, several reasons for the failure of the experiment were also discovered, such as that the team was not always able to apply SMART consistently, because of other, unrelated factors, such as a new Scrum Master, applying a more loose view of SMART and focusing on other meetings than the retrospective. It is also possible that the effects of the intervention are delayed, since adapting SMART into the retrospective practice requires time and repetition. Both Scrum Masters agreed that they would continue to apply what they learned from the training, if possible.

5. Discussion

5.1 Interpretation of results

The goal of the study was to support the continuous improvement of three Scrum teams with an intervention to improve the team reflexivity, by improving the quality of their goals, so that they are better equipped to realize the potential of the retrospectives. The research question was: *How can we design an intervention that supports Scrum teams to enhance team reflexivity during their retrospectives?* The research question was answered by following three steps: first, studying what elements need improvement, second, designing the intervention and third, testing the effect of the intervention. The main outcomes of this study are the following: First, this study has found that there is room for improvement in the continuous improvement through retrospectives, as a lot of examples of low quality team reflexivity were identified. What is more, the results indicate that the intervention was probably not enough to solve the team's issues with continuous improvement since the results show that the intervention only led to small, temporary changes. More information is needed for methods that can affect real change. Lastly, this study gives insight into team reflexivity during retrospectives in the real-life work context of SD teams working with Scrum.

To elaborate, the results indicate that the continuous improvement during retrospectives can be improved, which is in line with previous studies that have found that the potential of the retrospective is often not fully realized (Moe, 2013; Ringstad et al., 2011; Salo & Abrahamsson, 2006). Most importantly, in the beginning, the quality of planning was only moderate, for all participating groups. This is consistent with Ringstad et al.'s (2011) and Moe and Dingsøyr's (2008) studies who found that in agile teams, planning is often inefficient and that teams deviated from the Scrum method. At this point needs to be noted, that it is characteristic for agile methods to plan few things up front due to their complex tasks (Moe, Dingsøyr & Dybå, 2008). The planning itself usually takes place during the sprint planning meeting, during which the plan is laid out for the next sprint, or during the daily scrum, in which the progress is monitored (Schwaber & Sutherland, 2013). However, this should not be used as an excuse to not think about how to implement future improvements because otherwise, the team could set goals they cannot carry out because they did not think about the implementation beforehand. Moreover, it was found that few episodes were concluded with a wrap-up. The percentage of episodes that included a wrap-up was ranging from 11% to 25%, which is in line with findings from Raes et al. (2015). The low percentage of wrap-ups per episode could also be explained by the nature of retrospectives, in which, the beginning usually consists of a short sharing session of all topics the team could come up with. These short introductions of the topics were often swift and did not end with a conclusion, as their purpose was to give an overview rather than to have a discussion. Sometimes, the topics were summarized, however, this was not often the case. Later, the team would prioritize the topics and only discuss the few most important topics.

This study has also led to the design of an intervention to teach the use of SMART in retrospectives. However, the data shows that the intervention probably was not enough to solve the team's issues with continuous improvement. There were some small, temporary changes in the quality of planning for the teams, which is in line with findings from Marsland and Bowden's (2010) study, who also found modest changes. The data analysis revealed that there was an increase in quality of planning for all participating teams, when comparing scores from before and after the intervention, regardless of the assigned condition. The strongest increase was for team C. However, the effects declined again after the second observation, indicating that the effect of the intervention has decreased over time. Team A showed a continuous increase in their quality of planning. The control group also had a slight increase in the quality of their planning which decreased strongly again after the second observation. Additionally, this study found that the most often coded SMART criteria were Relevant and Specific, while Timely and Measurable were assigned the fewest. This is also in concurrence with Marsland and Bowden's (2010) study. It needs to be pointed out, that the control group also showed an increase in the percentage of wrap-ups per episode as well as in the quality of planning. Team B consistently increased their use of wrap-ups per episode, while the teams in the other condition either first increased, then decreased or stayed the same. Team B was a relatively new team which was not as used to working together as the other teams. It is possible that this team has managed to formulate more conclusions to discussions because the team members have improved their ability to work together during that time. Also, it is possible that they improved either being observed by a researcher with a camera, or due to the introduction of a new Scrum Master who witnessed the intervention with the teams in the experimental condition and used the knowledge from the intervention in the control group. Also, the Scrum Master used the premise of formulating AP's at the end of an retrospective which were 'observable improvements'. The concept of observable improvements is largely concurrent with some of the SMART criteria which could explain the increase in DoS scores of team B. It seems that a lot more information is needed to see real change.

The interview revealed several possible explanations for why the intervention did not achieve the expected effect. Firstly, adapting to SMART takes long time and repetition (Marsland & Bowden, 2015; Swanson, 2016). Due to the limited scope of this study and the different facilitation methods of the new Scrum Master for Team A, this was not possible. What is more, both teams proceeded after the intervention by applying a more loose use of SMART, such as by focusing on merely the feasibility of items and neglected other criteria of SMART, especially timeliness and measurability. Team C also chose to focus on improving their planning more during the refinement, than during the retrospective. However, these effects could not be accounted for. Finally, although both Scrum Masters claimed that they want to continue using SMART, it is doubtful if they will be able to do so because of their high workload. Furthermore, it is more probable that they would continue to apply a more loose use of SMART and neglect more difficult SMART criteria such as timeliness and measurability.

Finally, this study gives insight into team reflexivity during retrospectives in the real-life work context of SD teams working with Scrum. First, this study has led to the development of a coding scheme of video observations to assess the quality of goals formulated during retrospectives, adding to Bowman et al's (2015) goal evaluation method for written goals. Second, this study has identified and discussed examples of the first two stages of team reflexivity, reflection and planning, and their respective qualities in a work context. The identified examples indicate the importance of explicating knowledge in retrospectives to improve team reflexivity, which is in line with previous studies (Andriyani et al., 2017; Matthew & Sternberg, 2009; Müller et al., 2009). Finally, this study shows possible pitfalls of an intervention in the environment of SD teams working with Scrum.

5.2 Limitations and recommendations

One general limitation of this study is the small number of teams that were studied. Additionally, the training lasted merely one hour, as the teams had requested a time-efficient intervention. Yet, that is a relatively short duration for sustainable learning to take place and could be extended when the training is used again in the future. What is more, the data analysis included only 3 out of 5 observation to allow more in-depth analysis. However, more detailed analysis of the remaining two meetings could have given additional insights. It also needs to be added, that the results of this study are context specific since only one firm was studied. Hence, the generalizability is limited to the three teams of this Dutch firm. Additionally, the documentation of the AP's was not part of the analysis as the study is about utterances from the teams. Also, in observation 3, all of the participating teams only formulated one AP which is a very small sample and makes it difficult to draw conclusions on the quality of the AP's.

At that point has to be mentioned, that the concept of reflection was not well understood enough to further differentiate between different qualities of reflection. Future studies could investigate how to discern between the varying qualities of reflection, such as the three depths of reflection. The recommendation is to extend the coding scheme for reflection, to include the three depths of reflection (Schippers et al., 2007). Additionally, the action stage of team reflexivity could not be accounted for, given the particular focus of this study on retrospective meetings that serve to reflect and to plan.

Related to that, this study could not test if the overarching goal of the study, namely to improve the quality of teamwork of SD teams so they can better deal with their demanding environment, was really reached. That is because there are too many variables that change, such as turnover of team members, or shifting focus inside the company and so forth and investing the teamwork quality is outside of the scope of this study. For example, one AP that was formulated SMART could not be enacted because the project it concerned was cancelled. It is still unclear if the quality of the goal itself can support the continuous improvement, which, in turn, can improve the quality of teamwork. This could only be evaluated with a longer-term study of the implementation of the goals that were either formulated SMART or not and their effect on the teamwork quality, for example using Hoegl and Gmuenden's (2001) measure of teamwork quality.

Finally, this study could not account for the kind of leadership of the Scrum Masters. However, this could be interesting, since the same facilitation methods of the new Scrum Master Anthony had different effects on the teams. During this study, he began facilitating all of the three teams. Two teams liked his facilitation, while one team asked him to stop with his facilitation, to continue with their already established retrospective routine. Further topics could investigate what kind of facilitation methods fits what kind of team and why.

5.3 Practical implications

This study sheds light on how reflection and planning look like in the real-life work context of retrospectives. What is more, this study gives examples of ways to improve team reflexivity in retrospectives. This information can help HRD practitioners, to improve the team reflexivity of agile teams and support the teams to improve continuously. The intervention and the SMART roadmap could also be used for other agile teams, to improve how they formulate AP's. Also, the ways of improving team reflexivity highlight, what teams need to focus on, in order to improve their retrospectives and support their continuous improvement. For example, the teams could investigate if the quality of their reflection is high enough to discover the root causes, and if not, use reflection activities such as root cause analysis (Lehtinen et al., 2017).

The training can be used by agile teams to train planning skills. The information can help agile teams, to critically assess their retrospectives, if they invest enough in reflection and planning and if it is necessary to improve their formulation of AP's with SMART. For example, if a team struggled with their improvement, because the retrospectives do not lead to change, they can critically assess the quality of their points of improvement in order to better come to concrete, feasible AP's. However, it is being advised that the training and assessment are given in cooperation with a Scrum coach, or best in the context of a larger organizational change initiative.

Finally, the firm could investigate how the organization could help to make the AP's more SMART. One possibility is to establish a documentation system of reflection results and AP's that were established during the retrospective. This system could support the construction of shared cognition, as a failsafe to monitor efforts on previous improvement efforts, to make sure that old efforts are continued and information can be stored.

5.4 Conclusion

This study set out to discover how agile SD teams could improve the quality of their teamwork by supporting their continuous improvement process, with the help of an intervention to improve team reflexivity. To answer that question, this study builds on research on software process improvement in agile SD teams that have found that the potential of the retrospective is not fully realized, such as Ringstad et al. (2011). An intervention was designed to support team reflexivity during retrospectives, a promising concept for improving team performance and learning, especially for innovative teams. The intervention centred around teaching SMART to improve the quality of the goals and was designed

similar to other SMART interventions, such as Marsland and Bowman (2010). This study shows the effect of such an intervention and highlights typical pitfalls to watch out for in future research on retrospectives. Results indicate that the intervention had a limited effect and that more information is needed to affect real change. It also argues for the involvement of the organization in the improvement processes of SD teams, to support their efforts and to acknowledge that adapting SMART takes time and repetition. Further studies could investigate the effects of various qualities of reflection, how AP's during retrospectives are adapted, the effect of continuous improvement on teamwork quality, the role of the Scrum Master and ways to integrate the continuous improvement of the teams with an organizational change initiative or a documentation system.

To conclude, this study extends the conceptual and practical knowledge of team reflexivity and ways to improve team reflexivity in a real-life work environment. Moreover, this study gives a new scientifically relevant approach to improve the quality of teamwork of agile teams through continuous improvement. In the complex and challenging work environment that many SD teams face, this study can support them to improve their teamwork and excel at developing innovative software.

References

Albrechtsen, E., & Hovden, J. (2010). Improving information security awareness and behaviour through dialogue, participation and collective reflection. An intervention study. *Computers & Security*, *29*(4), 432-445.

Andriyani, Y., Hoda, R., & Amor, R. (2017, May). Reflection in agile retrospectives. In: Baumeister H., Lichter H., Riebisch M. (Eds.), Lecture Notes in Business Information Processing: Vol. *283. International Conference on Agile Software Development* (pp. 3-19).

Babb, J., Hoda, R., & Norbjerg, J. (2014). Embedding reflection and learning into agile software development. *IEEE software*, *31*(4), 51-57.

Bowman, J., Mogensen, L., Marsland, E., & Lannin, N. (2015). The development, content validity and inter-rater reliability of the SMART-Goal Evaluation Method: A standardised method for evaluating clinical goals. *Australian occupational therapy journal*, *62*(6), 420-427.

Bron, R., & Endedijk, M. D. (2016). The dynamic and emergent nature of team learning: Using machine learning techniques to automatically discern team learning processes from team conversations. Paper presented at the EARLI SIG 14 Conference, Regensburg, Germany.

Cannon, M. D., & Edmondson, A. C. (2005). Failing to learn and learning to fail (intelligently): How great organizations put failure to work to innovate and improve. *Long range planning*, *38*(3), 299-319.

Chow, T., & Cao, D. B. (2008). A survey study of critical success factors in agile software projects. *Journal of systems and software*, *81*(6), 961-971.

Cohen, S.G. & Bailey D.E. (1997) What makes teams work: group effectiveness research from the shop floor to the executive suite. *Journal of Management 23*(3), 239-290.

Decuyper, S., Dochy, F., & Van den Bossche, P. (2010). Grasping the dynamic complexity of team learning: An integrative model for effective team learning in organisations. *Educational Research Review*, *5*(2), 111-133.

DeShon, R. P., Kozlowski, S. W., Schmidt, A. M., Milner, K. R., & Wiechmann, D. (2004). A multiple-goal, multilevel model of feedback effects on the regulation of individual and team performance. *Journal of applied psychology*, *89*(6), 1035-1056.

Dick, W. (2002). Evaluation in instructional design: The impact of Kirkpatrick's four-level model. In R. Reiser & J. Dempsey (Eds.), *Trends and issues in instructional design and technology* (pp. 145-153). Retrieved from http://jhollenbeck.com/courses/hrd480/pdf/handouts/06b_dick_eval.pdf

Dingsøyr, T., Fægri, T. E., Dybå, T., Haugset, B., & Lindsjørn, Y. (2016). Team performance in software development: research results versus agile principles. IEEE Software, 33*(4)*, 105 – 110.

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

Dingsøyr, T., & Lindsjørn, Y. (2013, June). Team performance in agile development teams: Findings from 18 focus groups. In: H. Baumeister, & B. Weber (Eds.), Lecture Notes in Business Information Processing: Vol. *283. International Conference on Agile Software Development* (pp. 46-60). Springer, Berlin, Heidelberg.

Dybå, T., & Dingsøyr, T. (2008). Empirical studies of agile software development: A systematic review. *Information and Software Technology*, *50*(9–10), 833–859. http://doi.org/10.1016/j.infsof.2008.01.006

Dybå, T., Dingsøyr, T., & Moe, N. B. (2014). Agile project management. In G. Ruhe, C. Wohlin (Eds.) *Software project management in a changing world* (pp. 277-300). Springer Berlin Heidelberg.

Ennabih, A., Van Riel, A. C., & Sasovova, Z. (2016, January). Leadership And New Product Development Project Performance: A Team Cognition Perspective. In *Academy of Management Proceedings* (Vol. 2016, No. 1, p. 16920). Academy of Management.

Gabelica, C., Van den Bossche, P., De Maeyer, S., Segers, M., & Gijselaers, W. (2014). The effect of team feedback and guided reflexivity on team performance change. *Learning and Instruction*, *34*, 86-96.

Gurtner, A., Tschan, F., Semmer, N. K., & Nägele, C. (2007). Getting groups to develop good strategies: Effects of reflexivity interventions on team process, team performance, and shared mental models. *Organizational Behavior and Human Decision Processes*, *102*(2), 127-142.

Hoda, R., Noble, J., & Marshall, S. (2012). Developing a grounded theory to explain the practices of self-organizing Agile teams. *Empirical Software Engineering*, *17*(6), 609-639.

Hoegl, M., Parboteeah, K. P., & Gemuenden, H. G. (2003). When teamwork really matters: task innovativeness as a moderator of the teamwork–performance relationship in software development projects. *Journal of Engineering and Technology Management*, *20*(4), 281-302.

Hoegl, M., & Parboteeah, K. P. (2006). Team reflexivity in innovative projects. *R&D Management*, *36*(2), 113-125.

Kim, Y. (2007). Analyzing scrum agile software development with development process, social factor, and project management lenses. *AMCIS 2007 Proceedings*, 81.

Kirkman, B. L., & Rosen, B. (1999). Beyond self-management: Antecedents and consequences of team empowerment. *Academy of Management journal*, *42*(1), 58-74.

Kirkpatrick, D. (1996). Revisiting Kirkpatrick's four-level model. *Training & Development*, *50*(1), 54-57.

Kleingeld, A., van Mierlo, H., & Arends, L. (2011). The effect of goal setting on group performance: A meta-analysis. *Journal of Applied Psychology*, *96*(6), 1289-1304

Lawlor, K. B. & Hornyak (2012). Smart goals: How the application of smart goals can contribute to achievement of student learning outcomes. *Developments in Business Simulation and Experiential Learning*, Vol. *39* (pp. 259-267). Retrieved from https://journals.tdl.org/absel/index.php/absel/article/view/90/86

Lehtinen, T. O., Itkonen, J., & Lassenius, C. (2017). Recurring opinions or productive improvements - what agile teams actually discuss in retrospectives. *Empirical Software Engineering*, *22*(5), 2409-2452.

Locke, E. A. (1968). Toward a Theory of Task Motivation and Incentives. *Journal of Organizational Behavior and Human Performance, 3*, 157-189.

Locke, E. A., & Latham, G. P. (1990). *A theory of goal setting and task performance.* Englewood Cliffs, NJ: Prentice-Hall, Inc.

Marsland, E., & Bowman, J. (2010). An interactive education session and follow-up support as a strategy to improve clinicians' goal-writing skills: a randomized controlled trial. *Journal of evaluation in clinical practice*, *16*(1), 3-13.

Matthew, C. T., & Sternberg, R. J. (2009). Developing experience-based (tacit) knowledge through reflection. *Learning and Individual Differences*, *19*(4), 530-540.

McAvoy, J., & Butler, T. (2007). The impact of the Abilene Paradox on double-loop learning in an agile team. *Information and Software Technology, 49*, 552–563.

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

Moe, N. B. (2013, June). Key challenges of improving agile teamwork. In H. Baumeister & B. Weber (Eds.), Lecture Notes in Business Information Processing: Vol. *149. International Conference on Agile Software Development* (pp. 76-90). Springer, Berlin, Heidelberg

Moe, N. B., Aurum, A., & Dybå, T. (2012). Challenges of shared decision-making: A multiple case study of agile software development. *Information and Software Technology*, *54*(8), 853-865.

Moe, N. B., & Dingsøyr, T. (2008). Scrum and team effectiveness: Theory and practice. In P. Abrahamsson, R. Baskerville, K. Conboy, B. Fitzgerald, L. Morgan & X. Wang (Eds.), Agile Processes in Software Engineering and Extreme Programming (Vol. 9, pp. 11-20): Springer Berlin Heidelberg

Moe, N. B., Dingsøyr, T., & Dybå, T. (2008, 26-28 March 2008). Understanding self-organizing teams in agile software development. Paper presented at the 19th Australian Conference on Software Engineering, 2008. ASWEC 2008.

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. http://doi.org/10.1016/j.infsof.2009.11.004

Molenaar, I. (2011). Metacognitive activities embedded in interaction. In I. Molenaar (Ed.), It's all about metacognitive activities: Computerized scaffolding of self-regulated learning. Enschede: Ipskamp Drukkers.

Monaghan, J., Channell, K., McDowell, D., & Sharma, A. K. (2005). Improving patient and carer communication, multidisciplinary team working and goal-setting in stroke rehabilitation. *Clinical rehabilitation*, *19*(2), 194-199.

Müller, A., Herbig, B., & Petrovic, K. (2009). The explication of implicit team knowledge and its supporting effect on team processes and technical innovations: An action regulation perspective on team reflexivity. *Small Group Research*, *40*(1), 28-51.

Nerur, S., Mahapatra, R., & Mangalaraj, G. (2005). Challenges of migrating to agile methodologies. *Communications of the ACM*, *48*(5), 72-78.

Raes, E., Boon, A., Kyndt, E., & Dochy, F. (2015). Measuring team learning behaviours through observing verbal team interaction. Journal of Workplace Learning, 27(7), 476-500. doi: 10.1108/JWL-01-2015-0006\

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

Ringstad, M. A., Dingsøyr, T., & Moe, N. B. (2011, June). Agile process improvement: Diagnosis and planning to improve teamwork. In R. V. O'Connor, J. Pries-Heje & R. Messnarz (Eds.), Systems, Software and Service Process Improvement: Vol. 172. *European Conference on Software Process Improvement* (pp. 167-178). Springer, Berlin, Heidelberg.

Salas, E., Sims, D. E., & Burke, C. S. (2005). Is there a "big five" in teamwork?. *Small group research*, *36*(5), 555-599.

Salo, O., & Abrahamsson, P. (2007). An iterative improvement process for agile software development. *Software Process: Improvement and Practice*, *12*(1), 81-100.

Schippers, M. C., Den Hartog, D. N., & Koopman, P. L. (2007). Reflexivity in teams: A measure and correlates. *Applied psychology*, *56*(2), 189-211.

Schippers, M. C., West, M. A., & Dawson, J. F. (2015). Team reflexivity and innovation: The moderating role of team context. *Journal of Management*, *41*(3), 769-788.

Schwaber, K. & Sutherland, J. (2016). The Scrum Guide™. The Definitive Guide to Scrum: The Rules of the Game. Retrieved on 18.11.2016 from: http://www.scrumguides.org/docs/scrumguide/v1/scrum-guide-us.pdf

Stray, V. G., Moe, N. B., & Dingsøyr, T. (2011, May). Challenges to teamwork: a multiple case study of two agile teams. In A. Sillitti, O. Hazzan, E. Bache & X. Albaladejo (Eds.), Agile Processes in Software Engineering and Extreme Programming (Vol. 77, pp. 146-161): Springer Berlin Heidelberg.

Swanson, M., "Implementation of a SMART Goal Intervention for Diabetic Patients: A Practice Change in Primary Care" (2016). *Doctoral Projects*. 10. Retrieved from: https://scholarworks.gvsu.edu/kcon_doctoralprojects/10

Tjosvold, D., Tang, M. M., & West, M. (2004). Reflexivity for team innovation in China: The contribution of goal interdependence. *Group & Organization Management*, *29*(5), 540-559.

Vlietland, J., Van Solingen, R., & van Vliet, H. (2016). Aligning codependent Scrum teams to enable fast business value delivery: A governance framework and set of intervention actions. *Journal of Systems and Software*, *113*, 418-429.

Volet, S., & Mansfield, C. (2006). Group work at university: Significance of personal goals in the regulation strategies of students with positive and negative appraisals. *Higher Education Research & Development*, *25*(4), 341-356.

West, M. A. (2000). Reflexivity, revolution and innovation in work teams. In D. A. Johnson & S. T. Beyerlein (Eds.), *Product development teams* (pp. 1-29). Stamford, Connecticut: JAI Press

Widmer, P. S., Schippers, M. C., & West, M. A. (2009). Recent developments in reflexivity research: A review. *Psychology of Everyday Activity*, *2*(2), 2-11.

Wijga, M., & Endedijk, M. D. (2016). Social Regulation at the Workplace: The Development of an Analysis Tool to Understand Regulation of Learning in Teams. Paper presented at the International Conference: 8th EARLI SIG 14 Learning and Professional Development, Regensburg. http://doc.utwente.nl/102694/

Yu, X., & Petter, S. (2014). Understanding agile software development practices using shared mental models theory. *Information and Software Technology*, *56*(8), 911-921.

C. D. Patzak

Appendix A

SMART roadmap

SMART doelen

SMART is een middel om doelen zoals jullie actiepunten concreter te maken. Door het gebruik van SMART maak je duidelijke, haalbare en betekenisvolle doelen en ontwikkel je motivatie, een actie plan en advies voor ondersteuning om deze doelen te behalen. SMART staat voor:

- S pecifiek
- M eetbaar
- A anvaardbaar / acceptabel
- R elevant
- T ijdsgebonden

Jullie maken jullie actiepunt concreter door deze vijf stappen af te gaan. Bij ieder stap worden vragen aangereikt om de actiepunt SMART te maken.

1) Specifiek

Vage beloven voor een verbetering brengen over het algemeen weinig resultaten. Gebruik eerder deze vijf vragen om het doel zo specifiek mogelijk te maken:

Wat willen wij precies bereiken?

Wie is betrokken in het behalen van dit doel?

Wanneer gaan wij bezig met de acties om het doel te behalen?

Waar moeten wij zijn om dit doel te bereiken?

Waarom willen wij dit doel bereiken (motivatie)?

2) Meetbaar

Het moet meetbaar zijn of jullie actiepunt behaald is of niet om voortgang vast te stellen. Daarom is het belangrijk om het doel dermate te formuleren.

STEL JE VOOR dat ieder sprint een vast tijdsafstand is binnen welke een wetenschappelijk experiment uitgevoerd wordt. Aan het begin stellen jullie hypothesen op over wat jullie willen doen tijdens een bepaald tijdsduur. Na ieder sprint evalueer je op basis van relevante data de voortgang van je experiment en stel je de waarde van ieder experiment vast. Daarvoor moet je weten welke data relevant ervoor is. Jullie kunnen de actiepunt meetbaar maken door deze vragen te beantwoorden:

Wat moet gedaan zijn dat de actiepunt gehaald is (**Definition of Done**)? Hoe weten **wij** dat wij het doel hebben gehaald?

Hoe kunnen wij **voortgang** op dit gebied **vaststellen**?

3) Aanvaardbaar / Acceptabel

Om een doel te bereiken moet het doel realistisch en haalbaar zijn. Probeer hierbij jullie **uit te dagen** en te **leren**, maar let erop dat het doel **steeds nog te doen** is. Een haalbaar doel maken jullie door de volgende vragen te beantwoorden:

LET OP: het behalen van het doel moet **in jullie vermogen liggen**. Als het de beslissing van een ander persoon is of je het doel bereikt maken jullie het halen van het doel afhankelijk van iemand anders. Zorg daarom ervoor dat jullie zelf iets kunnen veranderen aan de situatie.

Hebben wij voldoende vaardigheden of resources om dit doel te bereiken?

Hoe realistisch is dit doel, gebaseerd op andere beperkingen zoals financiële factoren?

Wat zijn eerdere ervaringen met het behalen van soortgelijke doelen?

Hoe precies kunnen wij dit doel bereiken?

4) Relevant

Deze stap gaat erover dat jullie actiepunt **belangrijk** is **voor jullie team** en anderen. Want, vaak hangt een actiepunt samen met andere relevante doelen en hebben jullie ondersteuning nodig.

LET OP: Als jullie ondersteuning halen bij andere partijen is het belangrijk dat jullie steeds nog de controle bewaren over jullie doel

Zorg daarom ervoor dat jullie plan ieder geïnvolveerde persoon naar voren brengt maar dat **jullie steeds nog verantwoordelijk zijn** om **jullie eigen doel te behalen.**
Hiermee voorkom je dat de vraagstelling voor andere doelstellingen wordt gebruikt en jullie belang wordt genegeerd.

Gaat het behalen van ons doel ons daadwerkelijk iets opleveren?

Kunnen wij door het halen van het doel het wezenlijke probleem oplossen?

➔ Wees niet bang om een scherpe verandering aan te streven en 'out of the box' te denken. Verandering is het hart van agile!

Ook zouden jullie jullie af kunnen vragen of **iemand anders misschien beter het doel kan bereiken** en of jullie de opdracht aan hem over kunnen dragen.

5) Tijdsgebonden

Deze stap wordt gedaan om ervoor te zorgen dat de **uitvoering** van jullie actiepunt een **duidelijke begin** en een **duidelijk einde** heeft.

➔ OF is jullie het behalen van jullie doel binnen een bepaald tijdspunt gelukt
➔ OF jullie hebben het behalen van het doel opgegeven en een duidelijke einde eraan gezet.

Als jullie een lange termijn doel willen behalen is het belangrijk om jullie korte termijn actiepunten op tijd af te ronden. Stel daarom deze vragen:

Tot welk exacte datum willen we deze actiepunt behaald hebben?

Wat kunnen wij zes maanden van nu doen?

Wat kunnen wij zes weken van nu doen?

Wat kunnen wij vandaag doen?

Oefening:

Oefen het maken van een SMART doel even door met het team een doel op te stellen. Ga even na of je doel ook aan alle SMART eisen voldoet en leg kort uit hoezo.

SMART doel:

_____

_____

Specifiek: -

_____

Meetbaar: _____

Aanvaardbaar: _____

Relevant:

_____

Tijdsgebonden: _____

Appendix B

Interview scheme (Dutch)

Interview schema

Het doel van dit interview is om het effect van het training te onderzoeken, als aanvulling op de video analyse. In het kader van het trainingsplan werden algemene trainingsdoelen opgesteld (trainingsdoelen zijn te vinden in Appendix D). Het interview is bedoeld om achteraan te gaan of het training succesvol was in het bereiken van deze trainingsdoelen. Daarnaast wordt de effectiviteit en ervaren nut van trainingsoefeningen en het training zelf onderzocht. Tot slot worden vragen gesteld om een mogelijk Hawthorne effect vast te stellen.

De geïnterviewde is een leidinggevende of een teamlid welke aanwezig was tijdens de hele duur van het onderzoek. Het interview is semigestructureerd en opgedeeld in zeven delen. Aan het begin van ieder deel van het interview staan de onderwerpen welke tijdens het interview aan bod komen. In cursief staat vervolgens een tekst met bij de onderwerpen horende hoofdvragen en alternatieven welke de onderzoeker aanhoudt tijdens het interview.

1) Introductie en formaliteiten

Hallo, ten eerste heel erg bedankt voor je bereidheid om dit interview met mij te doen. Ik verwacht dat dit interview rond een uur tijd van je gaat vragen. Daarnaast wil ik je vragen of ik het interview op mag nemen en de data voor mijn onderzoek mag gebruiken. De data wordt geanonimiseerd zodat je naam niet terecht komt in het uiteindelijke verslag.

Zoals eerder uitgelegd is het doel van dit interview om achteraan te gaan of het training het gewenste effect heeft bereikt. Daarvoor heb ik vragen voorbereid die ik je ga stellen om te onderzoeken of het doel van het training is bereikt of redenen te identificeren voor hoezo dit misschien niet gelukt is. Maak je geen zorgen, dit is geen test maar eerder bedoeld om achteraan te gaan wat is gebleven van de training. Dus geef gewoon vooral een eerlijk antwoord. Voor we beginnen: heb je nog vragen over het interview?

2) Wat men vond van het training

Onderwerpen:

- Nut training en SMART
- Identificatie barrières voor gebruik tijdens training geleerde vaardigheden
- Algemene indruk van het hele team

Ik ben benieuwd naar hoe je en je team de nut van de interventie hebt ervaren.

Hoofdvragen:

a) Hoe vond je het training? Goede en slechte dingen, verbetervoorstellen?
b) Heb je het gevoel dat het training jullie iets van belang heeft opgeleverd voor jullie retro's? Weet je nu bij voorbeeld beter hoe je een retro moet doen dan daarvoor?
c) Denk je dat het leren van SMART handig was voor jullie als team? Gebruiken jullie bij voorbeeld nog SMART of het hand-out om jullie actiepunten te concretiseren?

3) Bereiken van de trainingsdoelen en lange termijn effecten

Onderwerpen:

- Retrospective gedrag van team vóór het onderzoek
- Kennis van SMART en van hoe men het voor de retrospectives kan gebruiken
- Kennis van wat SMART voor het verbeteren van de retrospectives kan betekenen
- Ervaren nut SMART voor retrospectives
- Aanwezigheid gedeeld mentaal model voor een bij het team passende manier om een retro te doen
- Identificatie barrières voor gebruik tijdens training geleerde vaardigheden

Tijdens het training had ik het met jullie over SMART, heb ik jullie wat feedback gegeven over de manier op welke jullie retrospectives doen en hebben we samen een oude actiepunt concreet gemaakt. Inmiddels zijn tien weken verstreken na het training en was ik benieuwd naar wat jullie hebben geleerd door het training.

Hoofdvragen:

a) *Hoe heeft het team tot aan het onderzoek samen gewerkt?*
b) *Denk je dat je team nu weet wat SMART is en hoe je het kan gebruiken voor de retrospectives?*
c) *Heb je het idee dat het team heeft begrepen hoe SMART kan hulpen om de retrospectives te verbeteren?*
d) *Ben je situaties of barrières tegen gekomen door welke jullie SMART niet konden gebruiken? (Mogelijkheden: (onvoldoende) tijd, het team / de Scrum Master,…)*
e) *We hopen natuurlijk dat het training niet alleen op de korte termijn maar ook iets op de lange duur gaat opleveren. Denk je dat jullie SMART nog gaan gebruiken nadat de camera's en ik weg zijn? En hoezo wel of hoezo niet?*

4) Vragen gericht op vaststellen van het Hawthorne effect

Onderwerpen:

2) Effect onderzoek, onderzoeker en camera

De volgende vragen zijn daarop gericht om te onderzoeken of de verandering in het gedrag van het team misschien door andere factoren komt dan door de interventie.

Hoofdvragen:

a) *Ik snap natuurlijk dat ik van buiten kom als onderzoeker en dat dit een invloed heeft. Denk je dat de dynamiek van het team daardoor daadwerkelijk heel anders is geworden, omdat ik aanwezig was?*
b) *Hoe denk je over de camera's, denk je dat het team zich in het begin substantieel anders heeft gedragen tijdens de retrospectives omdat zij werden opgenomen? Heb je een verschil opgemerkt tussen retrospectives zonder en met een camera?*

5) Leiderschap binnen het team / Rol Scrum Master

Daarnaast wil ik je graag vragen stellen over hoe je de rol van de Scrum Master tijdens het onderzoek hebt ervaren. Onderzoek wijst aan dat de leider een belangrijke rol voor de tevredenheid en effectiviteit van het team kan spelen, daarom wil ik dit graag onderzoeken.

Onderwerpen:

- Ervaren leiderschap tijdens onderzoek
- Inventariseren voorkeur voor sociaal gedeelde regulatie of externe regulatie
- Onderzoeken leiderschapsstijl (participatief, directief en ambidextriek) van verschillende Scrum Masters tijdens het onderzoek (Ennabih et al., 2016)

Hoofdvragen:

a) *Hoe zie je de rol van een Scrum Master in een retrospective?*
b) *Ik ben even benieuwd naar hoe je de stijl van de Scrum Masters zou beschrijven als je moet beslissen tussen twee verschillende leiderschapsstijlen of een mengsel ervan. Ik ga even drie verschillende leiderschapsstijlen beschrijven en je kiest een voor ieder Scrum Master.*
    i. *Jullie Scrum Master laat ieder zijn mening zeggen, betrekt het hele team bij beslissingen en schept ruimte voor nieuwe en creatieve oplossingen binnen het team. Jullie Scrum Master heeft belang voor het welzijn van ieder teamlid en wil dat iedereen erbij hoort. Daardoor zijn mensen tevreden en gemotiveerd.*
    ii. *Jullie Scrum Master schept heldere communicatie lijnen, heeft tijdens meetings altijd een oog op de tijd, is effectief in het opstellen van doelen en benadrukt het halen van deadlines. Hij is vaardig in het structureren van zowel doelen als ook het proces. Daardoor weet het team altijd wat het meest belangrijk is en zijn ze bewust ervan als zij doelen niet behalen.*
    iii. *Jullie Scrum Master is de beste. Hij weet wanneer structuur en effectiviteit het meest belangrijk zijn. Maar, hij weet ook wanneer hij een stap terug moet nemen en het team erbij moet betrekken zodat iedereen gemotiveerd is om aan hetzelfde touw te trekken. Hij is in staat om zijn leiderschapsstijl aan de situatie aan te passen.*

6) Afsluiten interview

Er zijn nu alle vragen aan bod gekomen en wat mij betreft zijn we klaar met het interview. Heel erg bedankt voor je tijd. Heb je nog laatste vragen of opmerkingen?

Appendix C

Evaluation questionnaire

To be found at this link: https://utwentebs.eu.qualtrics.com/jfe/form/SV_5vdesZZEbrRP55P

p. 1

Hallo lezer,

In de afgelopen maanden heb je in het kader van mijn masterthesis onderzoek een training van mij gevolgd. In deze korte vragenlijst van 5 minuten kan je je mening en opmerkingen over het training kwijt en verbetervoorstellen maken. De data wordt anoniem verzameld. Bedankt!

Met vriendelijke groet,
Cosima Patzak

p. 2

Je wordt gevraagd om drie algemene vragen over het training te beantwoorden, op een schaal van 1 (1 = ik vond het helemaal niks) tot 7 (7 = het kan niet beter).

Hoe heb je de het training in het algemeen ervaren?
Hoe vond je de inhoud van het training?
Hoe vond je de manier op welke het training werd gegeven?

p. 3

De volgende vragen gaan over het ervaren nut van het training en de SMART doelen. Je gaat twee vragen beantwoorden op een schaal van 1 (1 = Helemaal niet nuttig) tot 7 (7 = enorm nuttig).

Hoe heb je het nut van het training ervaren, als je nadenkt over wat dit training je en je team heeft opgeleverd?
Hoe nuttig heb je de SMART doelen gevonden om te leren om betere actiepunten te maken?

p. 4

Nu heb je de mogelijkheid om een verbeteringsvoorstel te geven. Stel ik geef dit training aan een ander Scrum team. Heb je voor dat geval concrete voorstellen hoe ik mijn training kan verbeteren?

---

p. 5

Heb je nog laatste opmerkingen over dit training of wil je iets kwijt? Zo ja kan je dit hier omschrijven of contact opnemen met mij via c.d.patzak@student.utwente.nl.

---

p. 6

Bedankt voor je responsen en je deelname aan mijn onderzoek!
Met vriendelijke groet, Cosima Patzak