Universiteit Leiden

ICT in Business and the Public Sector

Understanding the benefits and challenges of introducing a dashboard to help Agile teams improve their work

Name: Yassin El-Baz
Student-no: s1689398

Date: 24/12/2022

1st supervisor: Dr. Christoph J. Stettina
2nd supervisor: Prof. Dr. Ir. Joost Visser

MASTER'S THESIS

Leiden Institute of Advanced Computer Science (LIACS)
Leiden University
Niels Bohrweg 1
2333 CA Leiden

Abstract

Purpose

This research aims to understand what the benefits and challenges are of introducing metrics in the form of a dashboard to Agile teams. The research question reads as follows: *What are the benefits and challenges of introducing a dashboard containing Agile metrics to Agile teams, in order for the teams to improve their work?*

Research Approach

The action research methodology is employed in order to test a PowerBI dashboard with seven Agile teams during a period of eleven weeks. The dashboard contains data about the team's quality, workflow health, responsiveness and employee & customer satisfaction.

Findings

The benefit of introducing a dashboard with Agile metrics was that it helped teams in their Agile process and administration, especially teams with a lower Agile maturity. The dashboard had the unintended positive effect of making scrum masters aware of how important data quality is to leverage value from metrics.

The challenges of the introduction of a dashboard containing Agile metrics were a) poor data quality decreased the value scrum masters could leverage from the dashboard, b) integrating the dashboard in the processes and routines of teams is difficult to achieve without proper change management, c) it is difficult to strike a balance between creating a generic dashboard template and accommodating the need for customization for each team and d) the dimensions 'customer satisfaction' and 'employee satisfaction' in the dashboard rely on subjective and manual input.

The dashboard proved useful to support the scrum master and their team (especially teams with a lower level of Agile maturity) in their Agile process and administration, but it cannot be concluded the dashboard contributed to improving the work of Agile teams.

Research limitations

This research represents progress in the understanding of how Agile maturity impacts the usage of metrics by Agile teams to improve their work. However, the conclusions drawn in this research calls for further investigation on a more quantitative basis to better understand and study this relationship between the concepts of actionable metrics and Agile maturity.

Practical implications

This research identifies how Agile metrics as part of a performance framework could help Agile teams to improve their Agile process and administration, by: (1) identifying the challenges in introducing such a dashboard, (2) how and when these metrics could help the teams in their Agile processes, (3) how Agile maturity of teams plays a role in how these teams view and use metrics and (4) how employee and customer satisfaction surveys can be integrated in a team's Agile process in an integrated manner.

Originality and value

This research appears to be unique in the fact that Action Research as a method was used to understand the benefits and challenges in introducing a dashboard to Agile teams. Additionally, this research qualitatively demonstrates how Agile teams use and perceive Agile metrics, considering their Agile maturity. This research also appears to be unique in the fact that an attempt was made to create customer and employee satisfaction surveys to be used in an Agile environment, based on existing contemporary satisfaction models.

Acknowledgements

First of all, I would like to express my gratitude to my first supervisor Dr. Christoph J. Stettina. Throughout my thesis internship and the writing of my thesis, he always provided me with insightful and helpful feedback. Dr. Stettina was always available to quickly talk and offer advice when I was stuck or unsure of something; any student would be lucky to have him as his or her first supervisor. Second, I would like to thank Prof. Dr. Ir. Joost Visser as my second supervisor. He also provided me with insightful feedback and advice during key moments of the writing process. Thanks to Gijsbert C. Boon as well for our informative talks and brainstorm sessions.

I would also like to express my gratitude to the host company and the supervision I received from my colleagues. Thank you for the opportunity to have written my thesis under the supervision and mentorship of all the enthusiastic colleagues. I really enjoyed my time at the host company and working with everyone, whether I worked with you directly or indirectly.

Finally, I would like to thank my family for their relentless encouragement and support, both during this thesis internship and throughout my academic career.

Table of contents

List of Tables

List of Figures

1 – Introduction

A dashboard with metrics can help Agile teams to objectively measure their performance (Davis, 2015). Measurement is crucial because it provides an objective view of the current situation, from which one can improve upon (Boon & Stettina, 2022). However, it is unclear a) which metrics in the form of a dashboard could help Agile teams to improve their work, and b) what are the benefits & challenges of introducing such a dashboard with Agile metrics to Agile teams.

The available literature on Agile metrics is abundantly discussed, along with the potential benefits of using such metrics in practice. However, to leverage the most value from these metrics, it is important to understand the bigger picture of how a dashboard with Agile metrics contributes to Agile teams and the improvement of their work and what the challenges are of introducing such a dashboard. If this is not adequately researched and answered, the introduction of such Agile metrics or a dashboard will most likely be less successful or will take more time to achieve the desired results.

A potential solution to the aforementioned problem could be to test the introduction of a dashboard containing Agile metrics to a selection of Agile teams, through the action research method. This dashboard would be used by the scrum masters and their teams to improve their work (e.g. optimize their workflow). A way of categorizing the metrics in the dashboard in a coherent manner would be to use either an existing performance management/measurement framework or create a new one.

By developing a dashboard with Agile metrics (both using the available literature on Agile metrics and considering what is technically feasible) and testing the introduction and use of these dashboards, the benefits and challenges of introducing such a dashboard can be researched. By including a variety of Agile teams with different Agile maturity levels in this 'experiment', one can draw conclusions on *why* certain teams experience more benefits or challenges than others.

Placing and connecting the selected individual Agile metrics in a broader performance management or measurement framework can help in categorizing these Agile metrics and connecting the categories to a more tactical or strategical level. For example, the Agile metrics can be placed in a performance measurement framework (divided into several categories); these categories can then be used as input to track and monitor objectives & key results.

To attempt to solve this problem with the potential solution, this research question is posed:

> ➢ *What are the benefits and challenges of introducing a dashboard containing Agile metrics to Agile teams, in order for the teams to improve their work?*

The first stage to answer the aforementioned research question is to discuss relevant literature on performance management/measurement and Agile metrics in chapter 2. Since the main purpose of an Agile project is to maximize delivery of business value (Racheva et al., 2009), assessment of customer satisfaction is also discussed in chapter 2. To ensure longevity of the team, it could be argued employee satisfaction is also important; this is also discussed in chapter 2. The gap in literature and methodology are discussed in chapter 3 and 4 respectively. In chapter 5, the action research framework is applied, along with the performed activities and the associated results. Chapter 6 contains the reflection & discussion. The conclusion is presented in chapter 7. The bibliography and appendices are to be found at the end of this research paper.

2 – Literature Review

In this chapter, the existing literature is explored on several topics. First of all, a comparison will be made between 'traditional project and product management' and 'Agile project and product management'. There are various practical frameworks on how to apply Agile, two of the most popular ones are Scrum and Kanban; both these frameworks will be briefly discussed. Since projects and products require some sort of governance process, a brief explanation of the Agile governance process will be given. An overview of measurement practices and metrics used in an Agile context is also discussed. This research focuses in part on Agile metrics that help and enable Agile teams to improve their work. This relates to the concept of 'performance management', this is also discussed in relation to performance management and measurement frameworks.

2.1 – Agile project & product management

As the title of this thesis suggests, Agile is an important theme in this research. This section will focus on what Agile entails and how it is used in project and product management. Additionally, this Agile methodology will be compared to more traditional forms of project and product management, since Agile methodologies promise advantages over traditional methodologies. An example of this promised advantage would be an increase in customer satisfaction, paired with lower defect rates and quicker time to develop (Sharma et al., 2012). It is thus interesting to quickly provide an overview of the key differences between traditional project management and Agile project management before delving deeper into the Agile methodology.

Table 1: Key differences between traditional project management and Agile project management (Engelhardt, 2019; Kaur & Bahl, 2015)

Category	Traditional Project Management	Agile Project Management
Development model	Traditional life-cycle model	Iterative
Focus	Process	People
Management	Controlling and commanding	Facilitating, leadership and collaboration, autonomous teams
Customer involvement	Requirements gathering and delivery phases	Constantly involved
Developers	Work individually within autonomous teams	Collaborative and in pairs
Technology	Any	Mostly object oriented
Product features	All included	Most important first
Testing	End of development cycle	Iterative, continuous and/or drives code
Documentation	Thorough	Only when needed
Communication	Formal	Informal

Before exploring the Agile methodology in-depth, the roots of project and product management will be discussed to better understand the reasoning and history of Agile.

Defining project management

It is useful to define what a 'project' is, so we can contrast this with the definition of a 'product' later on. A project is "*the organization of people and resources to achieve a defined objective and purpose*" (Lockett et al., 2008); it is characterized by having a defined time for completion, limited budget, well defined and preset objectives, and a series of activities to achieve those objectives

(Lockett et al., 2008).

The traditional management of projects is defined as "*the application of knowledge, skills, tools and techniques to project activities to meet project requirements*" (Project Management Institute, 2013). Traditional project management involves the completion of five phases: initiating, planning, executing, monitoring & control and eventually closure. Two of the most common implementations of traditional project management are the spiral model and the waterfall model (Ahmad et al., 2016).

History of project management
In 2005, Kwak identified four distinct periods in the history of modern project management. The first period identified is 'prior to 1958'. Kwak argues the origins of modern project management originated between the 1900s and 1950s. It transformed to what Kwak calls 'Human Relations Administration'. Better telecommunication and transportation systems enabled higher mobility and faster communication. Gantt charts (a classic project management tool) were also developed and in use during that period. The second period spans from 1958 to 1979, the main theme in this period was the application of Management Science and technological advancements. The institutionalization of project management started with what we now know as the International Project Management Association (IPMA). A few years after the founding of IPMA, the Project Management Institute (PMI) was founded. PMI is the publisher for the Project Management Body Of Knowledge (PMBOK). The third period, spanning from 1980 to 1994, is characterized by the multitasking abilities of personal computers (PCs). The efficiency of these personal computers enabled developers to build software which is capable of organizing and handling complex and large amounts of data to manage projects. With these increasing complexities and ability to multitask, Scrum was introduced as an Agile software development method. The fourth and last period spans from 1995 to present; technology continues to disrupt and enable change, which has a great impact on what project managers do. In this fourth period, the Agile Manifesto was written. (Seymour & Hussein, 2014).

Defining product management
A product is defined as "an item or service offered […] and designed to meet the needs or wishes of customers" (Airfocus, 2022). In the context of product management, a product is defined as "the result of management activities that provide users with unique opportunities (i.e. value to customers)" (Airfocus, 2022).

Products (in this context, software products in particular) are usually part of an overarching portfolio. Each product has a release sequence of past, present and future releases, and each release has multiple requirements. Since product management differs on each distinct hierarchy level, software product management can be divided into four process areas (Weerd et al., 2006)

- *Portfolio management* to handle the product portfolio which contains products;
- *Product roadmapping* to handle the product's different releases;
- *Release planning* to handle requirements gathering of each release;
- *Requirements management* to handle each individual requirement's administrative data.

Considering the just explained definitions and roots of project and product management, the transition from a traditional perspective to an Agile one is able to take place. Agile is an iterative approach to project & product management and software development that helps teams deliver value to their customers; it's a collection of methods that concentrate on a commitment to improving continuously and providing feedback (Atlassian, n.d.-c).

As discussed in the history of project management, the Agile Manifesto was written in 2001 (Seymour & Hussein, 2014) and provides the best starting point to dive into the theory of Agile (Fowler & Highsmith, 2001)

The Agile Manifesto
This Agile Manifesto defines values and basic principles for better software development; it has been widely adopted by mainly developers in organizations which focus on software development, but also outside the world of IT (Hohl et al., 2018). While there are 12 principles to be found in the Agile Manifesto, it would be cumbersome and excessive to list them all in this thesis. Instead, only the most relevant principles (Fowler & Highsmith, 2001) to this thesis will be highlighted and expanded upon.

1. "**Our highest priority is to satisfy the customer through early and continuous delivery of valuable software.**" This principle highlights the customer-centric approach in the Agile methodology. An Agile team may develop state of the art applications for its customer, but if the customer is not satisfied and does not see the value of certain decisions, one can ask how useful the expended effort was by the Agile team.
2. "**Business people and developers work together daily throughout the project.**" This principle shows the fact that it is not only about releasing a new version as fast as possible, but also about the dynamics of stakeholders and how important this is for the success and value of the project. Collaboration and trust are important throughout the project.
3. "**The most efficient and effective method of conveying information with and within a development team is face-to-face conversation.**" As shown in table 1, the communication in an Agile environment takes place in an informal way, as opposed to a formal way in traditional project and product management environments.
4. "**At regular intervals, the team reflects on how to become more effective, then tunes and adjusts its behaviour accordingly.**" Not only does the team develop a product or service for the customer in an iterative way, but also reflect on themselves to continuously improve and excel.

Agile product management
While there is a stark contrast between traditional project management and Agile project management, this contrast seems less prevalent when comparing traditional product management to Agile product management. The four principles of traditional product management (portfolio management, product road-mapping, release planning and requirements management) are applicable to Agile product management as well. The difference would then be on how exactly these four principles are used in practice. The Agile frameworks such Scrum and Kanban give practitioners a way to implement the four product management principles in a truly Agile fashion. For example, requirements management is the fourth principle of traditional product management, but in an Agile context this should be done as lightweight as possible because of the focus on value instead of extensive unnecessary documentation.

Agile project management
Agile Project Management is characterized by short cycles of incremental delivery of features and code changes which are integrated continuously (Dybå et al., 2014) in a software development environment. While there seems not to be a consensus in the literature on what the exact definition is of Agile project management, there are four principles which cover the essence of Agile project management (Dybå et al., 2014):

- *Minimum critical specification*: Only that which is essential and critical to the overall success, should be specified;
- *Autonomous teams*: teams should be responsible for monitoring and managing their own task execution and processes;
- *Redundancy*: Team members should be skilled in more than one field;
- *Feedback and learning*: Feedback and learning should be integral during project execution.

This approach is the complete opposite of the traditional project management where well-defined activities are executed linearly as discussed before. Agile project management also introduces changes in management roles as well in practices. While traditional management was viewed as something that controlled processes, Agile project management focuses on facilitating people (Hoda et al., 2008).

Agile project governance
Now the contrast between Agile and traditional approaches to product and project management have been made clear, it may not be as obvious as how Agile connects to performance management and measurement.

To understand performance measurement in the context of Agile teams and how metrics are embedded in Agile project governance, it is useful to look at the work of Lappi et al. (2018) on Agile project governance. Lappi et al. (2018) conducted research on governance-related practices used in Agile projects and products. Lappi et al. (2018) identified six key governance practices based on an extensive literature study:

1. *Goal-setting practices*: concerned with defining a project's goal, purpose and alignment on a strategical level.
2. *Incentive practices*: concerned with increasing the team's motivation and incentives.
3. *Monitoring practices*: concerned with tracking the team's progress in a visual manner, such as with team performance-related KPIs.
4. *Coordination practices*: concerned with facilitating the mechanisms of communication and involving customers and other stakeholders.
5. *Roles and decision-making power practices*: concerned with the autonomy of the team and its responsibility.
6. *Capability building practices*: concerned with building the ideal distribution of capabilities within the team and the process of exchanging knowledge.

This research will study the third above mentioned principle (monitoring practices), with a strong focus on team performance-related metrics. Due to this aim, a thorough explanation and comparison of performance measurement and management frameworks will be given in chapter 2.2.

2.1.1 – Agile Frameworks
As previously mentioned, there have been several frameworks developed based on the Agile methodology.

Agile Scrum Framework
One of the most popular Agile frameworks is the Agile Scrum Framework.

Scrum is a framework that assists teams and organizations to create value with adaptive and iterative cycles. It uses an iterative and incremental way to increase the predictability and to decrease the potential risks. (Schwaber & Sutherland, 2011)

According to the Scrum Guide, Scrum is based on empiricism and lean thinking. In practical terms, empiricism in a Scrum context states that knowledge is sourced from experiences and basing decisions on what is observed. Lean thinking focuses on reducing waste (e.g. unnecessary waiting) and an emphasis on the essential value-added components of a process. (Scrum.org, 2022)

In a nutshell, the Scrum process can be visualized in a simple manner, as is shown in figure 1 (Scrum.org, 2022):
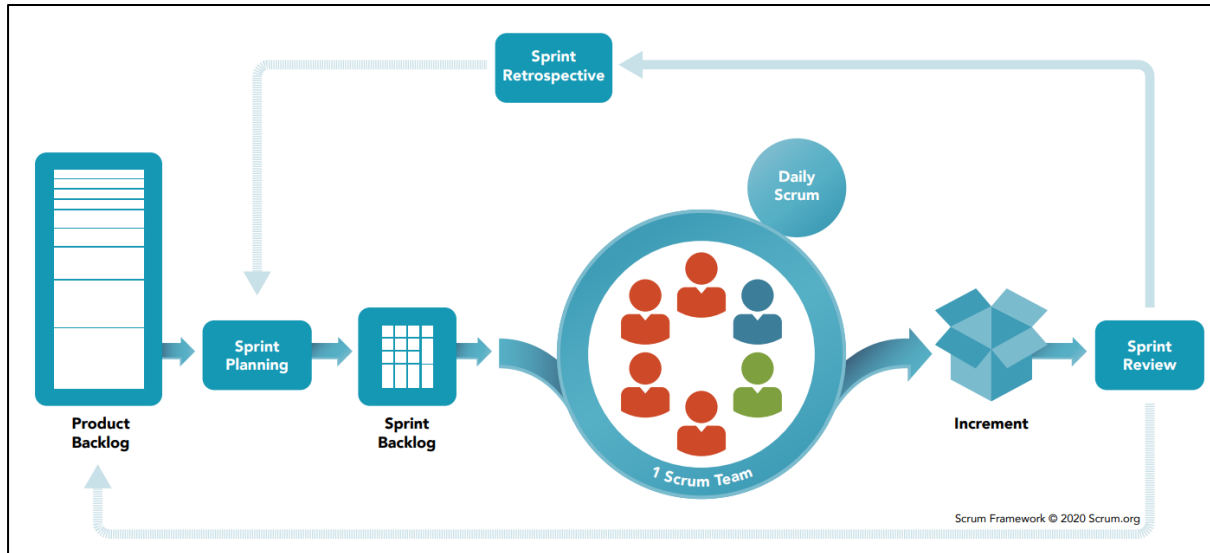
Figure 1: An overview of the Agile Scrum process (Scrum.org, 2022)

The product owner orders the work to be done for a complex problem that the team is solving into a product backlog. A sprint planning takes place to determine which work from the backlog will be performed for the upcoming sprint. This selection of work from the product backlog moves to the sprint backlog. Then, the team will perform the actual work during the sprint, with a daily scrum as ceremony; this is a daily stand-up to inspect the team's progress toward the goal of the sprint and if necessary, adapt the sprint backlog (e.g. if the selected work turned out to be more time intensive than initially thought, it will be moved to the next sprint's backlog). At the end of the sprint, the team has completed another increment of the specific product they are responsible for. This will be shown to the stakeholders to inspect the results and gather possible feedback for the next sprints. A sprint review takes place after every sprint, in addition to a sprint retrospective. The purpose of the sprint review is to inspect the outcome of the sprint (i.e. the increment), which could provide input for future adaptations for the product/increment. The sprint retrospective is used to reflect on the overall process of the tools, people, communication and team members. (Scrum.org, 2022)

Kanban
While Scrum has its origin in software development, Kanban has its origin in Lean manufacturing and was pioneered by Toyota with its Toyota Production System (Siderova, 2022). While manufacturing is not exactly the same as software development, Kanban is still applicable. Software development teams can leverage the 'Just-in-time'-principles by matching the amount of work in progress (WIP) to the capacity of the team. This allows the team to plan more flexibly, to have a faster output and achieve increased transparency during the iterations. The work of Kanban teams is contained on a Kanban board, a tool to visualize the team's work and to optimize the team's workflow. (Atlassian, n.d.)

Scrumban

It may happen that teams mix and combine the concepts of Scrum and Kanban into something that's called 'Scrumban'. Scrumban is a relatively new hybrid approach that consolidates the technical and theoretical compromises between the two frameworks. Additionally, Scrumban attempts to optimize the effort of the team to achieve certain quality standards. (Stoica, 2016)

In Scrumban, defining long-term development goals is done with bucket size planning. Each bucket represents a development plan with definitions of tasks. The nearer a bucket is to this point in time, the more detailed the tasks are. This corresponds to the Just-In-Time principle; details should be made and considered as late as possible. Scrumban limits the amount of work in the 'Work In Progress'-state. Similar to Scrum, Scrumban also has 2-week iterations. One new addition compared to Scrum would be that Scrumban has the concept of 'Feature Freeze' (FF). This FF entails that when the team is nearing the end of the current iteration, the team stops the work on new features and only focus on finishing the features that are almost complete. (Brezočnik & Majer, 2016)

2.1.2 – Individual Agile metrics and visualizations

Before the various individual Agile metrics and visualisations in the literature are discussed, it is useful to understand why measurement is necessary in the first place. It has already been discussed how metrics fit into the governance of Agile projects in section 2.1.1, but even outside this Agile context, measurement is still useful.

Scope of explored metrics

While financial metrics (e.g. revenue per employee and product cost ratio) can be useful indicators for a project's success, this falls outside of the scope for this research project.

Oza & Korkala (2012) classified metrics used in the Agile software development process in three categories: code level (e.g. test automation, code quality), productivity/effort level (e.g. burndown charts, velocity) and economic metrics (e.g. return on investment). The scope of this research will be focused on the productivity/effort level, which is synonymous with performance in this research.

Why is measurement useful?

Measurement is fundamental to justify change and it provides an objective reference from which to understand, learn and improve upon (Boon & Stettina, 2022). Furthermore, the use case for metrics are also related to supporting communication between stakeholders and decision making (Kupiainen et al., 2015). Kupiainen et al. provide a summarization of motivations for using metrics, based on the work of Pulford et al. (1995) and Grady (1992):

- Project estimation and planning;
- Project tracking and management;
- Understanding business and quality objectives;
- Improves software development communication, tools and processes.

The reasoning for using metrics is clear, but what exactly constitutes a suitable metric to be used in an Agile context? For this, we look at the heuristics for good Agile measurements.

Heuristics for good Agile measurements

Ktata & Lévesque (2010) suggest nine heuristics for good Agile measurements, based on the works of Deborah Hartmann & Robin Dymond (2006) and Levison (2009). A good Agile metric:

- Reinforces and affirms Lean and Agile principles;
- Measures outcome, not output;
- Follows trends, not numbers;
- Belongs to a small set of diagnostics and metrics;
- Is easy to collect;
- Reveals instead of conceals its context and significant variables;
- Provides triggers for meaningful conversation;
- May measure the product (value) or process;
- Stimulates 'good-enough' quality.

List of individual Agile metrics and visualizations

Table 2 contains Agile metrics discussed in scientific literature. The table only contains a brief explanation on what each metric or visualization shows or entails. The detailed explanation of each of the discussed metrics in table 2, can be found in Appendix B. Appendix B contains an extensive explanation for each metric, answering these three questions: (1) What does it show?; (2) What are the merits of using it?; (3) What are the pitfalls & considerations?.

Table 2: Overview of productivity/performance metrics in an Agile context, based on Appendix B

Name	What does it show or entail?
Velocity	An indication of the average amount of product backlog turned into an increment of a product during a sprint by a scrum team.
Throughput	Average number of work items processed per unit of time (e.g. # of user stories finished in a sprint).
Lead time	Amount of time passed from a the registration of the request to the completion of the request.
Cycle time	Amount of time passed from when work actually started on the request to completion of the request (cycle time is a component of lead time).
Flow efficiency	Amount of active time used in 'working' status and the amount of non-active time in 'waiting' status. A good flow efficiency starts from 40%. (Mas et al., 2020)
Mean time to repair	Average time it takes from when an error or bug is discovered to when it is solved (Scrum.org, 2020).
Open defect severity	A visualization of how many defects/bugs are currently open and their associated severity level.
Sprint Burndown chart	A visualisation of how the planned and actual work of the sprint has been completed. In an ideal situation, there would be zero deviation from the planned completion.
Release Burndown chart	Similar to the sprint burndown chart, however this release burndown chart documents the progress of the release instead of sprint.
Product Burndown chart	Similar to the sprint burndown chart, however this product burndown chart documents the overall product progress.
Cumulative Flow Diagram	A stacked area chart showing at each time interval the number of items in each stage of the process (e.g. backlog, in progress, review, done).
Accuracy of estimation	Shows how accurate the team is in their estimation of work items during a sprint/multiple sprints. This is calculated as the difference between estimated work and actual work (in hours, story points or days).
Release stabilization period	The time spent correcting product problems between when the developers say it is ready to release, to the point where it is actually released to the customers or end-users (Scrum.org, 2020)
Time in status	The average time a task spends in a specific status (e.g. in progress, review).
Number of bugs	The number of bugs, both at this moment in time (metric) and over time (visualization).
Technical debt	"Design or implementation construct that is expedient in the short term, but sets up a technical context that can make a future change more costly or impossible" (Leffingwell, 2021)
Productivity index	A metric that shows how much of the team's time is actually spent on the work, compared to non-productive time spent.
Defect removal efficiency (DRE)	The % of bugs detected internally compared to the amount of bugs that were detected externally. The DRE's formula is "(number of defects found internally / number of defects found internally + number of defects found externally) * 100%" (TestMatick, n.d.)
Backlog size health	Cumulative number of story points with the status 'ready to pick up' divided by the average velocity. A optimal value should lie between 2 and 3 (Agile Tools, n.d.)
Innovation rate	Percentage of effort/cost spent on new product capabilities, divided by total product effort or cost.

Impediments / blocker overview	The amount of impediments or blockers the team is experiencing that hinders (a part of) their work.
Bugs solved	Shows the (historic) amount of bugs that have been solved by the team.
Work Item Age	Amount of time passed between a work item that was started and the current time. It complements cycle time, which is only applicable for finished work items (Kanbanize, n.d.), while work item age is (also) applicable to work items in progress.
Customer Usage Index	Degree of usage of features to help decide the degree to which customers find the product/feature useful and whether actual usage meets their expectations. (Scrum.org, 2020)
Number of releases	The (historic) number of releases completed by the team.
Stakeholder satisfaction/trust	The (periodic) assessment of stakeholder satisfaction with the team and the product.
Customer/user satisfaction gap	The difference between a customer's current experience and their desired experience. (Scrum.org, 2020)
Scheduled Performance Index	How much work the team has completed each sprint.
Enhancement Rate	The enhancement rate shows for each sprint what percentage of the previous sprint has gone into production or acceptance in the current sprint. (Boerman et al., 2015)
Scope Prognosis	The scope prognosis is the average of the enhancement rate, combined with the remaining project size and the last expected iteration. It shows an indication of the anticipated degree of functional completeness. (Boerman et al., 2015)
Project size remaining	The remaining project size shows the amount of work that is remaining for the project. (Boerman et al., 2015)
Changed PBIs	It shows if the description of a PBI is different from the description of the PBI in the previous sprint. It can be shown as a percentage; the percentage of changed PBIs compared to the total number of PBIs. (Boerman et al., 2015)
Added PBIs	Added PBIs indicates the percentage of PBIs which are added to the backlog of a sprint, which were not present when the iteration started. (Boerman et al., 2015)
Rejected PBIs	The number or percentage of PBIs which have been rejected from the backlog compared to the project size. (Boerman et al., 2015)
Project size	The size of the backlog. Either by using the total effort estimation or simply counting the number of PBIs. (Boerman et al., 2015)
Time prognosis	The time prognosis shows in which iteration the product is likely to be finished. (Boerman et al., 2015)
Priority shift	Priority shift simply shows if a PBI was assigned a different estimation in the backlog (Boerman et al., 2015).
Estimation shift	The estimation shift shows the difference (positive or negative) in the estimated effort of a sprint compared to the previous sprint; it returns the sum of these. (Boerman et al., 2015)
Effort at risk	Before a PBI is pushed to production and adds value to the software product, there is a risk that any effort made is lost, for example due to changing priorities or budget cuts. Thus, the effort at risk metric shows the effort that has not yet been converted to product value. This metric has three categories: low risk, medium risk and high risk. As the PBI moves from the design-phase to the production-phase, the risk decreases. (Boerman et al., 2015)

What has been observed elsewhere in applying metrics?

While metrics in project and software development are often mentioned in relation with Agile methodologies, the Agile method is not the only one that can be used to deliver a project or product effectively or efficiently. While discussing every methodology is outside the scope of this research, we will quickly touch upon the Waterfall model. While the discussed metrics in Appendix B are primarily focused on an Agile Scrum environment, these metrics can also be (partly) applied in a Kanban or Scrumban environment. Sometimes this is the case because some of these metrics were used in a Kanban environment before the Agile (Scrum) methodology adopted and used these metrics as well.

The Waterfall model is an approach to project and software development, where a project is split up into multiple sequential phases. These phases are (1) Requirements, (2) Design, (3) Coding, (4) Testing and (5) Deployment/Maintenance. (Andrei et al., 2019)

Metrics can be collected and used at each one of these five stages (Paul et al., 1999). For the Requirements stage, requirements metrics are applicable; these can range from traceability to completeness and from size to unambiguity. To measure and track these areas, automated tooling can be used, such as Dynamic Object Oriented Requirements Systems and IBM Rational Rose. (Bokhari & Siddiqui, 2011)

For the second stage (Design), metrics can be used to compare the quality of the various drafted designs (Paul et al., 1999). While deciding how to measure the design can be subjective, the literature provides some guidance on how to quantify design. In the case of a web page, Ivory et al. (2001) use metrics like 'Link Count', '% of page bytes that are for graphics' and 'text positioning count' to measure the design of web pages.

For the third stage (Coding), plenty has been written on how to measure and track the actual development of software. Aspects such as maintainability, quality and architecture (Paul et al., 1999), can be measured with a range of code-level metrics.

Testing – the fourth stage – is also an oft-tracked and -measured concept. Nirpal & Kale (2011) offer a range of metrics to be used to test software; for example the defect rejection rate, defect severity index and the test execution productivity.

The last stage – Deployment/Maintenance – is less about a process and more about an action. After thoroughly testing the product, the team can deploy it to the customer. Simple metrics can focus on the number of failed or successful deployments (Pfeiffer, 2021), but overall the literature does not seem to consider this an important part to track and measure in the context of the waterfall methodology.

Actionability of metrics

There is abundant literature on Agile metrics, explaining the purpose and benefits of each Agile metric. However, the literature barely covers the topic of actionability of metrics in an Agile context. There are only a handful of studies that address this topic.

The research of Ram et al. (2020) attempted to conceptualize and define the concept of an 'actionable metric'. They did this through a questionnaire, asking respondents whether they agreed an actionable metric should meet certain criteria. Their results showed an actionable metric should inform decision-making, be practical and exhibit high-quality data characteristics (Ram et al., 2020). They did not discuss which of the existing metrics could be classified as actionable, that was outside the scope of their research.

In the work of Vacanti & Vallet (2014), actionable Agile metrics are studied and discussed. They found that using traditional Agile metrics, such as velocity, never lived up to the promises of greater transparency and predictability. In their case study of Siemens Health Services, they found that using actionable metrics of flow (i.e. cycle time, throughput and work in progress) resulted in greater efficiency and a reduction in lead time of 42%. The reason for this, as stated by the authors, is the fact that these metrics provide a better view of the teams' progress. Additionally, these metrics were more actionable because ownership of the team was possible; individual team members could act upon those flow metrics to improve the overall performance of their processes.

In the book 'Actionable Agile Metrics for Predictability' by Vacanti (2016), the emphasis is placed on flow metrics such as cumulative flow diagrams, cycle time scatterplots and throughput. This book is inspired by the paper mentioned in the previous paragraph, so the key takeaways are the same.

The work of Tekin et al. (2020) explored several software solutions for Scrum Software Development metrics. They found using four flow metrics (cycle time, throughput, work in progress and work item age) of one particular software package scored the highest in their assessment.

Overall, the literature on actionable metrics and the actionability of metrics is quite limited.

2.2 – Performance management

As Lebas (1995) states, few people agree on what performance really means; from efficiency to robustness. According to Lebas (1995), since performance cannot be defined objectively (since everyone seems to have a differing opinion on what constitutes performance, let alone how to manage it), it is important to posit a conceptual definition for this thesis and work from there. Furthermore, the difference between performance management and performance measurement will be clarified and existing frameworks will be discussed. Since customer and employee satisfaction is included in some of the frameworks, these two concepts will be explored after the existing frameworks have been discussed.

2.2.1 – Defining performance management

Searching through the available literature on the topic of performance management yields vastly different definitions. These definitions can be clustered in two ways: Human Resources and team & organizational performance.

Human resources or talent management perspective

The first way to approach the subject of performance management, is from a human resources or talent management point of view. Here, performance management is about the performance appraisal and management of employees on issues relating to pay, training, performance reviews and individual objectives (Dransfield, 2000; Den Hartog et al., 2004). As Pulakos (2004) states; performance management could provide HR-professionals with useful guidelines to develop and implement performance management systems. In this context, performance management is synonymous with how to measure and reward performance. The book of Armstrong & Murlis (2007) extensively uses the term performance management as a synonym to reward management in his book 'Reward management: a handbook of remuneration strategy and practice'.

Den Hartog et al. (2004) define performance management in the context of human resource management as "*performance management deals with the challenge organizations face in defining, measuring and stimulating employee performance […]*". Dransfield (2000) states that "*performance management is a human resources management […] concerned with getting the best performance from individuals in an organization […]*".

Another definition of performance management is from Armstrong & Baron (1998), which states performance management is a "*strategic and integrated approach to delivering sustained success to organizations by improving the performance of the people who work in them and by developing the capabilities of teams and individual contributors*". While this is an extensive definition, it does seem to look at performance management through a human resources-lens because of the "*..developing the capabilities of teams and individual contributors*" and because the book of Armstrong & Baron is primarily written from an HR-perspective. The scope of this research project does not consider this and would thus not be the best definition to use.

Efficiency and effectiveness of teams on a process-level
The second way to view performance management is from a non-HR perspective. For lack of a better category, this second perspective focuses on the efficiency and effectiveness of teams on a process-level. Here, performance management may still apply to individual employees, but is not concerned with HR-aspects such as pay and performance reviews. It is more about how the employee in the context of a team and the team itself can become more efficient and effective by measuring and managing their performance. For example, cycle time can be a measure for performance (Ghaliyini & Noble, 1996) in Agile teams, which can be used to understand and improve the process.

Armstrong (2006) defines performance management as a "*systematic process for improving organizational performance by developing the performance of individuals and teams*". It is a way of achieving better results from stakeholders by understanding and managing performance within an agreed upon framework of goals, requirements and standards (Armstrong, 2006).

While the definition of Armstrong is certainly not the only definition for performance management, it does seem to cover more theoretical ground than other definitions in literature. For example, the definition of Mohpman & Mohrman (1995) states that performance management is "*managing the business*", which may be true in essence, but is not very helpful in applying it in practice.

Thus, I would argue the posited conceptual definition of performance management is "*a systematic process for improving organizational performance by developing the performance of individuals and teams*" (Armstrong, 2006). This definition acknowledges performance management is a process (continuous), and looks towards the individual- and team-level (the scope of this research) to increase their performance in a systematic manner (framework).

2.2.2 – Difference between performance management and performance measurement
While performance management has already been defined for the scope of this thesis, the term performance measurement appears to accompany performance management in the literature. For the sake of clarity, it is useful to highlight the distinction between and similarity of these two terms.

According to Lebas (1995), performance management precedes performance measurement and gives it meaning. In more practical terms, this means performance management is a philosophy which is supported by performance measurement. They follow one another in an iterative process. As figure 2 shows, performance management and performance measurement are closely intertwined according to Lebas (1995).
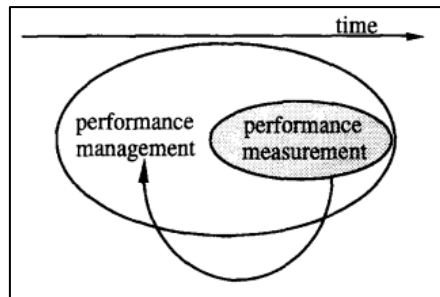
Figure 2: Performance management creates the context for measurement. Performance management precedes and then follows performance measurement (Lebas 1995)

Performance measurement is a component and step in the process of organizational performance management (Armstrong, 2006).

Amaratunga & Baldry (2022) state it similarly: "*In order for an organization to make effective use of its performance measurement outcomes it must be able to make the transition from measurement to management*". Measurement is thus not an end in itself, but a tool for effective management.
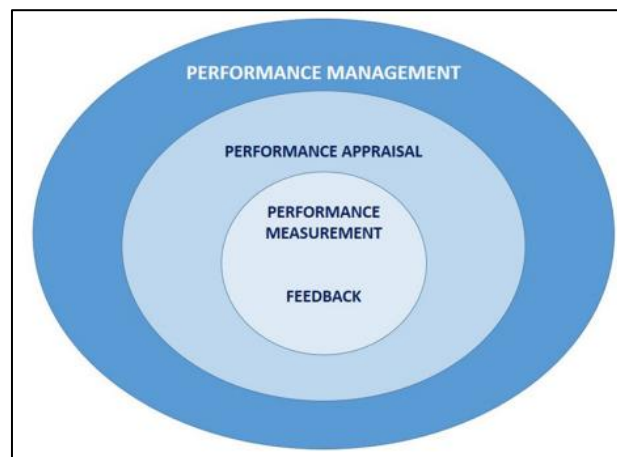
Figure 3: The relation between performance management and performance measurement from an HR-perspective (Justin & Joy 2022)

Justin & Joy (2022) seems to agree with the notion that performance measurement is part of performance management, as seen in figure 3. Although figure 3 seems to look at the relation between performance management and measurement in light of a human resources perspective.

Since we have established performance measurement is a component of performance management, we can define performance measurement as "*the process of quantifying effectiveness and efficiency of actions*" (Neely, 1999). Its purpose is to monitor and improve the performance of these actions on a continuous basis; it is a mechanism for continuous improvement (Eriksson, 2017).

2.2.3 – Existing performance management & measurement frameworks

The conceptual definition of performance management has been stated and the possible confusion between performance management and measurement has been cleared up. But before the existing performance management frameworks will be explored, it is important to understand which issues or questions these frameworks try to address.

According to Otley (2008), there are several issues that are addressed in developing such a performance management framework:

1. What are the key objectives that are central to the organization's overall future success, and how does the organization evaluate its achievement for each of these objectives?
2. What strategies and plans has the organization adopted and what are the processes and activities that will be required for it to successfully implement these strategies and plans? How does it assess and measure the performance of these activities?
3. What level of performance does the organization need to achieve in each of the areas defined in the above two questions, and how are appropriate performance targets set?
4. What rewards will managers and other employees gain by achieving these performance targets, or which consequences will they suffer if they fail to achieve them?
5. What are the information flows, feedback and feed-forward loops that are necessary to enable the organization to learn from its experiences, and to change its current behaviour in the light of those experiences?

Comparing performance management and measurement frameworks

The following section of this research provides a short breakdown of the major performance management and measurement frameworks which are found in scientific literature. However, the discussed frameworks adopt and consider wildly different perspectives and considerations. For example, on the one hand Çiçek et al. (2005) propose a performance measurement model to provide feedback to teams with a focus on continuous improvement. On the other hand, the Performance Prism proposes a list of questions for organizations and teams to use when defining performance measures. These questions assist in making sure all aspects of stakeholder satisfaction in relation to performance are sufficiently covered.

From this example, we are able to see that two models that both attempt to help the practitioner in the field of performance measurement, adopt a different focus and perspective to achieve this goal. It was thus deemed necessary to compare the discussed frameworks, both to gain a better understanding of these frameworks and to observe on a high-level which perspectives or dimensions these frameworks (not) cover.

This comparison makes use of six dimensions of organizational performance: Productivity, Responsiveness, Quality, Workflow health, Employee satisfaction & engagement and customer satisfaction (Stettina et al., 2021). For each of these six dimensions of organizational performance, an examination was done on how these six dimensions are covered by each performance framework. This mapping can be found in Appendix A.

To understand performance management and measurement from a theoretical perspective, several existing frameworks have been identified from the literature. The following have been identified and explained in this thesis: Economic Value Added, the Balanced Score Card, the Software Value Map, Çiçek's Team Performance Model, Keegan's Performance Measurement Matrix, Fitzgerald's Framework of Performance Measurement, the SMART Pyramid, Brown's Macro Process Model, the Performance Prism, the EFQM Business Excellence Model, Kanji Business Excellence Measurement System and the Evidence Based Management Guide.

Economic Value Added

Economic Value Added (EVA) is intended as an overall measure of financial performance that focus the managers' attention to the delivery of shareholder value. Traditional measures of financial performance (e.g. return on investment, profit) are clouded by inherent defects which may cause

dysfunctional thinking and decision-making by managers. One of these problems was the fact that managers do not understand where and how value is ultimately created. (Otley, 1999)

In order to solve the inherent defects of traditional financial performance measures, Stern and Stewart developed the Economic Value Added performance measure in 1991. (Sabol & Sverer, 2017; Otley, 1999)

EVA is defined as "accounting profit less a charge for capital employed" and is claimed to be less affected by these aforementioned inherent defects. (Otley, 2008)

While there has been extensive research conducted on the merits and deficiencies of EVA as a financial performance measure (such as the evaluation of EVA by Crowther et al. in 1998), financial measures are outside of this thesis' scope.

Balanced score card

The Balanced Score Card (BSC) is a strategic tool used to align business activities to the strategy and vision of the organization, improve communication and monitor performance of the organization against strategic objectives. It is a tool used for performance measurement that considers not only traditional financial measures but also business processes and customer satisfaction. (Nzuve & Nyaega, 2013) Kaplan & Norton introduced the BSC in 1992 because they believed measurement was fundamental to managers; if organizations wanted to improve the management of their intangible assets, they would need to accommodate measures of these intangible assets into their existing management systems (Crowther et al., 1998; Kaplan, 2009).

The original scorecard required managers to answers four basic questions, essentially looking at the business from four related perspectives (Kaplan & Norton, 2005; Kaplan, 2009):

- Customer/external perspective → "*To achieve our vision, how should we appear to our customers?*"
- Internal perspective → "*To satisfy our shareholders and customers, what business processes must we excel at?*"
- Innovation & learning perspective → "*To achieve our vision, how will we sustain our ability to change and improve?*"
- Financial perspective → "*To succeed financially, how should we appear to our shareholders?*"

In each of these four perspectives, multiple related goals with associated quantifiable measures are formulated to measure an organization's progress towards achieving those perspectives. One example would be the goal 'Succeed' with the accompanying measure 'Quarterly sales growth and operating income by division' for the financial perspective (Kaplan & Norton, 2005).

A theoretical model without any empirical evidence of its merits will not inspire potential teams and organizations to adopt and use the model in practice. Multiple studies have been conducted on the BSC's effect on performance. One of such studies is (De Geuser et al., 2009); they surveyed 76 business units and the results indicate that the BSC has a positive impact on organizational performance. More specifically, they state "*the BSC improves the integration of the management processes and empowers people*" (De Geuser et al., 2009). The study of Malina & Selto (2001) also indicates the BSC in their case study was an effective method to control corporate strategy. Additionally, this study of Malina & Selto observed disagreement and tension between top and middle management with regards to how appropriate the BSC was as tool to communicate, control and evaluate. Overall, the results of their case study were positive, indicating changes in processes and improvements in customer-oriented services.

A study of Neely et al. (2004) came to a different conclusion; the introduction of the BSC had no significant impact on branch performance (gross profit or sales). One possible explanation offered by Neely et al. for this lack of perceived benefits is the managers (in)ability to take actions based on the data. Data was not always actionable (e.g. customer retention of 74%) and this aggregated information could be the reason the managers did not and could not take action based on the data contained in the BSC.

Tableau de Bord

The Tableau de Bord (TBD) was first introduced in France, in the 1930s. It was described being similar to a 'dashboard' (hence the name tableau de bord) and it was used by managers to guide their organizations to their desired destinations. The TBD would often contain graphs and ratios which managers could to understand the organization's current situation and to help guide their organizations to their desired situation. The TBD included a combination of financial and non-financial indicators. (Bessire & Baker, 2005)

The Tableau de Bord contains a hierarchy of related measures that map to different levels of the organization; this forces business units and divisions of the organization to align themselves with the company's overall strategy. (Kennerly & Neely, 2002)

The TBD is the French equivalent to the BSC because of the many similarities; they both focus on avoiding the dominance of financial performance measures, they both assume anticipation has more merits than mere reaction and both recommend a selection of performance measures instead of overloading the user of the model. (Bourguignon et al., 2001; Epstein & Manzoni, 1997)

Epstein & Manzoni (1997) state that creating a TBD involves translating the business unit's mission and vision into objectives. These objectives are then used to create key success factors (KSF). These KSF's are then translated into quantitative key performance indicators (KPIs). However, TBD is more an approach on the alignment of strategy and operations; it does not specify perspectives such as the BSC. Thus, the TBD will not be included in Appendix A.

Software Value Map

Perhaps more related to value management, but relevant nonetheless is the Software Value Map (SVM) introduced by Khurum et al. (2013). It builds upon the Balanced Score Card, but ensures a broader view of the organization's health by focusing on four perspectives just like the BSC: customer, financial, internal business and innovation & learning. (Korpivaara et al., 2021)

The SVM uses the four perspectives from the BSC as 'value perspectives' (VP). Each VP has a nested cluster of value aspects (VAs); each VA can have multiple sub-value aspects (SVAs). Each SVA can have end nodes called value components (VCs). (Khurum et al., 2013). As one can derive from figure 4, the SVM provides a complete view of value constructs relating to four perspectives that are important when making decisions about software product development and management. Each of the value components (the last level in the hierarchy of nodes) can be measured with 'measured through'-attributes, which the authors provide. For example, the value component 'Architectural usability value' uses the sub-value aspect 'operability'. This operability can be assigned a value ranging from -2 to +2, where +2 signifies the operability greatly improves the architecture's usability. (Khurum et al., 2013)

In the SVM, interrelationships are common and possible among the value aspects. For example, the value of the product architecture from an internal business perspective is related to the perceived quality by the customer from an external perspective (Khurum et al., 2013).

While extensive research has been conducted on the topic of value management, there seems to be a lack of empirical evidence supporting this Software Value Map. Khurum used a case study to showcase the merits of this novel SVM, but empirical studies confirming (or denying) the benefits of using this specific model seem to be lacking. However, Alahyari et al. (2017) note that Khurum does not look into how different organizations interpret and give meaning to the 'value', nor which value aspects are important to achieve in different contexts.

Moreover, the SVM does what its name implies; it maps value of software using the four BSC perspectives, but is less clear from a performance point of view. Thus, it can be used as input for performance measurement, but is not a encompassing performance management tool like the BSC for example.
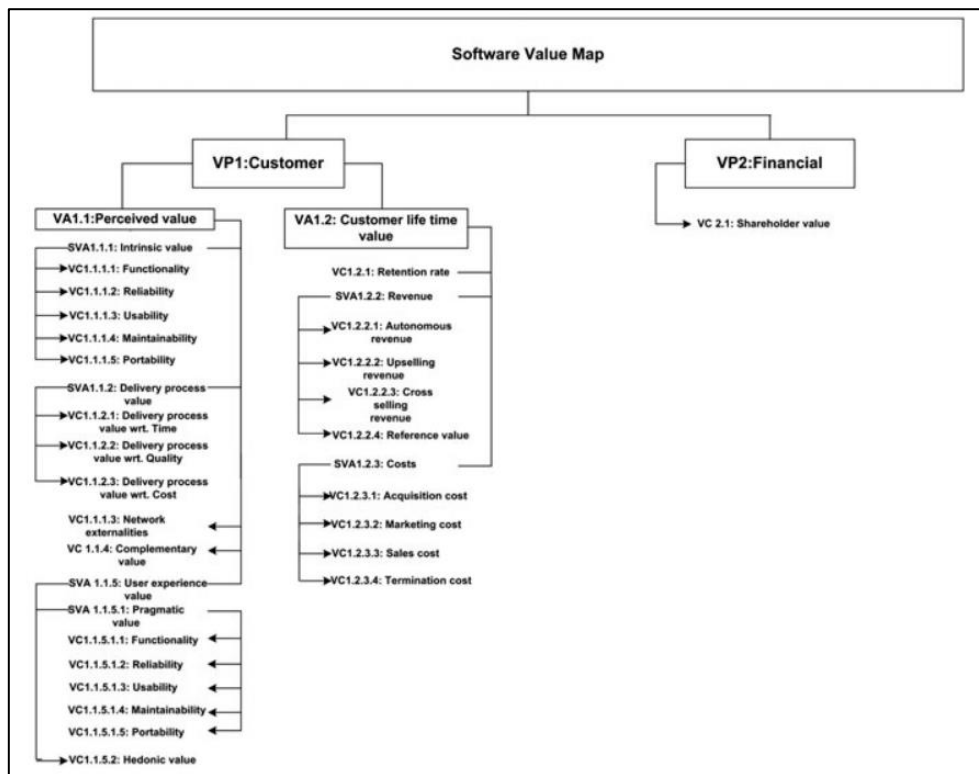
Figure 4: Example of the Software Value Map (Khurum et al., 2013)

Çiçek et al.'s Team Performance Measurement Model

Çiçek et al. (2005) propose a performance measurement model for providing feedback to total quality teams with continuous improvement as the main focus and purpose. It empowers teams to use their own performance measurement, using Çiçek et al.'s broad but applicable model. Figure 5 shows how each component of Çiçek et al.'s model is related and connected. The Team Performance Measurement Model provides guidance on how to assess a team's current standing regarding each of the five components and practical ways to measure each component.
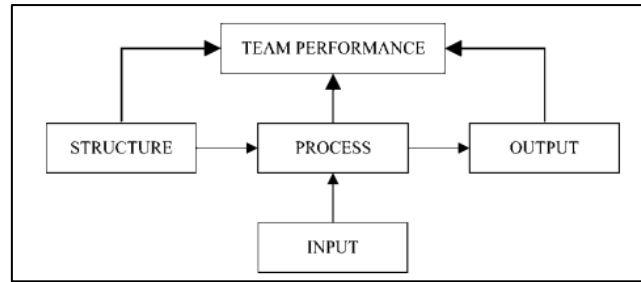
*Figure 5: Çiçek et al's Team Performance Measurement Model
(Çiçek et al., 2005)*

Çiçek et al. do not necessarily define which metrics are used for each of the four categories that have a(n) (in)direct effect on team performance. In their case study, the performance of a health care team was used. Each performance dimension (structure, process, output, input) had multiple measures which the health care team decided upon themselves (e.g. percentage of on-time patient care orders). The frequency of conducting measurements were two months for process, input and output; structure had a periodicity of six months.

These measures for each performance dimensions were sourced from several workshops with this specific team where they prepared process flowcharts to represent the various stakeholders and processes. Based on this flowchart, measurements for input, output, process and structure were identified. After this interactive workshop, the appropriate methods and means for data collection and analysis were developed with the health care team.

Keegan's performance measurement matrix
Keegan, Eiler and Jones developed a performance measurement matrix that in their eyes reflects a balanced view on measurement. It categorizes measures as either 'cost' or 'non-cost', and 'external' or 'internal'. It is a simple framework which enables an organization or business unit to plot its current measures and identify possible adjustment in measurement focus. (Kennerley & Neely, 2002)
An example of this would be an abundance of internal cost measures and the absence of measures for the other three categories. Figure 6 shows Keegan's performance measurement matrix (Neely et al., 2000).

*Figure 6: Keegan's performance measurement matrix
(Neely et al., 2000)*

One of the criticisms of Keegan's matrix in the literature is the fact that it does not explicitly show the relation between the four dimensions (Neely et al., 2000). This can also be seen in figure 6; just because there is an arrow linking the various dimensions to each other, does not make it explicit how they are related and how each dimension impacts the other.

Fitzgerald's framework of performance measurement

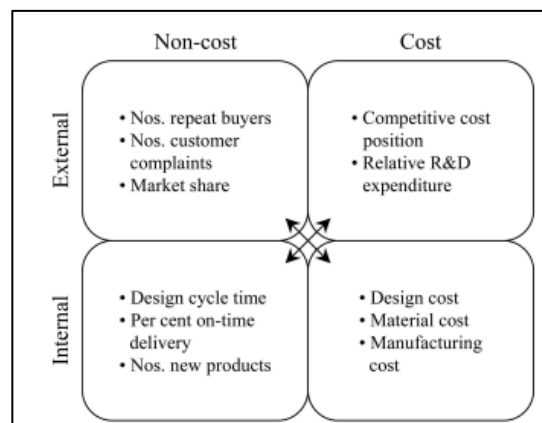Fitzgerald et al. propose a framework classifying performance measures into two types: measures relating to results (competitiveness, financial performance) and measures relating to the determinants of those results (quality, flexibility, resource utilization, and innovation). Results can be seen as the lagging indicators, while the determinants are the leading indicators. (Neely et al., 2000)

Results	Financial performance
	Competitiveness
Determinants	Quality
	Flexibility
	Resource utilisation
	Innovation

Figure 7: Fitzgerald's framework for performance measurement (Neely et al., 2000)

While not every possible result or determinant is included in the model as shown in figure 7 (e.g. employee or customer satisfaction is missing), it does provide a useful distinction between the aforementioned lagging indicators and leading indicators. (Neely et al., 2000)

Strategic Measurement And Reporting Technique (SMART pyramid)

The SMART-pyramid was developed by Wang Laboratories, Inc. as a result of their dissatisfaction with traditional measures of performance such as efficiency and productivity. Their objective was to create a management control system with performance indicators to define and sustain the organization's success. This pyramid is composed of five levels, four of which contain objectives and measures, as seen in figure 8. (Ghalayini, 1996)

Figure 8: The SMART-Pyramid (Ghalayini, 1996)
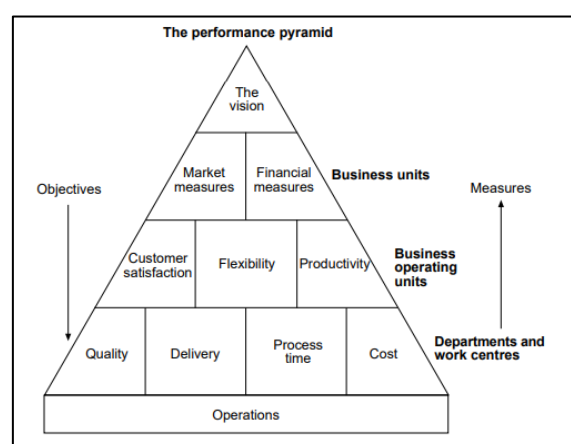
A criticism of this SMART-pyramid is that it does not provide a way to identify the measures. For example, the model prescribes 'Quality' as an objective, but does not provide any tangible measures. Another criticism is that continuous improvement is not explicitly mentioned in the SMART-pyramid, giving off the impression it is a one-time endeavor. (Striteska & Spickova, 2012)

Macro Process Model

Brown (2020) developed the concept of linking measures even further. In his Macro Process Model of the Organization, clear links are shown between the five stages in a business process and their respective measures. These stages are defined as inputs, processing system, outputs, outcomes and goals. Brown argues that each stage is the driver of the performance of the next stage. (Brown, 2020; Neely et al., 2000)

This is similar to Fitzgerald's framework for performance management, because Fitzgerald also stated the determinants (here: inputs, processing system, outputs) are the drivers of the results (here: outcomes, goals). The Macro Process Model is visualized in figure 9 (Neely et al., 2000).
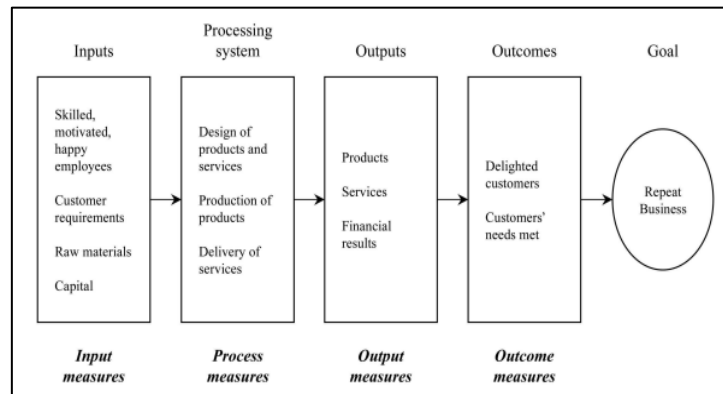
Figure 9: Brown's Macro Process Model (Neely et al., 2000)

Performance Prism

The Performance Prism adopts the view that stakeholders should play a key role in performance measurement. Typically, stakeholders in commercial organizations are thought of as shareholders. However, other stakeholder groups must also be considered (e.g. employees, suppliers, customers, regulators) (Neely et al., 2001). The performance prism proposes a set of questions in order for organizations and teams to use when defining performance measures. These questions assist in making sure all facets of stakeholder satisfaction in relation to performance are covered. Each facet of the Performance Prism is connected to a the five following questions (Kennerley & Neely, 2002; Neely et al., 2001)

1. Stakeholder satisfaction: Who are our key stakeholders and what are their needs and wants?
2. Strategies: What strategies do we need to satisfy the needs and wants of these key stakeholders?
3. Processes: What critical steps need to be taken to operate and enhance the processes?
4. Capabilities: What capabilities do we need to operate and enhance the processes?
5. Stakeholder contribution: What contribution do we require from the stakeholders if we want to develop these capabilities?

Figure 10 shows that the strategies, capabilities and processes that determine and assist in achieving stakeholder satisfaction.
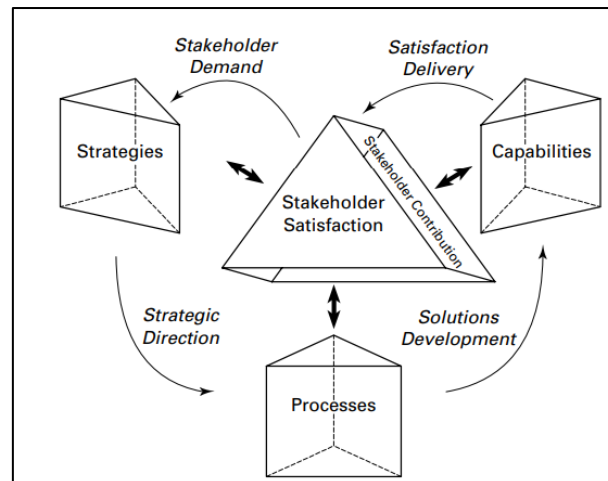
Figure 10: Performance Prism (Neely et al. 2000)

Neely et al. (2001) notes the Performance Prism is not a measurement framework in a prescriptive sense. Instead, it is a tool which management can use to influence their thought processes about what they want to address in the managing of their business. In a case study by Neely et al. (2001), the Performance Prism was flexible enough to be applicable to a wide variety of organizations.

Striteska & Spickova (2012) in their review of the Performance Prism note that the Performance Prism offers little guidance on how the performance measures are to be implemented in an organization or team, as well as a lack of logic among the measures (no link between the drivers and results).

European Foundation for Quality Management's (EFQM) Business Excellence Model
The EFQM Business Excellence Model is, along with the Balanced Scorecard, the most widely adopted performance measurement system (Striteska & Spickova, 2012). The EFQM model is a non-prescriptive system; it proposes to help organizations in assessing their progress towards achieving continuous improvement and business excellence. It is based on eight fundamental concepts: (1) results orientation, (2) customer focus, (3) people development and involvement, (4) continuous learning, innovation and improvement, (5) leadership and constancy of purpose, (6) partnership development, (7) management by process and facts and (8) public responsibility. These eight concepts are further specified in nine criteria, which are then divided into four results on the one hand and five enablers on the other hand in order to measure excellence (Striteska & Spickova, 2012; Calco-Mora et al., 2005).

Although not designed as a performance measurement framework, the EFQM take a broader view of performance, addressing some areas of performance not considered by for example the balanced score card. It is a broad model that explicitly highlights the enablers for performance improvement and indicates result areas that should be measured, as seen in figure 11. (Kennerly & Neely, 2002) However, the EFQM model is a self-assessment tool rather than an objective measurement. Another potential flaw is the fact that the four categories that fall under 'Results' are quite broad. Additionally, not every aspect of this framework is readily measurable, such as the enablers 'Policy and strategy'. (Kennerly & Neely, 2002)

Figure 11 (Neely et al., 2000) shows the nine criteria which can be mapped to the eight fundamental concepts. However, these terms in this framework are so open to interpretation, it can mean anything and everything.
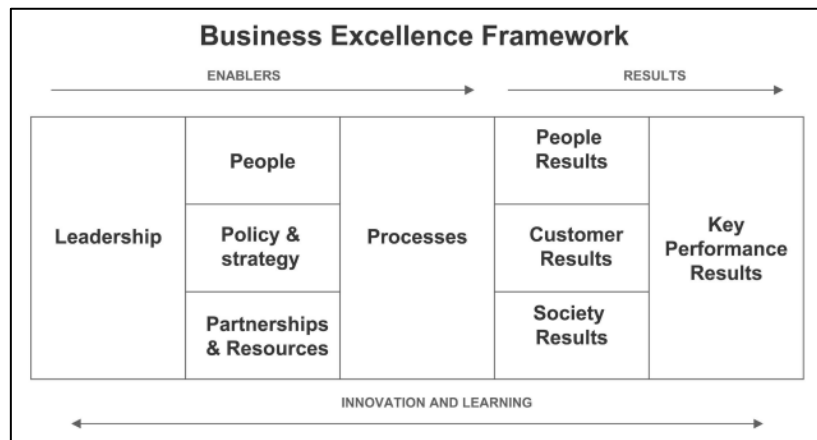
Figure 11: European Foundation for Quality Management's Business Excellence Framework (Neely et al. 2000)

As already mentioned, it is a self-assessment framework to measure the strengths and areas to improve upon of an organization. (NEF Consulting, 2021)

Another criticism, by Striteska & Spickova (2012), is the fact that the EFQM model does not provide guidelines on how to design and use effective performance management; hence the non-prescriptive characteristic. While it provides a useful view of how enablers (leadership, people, policy & strategy, partnerships & resources) drive results (people results, customer results, society results), the model does not provide much more than that.

Kanji Business Excellence Measurement System
The Kanji Business Excellence Measurement System (KBEMS) consists of the Kanji Business Excellence Model (KBEM) and the Kanji Business Scorecard (KBS). Both are made up of several criteria that can be mapped to Critical Success Factors (Kanji, 2002). The KBEM is used for the measurement of performance by internal stakeholders, while the Kanji Business Card is focused on the performance from the perspective of external stakeholders. After taking both of these into account, the Organizational Performance Excellence Index (OPI) can be calculated. This OPI is an aggregated measure of the organization's excellence in managing the various critical success factors. (Striteska & Spickova, 2012)

The KBEM consists of ten items (leadership, customer delight, customer focus, management by facts, improvement of process, people-based management, people performance, continuous improvement, culture of continuous improvement, performance excellence). KBS consist of five other elements (organizational values, process excellence, delight the stakeholders, performance excellence). (Striteska & Spickova, 2012)

Striteska & Spickova (2012) again identified several strong and weak points of the KBEMS. One of its merits is the fact that it includes stakeholder groups (employees, suppliers, etc.) who are usually neglected in the field of stakeholder management. One of the criticisms is that it offers little guidance how the performance measures are to be implemented. Additionally, it does not provide

metrics but instead provides a framework for which to group the measurements and metrics under. If an organization would like to measure their organizational values, that organization is required to come up with their own questionnaire.

2.2.4 – Customer Satisfaction

One of the principles of Agile and part of the Agile heuristics is the fact that the focus should be on creating value for the customer (Buresh, 2008). Agile already facilitates shortened feedback loops; the needs and wants of the customer are of the highest priority. Just like employee satisfaction, if all other metrics show improvement, but the customer is not satisfied, how much value is the Agile team actually delivering? Thus it is important to understand the concept and measurement of customer satisfaction from the existing literature.

Proxy measures of customer satisfaction
One very simple way to gauge customer satisfaction is the thumbs-up rule metric. The thumbs-up rule metric can be used to measure the customer satisfaction at the end of each sprint (Padmini et al., 2015). Another simple method would be to use the number of defects reported by a customer. While the number of defects has already been discussed in the context of Agile metrics, it can also be used as an proxy measure of customer satisfaction.

These simple ways of measuring customer satisfaction are however prone to error and misinterpretation. Just because the number of defects reported by a customer is zero, does not mean the customer is satisfied with the product. Thus, it is important to look towards the literature on how customer satisfaction is measured.

Net Promoter Score (NPS)
Three prevailing metrics can be found on the topic of customer satisfaction. The first is the Net Promotor Score (NPS), which measures how likely a customer would recommend a specific brand to a friend or colleague. Based on their response on a scale from 0 to 10, the respondents are grouped into 'promotors' (9-10 rating), 'passively satisfied' (7-8 rating) and 'detractors' (0-6 rating). (Fisher & Kordupleski, 2019)
While useful when comparing brands from a market competitor standpoint, this metric can also be applied in an Agile context. Instead of asking 'How likely would you recommend this specific brand to a friend or colleague?', the question can be adapted into 'How likely would you recommend this development team to your peers?'.

Customer Satisfaction Score (CSAT)
The second prevailing metric in the literature is the Customer Satisfaction Score (CSAT). It measures the customers' satisfaction with the service they receive; usually after an interaction with someone from a support team asking them to rate the help they received. It is calculated with the following formula (Geckoboard, n.d.):

CSAT (%) = (number of positive responses / total number of responses) * 100%

While useful in a customer service/support setting, this simple percentage would probably be too one-dimensional to use and derive insights from, when used in the context of an Agile team.

Customer Effort Score (CES)
The last individual customer satisfaction metric, would be the Customer Effort Score (CES). CES is defined as the customers' intention to continue doing business with the company, increase their spending or spread positive word of mouth to others (Dixon et al., 2010). As with the CSAT and NPS, this metric is more applicable when looking at various businesses in their respective markets rather

than assessing whether a(n internal) customer is satisfied with an arbitrary sprint of an Agile product team.

Swedish Customer Satisfaction Barometer (SCSB)

In 1992, Claes Fornell developed The Swedish Customer Satisfaction Barometer (SCSB). This model was the first to measure the satisfaction of customers on a national level; it measures the customer satisfaction in more than 30 industries and for more than 100 companies (Fornell, 1992). Fornell argues that the extent to which a company is able to satisfy its customers, is an indication of the company's overall health and repeat business. A manifestation of this future prospect is the fact that a high customer satisfaction could serve as a barrier for competitors to take away the company's customers. (Fornell, 1992)

While Fornell recognizes that there is no consensus on how to measure customer (dis)satisfaction, the SCSB identifies three facets of customer satisfaction (Anderson et al., 1994):

1. The degree of general satisfaction;
2. The confirmation of expectations;
3. The distance from the customer's hypothetical ideal product.

Fornell argues that there are some advantages of the SCSB over traditional approaches of measuring satisfaction. The first one is that the causes of satisfaction are not confused with the phenomenon itself. Additionally, the unreliability of measures is taken into account. This way, the indicators defining customer satisfaction can be weighted in such a way that their combination (i.e. the SCSB) has a maximal impact on loyalty and customer retention. (Fornell, 1992; Grigoroudis & Siskos, 2004)

Figure 12 shows an example of the questionnaire used for the car industry in the SCSB (Grigoroudis & Siskos, 2004).

Figure 12: Questionnaire used for the car industry in the Swedish Customer Satisfaction Barometer (Grigoroudis & Siskos, 2004)

The American Customer Satisfaction Index (ACSI)

The American Customer Satisfaction Index (ACSI) builds further on the previously discussed Swedish Customer Satisfaction Barometer. Fornell et al. (1996) state that by using the literature on quality and making the distinction between customization and reliability, the ACSI introduces the concept of 'perceived quality', alongside the original 'perceived value'. An additional improvement of the ACSI over the SCSB is there are now three operationalising measures for customer expectations (compared to SCSB's one measure).

The ACSI is a market-based measure of performance for national economies, industries and companies. It is a customer-based measurement system to evaluate and enhance the performance of these national economies, industries and companies. Annually, the ACSI system estimates a company-level customer satisfaction index for each company and weights those company-level indices to calculate the national, industry and sector indices. (Fornell et al., 1996)

The ACSI model is outlined in figure 13 below:
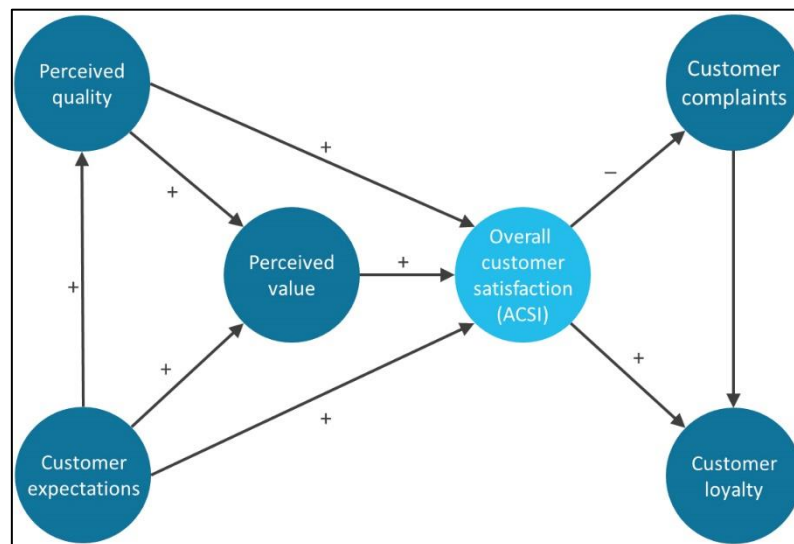
Figure 13: The American Customer Satisfaction Model (EURIB, n.d.)

In figure 13, several antecedents influence the 'overall customer satisfaction (ACSI)'. In turn, an increasing customer satisfaction index will decrease the customer complaints and increase customer loyalty. To achieve these desired outcomes, a national economy, a sector or a company will need to understand its perceived quality, the expectations of the customers and the perceived value by its customers. This model applies both to goods and services, as outlined in the work of Fornell et al. (1996).

The original intent of the ACSI creators was to use this as a representative of the American national economy; a measurement of the quality of goods and services experienced by the customers (EURIB, n.d.). However, abundant literature has been written on using this ACSI as a framework or inspiration to assess the customer satisfaction in other fields and applications. For example, Hsu in 2008 used the ASCI to develop a customer satisfaction index for electronic and online experiences. Deng et al. (2013) adapted the ACSI to assess the customer satisfaction for international tourist hotels. A wholly different application of the ACSI was done by Yazdanpanah & Feyzabad (2017) to investigate the satisfaction of Iranian farmers with agricultural extension programs.

European Customer Satisfaction Index (ECSI)

Inspired by the successful experiences of the Swedish and American customer satisfaction indices, the European Customer Satisfaction Index (ECSI) was developed by the European Organization for Quality, the European Foundation for Quality Management and the European Academic Network for customer-oriented Quality Analysis, supported by the European Commission. (Kristensen et al., 2000)

The result of this multidisciplinary effort resulted in the creation of the ECSI, which is shown in figure 14 (Haaften, n.d.).
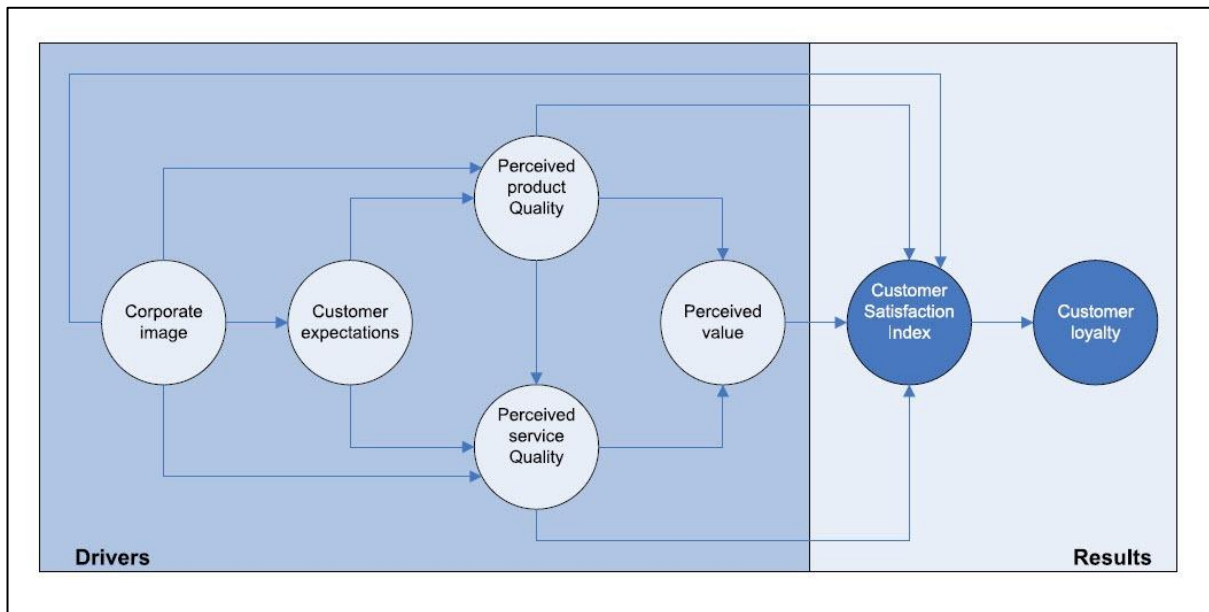
Figure 14: The European Customer Satisfaction Index (Haaften, n.d.)

As is evident from figure 14, a distinction is made between drivers (determinants) and results (consequences) in the ECSI. An important difference between the ACSI and ECSI is the fact that the ECSI divides 'perceived quality' into two elements: 'hardware' which consists of the quality of the product/service itself and the 'humanware' which represents the interactivity in the process of customer service (Kristensen et al., 2000). Another addition in the ECSI compared to the ACSI is the antecedent 'Corporate image', which both affect some determining antecedents and resulting antecedents.

Similar to the ACSI, the ECSI is used in a variety of ways in the available literature. Aydin & Özer (2005) adapted the ECSI to be used in the Turkish mobile telephone market; they added the antecedent 'trust' to the model and re-arranged the relationships between the various antecedents. Askariazad & Babakhani (2015) reached the same conclusion; they applied the ECSI in a business to business context and discovered that adding trust as an antecedent leads to a better explanation of loyalty compared to the original ECSI.

The aforementioned measurement methods (NPS, CSAT, CES, SCSB, ACSI) can be both applied within and outside an Agile environment. Interestingly enough, there is sparse literature on the topic of customer satisfaction in a specifically Agile environment, perhaps because of the implicit customer centrality in the Agile methodology. Most literature on this topic focuses on how the Agile (scrum) methodology can increase customer satisfaction, but not on how to measure this periodically. One of the frameworks which addresses this gap in literature is the Scrum Team Survey.

Scrum Team Effectiveness: Stakeholder satisfaction survey

Verwijs & Russo (2021) have conducted research and propose a theory of scrum team effectiveness that explains what makes some Scrum teams more effective than others. Based on their research of what makes scrum teams effective, they developed multiple surveys for teams, stakeholders and supporters.

One of these surveys is a stakeholder survey consisting of 12 questions, intended for the customers/users of the Agile team (Verwijs, n.d.). These twelve questions are taken verbatim from Verwijs' Stakeholder satisfaction survey (Verwijs, n.d.):

1. The frequency of new releases is good enough for my needs.
2. I have a good sense of what this team is working on.
3. I am happy with the value that this team delivers every Sprint.
4. I am satisfied with the quality of what this team delivers.
5. When I have an idea or suggestion, members of the team are available to listen to me.
6. I am satisfied with how often new versions are released.
7. The team frequently asks for my feedback, ideas or thoughts.
8. I am satisfied with the value that this teams delivers.
9. I frequently meet or interact with members of this team.
10. When the team delivers a new version, it is usually free of serious bugs.
11. This team frequently delivers new versions.
12. What this team delivers is of high quality.

Each of the twelve questions stated above are answered by the customer on a 7-point bipolar Likert-scale, ranging from completely disagree to completely agree (Verwijs, n.d.). While this survey is scientifically validated by the creators, one can ponder on how useful it is to use this exact questionnaire after every sprint iteration. This questionnaire looks more suited to a monthly or quarterly basis than to use it after every sprint.

2.2.5 – Employee satisfaction

If some or all metrics are improving sprint after sprint, it does not mean the employees are satisfied with their work or their colleagues. And even if the customer – whether external or internal – is satisfied with the output of the Agile teams, it does not mean the team itself is satisfied. For the health and longevity of the team, it is important to measure and keep track of the employees' satisfaction. Higher employee satisfaction results in positive outcomes for the organization (Tripp et al., 2016). Additionally, it has been claimed that using Agile methods creates more satisfied employees compared to more traditional methods of software development (Tripp et al., 2016).

The degree of an employee's satisfaction with their job depends on the employee's motives and values (Bin & Shmailan, 2015), but employee satisfaction and performance also have a cyclical relationship (Alromaihi et al., 2017). This means job satisfaction has influence on their performance; at the same time performance of an employee (or team in a broader sense) has an influence on job satisfaction. Members of an Agile team rate their work environments significantly higher than general IT professionals (Marchesi & Succi, 2003). This is positive to read from the literature, but it remains important to understand what contributes to employee satisfaction and possible ways to measure this in order to facilitate communication and improve employee satisfaction.

The following theories, models and frameworks will briefly be discussed which relate to job, employee and team satisfaction:

- Job Characteristics Model (JCM)
- Happiness metric & morale
- Minnesota Satisfaction Questionnaire (MSQ)
- Herzberg Two-Factor theory
- Job Descriptive Index (JDI)

Job Characteristics Model (JCM)

The job characteristics model (JCM) is a useful model to understand job characteristics that influence workers' perceptions of and attitude about their job (Tripp et al., 2016). These five core job characteristics are (Tripp et al., 2016):

1. *Task significance*: the extent to which people believe that their job impacts the lives of people (either society at large or in an organization).
2. *Task identity*: the extent to which a job's tasks are 'whole' or involve a completion of an identifiable outcome.
3. *Skill variety*: the extent to which one perceives a job as requiring a variety of skills, talents and experiences.
4. *Job autonomy*: the extent to which an employee has the discretion about how to complete the work required and set a schedule for completion.
5. *Feedback*: the extent to which the process of completing the work provides employees with information through which they can evaluate their own performance.

These five core job characteristics can be combined into a single index of motivating potential score (MPS) that reflects the overall potential of a job to influence the individual's feelings and behaviours (Bahrami et al., 2016).

The JCM is useful for job redesign purposes if the MPS outcome is low or lower than expected (Wall et al., 1978). With five clearly identified components of what impacts an employee's perceptions of the job and their attitude about it, organizations are able to (re)design jobs in order to maximize the MPS. Consequently, this has a positive effect on the job performance, as Tripp et al. (2016) stated.

While useful to (re)design jobs and identify the potential characteristics of what contributes to an employee's high or low job satisfaction, this model does not focus or show patterns such as team dynamics and communication. Trying to ascertain how Agile a team performs and how happy they are with their orientation on improvement efforts will not be possible by using the JCM. The JCM does also not include anything on the perception of other team member's performance.

Happiness metric & morale

The happiness metric was developed to measure happiness as a proxy of a team's wellbeing, although there is no formalized approach for measuring happiness in Agile teams. Most commonly is to ask team members to rate their current happiness on a 5-point Likert-scale. (Verwijs, 2020) While the benefits of such an approach are numerous (e.g. they emphasize the human factor, provide input for retrospectives), the usefulness depends on the ability to capture actual team well-being and whether it can be used as input for future improvements (Verwijs, 2020). The biggest pitfalls of using happiness metrics are (Verwijs, 2020; Manning, 1991):

- The Happiness metric is too subjective to be meaningful: happy is interpreted differently by everyone, it's subjective.
- The Happiness metric is not task-oriented: It is more productive to ask team members if they are happy with their work or tasks instead of asking about something which cannot be easily

changed. A team member can be unhappy in a general sense, but still be happy with the task he/she is performing.

- The Happiness metric is not team-oriented: Perhaps not common in practice, but it is possible to have a happy team, even though they are not productive or performing well.
- The Happiness metric does no justice to the reality of the work environment: Sometimes there are boring tasks that have to be picked up by someone. Sometimes, members of a team are required to be a team-player, whether they are happy with this or not.
- The Happiness metric is statistically bad metric: Because of its subjectivity, it is difficult to classify it as 'scientific'. Questions about this topic are too open to interpretation to conclude something concrete.

An alternative proposed by the same author (Verwijs, 2020) would be to focus on team morale, because morale is more task- and team-oriented, it includes happiness (in an implicit manner), and is less susceptible to changes in mood. Team morale is defined as "the enthusiasm and persistence with which a member of a team engages in the prescribed activities of that group" (Manning, 1991).

To put team morale in the perspective of performance management, teams with high morale usually have the following traits (Verwijs, 2020; Manning, 1991):

- Team members are willing to help each other out, the nature of the task is irrelevant;
- Team members are generally happy and enjoy working in that team;
- Team members are proud of their team and the work they perform.

Thus, "*measuring team morale puts a focus on the task at hand and how the team feels about its capabilities to deal with it*". (Verwijs, 2020)

This team morale questionnaire below is an adapted version of the Utrecht Work Engagement Scale (UWES), a questionnaire to assess work and well-being which is psychometrically validated and reliable (Verwijs, 2020; The Liberators, 2022):

1. I am enthusiastic about the work that I do for my team.
2. I find the work that I do for my team of meaning and purpose.
3. I am proud of the work that I do for my team.
4. To me, the work that I do for my team is challenging.
5. In my team, I feel bursting with energy.
6. In my team, I feel fit and strong.
7. In my team, I quickly recover from setbacks.
8. In my team, I can keep going for a long time.

Minnesota Satisfaction Questionnaire (MSQ)
The Minnesota Satisfaction Questionnaire (MSQ) is a questionnaire designed in 1967 to measure an employee's satisfaction with their job. The MSQ provides aspects of a job that an employee may find rewarding; it can also be used to explore vocational needs and in generating information about job reinforcers. (Weiss et al., 1967)

The MSQ is self-administered and has both a long and short form. The long form consists of 100 statements (mapping to 20 different facets of job satisfaction); the short form consists of 20 statements (each statement maps to a different facet of job satisfaction) (Weiss et al., 1967). Each statement can be answered on a 5-point Likert scale (Decker & Borgen, 1993). The 1977 version of the MSQ short form uses a bipolar 5-point Likert scale, but the 1967 version of the MSQ short form

uses a unipolar 5-point Likert scale. The authors argued that the bipolar scale tends to result in negatively skewed responses (i.e. ceiling effect), so to adjust for this, they used (and recommend) a unipolar scale (Weiss et al., 1967). The 20 items that are assessed both in the long and short form of the MSQ are:

1. Ability Utilization
2. Achievement
3. Activity
4. Advancement
5. Authority
6. Company Policies
7. Compensation
8. Co-workers
9. Creativity
10. Independence
11. Moral Values
12. Recognition
13. Responsibility
14. Security
15. Social Status
16. Social Service
17. Supervision--Human Relations
18. Supervision--Technical
19. Variety
20. Working Conditions

The MSQ has been extensively used in literature and case studies to assess job and employee satisfaction. One example is the empirical investigation of the MSQ by Hancer and George (Hancer & George, 2003). They examined job satisfaction of restaurant employees working in non-supervisory positions; they used the short form of the MSQ.

Similarly, the MSQ was used to investigate the job satisfaction among health care workers employed at public health centres in Turkey (Bodur, 2002). The results of this survey helped identify which groups among the health care workers are the least satisfied and which of the 20 items in the MSQ could be improved to raise the overall job satisfaction of these least satisfied groups.

Herzberg's Two-Factor Theory

In 1959, Herzberg and his colleagues published the two-factor model of work motivation and developed the theory of motivation-hygiene (Alshmemri et al., 2017). The factors that have an effect on job satisfaction are split into two sets of categories: hygiene factors and motivation factors. Hygiene factors are extrinsic and considered less important than motivation factors. Hygiene factors reduce the level of job dissatisfaction (or are at least correlated with reducing it), while motivation factors directly influence an employee's motivation and satisfaction. Motivation factors lead to job satisfaction. (Alshmemri et al., 2017)

Alshmemri et al. (2017) also provide an overview of the factors in Herzberg's theory, as seen in table 3:

Table 3: Summarization of Herzberg's theory factors, by Alshmemri et al. (2017)

Motivation factors	Hygiene factors
Achievement	Salary
Work itself	Interpersonal relationship
Recognition	Supervision
Responsibility	Working conditions
Advancement	Policies and administration
Possibility of growth	

Job Descriptive Index (JDI)
The Job Descriptive Index (JDI) is a questionnaire with 72 items designed to measure the level of satisfaction, distributed over five facets that relate to an employee's satisfaction: promotion, supervision, co-workers, pay, and work. (Hancer & George, 2003)

Similar to the Minnesota Satisfaction Questionnaire, the amount of questions in the JDI can be seen as quite a work to fill in. To remedy this, Stanton et al. (2002) developed an shortened version of the JDI: the Abridged Job Descriptive Index (AJDI). This AJDI consists of 25 items instead of 72.

Multiple studies have been conducted on the review and validity of the JDI.
Kinicki et al. (2002) investigated the construct validity of the JDI using a meta-analysis that surveyed antecedents, correlations, and results of job satisfaction. Their study concluded that the construct validity of the JDI was sufficiently supported.

2.2.6 – Change management and performance information
Providing performance information is not sufficient to improve business performance results. The real success lies in people's behavior in actually using this performance information. Many academics and executives believe that the main reason why performance measurement is short-lived is because of people's behavior when confronted with this new information that requires potential change. (Nudurupati, 2011)

Change needs to be managed properly as it can impact the adoption of performance measures, especially if a team has never worked with such metrics.

2.3 – Performance management in an Agile context
The principle from the Agile manifesto "*at regular intervals, the team reflects on how to become more effective, then tunes and adjusts its behaviour accordingly*" (Hohl et al., 2018) is synonymous with continuous improvement. The key takeaway of continuous improvement is to continuously search for ways in which processes and products can be improved. This is done to achieve greater value delivery to the customers at an increased level of efficiency. However, before any organization or team knows what to improve, they has to establish where its current performance is lacking and why; this is achieved with performance measures. (Eriksson, 2017; Neely, 1999)

The Beyond Budgeting model (BBM), the Software Value Map (SVM) and the Evidence Based Management Guide (EBMG) are all performance management models that have been studied in an Agile context according to the literature.

Beyond Budgeting
Highsmith (2006) states in his research that the practitioner needs a new, adaptive performance management system (APMS) which has two attributes: (1) to focus on an enterprise group (team, business unit, division, company), and (2) to encourage those groups to perform at a higher level than they currently do. He states that the existing performance management systems need to be changed if we want to apply performance management in truly Agile organizations. According to Highsmith, Beyond Budgeting is one of the existing performance management frameworks which is suitable to be used as and adapted to an adaptive measurement system. As Lohan et al. (2010) states, Beyond Budgeting adopts an Agile and adaptive perspective.

In 1997, the Beyond Budgeting performance management model was introduced as an alternative to the traditional command and control type models. Beyond Budgeting is oriented towards fast changing environments and uses a 'sense and respond' control mechanism, which allows an

organization or team to not get left behind by these fast changing environments. The model consists of six leadership principles and six process principles when taken together and are used holistically to help improve performance management within in a team or organization. (Lohan et al., 2010)

Table 4 lists the twelve principles of the Beyond Budgeting model (Lohan et al., 2010).

Table 4: The twelve principles of the Beyond Budgeting Model (BBM) (Lohan et al., 2010)[1]

Leadership Principles	Process Principles
Customers: Focus everyone on improving customer outcomes, not on hierarchical relationships.	**Goals**: Set relative goals for continuous improvement; do not negotiate fixed performance contracts.
Organization: Organize as a network of lean, accountable teams, not around centralized functions.	**Rewards**: Reward shared success based on relative performance, not on meeting fixed targets.
Responsibility: Enable everyone to act and think like a leader, not merely follow the plan.	**Planning**: Make planning a continuous and inclusive process, not a top down annual event.
Autonomy: Give teams the freedom and capability to act; do not micro-manage them**.**	**Controls**: Base controls on relative indicators and trends, not variances against a plan.
Values: Govern through a few clear values, goals and boundaries, not detailed rules and budgets.	**Resources**: Make resources available as needed, not through annual budget allocations.
Transparency: Promote open information for self-management; do not restrict it hierarchically.	**Coordination**: Coordinate interactions dynamically, not through annual planning cycles.

Each principle is accompanied with several questions to assess the effect organizational practices have on the performance of Agile software development teams (Lohan et al., 2010).

The Beyond Budgeting model has been applied multiple times by researchers in the context of Agile software development teams, such as in the research of Lohan et al. (2013) & Lohan et al. (2010).

Software Value Map in Agile organizations
Alahyari et al. (2017) examined the Software Value Map in an Agile environment. The most prioritized value perspective was 'customer value' in an Agile context; 'delivery time' and 'quality' were correspondingly the most important value constructs. While the Software Value Map has already been discussed in section 2.2.3, Alahyari et al.'s examination of the Software Value Map focuses on the interpretation of (customer) value in Agile organizations and not necessarily on the concept of performance measurement as a whole.

Evidence-Based Management Guide (EBMG)
One of the more applicable frameworks for performance management in an Agile context is the Evidence-Based Management Guide (EBMG), which helps the practitioner in measuring value to achieve improvement and agility. It is an empirical approach that assists organizations to continuously improve customer satisfaction, business results and capabilities under conditions of

---

[1] Table directly copied from Lohan et al. (2010)

risk and uncertainty. It provides a framework for organizations to improve their ability to deliver value through achieving strategic goals. (Scrum.org, 2020)

The EBMG provides advice on how to use experiments and hypotheses to set and achieve goals. However, the most interesting and relevant part is the fact that the EBMG provides a set of value perspectives that correspond to the organization's ability to deliver value. These perspectives are called Key Value Areas (KVAs). These four KVAs are shown in figure 15 (Scrum.org, 2020).
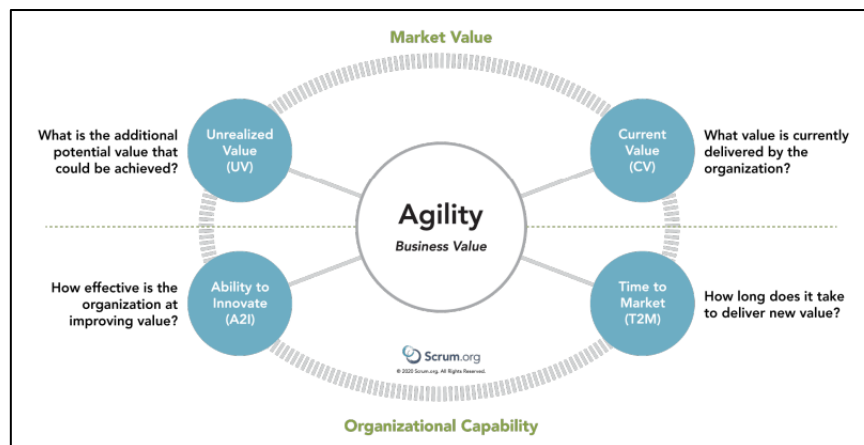

Figure 15: The Evidence Based Management Model, with four Key Value Areas (Scrum.org, 2020)

Each KVA focuses on a different aspect of either value or the ability of the organization (or team for that matter) to deliver value. The KVA's and their descriptions are sourced from Scrum.org (2020).

Current value: "The value that the product delivers today"
The purpose of current value (CV) is to understand the value that a team or organization delivers to its customers and stakeholders at this moment in time; neither the past nor present is taken into account with this KVA. Aspects such as happiness of users/customers, the team members and other stakeholders are especially applicable.

Unrealized value: "The potential future value that could be realized if the team met the needs of all potential customers/users"
Unrealized value (UV) helps teams and the organization to maximize the value from their product or service over time. When customers experience a gap between their current experience and the desired experience, it represents an opportunity for the business which is measured with unrealized value.

Time-to-Market: "The team's ability to quickly deliver new capabilities, services or products"
Time-to-Market (T2M) is about minimizing the time it takes to deliver value. Without managing this KVA, the ability to deliver value in the future is uncertain.

Ability to innovate: "The effectiveness of an organization to deliver new capabilities that might better meet customer needs"
The ability to innovate (A2I) is used to maximize the ability to produce innovative solutions and new capabilities. Improving A2I helps teams become more effective in ensuring that the work it does improves the value of the products/services for the customers.

Individual performance measurement to evaluate software engineers in an Agile context
Besides the literature on performance management frameworks, additional research has been conducted on the actual metrics in an Agile context. As one may have read from all the frameworks which are discussed; they provide various perspectives and categories without the actual metrics that the practitioner should use when applying those perspectives.

Thus, some studies have proposed models and metrics to specifically measure the performance of software engineers in an Agile context. One of such studies has been conducted by Alnaji & Salameh (2015), which propose a performance measurement framework to evaluate software engineers in an Agile software development context. Alnaji & Salameh state that traditional evaluation criteria for the performance of software engineers are incompatible with principles of Agile development. Thus, their study proposes a performance measurement framework which is aligned with the Agile core values and principles.

This framework consists of three main categories. Each category has multiple input values and resulting calculations. The input values for the category Productivity are 'team velocity', 'engineer planned story points', 'engineer completed story points' and 'team completed points'. The productivity as a resulting calculation is then calculated by measures such as 'productivity index'. The same principle is applied to the other categories. (Alnaji & Salameh, 2015)

Table 5 provides a succinct overview of the main measures, input values and metrics:

Table 5: Performance measurement framework to evaluate software engineers in an Agile context (Alnaji & Salameh, 2015)

Category	Measures
Productivity	Planned productivity for a SE during a given sprint (PPSE)
	Actual productivity for a SE during a given sprint (APSE)
	Difference between a SE's planned and actual productivity (PV)
	Productivity Index
Efficiency	Defects escaped rate for the team
	Defect cycle-time for the team
	User stories cycle-time for the team
	Spill-over rate for the team
Social-skill	Mentorship and team-collaboration
	Breadth-of-Knowledge

Additionally, in the paper of Al-Heyasi (2018), four types of metrics are proposed to characterize the performance of software engineers on an individual level: (1) contribution measures, (2) influence measures, (3) impact measures and (4) impression measures.

3 – Gap in literature

In the previous chapter, the findings of the literature review were discussed. Various theories, models and definitions were compared to each other and synthesized. With this theoretical understanding, the potential gap in literature will be explored in this chapter.

Performance management & measurement frameworks

Various performance management and measurement frameworks have been discussed and analysed. Additionally, these frameworks were mapped to and compared with Stettina et al.'s (2021) dimensions of organizational performance. Through this extensive analysis and mapping of the frameworks, it was concluded that all of the frameworks provide categories or perspectives to look at performance and how to manage this. However, only a few of these frameworks actually provide examples of metrics the practitioner can use. For example, the Balanced Score Card provides four perspectives (financial, customer, internal process, learning & growth) on how to look at performance, but the authors do not prescribe the actual metrics or KPIs to use within these four categories (except for a few examples). The same lack of prescribed metrics is applicable for all the previously discussed performance management frameworks in chapter 2, with the exception of a few which provide only some examples of metrics the practitioner could use.

Agile metrics & visualizations

Both to make up for the lack of metrics provided by the performance management frameworks and because the research setting is an Agile one, a literature study was conducted on Agile metrics and visualizations. Numerous Agile metrics and visualizations were explored in chapter 2 and explained more in-depth in Appendix B. The reviewed literature is primarily focused on explaining the various Agile metrics and their potential use cases. However, research on how these Agile metrics and visualisations are used in practice and what their impact is on the team is scarce to non-existent.

Employee satisfaction

Higher employee satisfaction results in positive outcomes for the employing organization (Tripp et al., 2016). Multiple theories and models relating to employee satisfaction have been discussed in chapter 2. Each theory or model provides a unique way of looking at what constitutes the satisfaction of employees. The gap in literature is observed in the fact that employee satisfaction in an Agile context is under-exposed; the team morale questionnaire is the only model that is specifically designed for Agile teams. Moreover, the literature does not provide any insight in the medium of assessing the satisfaction of employees. Traditional approaches such as surveys sent via e-mail or conducted by interviewing employees, are not compared to more interactive approaches (e.g. an interactive satisfaction assessment during the retrospective).

Customer satisfaction

If the velocity is stable and the number of features is steadily increasing, it does not automatically translate to delivering value to the customer. Thus, it is important to assess and understand if the customer/end-user of the product is satisfied with both the team and the product. Similar to the employee satisfaction, multiple models and theories have been explored in chapter 2 which try to capture the abstract concept of customer satisfaction. The only customer satisfaction model that has been specifically adapted to an Agile environment is the 'Stakeholder satisfaction survey' from the Theory of Scrum Team Effectiveness (Verwijs, n.d.). The other models and questionnaires are more geared towards external customers (e.g. consumers of a certain product), in addition to the non-consideration of an Agile context. Since customer feedback and satisfaction are already present implicitly in the sprint demo and review sessions, it may explain why the various explored models have not been adapted to fit an Agile context.

In conclusion, the main gap in literature is observed for the following topic:

- Extensive literature has been written on how Agile metrics can be used by Agile teams, but none of the literature focuses on the introduction of such metrics to see what the overall benefits and challenges are for Agile teams to improve their work.

Moreover, a few other topics are not covered in the literature. While less significant than the gap observed in how actionable and useful metrics are, these four topics are still relevant to somehow include in this research:

- Extensive literature has been written on the topic of metrics and visualizations in an Agile context, but provide barely information on how actionable or useful the metrics are for teams.
- Performance management and measurement frameworks only provide perspectives on how to understand (team) performance, but do not actually prescribe metrics for the practitioner to use for each perspective in the frameworks.
- Extensive literature has been written on the topic of metrics and visualizations in an Agile context, but none provide information on how actionable or useful the metrics are for teams.
- Several employee satisfaction models have been explored, but there is barely literature on how to assess the employee satisfaction periodically and systematically in an Agile context.
- Customer satisfaction is often elicited during the sprint review and demo sessions, but again there is hardly any literature on how to assess the customer satisfaction periodically and systematically in an Agile context.

To address these gaps in literature, the following research question has been formulated:

> *What are the benefits and challenges of introducing a dashboard containing Agile metrics to Agile teams, in order for the teams to improve their work?*

This research question addresses all aforementioned gaps in literature. First of all, by creating and testing a performance measurement framework in the form of a dashboard; this research will provide the future practitioner with evidence-based prescribed metrics which are both actionable and useful. Second, this performance measurement framework will visualize the results of employee and customer satisfaction surveys conducted in the case study organization. But before it is displayed, it is necessary to formulate questions that could assess both the employee and customer satisfaction in an Agile context which are also adapted to be conducted periodically (e.g. every 1 to 2 sprints). By creating and testing such a dashboard throughout a period of eleven weeks, the benefits and challenges of introducing and using a dashboard are researched, as well as how the dashboard and the contained metrics help the Agile teams to improve their work.

4 – Methodology

In this Methodology chapter, the research onion model will be used to determine the research approach. Additionally, a brief introduction is given on what action research is and why it has been chosen as a strategy compared to other alternatives. Furthermore, the research is conducted in an organization, so it essential the context of the case study organization is mentioned. Lastly, this research utilizes various data sources and analysis techniques; these are discussed as well.

4.1 – Research Onion Model

The research onion model (Saunders et al., 2009) is used to determine which research approach will be applied. The research onion model is shown in figure 16 (Saunders et al., 2009). The research onion model is a tool to help organize the research and to aid in developing the research design by 'peeling the layers' step by step, starting from the outside and working towards the centre (Melnikovas, 2018).
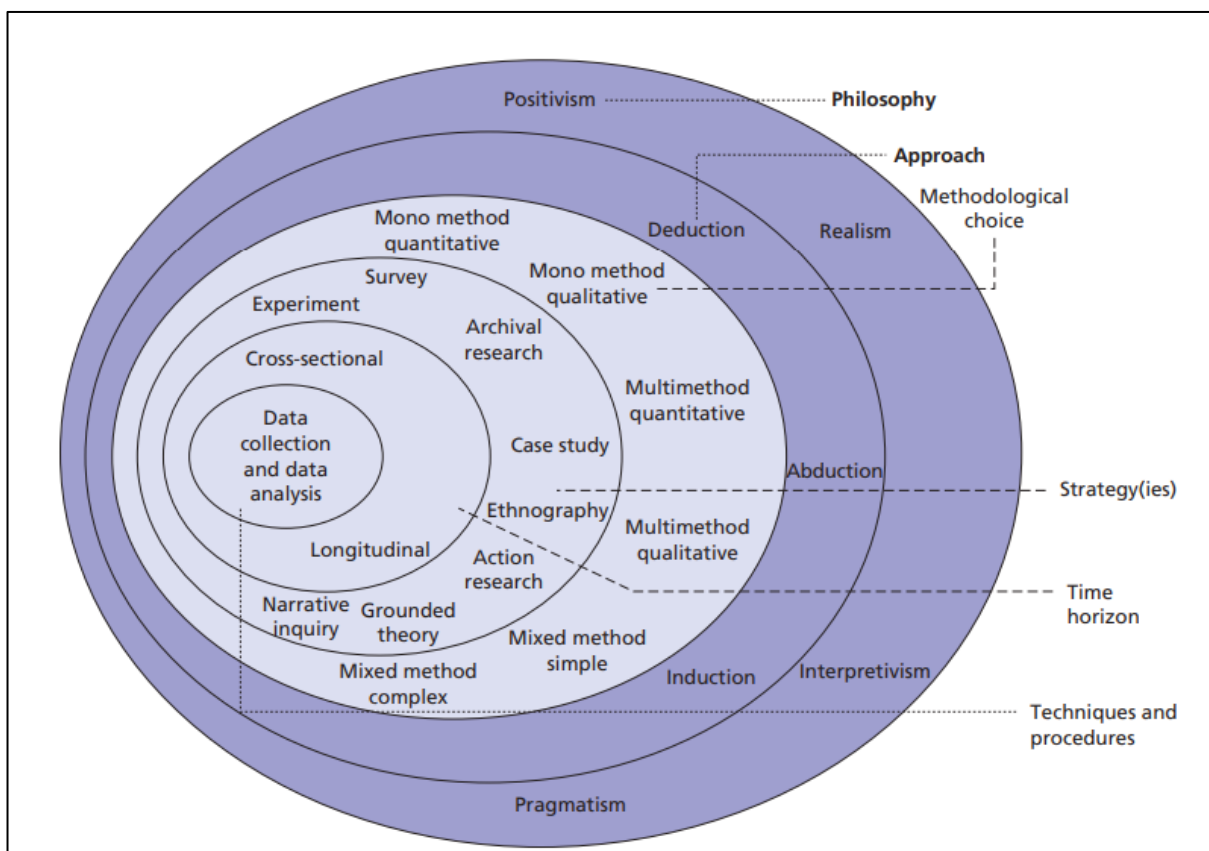
Figure 16: The Research Onion Model (Saunders et al., 2009)

Philosophy & approach

Similar to an onion, the layers are peeled (in this case explained) one by one. The first, most outer layer concerns the philosophy. This research philosophy are the assumptions about how one views and interprets the world; these assumptions will provide the basis of the research strategy and the methods as part of this strategy (Saunders et al., 2009). The research philosophy for this research is interpretivism. Interpretivism focuses on the social world of the research subjects and understanding their point of view; a highly appropriate philosophy for business situations because they are both complex and unique (Saunders et al., 2009). The objective of this research is to explore how metrics as part of a framework can be used to facilitate greater success and performance of

Agile teams. The applied research approach (second outer layer) is induction; induction focuses on generalising from the specific to the general; data collection is used to explore phenomena and to identify patterns to create a framework (Saunders et al., 2009). In this research, seven Agile teams are participating and the results from these seven teams will be generalised to answer the question on what the benefits and challenges are in introducing a dashboard which can be used to help Agile teams improve their work.

Methodological choice & Strategy
The methodological choice can be characterized as 'mono method qualitative'. The research focuses on seven participating Agile teams in a qualitative manner, as opposed to relatively larger number of teams (i.e. quantitative). The chosen strategy is to employ an exploratory action research approach. Through Action Research, researchers produce knowledge by working with practitioners to solve a practical problem and empower stakeholders (Huang, 2010).

Time horizon
The time horizon for this research is cross-sectional. It is focused on the study of a phenomenon at a particular point in time; it is time constrained as is usual for projects undertaken in the context of academic courses and projects (Saunders et al., 2009).

4.2 – Selecting Action Research as the method

What is Action Research?
Action research is a self-reflective cycle that consists of planning, action, observation and reflection (Heigham & Croker, 2009). It is a systematic approach that involves both action and research (University of Central Missouri, n.d.). As the researcher attempts to identify, explore and solve a challenge in the research context, the information is recorded is a systematic manner. After collecting this information, it is analysed and reflected upon periodically to understand what it proclaims and if any adjusting actions are necessary to then arrive at the most fruitful and impactful conclusion. (Heigham & Croker, 2009)
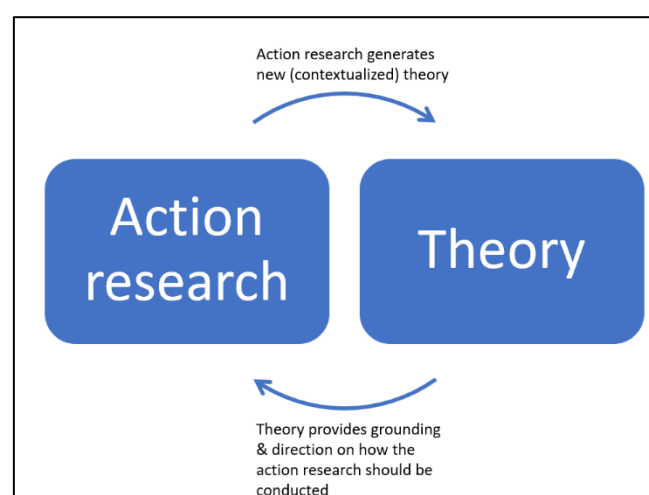

Figure 17: A visualisation of Heigham & Croker's (2009)
explanation of the relation between action research and theory

Benefits of Action Research

Susman & Evered (1978) list several benefits of action research in the context of positivist science deficiencies. Not all of these are generalizable or relevant for this research, but a few are.

First of all, Action Research is collaborative: there is an interdependence between the researcher and the client. The direction of the research process will be partly a result of the needs of the client and the competencies of the researcher.

Second, Action Research generates theory grounded in action (and vice versa). In action research, the theory provides a direction on what action should or could be taken to deal with the challenge at hand. In addition, action research contributes to that development of that very theory by taking actions and evaluating their effect on the challenge and the stakeholders. Figure 17 is a visual illustration of this process.

Third, Action research is situational. While this may seem more of a disadvantage than an advantage, in this case it would seem it is an advantage. The researcher knows and understands the context (i.e. relationships between stakeholders, events, the case study organization) of the research setting. With the knowledge of this situational context, the researcher is able to adapt the theory to fit this context in order to optimize both the quality and quantity of the results. Theories in literature are often generic, so the liberty for the researcher to test an approach and continuously analyse the results and adapt his approach is useful with so many variables at play.

Why Action Research (instead of a conventional experiment)?

Often, one cannot generalise from the results of action research. Action research projects are often undertaken in a small-scale, specific research context where the purpose is not to ensure external generalizability but rather to solve practical problems. Of course, this does not mean action research does not have the potential for generalizability, it just means that it is not the main purpose of it. (Costello, 2003)

This is both the disadvantage of action research and at the same time the reason why action research as a scientific method has been chosen for this challenge.

The fact is, the case study organization is characterized by several characteristics. Different variables and stakeholder engagement levels are at play. The challenge to solve with this research and the involved variables (e.g. stakeholders, culture, data literacy in the organization, etc.) all play a part and have an impact on the end result. It was unclear which variables could (likely) influence the research in a significant manner. With this uncertainty factor and the lacking ability to control and understand all these variables within a short amount of time, it has been decided it would be best to choose a pragmatic approach that could be adapted as the research progresses; the action research method.

Additionally, in experiments, researchers manipulate independent variables to test the effects on the dependent variables; control of variables is critical for the internal validity of the results (Bhandari, 2022). As the researcher with limited time and a limited number of participating teams, it was decided not to utilize the controlled experiment approach.

Thus, both because of action research' merits and the unsuitability of an experiment for the research setting, the action research approach was chosen.

4.3 – Case selection

This research was conducted at and in collaboration with a multinational global food corporation based in the United States of America, at their offices in the Netherlands. In April and May of 2022, several introductory conversations were held with the Project Management Office (PMO) EMEA Lead and an Agile coach working under this PMO EMEA Lead. The PMO plays a leading role in the setting and upholding of standardizations, practices and methods of project management (Silvius, 2021). The PMO conducted several internal exercises to map and understand which Agile metrics and KPIs they could use and prescribe to the Agile teams in the organization (PMO EMEA Lead, personal communication, May 2022). However, this exercise resulted in a myriad of metrics and KPIs (Agile Coach, personal communication, May 2022). While the literature does provide guidance on what metrics and KPIs exist in an Agile context, the literature scarcely provides further guidance on how useful and actionable those metrics are in practice and how to best introduce such metrics in the form of a dashboard to teams. The only discussed Agile metrics in the literature in the context of actionability are 'flow metrics' such as the cycle time and cumulative flow diagrams. The conducted literature review in the second chapter explores this topic in more detail.

4.4 – Data collection and data sources

This research uses various methods to collect data. Seven semi-structured interviews were conducted. PowerBI dashboards were built sourcing data from Azure DevOps and Microsoft Forms/Mentimeter, which were then used to observe and ask how the dashboards were used in practice by the seven scrum masters and their teams. Lastly, an interactive survey was conducted half-way through the experiment to receive feedback and advice from stakeholders.

Semi-structured exploratory interviews
First of all, seven semi-structured interviews were conducted with each team's Scrum Master and Product Owner. By conducting these interviews, the products the teams were working on were better understood, along with the current metrics and measurements they were using at that time. Current opportunities and challenges for each team were put into perspective and they gave their view on the action research which they would be participants in. The questions used for these interviews can be found in Appendix C.

Azure DevOps data & survey data
Furthermore, for each of the seven Agile teams in the case study organization, a PowerBI[2] dashboard was created. The data displayed in these dashboards came from two sources: the Azure DevOps[3] environment of the respective teams and the results of the customer & employee satisfaction surveys. Some of the teams used Mentimeter to assess the employee satisfaction and some of the teams used Microsoft Forms to assess the employee satisfaction, but the same questions were used regardless of the medium.

Weekly conversations with each of the seven scrum masters
To evaluate how the scrum masters used the dashboard and its containing metrics, weekly conversations were held with each of the seven scrum masters. During these conversations, which took around half an hour in most cases, the scrum masters were asked several questions to assess if and how they used the dashboard.

---

[2] PowerBI is Microsoft's platform for self-service and enterprise business intelligence (Microsoft, n.d.)
[3] Azure DevOps is a collection of tools to help software development teams track their work, manage their code, deploy their applications and run tests. (Sokolov, 2022)

Interactive survey to receive feedback

Lastly, an interactive and real-time survey was conducted half-way through the action taking-phase. This involves providing a link of the interactive survey to the participants. The participants are able to answer the questions one by one, while the results of the answers are shown per question. These answers are then used as input for discussing and clarifying the answers with all the participants. Numerous stakeholders were invited: all of the Scrum Masters and Product Owners of the seven teams, various Agile coaches from different regions and representatives of the PMO. This session was meant to showcase the results so far and receive the attendees' feedback on the progress so far and which additional steps to take in the research.

4.5 – Principles of Participatory Action Research

To ensure a credible and valid research method, the eleven principles of Participatory Action Research are applied (Kemmis et al., 2014).

1. *Establish working rules for the collaborating group*: The seven participating teams were informed about how the experiment is set up, what is expected of them and what value they can get out of participating in the research. Scrum masters were asked to open and use the dashboard to see how they used the metrics and how they interacted with the dashboard.

2. *Observe protocol*: Relevant stakeholders – both scrum masters and others such as management – were aware of this research. Backlog data of the seven teams was obtained by asking scrum masters for permission to extract and use the data. A non-disclosure agreement was signed to make all parties aware that confidential data should be handled with integrity.

3. *Involve participants*: Not only the scrum masters were seen as stakeholders. The team members of each of the seven teams and the product owners were informed and involved. The product owners were mostly involved with the customer satisfaction surveys, while the team members were introduced to the purpose of the research and the dashboard in the first weeks of the action taking phase.

4. *Negotiate with those affected*: Not everyone would want to be directly involved. The scrum master was seen as the primary contact point for each team. This way, each participating team could focus on their work as they normally do, while the scrum master supported their team with the dashboard and their other responsibilities.

5. *Report progress*: Half-way through the action-taking phase, a progress update in the form of a presentation and interactive Mentimeter session was conducted. This way, all relevant stakeholders could see and understand the progress that had been made so far. The Mentimeter session was used to collect input and feedback, based on what was shown in the presentation and what the second part of the action taking phase should take into account.

6. *Obtain explicit authorisation before you observe*: The scrum masters were interviewed on a weekly basis and during the Evaluating-phase, seven exit interviews were conducted. For all of these, explicit authorisation was requested (to record and transcribe these sessions) and were granted.

7. *Negotiate description of people's work and accounts of others' point of view*: It was made clear to the scrum masters that if they would like to amend anything from either the weekly conversations or the exit interviews, that would be possible. However, this was not used or requested.

8. *Negotiate reports for various levels of release*: Sometimes stakeholders not involved in the research were interested in which metrics were used in the research and what the dashboard looked like. For these situations, the dashboards and metrics were anonymized. Anonymisation of data and

visuals also was applied when an external organisation invited me to present the findings of this research.

9. *Accept responsibility for maintaining confidentiality*: when documents or data was presented or sent to stakeholders, it was either verbally or in writing mentioned that they have the responsibility to uphold confidentiality.

10. *Retain the right to report your work*: Through an amended non-disclosure agreement, the rights to anonymously report the results of this research were obtained. Both the final presentation that was conducted in the case study organization and possible future publications are anonymized, as to not cause harm (e.g. embarrassment) to those involved.

11. *Make your principles of procedure binding and known*: Before the action taking phase began, the scrum masters were informed through e-mail about what the intention was of this research, containing what value do they get by participating, what are the ground rules and what I will do with the information. If they had any questions, these were answered during the introductory interview sessions with the scrum masters.

4.6 – Research Design

While the scientific literature agrees on the purpose and benefits of action research, the four (or five) stages of action research are named and characterized differently. Heigham & Croker (2009) state that action research consists of planning, action, observation and reflection while Mertler (2009) provides the stages planning, acting, developing and reflecting. The action research stages of Kemmis et al. (2014) have been chosen for this research, because it provides the most appropriate stages to group the performed activities under: reconnaissance, planning, enacting & observing, reflecting. These four stages as described by Kemmis et al. (2014) are slightly changed in the wording, based on the research of Renee Boot (2022). The four stages of this action research along with the activities, are shown in figure 18 below:
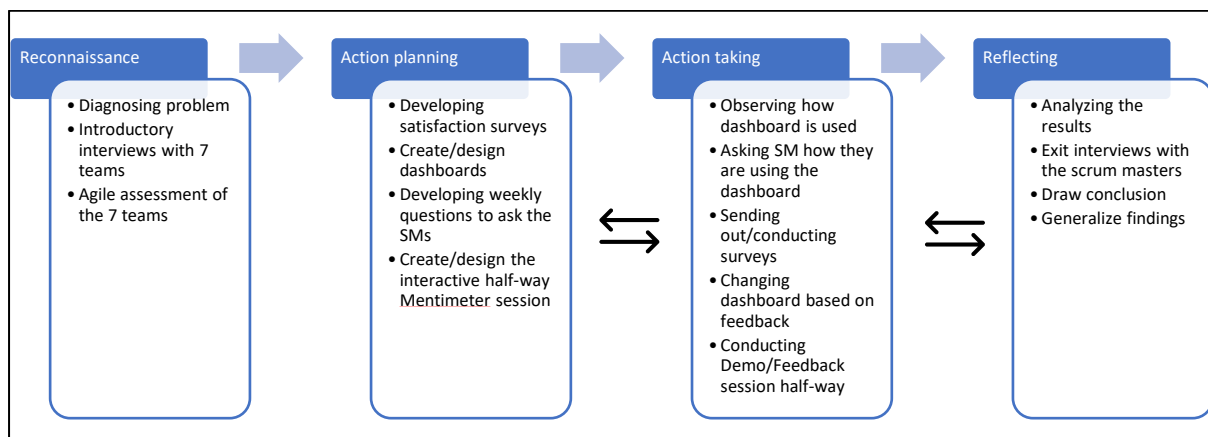

Figure 18: The Action Research framework with corresponding activities for this thesis

Reconnaissance
According to Kemmis et al. (2014), the reconnaissance stage involves exploring what the problems, needs, challenges and opportunities are of the stakeholders involved in and affected by the research. In the context of this action research, the reconnaissance stage involves understanding what the problem or challenge is, how it came about and what the potential merits are in solving this particular challenge.

The reconnaissance stage contains three activities as shown in figure 18; the first of which is diagnosing the problem and describing the context/teams. To diagnose the problem, conversations were held with the PMO EMEA Lead and his Agile Coach to understand why they would like to research which Agile metrics and visualizations could be used for Agile teams in the case study organization.

The second activity concerns conducting seven interviews with each of the seven participating teams' scrum master and product owner. The goal of these interviews was to understand the team, the business case the team is working on and what their experience is with using Agile metrics and visualizations.

The last activity in this reconnaissance phase was to assess the Agile maturity of each team. This was done so potential findings could potentially be linked to the Agile maturity of the teams. Additionally, since an even distribution was desired with which medium the team would be assessing their employee satisfaction (i.e. Microsoft Forms or Mentimeter), this Agile maturity mapping would help in creating an even distribution.

Action planning
Similar to the reconnaissance phase, Kemmis et al. (2014) provides some guidance on what the planning phase entails. Kemmis et al. (2014) states that the planning stage is about deciding which steps to take and how evidence will be gathered and documented.

In this action planning phase, several instruments and tools will be created and designed which will be used throughout the action taking/experiment phase. An adaptation of the American Customer Satisfaction Index was used to create the customer satisfaction survey. The Minnesota (Job) Satisfaction Questionnaire was the basis for the employee satisfaction survey.

Throughout the action taking phase (i.e. experiment), seven PowerBI dashboards were designed with data from both Azure DevOps and Microsoft Forms/Mentimeter. While these dashboards were developed for each team containing different data, the visuals remained mostly the same. The case study organization is interested in a dashboard they can quickly provide to Agile teams in the organization (Agile Coach, personal communication, August 2022). To guarantee the applicability of the research' results, it was imperative not to customize the dashboard in such a way that none of the dashboards are similar.

Since the purpose of the action taking phase is to observe what the benefits and challenges are of introducing the dashboard and which metrics are most actionable and useful, some questions had been developed for the researcher to ask the scrum masters on a weekly basis. Asking users about their use of the dashboard is not a new phenomenon, as Santiago Rivera & Shanks (2015) also surveyed users of a dashboard to support management of business analytics capabilities. However, Santiago Rivera & Shanks (2015) used a standard survey containing 12 questions, each involving a 5-point Likert scale, while this action research adopted open-ended questions during weekly calls with the scrum masters.

These open-ended questions were revised several times during the action taking phase, but the purpose of asking these questions was to assess if they opened their team's dashboard, what their thoughts were and if they took any action.

Since the weekly conversations were held with each of the scrum masters individually, the scrum masters were unaware of the general over-arching findings and conclusions. For this reason, halfway through the experiment/action taking phase, a presentation was held with all of the scrum masters, product owners, various Agile coaches and other stakeholders. This presentation was meant as an

update to showcase what had been done so far and to collect feedback on what the next steps should look like.

Action taking

Action taking involves acting on what had been planned in the previous stage and observing its effects during this phase. After a certain period of time, reflecting on what happened is necessary: the preliminary evidence and documentation are critically analyzed and interpreted. This provides input for potential re-planning. (Kemmis et al., 2014)

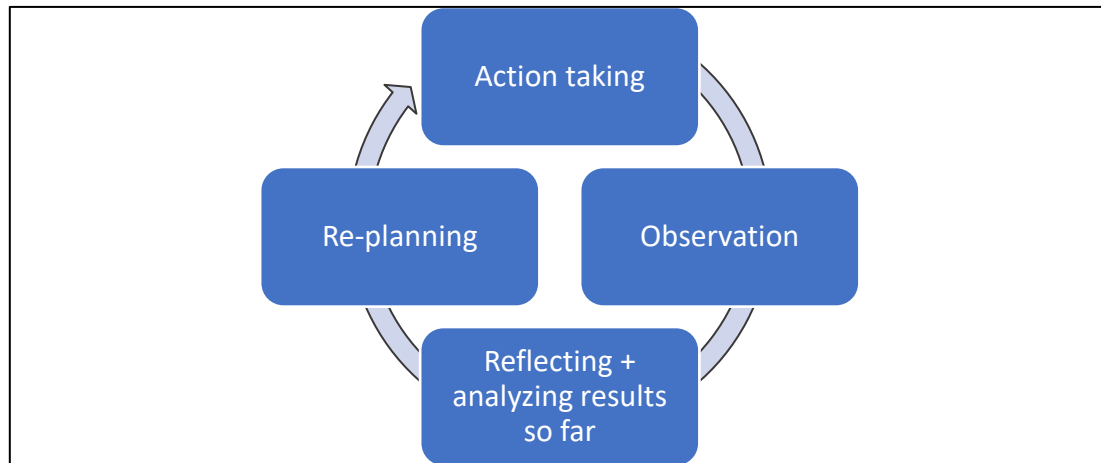This action taking-loop is visualized in figure 19:

Figure 19: The Action Taking-loop, based on Kemmis et al. (2014)

Several activities are carried out in this action taking phase. The first is observing how the dashboard is used by the seven teams. This is done both through attending some of their retrospectives and stand-ups and showcasing the dashboard to the scrum master every week and asking them what are their thoughts on the information displayed in the dashboard.

Second, during these weekly conversations with the scrum masters, they are asked questions about their use of and thoughts on the dashboard this week. Third, during this action taking phase, employee and customer satisfaction surveys were sent periodically and shown in the dashboard to the respective team and scrum master.

As showcased by figure 19, during this action taking/experiment phase, the results are periodically analyzed and reflected upon in order to see if any re-planning or adjustments are necessary.

Evaluating

While evaluating is a continuous process during the action taking phase, the last phase in this action research is also called evaluating. In this phase, everything comes together and is analyzed, synthesized, interpreted and explained (Kemmis et al., 2014). Conclusions are drawn for the specific case study organization and are generalized to help other practitioners and organizations who face the same or a similar challenge.

Besides the generic activities that are part of any research - analyzing the results and drawing conclusions – the action taking phase in particular will also be evaluated and reflected upon.

Since the Scrum Masters were the most involved stakeholders during the action taking phase, these scrum masters were interviewed at the end of the experiment to understand their thoughts and opinions on the dashboard and experiment.

5 – Applying Action Research

The high-level timeline of the executed action research and the four stages are outlined in table 6.

Table 6: A high-level timeline of the executed action research

	Reconnaissance	Action Planning	Action Taking	Reflecting
May, 2022	Introductory conversations with EMEA PMO Lead and the Agile Coach			
May & June 2022	Introductory interviews with 7 Agile teams			
June & July, 2022		Create/design dashboard		
July & August, 2022		Develop satisfaction surveys		
August, 2022		Create questions for SMs		
November, 2022		Create the interactive half-way Mentimeter session		
November, 2022:		Create questions to ask the SMs as a reflection on the experiment		
August 22 till November ??, 2022			Observe how dashboards are used in daily stand-ups and retro's	
August 22 - November 11, 2022			Ask SMs in a weekly call if and how they used the dashboard	
August 22 - November 11, 2022			Conduct employee/customer satisfaction surveys	
November, 2022				Analyze, synthesize the results
November, 2022				Generalize findings beyond the case study
November, 2022				Conducting exit interviews with SMs
December, 2022				Reflection & Conclusion

5.1 – Reconnaissance

In this section, the context in which the research is conducted, will be explained. The stakeholders are identified, the research problem is clarified and an overview of the seven teams is given.

Who are the stakeholders?
To examine the current situation in this Reconnaissance stage, several introductory and exploratory interviews were conducted with two important stakeholders. Stakeholder/interviewee A is the EMEA PMO Lead at the case study organization and is also the sponsor for this research. Stakeholder/interviewee B is an Agile Coach working in the EMEA PMO department.

Besides the sponsors of this research, the scrum masters of the seven participating Agile teams are also important stakeholders. They have used the dashboard, showed it to the team and had potentially taken action on it.

What is the challenge to solve & what has already been tried/tested?
At the moment, the case study organization is undergoing an Agile transformation journey. Originally the challenge to solve with this research was to determine what the right KPIs/OKRs are to measure the success of such an Agile transformation and how to act on these metrics. However, after several conversations with Stakeholders A and B, the challenge to solve was altered and the focus shifted to how the organization could develop a set of metrics and visualizations to facilitate the success and performance of the Agile teams. This main focus was then changed to what the benefits and challenges are of introducing such a dashboard to Agile teams to improve their work.

Before this action research, the case study organization conducted an internal exercise to attempt to answer that very question. They brainstormed with several stakeholders on what metrics they could use in the organization to track the progress, success and performance of the Agile teams (Stakeholder A, personal communication, May 2022). This internal exercise conducted with the tool Mural resulted in a longlist of metrics categorized in: cost, quality, speed, client experience, risk and leverage. However, this longlist of metrics did not provide the stakeholders with an understanding of how actionable these are in practice and what the benefits and challenges are of introducing such a dashboard to Agile teams. There was no simple way to transform this longlist to a shortlist without basing it solely on assumptions and gut-feeling. That is also why the action research methodology had been chosen; to test and iteratively adapt the approach in order to aid the organization with their higher-level Agile transformation journey.

Seven participating Agile teams for the experiment
To actually test which metrics and visualizations are useful or actionable for the Agile teams in the organization, some of the metrics from the literature review are tested in the action taking phase. Seven Agile teams are participating in this action taking experiment phase, where a dashboard has been created based on each team's Azure DevOps data and the relevant employee/customer satisfaction survey answers. This main focus of this research is to understand the benefits and challenges of introducing such a dashboard to help Agile teams improve their work.

But before the dashboards were created and tested in the action taking phase, semi-structured interviews were conducted (see Appendix C for the questions used in these interviews) with each team's scrum master and product owner. These interviews served several purposes. First of all, it was to introduce the researcher to the scrum master and to explain the challenge this research was trying to solve. Second, these interviews provided the researcher with information on the team composition, team Agile maturity and an understanding of what the product or service they are working on for their customers. Third, these interviews provided an opportunity to explain the merits of participating as a team in this action taking phase.

Thus, seven semi-structured interviews were conducted for each of the seven teams, with each interview attended by the respective scrum master and product owner. The most important findings are summarized below for each team in table 7:

Table 7: Most important findings from the conducted introductory interviews with the seven participating Agile teams

Team	Team size	Team start date	Agile maturity	Regular retro/ demos	Using metrics already?							
					Velocity	Burndown	Remaining hours vs. estimated	Value delivered	Bugs	Throughput	General health metrics	Cycle Time User stories
Team A	2 back-end, 2 front-end developers. All part-time	July 2021	Between low and medium	No, ad-hoc.								
Team G	2 back-end, 1 front-end developer, 1 BA. All part-time	January 2021	Low according to SM, hybrid Agile/waterfall	Retro part of a larger team, no demos because MVP just released		x						
Team F	9 developers, mix of competencies	Spring 2022	Low, according to SM/AC	Bi-weekly status update, that's it. Retro every 2 weeks	x	x	x					
Team B	13 developers, 8.5 FTE	2016. Team now is 2019	Very mature. DoD is there.	Yes, retro every 2 weeks. Demo in place	x	x		x	x			
Team E	~10 team members, 6 FTE	?	Low/ n/a: waterfall	More ad-hoc now								
Team C	5-6 part-time developers	February 2022	High	Yes, every 2 weeks.	x	x				x	x	x
Team D	6 FTE, spread across ~10 members	2020	Mature, 7/10 according to SM	Yes, retro is ad-hoc, no sprint review								

The teams' Agile maturity

As is evident from table 7, the participating teams have quite differing characteristics. Team sizes, Agile maturity and experience in using metrics all differ from team to team. What's interesting is the fact that each team uses different metrics, partly because of its perceived usefulness by the scrum masters/team.

Additionally, the scrum masters were asked to judge their team's Agile maturity. While the scrum masters may have a judgement, this can be quite subjective. For this reason, each team's Agile maturity and mindset was assessed through a short Microsoft Forms questionnaire. The framework used to assess the Agile maturity, is the Agile Transformation maturity model based on the work of Laanti (Stettina et al., 2021; Laanti, 2017).

Figure 20: The Agile Transformation Maturity model of Laanti (Laanti, 2017)

Based on this Agile transformation maturity model, the first question of the Agile assessment was asked to determine the Agile maturity of the teams (see Appendix D for the whole survey). Besides this question, twenty additional questions have been asked in this survey. These questions concern the Agile mindset. These twenty Agile mindset questions are inspired by the works of Eilers et al. (2022), since he has conducted extensive research on how to assess the Agile mindset and why it is so important. The wording of these twenty questions have been copied from the survey of Lim (2022), since he also attempted to assess the Agile mindset of Agile teams.

By cross-examining the results of the Agile maturity/mindset surveys and the findings from the introductory interviews with the each of the seven scrum masters, the following result is displayed in table 8:

Table 8: Agile maturity of each team, based on self-reporting of the Scrum Masters and the Agile assessment survey

	Team A	Team B	Team C	Team D	Team E	Team F	Team G
Agile maturity	Novice/ Fluent	Fluent/ Advanced	Fluent/ Advanced	Advanced	Beginner	Novice/ Fluent	Fluent

To conclude this Reconnaissance phase, it is evident that the organization does not have a clear framework to give the Agile teams with which they are able to facilitate and help the teams in increasing their success and performance. Various teams have been interviewed and analyzed, but each of these teams has a differing Agile maturity level and experience in using metrics.

5.2 – Action Planning

The action planning phase serves to decide on and design the various instruments and tools which will be used throughout the action taking/experiment phase to solve the research challenge. This includes the employee and customer satisfaction surveys, the PowerBI dashboard for the teams and the interactive Mentimeter session with numerous stakeholders half-way through the action taking phase.

5.2.1 – Employee satisfaction survey

While measuring the output, quality and responsiveness of the team may be useful or actionable, it's also important to understand whether the members of the Agile teams (i.e. employees) are satisfied. In chapter 2, multiple models relating to employee satisfaction have been discussed. The most discussed and the most practical model (from a pragmatic perspective) is the Minnesota Employee Satisfaction Questionnaire. This MESQ has both a short (20 questions) and a long version (100 questions), both corresponding to 20 categories (Weiss et al., 1967), as outlined in chapter 2.

However, even the short form of the MESQ is deemed too much for this research, since the survey should be both applicable in an Agile context and is conducted every 2 to 4 weeks. Asking the employees to answer a survey containing 20 questions every 2 to 4 weeks is too excessive and will most probably negatively impact the response rate.

Instead, several dimensions were identified together with stakeholders, which were deemed important to assess every one to two sprints.

Table 9: Mapping of the employee satisfaction survey questions to the Minnesota Employee Satisfaction Questionnaire

#	Dimension deemed important to assess	Maps to MESQ questions (short form version)	Corresponding question[4]
1	Collaboration	Q9 & Q18	How satisfied are you with the collaboration within the team?
2	Autonomy	Q15 & Q16	How satisfied are you with the ability as a team to take decisions that lead to better results?
3	Accomplishment feeling/prize	Q19 & Q20	How proud are you of your accomplishments as a team?
4	Ability	Q11	How satisfied are you with the opportunity to apply your skills and knowledge within your team?
5	Stress-levels	Q1, Q13 & Q17	How satisfied are you with the amount of work that was assigned to you?
6	General/overall satisfaction	-	How would you rate the overall satisfaction of your team?

---

[4] The response scales for question 1, 2, 4, 5 and 6: (1) Not at all satisfied, (2) Slightly satisfied, (3) Moderately satisfied, (4) Very satisfied, (5) Completely satisfied. The response scale for question 3: (1) Not at all proud, (2) Slightly proud, (3) Moderately proud, (4) Very proud, (5) Completely proud.

Unipolar vs. Bipolar response scale

Table 9 contains the questions, but not the corresponding response scale. The MESQ short form uses a 5-point bipolar Likert scale (i.e. very dissatisfied, dissatisfied, neutral, satisfied, very satisfied) (Weiss et al., 1967). This bipolar approach to the response scale is often based on the assumption that satisfaction and dissatisfaction can occur concurrently; if the respondent marks 'very satisfied', the rating is also interpreted that the respondent is not 'very dissatisfied'. The bipolar approach is thus unsuitable to assess the satisfaction; the feeling of satisfaction does not exclude the feeling of dissatisfaction (and vice versa). (Um et al., 2021)

 The second option for the response scale is a unipolar scale; this unipolar scale is measured from a zero point to an extreme (e.g. 'Not satisfied at all' as a zero point and 'Completely satisfied' as an extreme). (DeCastellarnau, 2018)

The reliability of unipolar response scales are higher than bipolar rating scales (Alwin, 2007). Additionally, the unipolar response scale has been extensively used to assess customer satisfaction in various service industries because of its powerful predictability and theoretical and practical reasoning. (Um et al., 2021)
For these reasons, a unipolar response scale is used to assess the employee satisfaction periodically.

5- vs. 7-point Likert scale

While the unipolar scale is used for this research instead of MESQ's bipolar scale, the MESQ uses a 5-point Likert scale. A Likert scale is a rating scale that quantitively assesses a respondent's opinion, behaviour or attitude (Scribbr, 2022). There has been much debate on whether a 5- or 7-point Likert scale is better. Joshi et al. (2015) explore some of the arguments for the 7-point Likert scale, such as an increased probability of meeting the respondent's reality.

However, even though there are clear merits in using the 7-point variant, this survey is repeated every two to four weeks with the seven participating teams. Since employees are already busy enough as it is, they may suffer from decision fatigue: an impaired ability to take decisions as a result of repeated decision-making (Pignatiello et al., 2020). The 7-point scale may provide a slightly higher reliability, but pragmatically the potential trade-off by causing decision-fatigue would not be worth it. The employee satisfaction questionnaire should be simple, uncomplicated and fast to fill out.

Since the survey is conducted periodically, it is important to design the survey to be as simple as possible to increase the possible response rate. The 7-point response scale (which has more options than a 5-point scale) may bring about decision fatigue. Thus, for the employee satisfaction survey, the 5-point unipolar Likert scale has been applied. A 7-point scale provides higher reliability (Joshi et al., 2015), but a 5-point scale could potentially be rescaled to a 7-point one with a simple scaling method (Dawes, 2008).

Mentimeter or Microsoft Forms as the medium

As previously mentioned, some of the Agile teams would be using Microsoft Forms (sent via e-mail) to stipulate their satisfaction and some of the Agile teams would be using an interactive Mentimeter session during their retrospectives. Based on the conducted interviews with the scrum masters and considering all the teams' Agile maturity, table 10 shows which team is using which medium and how often to assess the employee satisfaction.

Table 10: The team's Agile maturity and their designated medium and interval to assess the employee satisfaction

	Team A	Team B	Team C	Team D	Team E	Team F	Team G
Agile maturity	Novice/ Fluent	Fluent/ Advanced	Fluent/ Advanced	Advanced	Beginner	Novice/ Fluent	Fluent
Medium	Microsoft Forms	Menti Meter	Menti Meter	Microsoft Forms	Microsoft Forms	Menti Meter	Microsoft Forms
Interval	Every month	Every retro-spective (2 weeks)	Every retrospective (2 weeks)	Every month	Every month	Every retro-spective (2 weeks)	Every month

5.2.2 – Customer satisfaction survey

Multiple customer satisfaction frameworks/models have been discussed. For this action research, the American Customer Satisfaction Index (ACSI) has been selected as the framework to base the customer satisfaction surveys on. The reasoning behind this is twofold. First of all, from the literature it seemed the ACSI was the most influential and oft-cited framework. Additionally, the antecedents in the ACSI are general enough to base the questions in the customer satisfaction survey on. With stakeholders from the case study organization, it was agreed upon to use the ACSI as a basis for the design of the customer satisfaction survey. Thus, six questions were formulated based on four of ACSI's antecedents. Similar to table 9, table 11 contains the linkage between the questions used in the customer satisfaction survey in this action research and ACSI's antecedents.

Table 11: Mapping of the customer satisfaction survey questions to the antecedents of the American Customer Satisfaction Index

#	Antecedent of the ACSI	Corresponding question[5]
1	Customer loyalty	On a scale from 1 to 10, how likely would you recommend this development team to your peers?
2	Perceived Quality	How satisfied are you with the quality of the product?
3	Perceived Value	How satisfied are you with the value the product provides to you?
4	Customer expectations	How satisfied are you in having your overall expectations met? (i.e. in general, does the team/product meet your expectations?)
5	Customer expectations	How satisfied are you in having your expectations met with regards to communication (e.g. does the team communicate in a timely and honest way?)
6	Customer expectations	How satisfied are you in having your expectations met with regards to predictability (i.e. did the team deliver what they promised to do)?

---

[5] The response scales for question 2 to 6: (1) Not at all satisfied, (2) Slightly satisfied, (3) Moderately satisfied, (4) Very satisfied, (5) Completely satisfied.

The antecedent for question 1 is 'customer loyalty', which is the result of the overall customer satisfaction. This first question was formulated from a Net Promoter Score-perspective.

5.2.3 – PowerBI Dashboard

Besides the various surveys that are conducted periodically, the action taking phase also accommodates the use and testing of the PowerBI dashboard in collaboration with the seven participating Agile teams. Before this dashboard is used in practice by the various teams, the dashboard had to be designed and the data had to be connected in a logical manner. The dashboard contains data from three data sources: the Azure DevOps environment for each team, the Microsoft Forms/Mentimeter employee satisfaction survey results and the Microsoft Forms customer satisfaction survey results.

Azure DevOps

As previously mentioned, Azure DevOps (ADO) is a collection of tools to help software development teams track their work, manage their code, deploy their applications and run tests (Microsoft, n.d.). With this tool, the various Agile teams are able to create dashboards, manage their backlog, organize their Scrum board, keep track of bugs, assign work to team members in an organized manner, conduct their retrospectives and much more. These processes in ADO create myriads of data.

ADO provides users with the 'Analytics views' functionality. This functionality enables users to specify which Azure Boards data the user would like to export for a PowerBI report (Microsoft Learn, 2022). For example, a custom view can be created that only shows and exports all data related to bugs (e.g. creation date, solved date, severity, title, etc.). This data can then be accessed through PowerBI, providing a simple and easy way to visualize ADO-data in PowerBI .

Thus, for each team, several custom 'views' had been created: Bugs, Epics, Features, User Stories, Tasks and Impediments.

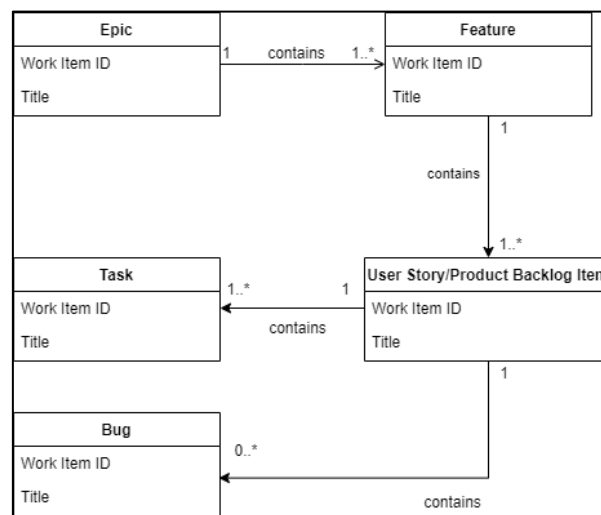The simplified data model can be seen in figure 21:

Figure 21: Class diagram of the PowerBI data model

The objects Feature, User Story, Task and Bug all include a field 'Parent Work Item ID', which serves as the foreign key, while the Work Item ID serves as the primary key.

Displaying employee & customer satisfaction survey results in the dashboard
The second and third source of the dashboard are the results of the employee and customer satisfaction results. These results are exported from Microsoft Forms/Mentimeter to an Excel workbook. This Excel file is placed in a SharePoint environment, after which PowerBI is able to connect and load this data into the dashboard.

Dashboard design
Since considerable amounts of data is exported from ADO to PowerBI, numerous metrics and visualizations could be incorporated. However, the EMEA PMO Lead mentioned he would like the dashboard to be one page (EMEA PMO Lead, personal communication, August 2022). This instruction, paired with multiple rounds of feedback and design iterations, resulted in the dashboard shown in figure 22.
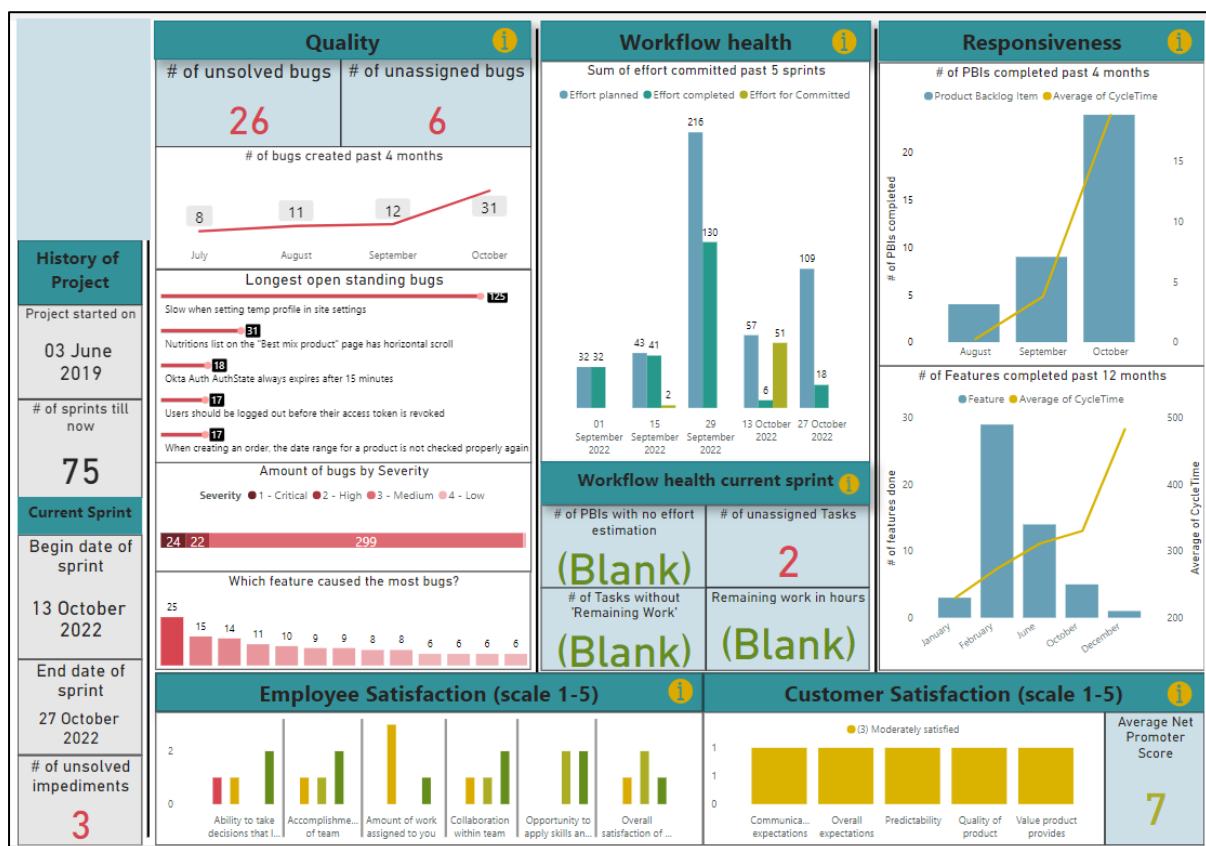

Figure 22: The PowerBI dashboard for the Team B team

While this dashboard is a screenshot of one of seven dashboards (corresponding to the seven Agile teams), the categories and the visualization method remains the same for all the dashboards. Each dashboard is divided into key sections corresponding to the categories quality, workflow health (current sprint), responsiveness, employee satisfaction and customer satisfaction. These categories have been chosen for several reasons. First of all, the categories were partly inspired by Stettina et al.'s dimensions of organizational performance (Stettina et al., 2021). Since all the discussed performance frameworks in chapter 2 were compared to these dimensions of organizational performance, these dimensions are also used as a basis to decide what to include in the dashboard.

Stettina et al.'s (2021) dimensions of organizational performance	Included in dashboard?
Productivity	No
Responsiveness	Yes
Quality	Yes
Workflow health	Yes
Customer satisfaction & engagement	Yes
Employee satisfaction	Yes

Productivity
It was discussed with the EMEA PMO Lead that explicitly mentioning and visualizing productivity could be interpreted as micro-managing and thus make the teams adverse to using the dashboard and participating in the experiment. While there may not be evidence for this, due to the concerns of the EMEA PMO Lead and the limited area of the dashboard canvas, the Productivity-category was not included.

As for the metrics and visualizations under each category in the dashboard, these were decided upon both through iterative sessions with the stakeholders involved and because of the limitation of available fields exported to PowerBI.

Quality
While quality is not exclusive to showing bugs, because of the limitations as to what is registered and exported, the quality category is constrained to data relating to bugs. The first two KPIs ('# of unsolved bugs' & '# of unassigned bugs') are intended to help the team visualize how many bugs there currently are and how many are not assigned yet. Furthermore, making sure bugs are assigned to team members promotes accountability; it is thus helpful to know whether there are bugs for which nobody is responsible yet. The next visual is '# of bugs created in the past 4 months'. This graph shows the team and its scrum master a proxy measure of the product's quality the past 4 months; is the number of bugs over time increasing or decreasing? The next visual is the 'Amount of bugs by Severity'. This shows the distribution of bugs and their severity; it shows whether the team only uses one severity level or do they actually differentiate between severity levels. The last graph for quality is 'Which feature caused the most bugs'. As mentioned previously, an epic consists of one or more features. If a certain feature has a relatively larger number of bugs than the other features, the quality control could be focused on these bug-ridden features.

Workflow health
Velocity graph shows the team how much effort is planned each sprint and whether or not they are completing their planned work. The 'Effort Planned' bar in this graph shows whether the planned workload is consistent across the past 5 sprints; the same applies for the bar 'Effort completed'. The velocity chart enables the team to know whether they are systematically over- or underestimating (and completing) their work.

Workflow health current sprint
These KPIs are health metrics, intended to aid the scrum master and the team in their effort to correctly administer information in ADO. The velocity chart is dependent on the fact that every user story has an effort estimated. These health metrics for the current sprint make sure the scrum master and the team did not forget to administer something, especially since some visuals are dependent on the correct administration of information in ADO.

Responsiveness

Responsiveness is concerned with the output of the team and how long it takes them to produce this output; both in user stories and features. These graphs show the team whether they have a consistent completion rate of features for the customer, and whether they are completing user stories within the 10 days a sprint usually lasts (i.e. average cycle time should be 10 days or less).

Employee satisfaction

The results of the employee satisfaction are visualized in the dashboard as well. This helps the team understand on a team-level how happy/satisfied they are and if there are areas in which improvement is needed. This will most probably be used by the scrum master to understand how the satisfaction can be increased in a particular area.

Customer satisfaction

A team may have a consistent output of features and have a stable velocity, but that does not automatically ensure the customer is satisfied. Thus, the results of the customer satisfaction survey are also visualized.

Limitations of the PowerBI dashboard

As with all research, sufficient reasoning has to be provided on why certain decisions have been made. First of all, the choice for a PowerBI dashboard instead of a native ADO dashboard is due to the fact PowerBI supports external data sources (such as survey results), while an ADO dashboard can only visualize the data contained in ADO. This comes with a catch; the exportation of data from ADO to PowerBI is limited. Not all available data from ADO is exported to PowerBI. An example would be creating a velocity chart in both ADO and PowerBI. ADO 'remembers' if a user story has been moved from one sprint to another, which is shown in the velocity graph in an ADO dashboard. However, this information is not able to be exported to PowerBI, thus in some cases the velocity chart in PowerBI displays incorrect data.

However, even though it may sometimes occur PowerBI cannot display the data in an accurate manner (of which the team knows), the purpose and goal behind the action taking phase is to observe how the dashboard is used and which metrics or visualizations are actionable or/and useful.

To assess whether the dashboard is used by the scrum masters and their teams, the scrum master was informally interviewed every week and asked whether or not they used the dashboard.

5.3 – Action taking

Action taking involves acting upon the plan made in the previous stage. (Kemmis et al., 2014)

5.3.1 – Employee satisfaction surveys

Table 13 shows for each of the seven teams what the responses were during the action taking phase. Each team (except for one) was asked to fill out the employee satisfaction survey via Microsoft Forms as a reference/zero-point (first row of the table). From the first iteration and onwards, each team either used Microsoft Forms (MF) or Mentimeter (MM) as a tool to assess the employee satisfaction.

If Microsoft Forms as a tool was assigned to a team, they were asked by e-mail to fill out the survey; the scrum master was subsequently asked to promote this survey in various meetings and communication channels. If the teams used Mentimeter as a tool to assess the employee satisfaction, the researcher was present during the team's retrospective and asked the same 6 questions in an interactive Mentimeter session. During this session, the results would be hidden for

each question by the researcher and would only be shown when every participant had answered the question, as to not influence the answers of the respondents.

Observations

During these retrospectives with Mentimeter as a medium, the scrum masters would mostly say those answers confirmed their 'gut feeling' as a scrum master, but it was helpful to have those answers as to quantify it.

As table 13 shows, teams using Microsoft Forms as a tool to assess the employee satisfaction showed a mostly decreasing response rate. Three out of the four teams which were using Microsoft Forms, had a decreasing response rate as the action taking phase progressed. This could be due to the fact employees think filling out surveys is cumbersome, or perhaps they are simply too busy.

In contrast, teams using Mentimeter as a tool to assess the employee satisfaction showed a consistent response rate; everyone who was present during the teams' retrospectives were asked to join the Mentimeter session and answer the questions.

Whether the teams were using Microsoft Forms or Mentimeter, the results would be shown in the dashboard which would be opened weekly by the scrum masters and shown to their teams. If this had not been the case, one could have explained the decreasing response rate for Microsoft Forms by saying the team does not know why the questions are asked and what the data will be used for.

Thus, the barrier to filling out the employee satisfaction survey is lower when the survey is conducted in an interactive manner when all the team members are already present. Conducting the survey by sending the survey link via e-mail resulted in less responses.

Table 13: Overview of the number of responses for the employee satisfaction surveys[6]

	Team A (MF)	Team B (MM)	Team C (MM)	Team D (MF)	Team E (MF)	Team F (MM)	Team G (MF)
Reference	4	x	4	3	2	10	5
Iteration 1	2	8	5	1	2	8	4
Iteration 2	0	x	5	1	1	6	4
Iteration 3	0	7	4	x	x	6	x
Iteration 4	x	x	4	x	x	x	x

5.3.2 – Customer satisfaction surveys

Unlike the employee satisfaction surveys, the customer satisfaction surveys were only conducted by using the Microsoft Forms tool. The link of this survey was periodically sent via e-mail to the customers/end-users of each team. Since the customer satisfaction surveys were sent less often than the employee satisfaction surveys, it is difficult to say whether the response rate is decreasing or not.

Similar to the employee satisfaction survey results, the results of the customer satisfaction surveys were also visualized in the dashboard. The scrum masters mentioned several times during informal conversations that quantifying and viewing the customer satisfaction is very useful and actionable for the scrum master and their team. Ultimately, it's about delivering value to the customer/end-user and this survey provides an indication on how well the team is doing in this regard. They also

---

[6] An 'x' indicates no survey was conducted.

mentioned these visualized results could help them in their conversations with other stakeholders to answer the question 'what value is this team delivering to its customers?'.

Table 14: Overview of the number of responses for the customer satisfaction surveys[7]

	Team A	Team B	Team C	Team D	Team E	P&S	Team G
Iteration 1	1	1	2	6	1	10	2
Iteration 2	1	2	x	2	2	x	2
Iteration 3	x	1	x	x	0	x	0

5.3.3 – Questions for the weekly calls with SMs

After 3 weeks into the action taking phase, this simple question of 'have you opened/used the dashboard this week?' would prove to be too one-dimensional. There is a difference in opening the dashboard and using the dashboard. A scrum master may open the dashboard, but may not see anything important to take action upon. Thus, after these first 3 weeks of the experiment, the questions were modified. Table 15 contains the questions and the reasoning on why it was deemed important enough to ask during these weekly informal talks with the scrum masters and what potential value the answers would provide.

Table 15: The questions asked during the weekly calls with the scrum masters and the reasoning on why it is asked

Question asked	Reasoning behind the question
Have you opened the dashboard and looked at it?	This question is aimed to understand whether or not they had the time or saw the merit in opening the dashboard in the first place.
Have you taken action based on the information displayed in the dashboard? If so, how?	If they opened the dashboard, an action could follow based on the information displayed. Since one of the goals of this action research is to study actionable metrics, this question fits this goal.
What are your thoughts on the information visualized in the dashboard this week? Anything that catches your eye or is interesting to look at upcoming week?	If they for example did not have time to open the dashboard or to take action, the weekly session would still prove useful by asking the scrum masters what their thoughts are on the information currently displayed. Will they take action now? What catches their eye?
What would you like to change (added/removed) from the dashboard?	While the dashboards would not deviate too much from the original design (unless the majority requests it), it is interesting to see what actionable or useful information is missing by asking them this question.
Have you used any KPIs or metrics that are not included in this dashboard I've developed for you (e.g. some metrics tracked by Azure DevOps that you use) this week? If so, which ones?	As previously mentioned, not all information could be correctly displayed in PowerBI. Thus, it is informative to ask whether the scrum masters used any other metrics or visualizations (for example in ADO) besides the PowerBI dashboard.
Could you give a rating for each visual/KPI in the dashboard on a scale from 1 to 5: (1) not at all useful, (2)	Half-way through the experiment, it was decided together with the client that it perhaps would be beneficial to ask a quantitative rating and see whether

---

[7] An 'x' indicates no survey was conducted.

Slightly useful, (3) Moderately useful, (4) Very useful, (5) Completely useful.	the answer to this question it changes or remains stable for the second half of the experiment. However, after several weeks, this question was removed because the scrum masters thought it was too annoying to rate all the metrics every week and their ratings mostly remained the same.
What are my own observations (during the retro, stand-up or during my talks with the scrum master)?	Sometimes, the researcher was present during the stand-ups and retrospectives of some Agile teams. This was done to facilitate and help the team in using the dashboard, interpreting the information correctly and to spur them in to using the dashboard more. For this question, the researcher's own observations could be answered here.

5.3.4 – Demo & feedback session

The action taking phase takes eleven weeks, during which the dashboard is used by the scrum masters and their respective teams. Each week, the scrum master is asked if and how they used the dashboard. However, these conversations are conducted individually on a weekly basis. This means the other scrum masters do not know how other scrum masters and teams are using their respective dashboard. They are oblivious as to what the overall findings, progress and feedback is from a researcher's point of view.

Thus, it was deemed important during this action taking phase to present the results so far in week 7 of the action taking phase and to gather feedback from the attendees. The invited attendees were the seven Scrum Masters, the seven product owners, various Agile coaches, stakeholders from PMO EMEA and some stakeholders from the leadership team.

This session would consist of a presentation to inform the attendees of what the learnings and findings were so far and which direction the research was heading. During this session, a short interactive Mentimeter session was conducted to gather the attendees' feedback and thoughts on the results so far and what potential steps could additionally be taken to enhance the value of the action research. These questions for the interactive Mentimeter session were formulated in collaboration with the company supervisor and the university supervisor of this thesis. Five questions were asked:

1. What do you like in the current set-up?
2. What are you missing?
3. Which metrics, KPIs or information would you like to see/show during the quarterly management review cycle?
4. Do you have any general thoughts and feedback based on what I've shared with you today?
5. On a scale from 1 to 10, how likely would you recommend this dashboard/metrics approach to your colleagues?

The questions 1 to 4 were open-ended, while question 5 was formulated as a Net Promoter Score-question; asking them how likely they would recommend this approach to their peers.

For each question, a summarization of the answers is given. Besides the summarization, a 'response' from a researcher point of view is given on why their feedback was (not) taken into consideration/implemented with appropriate reasoning.

Q1: What do you like in the current set-up?
While the question is worded in such a way that only positive answers would be given, it was surprising and encouraging to see thirteen answers to this question which were all very positive.

Q2: What are you missing?
Table 16 contains the answers to the second question asked during this interactive Mentimeter session, categorized in 4 categories.

Table 16: The answers submitted and their frequency during the interactive Mentimeter session

Category	Answer submitted	# of times submitted
Missing metrics	Value metrics/QMRC	3
	Lead time/cycle time	1
	Velocity: committed vs realized effort	1
Functionality	Possibility to change teams on the spot	2
	Drill-down functionality	1
Layout	Bigger font size	1
	Different dashboard for only actionable items	1
Adoption	Jira-conversion	1
	PMO support to build this dashboard for teams not in this experiment	1

Missing metrics
Cycle time is already present in some of the visuals in the dashboard. At the beginning of the action taking phase, lead time as a metric was deemed as not valuable enough to incorporate. During the development of the dashboard's first iteration, it was discovered that high lead times were mostly due to bad administration practices which increased the average significantly and did not reflect the reality. The most submitted answer would be the need for value metrics. These value metrics are not part of the team's dashboard, which is mainly focused on facilitating the team's success and performance. Instead, these value metrics are better suited for the quarterly management review dashboard (explained at the third question). Velocity is already present in the dashboard, so it is unclear why it was submitted as an answer.

Layout
Bigger font size has been implemented as feedback to make the dashboard more readable. Since the EMEA PMO Lead requested in the Reconnaissance phase that the dashboard for the team should be kept as one page for simplicity's sake, a different dashboard for actionable items was not created.

Functionality
Two stakeholders submitted the answer of that they would like to have the possibility to change teams on the spot. However, this is unfortunately not possible since each Agile team is using ADO in a different manner and thus each dashboard needs some level of customization in order to correctly display the data. Drill-down functionality has been partly implemented, by providing tooltips for some of the visualizations.

Adoption

In the organization, some teams in other regions use Jira to manage their projects as opposed to ADO. This action research is focused on testing a performance measurement framework and the data is sourced from ADO. While the same data could be sourced from JIRA, this is outside the scope of this action research. The last submitted answer is interesting; not only should a performance measurement framework in the form of a dashboard be tested, it should also be supported by someone after this action research has been completed. To increase the chance of success of other dashboards implemented in other Agile teams, it was imperative for a department such as PMO to support this endeavor.

Q3: Which metrics, KPIs or information would you like to see/show during the quarterly management review cycle?

Table 17: The answers submitted and their frequency during the interactive Mentimeter session

Category	Answer submitted	# of times submitted
Business value	Scope (change); what is completed and what is upcoming	5
	Business value delivered	3
	Customer satisfaction	3
	Business outcomes & achievements	2
	# of features developed	1
Financial metrics	Current budget, spent and left	3
	Return on Investment	2
	Budget forecast	2
	Team costs	1
	EBITDA	1
Product metrics	Cross-program roll-ups	1
	Adoption rate of the product	1
	Pending bug fixes	1
	Product growth	1
	Customer retention rate	1
Team-level performance metrics	Throughput	1
	Lead time	1
	High-level challenges team overcame	1
	Team satisfaction	1

This third question about the quarterly management review cycle requires some explanation. The main focus of this research is to understand and test what the challenges and benefits are of introducing a dashboard to Agile teams, in order to facilitate and help increase their efficiency, effectivity and ultimately success.

However, the EMEA PMO Lead was also interested in developing a dashboard template that the organization could use during quarterly management review cycles on a higher level. This dashboard template would not include for example velocity (since velocity is more of a team metric), but instead would include high-level information of what the team achieved past quarter and what the team is planning to achieve upcoming quarter.

Since the dashboard for the quarterly management review cycle would be used as a supportive tool, it cannot possible contain all the information that is discussed in this quarterly review. Thus, certain decisions need to be made on what is and is not suitable to show in such a dashboard.

Business value
As one can see from the number of answers submitted, business value is quite a popular category. Stakeholders are interested in seeing what has been completed on a high level (i.e. features) and what high-level work is upcoming for the upcoming quarter. This has been included in the dashboard. Customer satisfaction is also deemed as important, the customer determines whether value is delivered, and is thus important to quantify and include. Business outcomes are important, but not obtained from ADO, unfortunately. This should be then told by the scrum master/product owner when showing this dashboard, since it cannot be retrieved automatically.

Financial metrics
Similar to business value, financial metrics are deemed important to display by the stakeholders. Understanding how much is spent and how much is left is important so potential actions could be taken during these quarterly reviews (e.g. requesting more budget). Return on investment was mentioned twice, but this is very difficult to quantify since not all products or projects have a quantitative merit. Furthermore, it was determined in collaboration with the EMEA PMO Lead that EBITDA is not suitable to show in a visual dashboard.

Product metrics
While very relevant and applicable, the dashboard is limited to the data ADO (and perhaps a financial administrative system) is able to provide. Product/business metrics are important to keep track of, since they are a proxy of value delivered (e.g. adoption rate, new customers onboarded, etc.). While it has been included in the dashboard as an example/mock-up, the product owner or other stakeholders should supply these metrics so these can be visualized. Each team is interested in different product metrics and thus warrants a different approach and different visuals in each scenario.
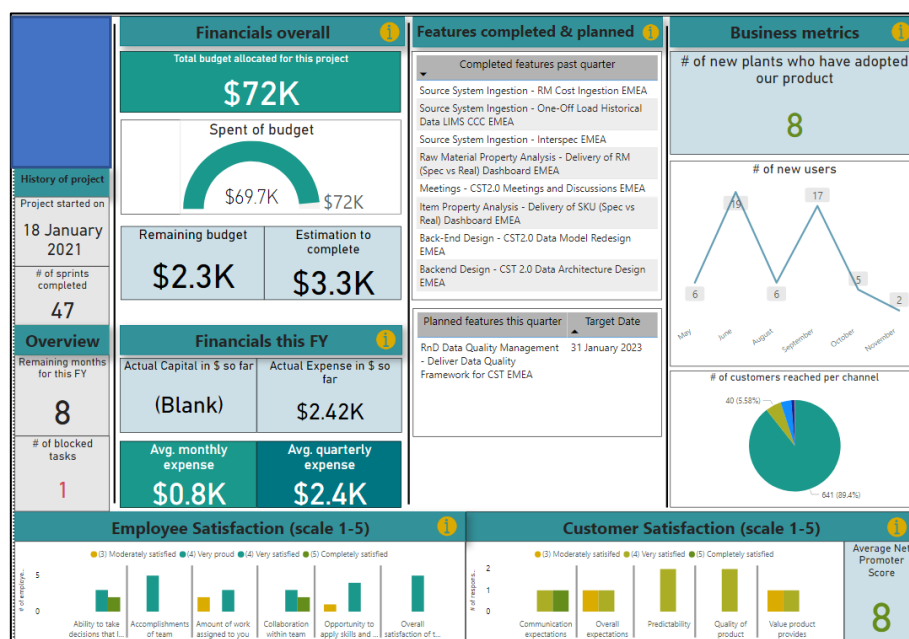
Figure 23: The dashboard template for the Quarterly Management Review Cycle

Team-level performance metrics
Since the team-level performance metrics such as throughput and lead time are too low-level, this would be unnecessary to display for such a high-level quarterly review. The team satisfaction is included in the dashboard, because it is deemed important to understand whether the team is satisfied or not for the longevity and health of the team. High-level challenges that have been solved is not something ADO provides, and would fit in a presentation instead of a dashboard.

Based on the feedback received on the third question and informal conversations with the EMEA PMO Lead, the dashboard template for the quarterly management review cycle was created, as shown in figure 23.

Q4: Do you have any general thoughts and feedback based on what I've shared with you today?

Table 18: The answers submitted and their frequency during the interactive Mentimeter session

Response	# of times
Thank you.	4
Ask Product Owners and business to see what information they would like to see as well in the QMRC dashboard.	1
Adaptation outside ADO	1
Provide more options for measurement to teams.	1
Verify what the possibilities are if it would only be shown in ADO.	1
Would like to see the literature review to see the purpose and benefits of each metric.	1

Q5: On a scale from 1 to 10, how likely would you recommend this dashboard/metrics approach to your colleagues?

Figure 24 shows the distribution of answers on this last question that was asked during the Mentimeter session. As one can see, the average is 7.5, which is quite positive!
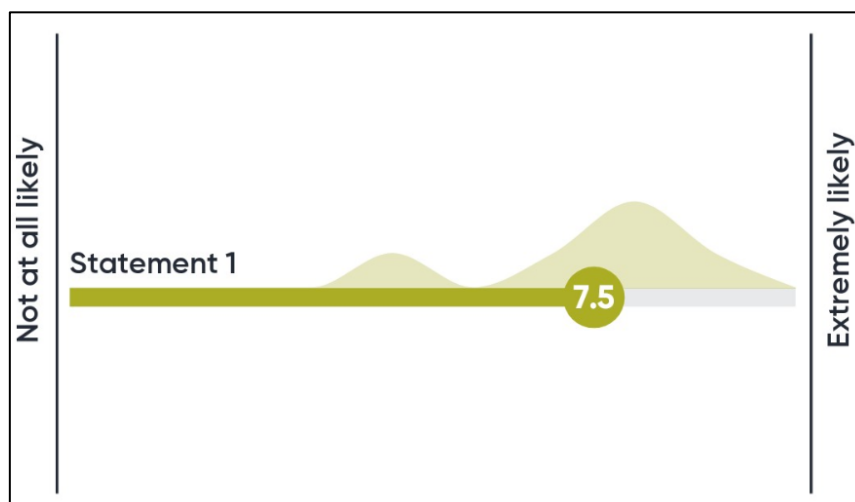
Figure 24: The distribution of answers for the fifth question

5.3.5 – Logbook of the experiment & dashboard use

During the experiment/action taking-phase, a logbook was maintained for each team, for each week. This logbook was established by taking notes during the each of the weekly calls with the seven scrum masters and based on the occasional participation in the teams' meetings such as stand-ups and retrospectives. During these weekly calls and meetings, notes were kept in an Excel file.

		Team A	Team B	Team C
Agile team maturity		Novice/Fluent	Fluent/Advanced	Advanced
Feedback week: 26 - 30 sept	Have you opened the dashboard and looked at it?	Yes, have opened the dashboard.	Yes, have opened the dashboard.	Yes, together during the bi-weekly retrospective.
	Have you taken action based on the information displayed in the dashboard? If so, how?	Yes, resolved the impediments that were shown in the dashboard.	Yes, used employee satisfaction in their Norway team-building and improvement talks. Since she was travelling, has not used it extensively.	No.
	What are your thoughts on the information visualised in the dashboard this week? Anything that catches your eye or is interesting to look at upcoming week?	It's weird to see that for the visual 'Workflow health: % of tasks completed', not all tasks have been completed in the previous sprints. However, after investigating, the scrum master discovered that PBIs were marked as complete, while some of the linked tasks to those PBIs were not yet set on complete; he will take a look at that.	She sees impediments; will look into that and see if they can be marked as complete. There is a slight increase in number of bugs this past month compared to previous months; but that is already known by the team and the SM.	Some impediments are not relevant anymore, will close them soon. Remaining work in hours is too much in detail and is too administrative to take action upon. The SM can't expect everyone to constantly update the remaining work hours for each task someone is working on. Trusting each other is more important and effective than tracking remaining work hours. '# of tasks without remaining work' is also quite excessive; the team member who works on those tasks understands how much time it takes and updates the team in the daily stand-up if necessary. It does not add extra value if you fill in for each task the remaining hours; already mentioned it's too excessive.
	What would you like to change (added/removed) from the dashboard?	Scrum master would like to see the Work Item ID of the unsolved bugs for quicker look-up in ADO. Would like to have shorter dates in the graphs (e.g. instead of September, it should be Sep.). Would like to replace the default '(Blank)' with 0 or N/A.	Since most bugs are linked to no feature, the graph does give a skewed view (since 100+ bugs have no feature linked it to it). Thus, would rather see in plain text, which top 3 features are responsible for the most bugs.	Remove 'Remaining hours' and '# of tasks without remaining hours'.
	Have you used any KPIs or metrics that are not included in this dashboard I've developed for you (e.g. some metrics tracked by Azure DevOps that you use) this week? If so, which ones?	No.	Yes, time to production (SLA), delivered value this sprint, planned vs. delivered effort.	His usual set of metrics, as discussed in the previous weeks.
	Could you give a rating for each visual/KPI in the dashboard on a scale from 1 to 5: (1) not at all useful, (2) Slightly useful, (3) Moderately useful, (4) Very useful, (5) Completely useful.	To be done.	Look in the mail.	x
	What are my own observations (during the retro, stand-up or during my talks with the scrum master)?	Team A is part of a larger team. They have some stand-ups throughout the week with this larger team. The SM of team A talked with and showed the dashboard to the SM of this larger encompassing team. He mentioned that the SM said this seems like micromanaging. SM of Team A is also very busy with his own daily work; SM does not want to overload the team members.	x	Team is helpful and willing to help me in my experiment, as is the SM. However, I can see the KPIs and visuals don't add that much value or insight to the SM/team on a daily or weekly basis. It's more related to administration of ADO than actual increase of performance.

Figure 25: Excerpt Excel file used for taking notes during weekly calls with scrum masters and occasional participatory observation during meetings of the team

Figure 25 is an excerpt from the Excel file with the notes kept throughout the period of eleven weeks per team. Based on this large Excel file, the logbook below has been written in a more readable format.

Team Team A (Agile maturity: Novice/Fluent)
August 29 – September 2
In the first week, the scrum master said he opened the dashboard and took action on the KPI '# of user stories without effort'. For the category 'Quality', he said it was interesting to see which features are responsible for the most bugs (visualization '# of bugs per feature').

September 5 – September 9
In the second week, the scrum master said he did not open the dashboard. The dashboard displayed the KPI # of open impediments/blockers'; he was not aware of this so he was helped by this and he said he would take action to resolve these impediments. The scrum master was reminded to assign efforts to user stories based on the KPI '# of user stories without effort', both to help the team and to ensure the velocity chart was displayed correctly.

September 12 – September 17
In the third week, the scrum master said he did not open the dashboard. He mentioned that the dashboard is not something that he would use every day, except for the category 'workflow health current sprint'. He said this category would be most interesting and suitable to take action on. The scrum master requested another metric: the amount of user stories completed compared to the amount of bugs opened. However, there was no room in the dashboard to add this metric.

September 19 – September 24
In the fourth week, the scrum master said did not open the dashboard. The reason for this, is that he

was too busy and did not have time. Last week, he mentioned he would take action to resolve the open impediments he was not aware of (KPI '# of open impediments'), but he said he is still trying to figure out what these impediments mean and how to solve them. The scrum master wanted some extra filters applied on visualizations, but this was not possible due to the limitations of exported data from ADO to PowerBI.

September 26 – September 30
In the fifth week, he said he had opened the dashboard and had taken action on it; he had resolved the impediments that were shown in the dashboard (KPI '# of impediments'). The visualization '% of tasks completed in the past 5 sprints' showed unsolved tasks in the previous 5 sprints, while the user stories had been completed. After further investigation, the scrum master discovered why; user stories are marked as complete while the linked tasks are still as in progress. He mentioned this should not be the case and that he would look into this to solve this administrative issue.

October 3 – October 7
In the sixth week, he said he had opened the dashboard and took action on it; he said he informed the team to mark work as complete if it is complete (i.e. a completed user stories should have all their linked tasks completed). This was due to the velocity chart; the velocity chart was not displaying information correctly because the team was not administrating data in ADO correctly; this triggered the scrum master to instruct the team members to pay attention to this. The scrum master mentioned that the visualization '# of bugs by severity' in category 'Quality' is useful to know, but not actionable for him as a scrum master.

October 10 – October 14
In the seventh week, no weekly call took place.

October 17 - October 21
In the eight week, the scrum master said he had opened the dashboard three times. He said he took action on it by completing some of the ongoing tasks of completed user stories in the past few sprints (visualization '% of Tasks completed past 5 sprints'). There were some unsolved bugs (KPI '# of unsolved bugs'), but he was aware of this and did not take any further action. Developers are still forgetting to assign effort to user stories (KPI '# of user stories without effort), which has an impact on the usability of the velocity chart.

October 24 – October 28
In the ninth week, the scrum master said he did not open the dashboard (only quickly before our weekly meeting). He said he had not taken any action on the information displayed. The reoccurring theme was the fact that the team members are still forgetting/struggling to assign effort to user stories (KPI '# of user stories without effort). For the category 'Quality', the visual 'Longest open standing bugs' is interesting; it helps him as a scrum master in case a bug takes too long to solve. The scrum master remarked that the category 'Workflow health current sprint' and the visual 'Longest open standing bugs' are the most actionable; it really helps him. Since the team is still struggling to mark tasks as complete when a user story has been completed, he requested to alter the visual from '% of tasks completed past 5 sprints' to '% of user stories completed past 5 sprints'.

October 31 – November 4
In the tenth week, the scrum master said during the weekly conversation he was unfortunately too busy to open the dashboard that week. When asked about his thoughts on the current state of the dashboard, he mentioned that there are some long open standing bugs (visual 'Longest open standing bugs') and some open user stories; but that's because they are waiting for a deployment.

The KPIs '# of user stories without effort' and ' # of tasks without remaining work' grabbed his attention, but after further investigation, no action was necessary.

November 7 – November 11
In the tenth week, the scrum master skipped the weekly conversation; he mentioned he was too busy and did not have time to open the dashboard. He also did not have time to attend our weekly conversation.

Team Team B (Agile maturity: Fluent/Advanced)
August 29 – September 2
In the first week, the scrum master said she did not open the dashboard. During this first weekly call, she remarked that it would be helpful to see not only planned and completed effort, but also committed effort in the Velocity graph. Committed user stories imply the development work is finished and the user story is now in testing with the QA-engineers.

September 5 – September 9
In the second week, the scrum master said she had not opened the dashboard and did not take any action on it. She mentioned the category 'Responsiveness' was not that useful for the team, since the team is constantly adding and removing user stories to the features. Thus, it does not make that much sense to show how many features have been released and how long it takes the team to release the features. The visual '# of user stories completed in the past 4 months' in the category 'Responsiveness' is more interesting than on a feature-level. The results of the employee satisfaction survey in the dashboard is interesting to know/see as a scrum master, she said.

September 12 – September 17
In the third week, the scrum master said she did not open the dashboard nor took action on it. She mentioned she is using different metrics that are (more) useful for her: business value delivered (small, medium, large) for each user story, velocity of the sprint and the whole increment, customer value (intuitively assigned to user stories) and SLA-level metrics (e.g. how long did it take to fix critical incidents).

September 19 – September 24
In the fourth week, the scrum master said she had opened the dashboard, but did not take any action on it. She mentioned the visual 'Longest open standing bugs' in the category 'Quality' is interesting; she said she would take a good look at it and take action if necessary.

September 26 – September 30
In the fifth week, the scrum master said she had opened the dashboard. She mentioned the team used the results of the employee satisfaction survey for their team-building and improvement talks. Due to travel, she said she did not have the time to take any action on the dashboard. The scrum master saw there are open impediments (KPI '# of open impediments); she said she would look into it to mark them as complete. The dashboard showed a slight increase in the number of bugs compared to previous months (graph '# of bugs created past 4 months), but this was already known by the scrum master.

October 3 – October 7
In the sixth week, the scrum master used the dashboard once, during one of the daily stand-ups where I was present as an intervention. She did not take any action on it. The scrum master gave her thoughts on the information displayed in the dashboard this week. She said that the workflow health of the current sprint is the most interesting in the beginning of the sprint. The 'Remaining work in hours' Is useful when you are dealing with hard deadlines; it can help to plan easier. 'Unassigned

tasks' is not that useful because in most cases we as a team do not know who will pick up those tasks yet. One of the team members mentioned that the 'Longest open standing bugs' are mainly bugs that have been ready for testing for a long time; it helps them identify if they need to take any further action to actually mark those bugs as complete. They mentioned someone should verify and clean up all the bugs in this 'Longest open standing bugs'. Unassigned bugs are usually low priority according to this developer; only critical ones are important to keep track of and assign.

October 10 – October 14
In the seventh week, the scrum master said she had opened the dashboard once on her own to get a management view of the team. She mentioned that the team is quite mature and experienced in reading information in ADO/backlog. Dashboard provides not that much value according to the scrum master. The dashboard is more suited for the scrum master to have a management overview of the team. Quality is only interesting if it applies to critical bugs; but the team is already aware of all critical bugs so it does not aid them that much with improving the quality of their product.

October 17 - October 21
In the eight week, the scrum master said she had opened the dashboard and took action; she said her team cleaned up the longest open standing bugs in order for the visual 'Longest open standing bugs' to display only relevant ones instead of the ones they already solved but forgot to mark as complete. She mentioned, similar to last time, that she cannot see a long-term use of the dashboard because of the team's maturity; it's not relevant on a daily basis.

October 24 – October 28
In the ninth week, the scrum master said she opened the dashboard once by herself that week. She said she took action on the dashboard; she closed bugs which had been open for a long time (visual 'Longest open standing bugs'). Scrum master mentioned that she is already active in the ADO backlog, so the dashboard does not help her that much. She mentioned the dashboard provides a good overview, but not for a daily use. She also saw the number of bugs increasing (visual '# of bugs created past 4 months), but that was explainable due to a different way of tracking bugs in ADO.

October 31 – November 4
The scrum master unfortunately did not have time to attend the weekly conversation that week.

November 7 – November 11
In the eleventh week, the scrum master said she did not have time to open the dashboard the past two weeks, due to a busy release schedule.

Team C (Agile maturity: Fluent/Advanced)
August 29 – September 2
In the first week, the scrum master said he did neither open nor use the dashboard. The scrum master did provide positive comments on the 'Responsiveness' category. The scrum master advised to make a distinction between the categories 'workflow health of the current sprint' and the 'workflow health of previous sprints'. He also advised to add tooltips so the dashboard becomes more actionable for the user.

September 5 – September 9
In the second week, no conversation was conducted.

September 12 – September 17
In the third week, the scrum master said he did neither open nor use the dashboard. He mentioned

during the weekly conversation that the employee and customer satisfaction are the most important parts of the dashboard for him as a scrum master. The scrum master is already using his own 'health metrics' similar to the metrics in the 'Workflow health current sprint' category. He mentioned the Velocity graph in 'Workflow health' was useful, but due to the limitations of the exported data, he prefers using ADO to visualize the velocity of the team. The metrics this scrum master is using are: throughput, velocity, cycle time of user stories, burn/down charts and general health metrics (e.g. user stories with effort/description). Those health metrics are more relevant to take action on daily. The other more historic visualizations are more intended for monthly or quarterly reports.

September 19 – September 24
In the fourth week, the scrum master said he did not open nor use the dashboard. It was noticed by the researcher that the scrum master is apprehensive to use the dashboard since he already has metrics which he and the team are familiar with.

September 26 – September 30
In the fifth week, it was observed the scrum master opened the dashboard in the retrospective, but did not take any action. Dashboard showed impediments (KPI '# of open impediments'), but they are not relevant anymore. The scrum master mentioned the health metrics in the category 'Workflow health current sprint' were too low-level/micromanaging according to the scrum master; the team does not use the field 'Remaining work in hours' for user stories since that is too much administrative work (KPI '# of user stories without remaining work). Trust for him is more important and effective than tracking hours. The health metric '# of tasks without remaining work' is also quite excessive according to the scrum master; the assigned team member understands and knows how much remaining work there is and updates the team on his tasks during the daily stand-up.

October 3 – October 7
In the sixth week, the scrum master said he did not open nor use the dashboard. When asked, he mentioned that the low Net Promoter Score could be used to facilitate the discussion with the Product Owner. Since the scrum master is already quite familiar with using metrics in ADO himself, he again stressed that employee and customer satisfaction are the most important metrics since those cannot be visualized in ADO.

October 10 – October 14
In the seventh week, the scrum master was observed opening the dashboard during the retrospective, but did not take action on it. When asked, the scrum master mentioned the velocity chart is useful and the visual '% of Tasks completed in the past 4 sprints' in the category 'Workflow health' are also useful from an administrative point of view. Visuals in the category 'Responsiveness' showed an increase in cycle time; it sparked a discussion with the team on what caused this but no action was taken.

October 17 - October 21
In the eight week, the scrum master said he had opened the dashboard but did not take any action since he thought it was unnecessary. There were blocked tasks/impediments shown in the dashboard (KPI '# of blocked tasks/impediments'), but the team was already aware of them.

October 24 – October 28
In the ninth week, the scrum master was observed opening the dashboard during the retrospective, but did not take any action. There were some blocked tasks (KPI '# of blocked tasks), but like last week, the team was aware of it. Scrum master mentioned he used the customer satisfaction survey results in the dashboard to structure his discussion with the product owner. Due to the cooldown period of the project, the health metrics of the current sprint were less relevant.

October 31 – November 4
The scrum master unfortunately did not have time to attend the weekly conversation.

November 7 – November 11
In the eleventh week, the scrum master said he did not open the dashboard. The project is soon migrating to another ADO environment. The KPIs '# of blocked tasks' and '# of unassigned tasks' were either already known or did not warrant any action. Increase in cycle time in the category 'Responsiveness', due to the administrative work of closing old user stories and features.

Team Team D (Agile maturity: Advanced)[8]
August 29 – September 2
In the first week, the scrum master said he neither opened nor used the dashboard. He mentioned the team is barely working on this project, so the category 'Responsiveness' is not really informative. The category 'Workflow health current sprint' was also not relevant he said, since the amount of work performed each sprint is minimal.

September 5 – September 9
In the second week, the scrum master said he had opened the dashboard, but did not take any action on the displayed information. He mentioned the team is in maintenance mode, so possible actions for the team are sparse. The category 'Employee satisfaction' is interesting to know as a scrum master, even though the employees are working part-time on this product. The scrum master saw there were some unassigned tasks in the current sprint (KPI '# of unassigned tasks'); he said he would look into this and assign those tasks.

September 12 – September 17
In the third week, the scrum master said he neither opened nor used the dashboard. Employee and customer satisfaction is informative, but not that actionable for the scrum master.

September 19 – September 24
In the fourth week, the scrum master said he had opened the dashboard, but did not take any action on it because he did not deem it necessary to do so.

September 26 – October 7
In the fifth and sixth week, no conversations were held.

October 10 – October 14
In the seventh week, the scrum master said he had opened the dashboard, but did not take any action on it. One task was still in progress even though the sprint end date was nearing; but the team was waiting for confirmation from the product owner before the task could be marked as complete.

October 17 - October 21
In the eighth week, the scrum master said he neither opened nor used the dashboard.

October 24 – November 11
From the ninth till the eleventh week, the scrum master said he had not opened the dashboard at all.

---

[8] Team D performed very little work during the experiment phase; the Team D project is almost complete so it may not have been the best choice to include as a team for the experiment.

Team Team E (Agile maturity: Beginner)[9]

August 29 – September 2
In the first week, the scrum master said she neither opened nor used the dashboard.

September 5 – September 9
In the second week, she said she opened the dashboard, but did not take any action because the project is currently on-hold. She mentioned the category 'Responsiveness' would be very useful when the work on the project resumes, in addition to the category 'workflow health current sprint' considering the low Agile maturity level of the team. The visual '% of tasks completed past 4 sprints' under the category 'Workflow health' would be also helpful for her as a scrum master.

September 12 – September 17
In the third week, the scrum master said she opened the dashboard, but did not take any action on it, since the project is on-hold.

September 19 – September 24
In the fourth week, no conversation was conducted.

September 26 – September 30
In the fifth week, the scrum master said the dashboard was not opened nor was it used. Scrum master mentioned that this week the project would resume and she was excited to use the dashboard from now on. She is also using metrics in ADO: the burndown and burnup chart. Additionally, considering the team's low Agile maturity, she is also using a custom made chart in ADO called 'Work assigned per person' (i.e. how many features, user stories, tasks, bugs are assigned to each team member in this sprint).

October 3 – October 7
In the sixth week, scrum master said she opened the dashboard, but had not taken any action on it. She mentioned again that the health metrics of 'workflow health current sprint' will help the team considering their low Agile maturity.

October 10 – October 14
In the seventh week, the scrum master said the dashboard was opened during the stand-up and outside the stand-up once. The dashboard showed that some user stories did not have any effort assigned (KPI '# of user stories without effort); she would take action on this.

October 17 - October 21
In the eighth week, the scrum master said she had opened the dashboard. She was using the 'Responsiveness' visuals to better monitor the backlog; to know whether certain epics, features or user stories had been open for a long time and if they should be closed. She also used the visual '% of PBIs completed in the past 4 sprints'; based on this, she completed open-standing user stories from previous sprints. She remarked that the velocity chart was not displaying the velocity correctly; perhaps due to the limitations of exported data to PowerBI.

October 24 – October 28
In the ninth week, the scrum master mentioned she had opened the dashboard and took action on it. She used the visuals in the category 'Responsiveness' to understand how the backlog had been managed and if she needed to close any user stories or features which had been open for a long

---

[9] Team E experienced a freeze in the summer months due to multiple problems. This had an impact in the first few weeks; the scrum master could not leverage that much action.

time. The KPI '# of user stories without effort' was quickly looked at, but she mentioned that specific user story is on-hold and thus no action is needed.

October 31 – November 4
In the tenth week, I participated in the stand-up and showed the dashboard to the scrum master and the team as in intervention. There was no action necessary. The scrum master mentioned that the refresh rate of the dashboard should be increased to see the effect of taking action quicker. The KPI '# of tasks without remaining work' was actionable this week, she said.

November 7 – November 11
In the eleventh week, the scrum master said she unfortunately did not have time to open the dashboard.

Team Team F (Agile maturity: Novice/Fluent)
August 29 – September 2
In the first week, the scrum master said he did not open nor use the dashboard.

September 5 – September 9
In the second week, the scrum master said he neither opened nor used the dashboard. He mentioned during the weekly conversation that the category 'Employee satisfaction' is interesting/informative to see. He requested to add the number of team members to the velocity chart, since the resources tend to fluctuate. He mentioned this could have an impact on the velocity.

September 12 – September 30
In the third, fourth and fifth week, the scrum master said he neither opened nor used the dashboard.

October 3 – October 7
In the sixth week, the scrum master said he had opened the dashboard once, but did not take any specific action on it. He mentioned that the category 'Workflow health current sprint' health metrics are useful/actionable, especially considering the team's (Agile) maturity. He noticed there is a decrease in planned effort in the velocity chart due to reduced resources, which was helpful for him to visualize. He requested to change the scope from feature to epic in one of the 'Responsiveness' visuals, since he said they are using ADO differently to categorize features. Furthermore, he mentioned he had used the standard burndown chart in ADO this sixth week as a metric besides the dashboard.

October 10 – October 14
In the seventh week, I showed the dashboard both during the stand-up and the retrospective. This intervention was made, because the opening and use of the dashboard for this team was deemed too infrequent to leverage value for this research. Based on the dashboard, I heard the scrum master mention to the team that the team is grossly overestimating the work for this sprint (Velocity chart); he instructed the team to estimate in a more realistic manner. The dashboard also showed some unassigned tasks (KPI '# of unassigned tasks'); these were assigned during the stand-up by the scrum master while the dashboard was shown. The scrum master also saw that there were several tasks without remaining work (category 'Workflow health current sprint'). However, this was not a problem since those tasks still were in the proposed state. He said he would keep an eye on the team's tendency to overestimate the effort.

October 17 - October 21
In the eight week, the scrum master said he did not open the dashboard. To gather enough feedback for that week, I was present in the team's stand-up the day after this conversation to showcase the dashboard. During this stand-up, the scrum master again mentioned the team is overcommitting; there is a large increase in planned effort compared to previous sprints (velocity chart). The scrum master also saw some unassigned tasks in the dashboard (KPI '# of unassigned tasks), but after discussing this with the team, it was decided that it is not necessary to assign those specific tasks.

October 24 – October 28
In the ninth week, he mentioned he had opened the dashboard. He said he told the team not to overcommit due to the high planned effort, as could be seen in the velocity chart. He also mentioned there were some tasks without remaining work (KPI '# of tasks without remaining work), he took action on this health metric. He requested to exclude retrospective-related tasks from the health metrics in the category 'Workflow health current sprint'.

October 31 – November 4
In the tenth week, the scrum master said he had opened the dashboard and reflected on how the team is estimating their work in this sprint. The KPI '# of tasks without remaining work' was looked at during this weekly conversation, but in this case it was already known and no action was needed. He mentioned the customer satisfaction survey results would be really useful to engage with the product owner and customers.

November 7 – November 11
In the eleventh week, the scrum master said he had opened the dashboard once, but did not take any further action. This scrum master announced he is leaving the company soon.

Team Team G (Agile maturity: Fluent)
August 29 – September 2
In the first week, the scrum master said he opened the dashboard. He also mentioned the took action on the dashboard; the visual 'Longest open standing bugs' of the category 'Quality' was very useful. He notified the developers in the team and asked them to look into these so they could potentially mark the bugs as complete. The scrum master mentioned to me that it is interesting to see the difference between 'Estimated hours' and 'Actual spent hours' of each sprint in the category 'Workflow health'; it could uncover a trend for him that indicated whether the team is over- or under-estimating the work. The employee satisfaction is also helpful/useful for him as a scrum master. He mentioned the category 'Responsiveness' is less relevant for the features, since there are not that many releases. However, he mentioned the user story-level of 'Responsiveness' is very interesting to see; the amount of user stories per month completed with their average cycle time. He said it's a positive sign for him as a scrum master to see the amount of completed user stories increasing per month, while the average cycle time is decreasing. He requested a burndown chart to be added to the dashboard, but that was not possible. The scrum master seemed very positive and genuine about the usefulness and applicability of the dashboard in his team. This team seemed the most enthusiastic in general about participating in this experiment and using the dashboard.

September 5 – September 9
In the second week, the scrum master told me he had opened the dashboard and looked at it. After that, he mentioned he did not have time to take action based on the dashboard. However, he did tell that the developers sent him a message that some of the longest open standing bugs could be closed, based on the information in the 'Longest open standing bug'-visual. The scrum master

mentioned that because of the team's high Agile maturity, the category 'workflow health of current sprint' is not really relevant/useful.

September 12 – September 17
In the third week, the scrum master told me he opened the dashboard and looked at it. He showed the dashboard to the team during one of their weekly meetings; he said the team was positive about the information and the visual aspect. The scrum master said he took action mainly on the category 'Quality': he assigned all non-assigned bugs to team members (KPI '# of unassigned open bugs') and solved some bugs which should have been done sooner (KPI '# of open bugs'). The scrum master gave his thoughts on the dashboard. The category 'Workflow health' is useful to understand whether the team is over- or under-estimating the work. The fact that the velocity and difference in planned and completed hours are shown historically (past 4 sprints) helps in planning for the next sprint. The category 'Employee satisfaction' is nice to have/show, but the team is quite close and tight-knit, so they already know these answers implicitly. The category 'Customer satisfaction' is also interesting and will be especially useful as the product matures; then it's interesting to see if the customer satisfaction increases or decreases.

September 19 – September 24
In the fourth week, the scrum master said he opened and looked at the dashboard. To observe how the dashboard was used, I was present during one of the team's weekly meetings. Based on the 'Quality' category, it sparked a discussion within the team. They discussed if the bugs shown in visual 'Longest open standing bugs' could be closed and some actually were closed based on this discussion. One of the team members mentioned that the visual 'Severity of bugs' is not that actionable, because the team only assigned the severity level 'Medium' to all bugs. This sparked a conversation about how the team could incorporate more severity levels when creating bugs in ADO. The graph '# of features completed past year' in the category 'Responsiveness' had some changes compared to last time; the scrum master said during this weekly meeting that it's mainly the result of cleaning up the ADO backlog. Thus, it does not represent the actual number of features completed and the actual time it took to complete them. Scrum master requested to have the field 'Work Item ID' added to the tooltips so he could easily search ADO for the relevant items.

September 26 – September 30
In the fifth week, the scrum master said he had opened the dashboard, but did not take any action on it. He mentioned they have a new way working in ADO, which has an impact on how the velocity is shown. Additionally, a key team member was sick, so that's why the categories 'Workflow health' and 'Workflow health current sprint' were quite similar to last week. The scrum master saw there were some tasks without remaining work (KPI '# of tasks without remaining work'); he said he would ask the team members about this KPI. He again mentioned the 'Responsiveness' category is not that actionable. He requested an additional KPI in the category 'Workflow health current sprint': the number of blocked tasks & bugs. The scrum master made an interesting remark in the weekly conversation; he suggested the dashboard be split into two; one focused on very actionable metrics/KPIs and the other provides a more historical and less actionable view of the team. Because not all KPIs or visualizations change with the same frequency.

October 3 – October 7
In the sixth week, I was present during their weekly team meeting and showcased the dashboard there as an intervention. The scrum master mentioned this weekly meeting is the perfect moment to open the dashboard and decide if any action is necessary. During this meeting, the scrum master focused on the category 'Quality'; a discussion ensued because of visual 'Longest open standing bugs'. The scrum master looked at those bugs in ADO and decided with the team whether they could

mark them as complete. The scrum master also looked at the team's velocity in combination with the current sprint's end date; he said to the team there were a few days remaining in the sprint, but a lot of work still has to be done. He advised and asked them to mark their assigned user stories as complete if they were to ensure he had a representative view of this sprint's progress in the dashboard. It was also observed that the scrum master addressed a team member because this team member's tasks did not have remaining work (KPI '# of tasks without remaining work). The team looked at the blocked tasks that were shown in the dashboard, but no action was necessary. The scrum master mentioned that the number of bugs is decreasing since the development of the product is less intensive compared to previous months (visual '# of bugs created past 4 months'). The scrum master mentioned during the weekly conversation that the difference in estimated and planned hours on the task-level is too low-level for him; he would rather use the velocity chart on the user story level. The scrum master also mentioned he was expecting to use the visualization '# of bugs per feature' more when he started to use the dashboard; it is less actionable than expected. The scrum master requested another KPI for the category 'Workflow health current sprint': the number of user stories without a description. The scrum master mentioned that not all information is relevant all the time; it depends on the phase of the project and the needs of the team.

October 10 – October 14
In the seventh week, the scrum master mentioned he opened the dashboard twice; once during the weekly meeting on Monday and once on Wednesday. When asked about actions taken, he responded it was not necessary on Monday. Wednesday he did take action by assigning story points to user stories (KPI '# of user stories without story points'). He mentioned during the weekly conversation that there was one long open standing bug which was dependent on other stakeholders in order to solve it. Scrum master mentioned he and the team looked at the category 'Workflow health of the current sprint', but there were no actions necessary. He also mentioned the team and the scrum master looked at blocked tasks (KPI '# of blocked tasks'); they are aware of it but could not remove them. The scrum master saw the team improved in planning their work; the difference between estimated hours and spent hours got smaller in the recent sprints. Scrum master said he told the team members to complete their user stories if possible, since the velocity chart was displaying no completed work. The scrum master requested me to alter the velocity chart; the velocity chart should only show the completed effort, instead of planned and completed. Planned effort only makes sense to show for the current sprint, he said. Scrum master mentioned again the ideal frequency of the dashboard would be 1 or 2 times per week.

October 17 - October 21
In the eighth week, the scrum master said he had used the dashboard during the weekly team meeting, where I was present as well. There was no action taken, because the scrum master said there was nothing that warranted immediate action. During the weekly team meeting, the scrum master focused on the health metrics of the category 'Workflow health current sprint', especially the KPIs 'User stories without story points' and 'Blocked tasks'. The scrum master requested to have more KPIs added, which would be '# of user stories completed this sprint'.

October 24 – October 28
In the ninth week, the scrum master said he opened the dashboard twice. Once during the weekly team meeting and again later that week. He took action on the dashboard by assigning a different severity level than the usual 'Medium' one, based on the visual 'Severity of all bugs'. He mentioned that the category 'Quality' was the most interesting/actionable part of the dashboard. He had a few requests on what should be changed in the dashboard. The team is now working with effort instead of story points; he requested the dashboard also accommodate this change in their way of working.

He requested two extra KPI's: '# of tasks without original estimate' and '# of remaining user stories for current sprint'.

October 31 – November 4
In the tenth week, the scrum master said he had opened the dashboard, but he did see something that warranted action.

November 7 – November 11
In the eleventh week, the scrum master said he unfortunately did not have time to open the dashboard that week. When asked about his thoughts on the information displayed currently, he mentioned the KPI '# of blocked tasks' are new, but already known. He also mentioned that the team is now using different levels of severity to assign to bugs, based on the fact the dashboard showed that the team was assigned the severity 'medium' to all bugs.

5.3.6 – Documentation of team & quarterly management review dashboards

The EMEA PMO Lead mentioned he would like to have extensive documentation of how the dashboards were created and maintained, in order to continue this project after this specific research ends (EMEA PMO Lead, personal communication, November 2022). To ensure the continuation and success of this research project, two extensive documentation PowerPoint-files were created, detailing for both the team and quarterly management review dashboards:

- Explaining the dashboard: for each metric or visualization, the following questions are answered:
    - What does it show?
    - How is it calculated?
    - What action could be taken?
- How to create 'Analytics Views' in ADO and how to connect PowerBI to these views.
- How to transform the data in PowerBI (ETL-process).
- Modifying, customizing the dashboard & explaining the functionality such al tooltips.

5.4 – Evaluating

This fourth and final phase of the action research framework is called 'Reflection' (Kemmis et al., 2014); however other authors such as Susman & Evered (1978) also call this phase 'Evaluation' or 'Evaluating'. Nevertheless, this phase is dedicated to analyze, synthesize, interpret and to draw conclusions (Kemmis et al., 2014). This reflection is done twofold: a) by conducting exit interviews with the seven scrum masters and b) reflecting on and evaluating the results of the research.

5.4.1 – Exit interviews with the scrum masters

Almost every week, the scrum masters were asked in an informal weekly conversation if and how they used the dashboard. But as this weekly inquiry had come to an end, it was deemed beneficial to evaluate their overall experience as a participant in the research. After all, they were the primary stakeholders during the research. To understand their experience as a participant of those eleven weeks and in what capacity their expectations have been met, several questions were drafted together with the client. These questions are shown in table 19.

Category	Question
Experiment overall	What was your experience as part of the team in taking part in this small experiment?
	What was your experience as a scrum master in this experiment?
Dashboard	Do you feel the dashboard was beneficial to use during these past ~11 weeks?
	How (often) do you expect to continue using the dashboard after the experiment ends?
	The dashboard is split up into multiple categories; do you feel like one or more categories should be removed? If so, why?
	Would you still use the dashboard as is and take ownership of maintaining after the experiment? (show documentation before these questions)
	Would you prefer continue using this dashboard in PowerBI or in ADO?
Employee satisfaction surveys	Your team used X tool to assess the employee satisfaction periodically; what did you think of using this tool to assess their satisfaction?
	What would be an ideal frequency to assess the employee satisfaction?
	How (often) do you expect to continue assessing the employee satisfaction after the experiment ends? Will you be using the same tool/method?
	What is your opinion on the questions asked in this employee satisfaction survey (both the number of questions and the questions themselves)?
Customer satisfaction surveys	The customer satisfaction was assessed with Microsoft Forms. What is your opinion on using this tool to assess this?
	How (often) do you expect to continue assessing the customer satisfaction after the experiment ends? Will you be using the same tool/method?
	What would be an ideal frequency to assess the customer satisfaction?
	What is your opinion on the questions asked in this customer satisfaction survey; both the number of questions and the questions themselves?
Closing questions	On a scale from 1 to 10, would you recommend the process of using these metrics, dashboard to measure and grow? What did you like and what would you improve?
	On a scale from 1 to 10, would you recommend participating in this experiment to other Agile teams within the organization?

Experiment overall
What was your experience as part of the team in taking part in this small experiment?
All seven scrum masters responded positively to this question. Everyone said it was a positive experience overall, with only the scrum master of Team D mentioning that he "was not able to leverage a lot out of the process", due to the stage of the project and the low amount of work involved. It added value; four out of seven scrum masters said the experiment with the dashboard provided them with a more holistic view compared to the routine work in ADO.

What was your experience as a scrum master in this experiment?
Their answers on this question were mostly the same as the previous one; very positive overall. Team D's scrum master noted that the workload to use and integrate this dashboard is not too heavy, it was fairly balanced. The scrum master of Team F noted that this is what everyone should do, since the Scrum methodology is empirical and the best way to work in line with this philosophy is to use data.

Dashboard overall
Do you feel the dashboard was beneficial to use during these past ~11 weeks?
All seven scrum masters answered very positively; they thought it was beneficial. The scrum master of Team A said that every time he opened the dashboard, he received value. The only exception would be the scrum master of Team D; he said it was not that beneficial for him as a scrum master, but he also said that may be because there isn't that much work being performed at the moment. The scrum masters of Team C and Team F said the employee and customer satisfaction were beneficial to include, alongside the ADO data.

How (often) do you expect to continue using the dashboard after the experiment ends?
For this question, the answers were quite similar as well. Except for Team D, the scrum masters said they would open it at least once every sprint. The scrum masters of Team C and Team E said they would open the dashboard themselves once a week, but every two weeks with the whole team. The scrum masters mentioned they would probably open and use the dashboard in the beginning or/and at the end of the sprint (i.e. retrospective).

The dashboard is split up into multiple categories; do you feel like one or more categories should be removed? If so, why?
Not every category was as useful or actionable for each scrum master during the experiment. The only scrum masters who answered with the fact that some categories could potentially be removed, are the scrum masters of Team E and Team G. They both said that they (should implicitly) know the employee and customer satisfaction, so those two categories were less interesting, useful and actionable for them than initially expected. The scrum master of Team A said that if he had to remove categories out of necessity, he would choose the employee and customer satisfaction.

Would you still use the dashboard as is and take ownership of maintaining after the experiment?
The main factor that determined whether or not the scrum masters were willing to take ownership of the dashboard, was their (perceived) PowerBI skills and knowledge. The scrum masters of Team G, Team F and Team B have sufficient PowerBI experience to take ownership of the dashboard maintenance. The scrum master of Team A said "the one responsible should need to have PowerBI expertise and knowledge". The scrum master of Team D said it would be too much work for him to maintain it. The scrum masters of Team C and Team E said it would be good if someone from the Project Management Office would be responsible for the maintenance.

Would you prefer continue using this dashboard in PowerBI or in ADO?
Each scrum master recognized that both PowerBI and ADO have their merits and demerits. The argument for using ADO is the fact that the data is already connected and collected in the same place; it's easier to navigate through and see the connection to the backlog. PowerBI has more user interface capabilities and provides the option to connect external data sources. PowerBI also offers the advantage to share the link of the dashboard with external stakeholders without granting them access to the ADO environment. The simple but most complex answer to this question would be: it depends.

Employee satisfaction surveys
Your team used X tool to assess the employee satisfaction periodically; what did you think of using this tool to assess their satisfaction?
Three teams used Mentimeter to assess the employee satisfaction every 1-2 sprints. The three scrum masters of these three teams (Team B, Team F, Team C) liked the tool because it was interactive and they are able to explain answers if needed. Four teams used Microsoft Forms to

assess the employee satisfaction every 2-3 sprints. The four scrum masters of these four teams (Team A, Team D, Team E, Team G) had differing answers. The scrum master of Team A wanted to use Mentimeter during the retrospective instead of Microsoft Forms, since the team members are then already present in the meeting and it ensures everyone gives their opinion. The scrum master of Team E had a similar statement about ideally using an interactive tool. The scrum master of Team D was okay with Microsoft Forms. The scrum master of Team G said Microsoft Forms is fine, but he already assesses implicitly the employee satisfaction during the retrospective, so he views it as unnecessary.

What would be an ideal frequency to assess the employee satisfaction?
The scrum masters of teams Team A, Team D, Team E and Team F said that once a month should be the ideal frequency to assess the employee satisfaction. The scrum masters of teams Team G, Team B and Team C said once a quarter or once every 6 months is the ideal frequency.

How (often) do you expect to continue assessing the employee satisfaction after the experiment ends? Will you be using the same tool/method?
With regards to the frequency, the scrum masters provided similar answers compared to the previous question. However, the scrum masters of the teams who used Microsoft Forms to assess the employee satisfaction said they would be using an interactive approach (Mentimeter/Mural) if they were to continue assessing the team's satisfaction.

What is your opinion on the questions asked in this employee satisfaction survey (both the number of questions and the questions themselves)?
All the scrum masters mentioned that six questions in this survey are adequate, except for the scrum master of Team E; she said the questions seem too similar and could be reduced to 3 or 4 key questions with the option for more user input. The scrum masters of Team F and Team G mentioned that question 5 ('How satisfied are you with the amount of work that was assigned to you?') should be phrased differently, since an Agile Scrum environment promotes the concept of allowing team members to pick up work themselves. The sixth question ('How would you rate the overall satisfaction of your team?') caused some confusion for the scrum masters of Team A, Team E and Team B. That question can be interpreted two ways; whether it is the employee's perception of the team's satisfaction or how satisfied that employee is in the team.

Customer satisfaction surveys
The customer satisfaction was assessed with Microsoft Forms. What is your opinion on using this tool to assess this?
Six out of the seven scrum masters answered that Microsoft Forms is the most appropriate tool to assess the customer satisfaction. Out of these six, one scrum master noted that with this tool, "they can reply whenever they have time". The only scrum master who preferred using a more interactive approach was the Team F scrum master; he said he prefers an interactive session where they are able to ask for clarifications about the questions asked in this survey.

How (often) do you expect to continue assessing the customer satisfaction after the experiment ends? Will you be using the same tool/method?
Five out of seven scrum masters expect to assess the customer satisfaction on a quarterly basis with Microsoft Forms. The first exception would be the scrum master of Team F; he said he would not mind doing it every one to two sprints. The second exception is the scrum master of Team B, she said she would love to do it twice a year if she had the power to do so. While the scrum masters of Team D and Team E said they expect to assess it every quarter, they noted that the release

frequency of significant features would also play a role in how often they would assess the customer satisfaction.

What would be an ideal frequency to assess the customer satisfaction?
The scrum masters mainly provided the same answer as in the previous question. The only exception was the scrum master of Team A; he said ideally it should be done once a month because of the product's release schedule. However, due to the fact that it takes time and effort to assess and follow-up on the answers, he expects to do it every quarter.

What is your opinion on the questions asked in this customer satisfaction survey; both the number of questions and the questions themselves?
Most of the answers to this question were focused on the survey's first question: 'On a scale from 1 to 10, how likely would you recommend this development team to your peers?'. The scrum masters of Team B, Team D, Team E and Finance mentioned that the customers/end-users are not really that engaged with the development team. Most of the customers only see the product and the product owner, instead of the development team. Thus, it really depends on the audience whether to include this first question or not. The fourth question ('How satisfied are you in having your overall expectations met?') also resulted in some comments from the scrum masters. The scrum master of Team B said the question was not specific enough; she was not sure she should be evaluating a recent addition to the product or the product in its entirety. The scrum masters of Team D and Team E said that for the sixth question ('How satisfied are you in having your expectations met with regards to predictability?'), it is not always applicable to ask because the releases are more of a notifying push towards the customers than an agreed upon pull from the customer.

Closing questions
On a scale from 1 to 10, would you recommend the process of using these metrics, dashboard to measure and grow? What did you like and what would you improve?
All scrum masters answered quite positively; the scrum masters of Team G, Team E, Team D and Team B answered with an 8. The scrum master of Team F provided a 10, Team A's scrum master a 7.5 and Team C's scrum master an 8.5. The scrum master of Team D said "before using metrics and dashboards, it is important to consider the product's stage and workload in order to maximize value". The scrum master of Team F who answered with a 10, said Scrum is empirical and the only way to know what has happened is with data. The scrum master of Team A said the "dashboard helps with the efficiency of the team".

On a scale from 1 to 10, would you recommend participating in this experiment to other Agile teams within the organization?
The answers to this question were also very positive. The scrum masters of Team A, Team C and Team E answered with an 8. The scrum masters of Team D and Team G answered with a 9; the scrum master of Team B with a 7 and the scrum master of Team F with a 10.

5.4.2 – Main findings of the experiment
The second activity in this Evaluating-phase is to analyze the results of the experiment's eleven weeks with the participating seven Agile teams to understand and arrive at the main findings and learnings. The main findings are categorized into the following categories, as shown in figure 26.
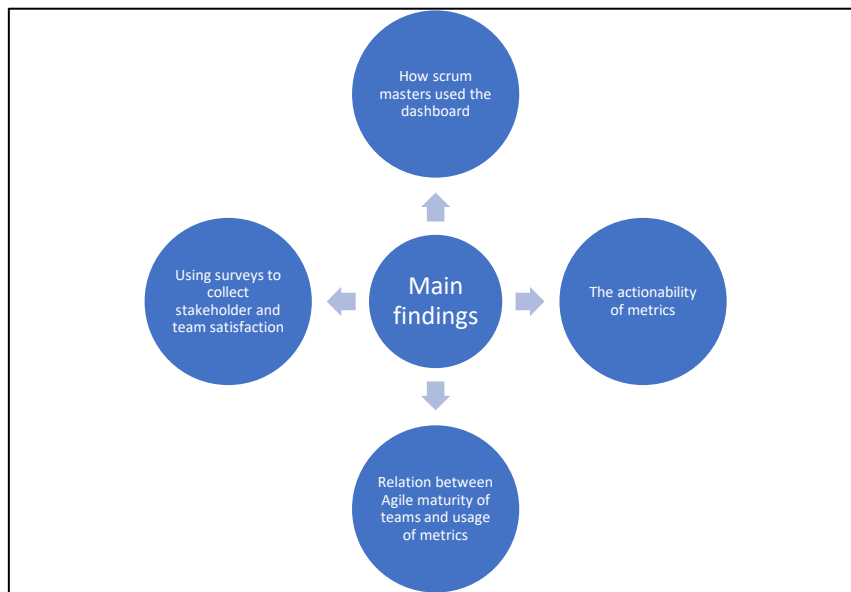
Figure 26: Categories of the main findings

Category A: How the scrum masters used the dashboard

A-1: Ideal frequency of opening dashboard is one to two times per week

Initially, it was expected the dashboard would be opened (almost) every day, for example during the daily stand-up to see if there are any actions necessary by the scrum master or the team. However, during the experiment, not one of the seven teams used the dashboard with this expected frequency. The scrum master of Team B said in the eighth week that she does not see how the dashboard could be used every day; it's not that relevant on a day-to-day basis. The scrum master of Team A said something similar in week 3; the dashboard is not something he would use very day (except maybe for the category 'workflow health current sprint'). The scrum master of Team G said in the sixth and seventh week that the ideal frequency of opening the dashboard would be once to twice a week. The exit interviews confirmed this; the scrum masters expect to use the dashboard once to twice per sprint. Some scrum masters said they expect to open it once individually and the other time with the whole team present, but the minimum they expect to open the dashboard at all is once per sprint.

A-2: Change management is important to successfully introduce the dashboard

As one may have read from the first few weeks of each team in the logbook, a lot of teams did not open the dashboard at all. It required interventions from the researcher and using these weekly conversations with the scrum masters as implicit accountability sessions to increase the chances of them opening the dashboard. Change can be difficult, because it requires altering familiar behaviour patterns. The interventions resulted in the scrum masters familiarizing themselves with the dashboard because they did not have any other choice. While change management would have been easier if each dashboard would have been exactly customized to the wishes of the scrum masters, the decision was made to use a dashboard template that could be used by the majority of teams.

A-3: The dashboard made some scrum masters aware of information they otherwise did not know

Everything the dashboard shows to the user (except for the satisfaction survey results), can also be found in ADO. However, not all scrum masters are proficient or have enough time to scour ADO for potential information to take action on. This was evident for some teams, including Team A. This team's scrum master said the dashboard helped him become aware of information he otherwise was not aware of. An example of this were the open impediments in the fourth week.

A-4: Framework with the five categories is useful, but received numerous requests to customize it to each team

The purpose of this research was to test out a dashboard to understand the benefits and challenges of introducing this dashboard to Agile teams; it did not really matter with which tool this dashboard was made. However, during the experiment, I received multiple requests from the scrum masters to apply some customized filters on the metrics because of their way of working. Additionally, each team had a slightly different requirements as to which metrics are relevant for the category 'Workflow health of current sprint'.

A-5: Some scrum masters used their own set of metrics in ADO alongside the dashboard

While the dashboard was received positively by some scrum masters because those scrum masters did not use any metrics at all, other scrum masters already had their own set of metrics in ADO that provided them with actionable/useful information. Examples would be the scrum masters of Team C and Team B that both had their own basic dashboard in ADO.

A-6: Dashboard was beneficial to use

Based on the answers to the question '*Do you feel the dashboard was beneficial to use during these past ~11 weeks?*' during the exit interviews, it can be concluded that the dashboard was beneficial to use according to the scrum masters. The information displayed was valuable. The only exception for this was the scrum master of Team D, because he said there wasn't that much work being performed to leverage value out of the dashboard.

A-7: PowerBI skills and knowledge is the main determinant of whether the scrum master will take ownership of the dashboard after the experiment

When asked about the planned ownership and maintenance of the dashboard, the scrum masters with previous PowerBI skills and knowledge were more eager and interested to pick up this task than the scrum masters with no PowerBI experience. Those with no PowerBI experience said someone well-versed in PowerBI or/and someone from the Project Management Office in the organization should be responsible for this task.

Category B: The actionability of metrics

B-1: Actionable metrics mainly present in categories 'Quality' and 'Workflow health current sprint'

Both through observation and conducting weekly conversations with the scrum masters, it became evident the categories 'Quality' and 'Workflow health current sprint' contained the most actionable metrics. For the team Team G, the scrum master said in the first and ninth week that the category 'Quality' is most useful and actionable for him. The team Team A used the category 'Workflow health current sprint' in the fifth and ninth week to adjust their administrative process in ADO. The team Team B intensively used the visual 'Longest open standing bugs' in the 'Quality' category. Additionally, the teams Team E in week 2 and team Team F in week 7 said the category 'workflow health current sprint' is very actionable/useful for them as their team is not that mature in their

agility.

The exception in the category 'Quality' would be that the visual '# of bugs per feature' was not that actionable or interesting for the Team G scrum master in week 6 and Team A scrum master in week 1.

B-2: Category 'Responsiveness' is not always a reflection of reality (relies on administration too much)

The category 'Responsiveness' contained two visuals: '# of features released past year and avg. cycle time' and '# of user stories completed past 4 months and avg. cycle time'. The initial idea was to visualize the team's output on both feature and user story level. However, during observation of the dashboard use and talking with the scrum masters, this 'Responsiveness' category did not always reflect the reality; features were often already finished and implemented at the customer's side, but it was not always marked as complete by the team in ADO. Thus, the average cycle time of how long a feature takes to develop was not an accurate representation. This could be observed in the fourth and fifth week of team Team G, when the scrum master said the increase in cycle time for features was mainly because the backlog was cleaned up by the scrum master. For the Team B team, this category was not that interesting, because the features are on-going, so it was not representative of the team's output in terms of features. The scrum master of Team E even used the visuals in the category 'Responsiveness' to know whether she could mark long open-standing features as complete. The scrum masters said 'Responsiveness' is more useful on a user story-level than on a feature-level, including the Team G scrum master who said so in the first week.

B-3: The category 'Workflow health current sprint' is most interesting in the beginning of the sprint

The scrum master of Team B said in week 6, the category 'workflow health current sprint' is the most relevant in the beginning of the sprint. Since this category contains health metrics such as whether all user stories have an effort assigned or whether all tasks are assigned to a team member, the beginning of the sprint is the time when these health metrics would matter the most.

B-4: The metrics on which action could be taken immediately and metrics that provided a historical overview serve different purposes

The dashboard contained a mix of both actionable metrics that were relevant immediately and showed historical views of e.g. development cycle time of features. The scrum master of Team C in the third week said the health metrics are more relevant to take action on daily, while the historic visualizations are better suited for monthly or quarterly reporting to show and communicate certain trends. The scrum master of Team G made a similar comment in the fifth week; the dashboard could be split into two with each a different focus. The first one would focus on very actionable metrics and the second dashboard would provide historical – but less actionable – trends of how the team performed. He said not all metrics change with the same frequency.

B-5: Health metrics in the category 'Workflow health current sprint' served multiple purposes

While it has already been discussed in this section that the health metrics helped the team with their ADO administration, the health metrics also served another purpose. This second purpose is the fact that the health metrics ensured the metrics in 'Workflow health' were displayed correctly. An example would be the health metric '# of user stories without effort' had an impact on the velocity graph displaying information correctly. After all, the velocity graph sums the effort of all user stories for a certain sprint. Thus, the health metrics indirectly were helpful, regardless of Agile maturity, because it ensured other metrics were shown correctly. In the sixth, eighth and ninth week, it was observed that the category 'Workflow health of current sprint' ensured the KPIs in 'Workflow health' were displayed correctly.

B-6: There is a difference in what the scrum masters deem as 'actionable' and 'useful'
Half-way through the experiment, it was decided to start asking quantitative questions during the weekly calls with the scrum masters; I asked them to rate each metric or visualization on a scale from 1 to 5 on both how actionable they thought it was and how useful they thought it was. After a few weeks of incorporating this question in the weekly calls, it was noted the answers to this question barely changed. Thus, it was removed from the questions I asked during these weekly calls with the scrum masters. While these numeric answers are not further explored or used, it was interesting to see that some metrics had a different rating on their 'actionableness' and 'usefulness'. An example would be the visual '# of bugs by severity' in the category 'Quality'. The scrum master of Team A said in the sixth week this was useful to know and see, but not actionable for him as a scrum master.

Category C: The relation between Agile maturity of teams and their usage of metrics

C-1: Teams with a lower Agile maturity were helped by the dashboard more than the teams with a higher level of Agile maturity
The Agile maturity of the teams was shown in table 8. During the experiment, it became evident that teams with a low Agile maturity were helped by the dashboard more than Agile mature teams. The dashboard helped these teams with a lower Agile maturity to optimize their work administration and helped in keeping track of the work performed. The caveat is that this does not automatically mean the team is more successful or productive.

This was evident for the Team F team; in week 6 the scrum master said that some parts of the dashboard (category 'Workflow health') are really useful, considering the team's Agile maturity. The same was said by the scrum master of Team E in the second week. On the other hand, the scrum master of Team G (an Agile mature team) said the category 'workflow health current sprint' is not that useful/relevant exactly because the team is mature in their agility. The caveat is that this does not automatically mean the team is more successful or productive.

C-2: The more mature a team is, the less relevant 'health metrics' become: trust and familiarity with the team & ADO plays a bigger role
While each team used slightly different health metrics in the category 'Workflow health current sprint', it was observed that these health metrics become less relevant and less interesting if a team is considered Agile mature. This was evident from the Agile mature teams. The scrum master of Team C said in the fifth week that the health metrics in the dashboard were too low-level and tend to imply micromanagement of the team. For him, trust was more important than tracking all those administrative metrics. The scrum master of Team B said something similar in the seventh week; the team is proficient in reading and dealing with the ADO product and sprint backlog. There is no need for such health metrics for that team. In the second week, the scrum master of Team G mentioned that because of the team's high Agile maturity, the health metrics of the category 'Workflow health current sprint' are not that relevant/useful. In the sixth week, the scrum master of Team G said the health metrics on the task-level were too low-level and requested to alter the health metrics to the user story level.

However, even though these health metrics are less relevant for Agile mature teams, sometimes these health metrics did help. For example, in the week 8, the scrum master of Team G used the

health metric '# of user stories without story points' in the category 'Workflow health current sprint'.

Category D: Using surveys to collect stakeholder and team satisfaction

D-1: Results of employee and customer satisfaction surveys were used less often than initially expected

Initially, it was thought the satisfaction of team members and customers would be of paramount importance. After all, releasing new features rapidly means little if both the team members and customers are not satisfied. However, the results of the employee and customers satisfaction surveys were used less often than expected. For the customer satisfaction survey results, the teams Team C and Team G used it to engage in their communication to the respective product owners. Other than that, the results of the customer satisfaction surveys were deemed as interesting, but nothing more than that. For the employee satisfaction survey results, some teams thought it was informative and interesting to quantify. The scrum master of Team B said it was interesting to know and see the employee satisfaction and she even used it as input for one of their team-building events. The scrum master of Team D said in week 3 that those results were informative, but not very actionable for him. The scrum master of Team G said the results of the employee satisfaction are already known implicitly, since it's a tight-knit team. Thus, it really depends on the team whether it was useful to quantify the employee satisfaction, but most of the time it was not used that much, partly because the scrum masters already implicitly know the answers to the questions.

D-2: Assessing the employee satisfaction with Microsoft Forms resulted in a decreasing response rate over time

Some teams used Microsoft Forms to assess the employee satisfaction, while other teams used Mentimeter during the retrospective in a more interactive manner. While the answers of the surveys cannot be compared because each sprint has different dynamics, workload and context, the number of responses can be compared. Using Mentimeter as a tool during the retrospective to assess the team member's satisfaction ensured everyone voiced their opinion. This is in stark contrast with using Microsoft Forms as a tool, because for almost all the teams, the response rate was either low from the beginning or dropped significantly after the first iteration.

Because the experiment lasted only eleven weeks, the choice was made to conduct these surveys as often as possible (while not irritating the respondents) in order to collect enough data for this research. However, if the research was not constrained by these eleven weeks, the frequency of both employee satisfaction and customer satisfaction surveys would definitely be altered to be less frequent. Partly because multiple scrum masters mentioned that as a scrum master, (s)he knows already how satisfied the team is due to the stand-ups, retrospectives and grooming sessions. The surveys only confirms this gut feeling most of the time, but does not provide a groundbreaking insight.

D-3: Some scrum masters felt satisfaction surveys were not useful or interesting

This finding corresponds to finding D-1; the results of the employee and customer satisfaction surveys were used less often than initially expected *because* some scrum masters felt these surveys were not very useful or interesting.

D-4: Majority of the scrum masters prefer to use an interactive tool to assess the employee satisfaction

When asked about what which tool they would prefer to assess the employee satisfaction in the future, the majority of the scrum masters said they would like to do this in an interactive manner

(e.g Mentimeter, Mural). The scrum masters of Team E and Team G were satisfied with the tool used during the experiment; Microsoft Forms.

D-5: There is no consensus on what the ideal frequency is to assess the employee satisfaction
It seems there is no relation between the Agile maturity of the team and the perceived ideal frequency to assess the employee satisfaction. Four scrum masters said once a month should be the ideal frequency, while the other three said once in three to six months would be the ideal frequency.

D-6: Microsoft Forms is the most appropriate tool to assess the customer satisfaction
The answers during the exit interviews should that every scrum master was content about using Microsoft Forms to assess the customer satisfaction (with the exception of Team F' scrum master). It enables customers to answer whenever they have time and does not require much time from the customer's end.

D-7: The release frequency may play a role in the timing of assessing the customer satisfaction
While five out of the seven scrum masters said they expect to assess the customer satisfaction every quarter, some of the scrum masters also said the release frequency also plays a role in determining what the frequency should be of assessing customer satisfaction. It could be better to assess their satisfaction after every major release instead of a fixed time of every quarter; the answers could have more potential for follow-up.

D-8: Customer satisfaction surveys should be customized to its audience
One of the questions asked during the exit interviews was '*What is your opinion on the questions asked in this customer satisfaction survey; both the number of questions and the questions themselves?*'. I received numerous points of feedback; this shows that the customer satisfaction survey is less suitable to be generalized and instead should be customized together with the scrum master and product owner to elicit the desired answers.

6 – Reflection & discussion

Teams with a lower Agile maturity tend to benefit more from the dashboard and Agile metrics than teams with a higher level of Agile maturity

While all seven of the participating Agile teams experienced some benefit in using the dashboard that contained Agile metrics, teams with a lower Agile maturity tend to benefit more from the dashboard than teams with a higher level of Agile maturity. The Agile maturity of the teams was assessed using Laanti's Agile transformation model (2021).

Teams with a higher level Agile maturity were less driven by metrics to assist them in their Agile work and administration process. Instead, concepts such as 'trusting each other to get the job done' and 'familiarity with the Agile backlog tool' were more relevant and important for them. This was both observed by the researcher, and made clear by the scrum masters of these teams with a higher level of Agile maturity. Korpivaara et al. (2021) states the level of Agile maturity could change the importance of metrics, which is in line with this research' findings.

Teams with a lower level of Agile maturity experienced benefits by using the dashboard to support their administration in Azure DevOps and their Agile processes. An example of how the dashboard supported the teams in their administration is the fact that health metrics reminded the team to assign an effort estimation to user stories or to assign unsolved bugs to team members. An example of how the dashboard helped the teams in their Agile process was the Velocity chart. The difference between 'estimated effort' and 'completed effort' of the past five sprints provided an indication to the scrum master on how accurate the team was in estimating their work (i.e. was the difference between estimated and completed effort decreasing).

These observations can be connected to the literature. Lappi et al. (2018) conducted research on Agile project governance practices, and he identified six key practices for Agile project governance: (1) Goal-setting practices, (2) Incentive practices, (3) Monitoring practices, (4) Coordination practices, (5) Roles and decision-making power practices and (6) capability building practices. While all six have been defined in the second chapter, only the third is relevant to mention here again. Monitoring practices are concerned with visually tracking the progress of projects and teams (Lappi et al., 2018). The dashboard assisted the teams with a lower Agile maturity in making decisions and retrieving information during each sprint. Of course, not all metrics are relevant all the time or at the same time. Some metrics are useful in the beginning of the sprint (such as health metrics), while other metrics are useful to observe throughout the sprint (such as the cycle time), or metrics that are useful at the end of a sprint (such as the velocity). Of course, these examples can be used interchangeably, depending on the needs of the scrum master and their team.

Additionally, while metrics are a helpful tool in the Agile governance process, it is advisable to select with the scrum master and the team which metrics they would like to use and include. It is important to avoid overloading users with metrics (Bourguignon et al., 2001).

Assessing customer & employee satisfaction enriches the Scrum team ceremonies and customer interactions

The primary principle of the Agile movement is the fact that customer satisfaction is of the highest priority (Buresh, 2008). Studies show adopting an Agile method instead of a traditional project/software development method (e.g. the waterfall method) can result in an increase in customer satisfaction (Mann & Maurer, 2005; Kohlbacher et al., 2011).

Due to its adaptive and iterative nature, the Agile approach prescribes to focus on customer collaboration and interactions (Beck et al., 2001). This is evident from the Scrum (the most popular

Agile method and also the method all participating teams in the experiment used) ceremonies. The sprint review serves to present the results of the sprint to key stakeholders, such as the customers (Schwaber & Sutherland, 2011). This demonstrates the embedded and implicit customer focus. However, what the Agile methodologies lack is a way to explicitly and quantitatively capture the customer satisfaction. The same argument can be made for employee satisfaction. The sprint retrospective serves to discuss within the team what went well, what problems were encountered and how the team could learn from this (Schwaber & Sutherland, 2011). But again, the Agile methodologies lack the way to explicitly and quantitatively capture the customer satisfaction.

The research showed both the results of employee and customer satisfaction surveys triggered discussions with relevant stakeholders, because it made their sentiments explicit. For example, some scrum masters used the results of the customer satisfaction surveys as input for their conversations with their customer(s). For the employee satisfaction, it was observed that the retrospective proved not only the ideal moment to conduct a satisfaction assessment, but also the ideal moment to discuss the results immediately. After all, the sprint retrospective is used to identify what went well, what did not work and what the team has learned from this sprint (Przybyłek & Kotecka, 2017). Because one of the factors to ensure an effective retrospective is to not conduct it in the same way repeatedly (Przybyłek & Kotecka, 2017), integrating such an (interactive) employee satisfaction assessment can increase the effectiveness and success of the retrospective.

Perhaps it is an obvious statement, but subjective data was needed for the dimensions 'Employee satisfaction' and 'Customer satisfaction'. This is in contrast with data used by other dimensions (Quality, Workflow health, Responsiveness), which are objectively and directly sourced from Azure DevOps; the tool the teams used for their Agile administration process.

To connect the topic of customer satisfaction surveys to the literature, the antecedents of the American Customer Satisfaction Index were used to categorize and formulate questions for the customer satisfaction surveys used in this research. Through this research, these antecedents proved useful as categories for the customer satisfaction surveys. However, the exit interviews with the scrum masters showed basing the questions on the antecedents is not as straightforward as it seems. The questions should be customized per team, in collaboration with the scrum master. Each team has a different context, different customers (e.g. internal or external customers) and a different product (e.g. an analytics dashboard or a web portal). An example of this would be the first question in the customer satisfaction survey: 'On a scale from 1 to 10, how likely would you recommend this development team to your peers?', based on the 'Customer loyalty' antecedent of the American Customer Satisfaction Index and inspired by the Net Promoter Score-concept. Some scrum masters remarked that this question is not formulated correctly, because in their case the customer was removed further from the development team than the Agile Scrum methodology prescribes. In those cases, it is still beneficial to assess the antecedent 'Customer loyalty', but closer scrutiny is needed to adapt the question in collaboration with the respective scrum master.

Assessing the team (member) satisfaction should be done in an interactive manner
For all seven participating teams, the customer satisfaction was assessed every 4 to 6 weeks by sending an e-mail to the relevant customers/end-users with the link to the survey. However, for the employee satisfaction, some teams used an interactive tool to assess their satisfaction during the retrospective, while other teams used the more conventional approach of sending every one to two sprints an e-mail to asking the team members to fill in the employee satisfaction survey.

The interactive tool – in the case of this research the tool was Mentimeter – was used during the retrospectives of some teams to assess their satisfaction with six questions. The questions and

subsequently the answers were shown on-screen during the retrospectives, in order for the scrum master and their team to see and discuss the results. These teams that used this interactive tool to assess and visualize their satisfaction, resulted in some cases in discourse between the team members. These discussions were aimed at understanding each other's answers and perspectives, based on the visualized and shown results during the retrospectives. It contributed to the purpose and idea behind the retrospective ceremony.

This is in stark contrast with the teams that used Microsoft Forms; a survey link sent via e-mail. For these teams, a decreased response rate was observed throughout the action taking-phase. While the results of these employee satisfaction surveys were visualized in the dashboard, no action or interest was observed during the eleven weeks by neither the scrum master nor the team members. Assessing their satisfaction and visualizing it afterwards was simply not actionable or useful for these teams.

For the practitioner, it is advised to assess the employee/team member satisfaction in an interactive and visual way in order to maximize the engagement and increase the likelihood of action taken on the results if necessary. For the customer satisfaction, both the action taking-phase and the exit interviews conducted with the scrum masters show a simple survey link sent to the customers is sufficient and appropriate; the answers can be submitted when the customer feels like it and the scrum master can review it in their own time to determine if any action is necessary.

To connect the topic of employee satisfaction surveys to the literature, the questions in the employee satisfaction surveys were mapped to six dimensions of the Minnesota Satisfaction Questionnaire (MSQ). These six dimensions were chosen or derived from the twenty MSQ dimensions in a pragmatic manner in collaboration with key stakeholders, because these dimensions had to be relevant enough to ask about every one to two sprints. However, for the future practitioner, one needs to consider how often and which of the twenty dimensions are relevant enough to base questions on. Perhaps the future practitioner could even alternate between the twenty MSQ-dimensions; each retrospective would focus on different MSQ-dimensions. This way, the assessment of employee satisfaction does not become stale and in turn, the retrospective will not be seen as repetitive (as Przybyłek & Kotecka in 2017 warned about).

Data quality management is important to ensure actionability of metrics
Agile teams use a tool to manage their sprint and product backlogs, whether this is a digital product such as JIRA or Azure DevOps, or a physical whiteboard with sticky notes on it. Even without metrics, it is important for these teams to fill out everything correctly (e.g. the estimated story points for user stories are filled out correctly). But when metrics make their entrance, it adds another layer of importance to make sure data quality is managed efficiently and effectively. For example, if a feature is completed and pushed to the customer, the team must make sure that this is also reflected in the backlog; the feature should be marked as completed on that date. Otherwise, the feature continues to be 'open' and the cycle time increases every day the feature is not marked as 'complete'. When the team wants to leverage the cycle time metric in the future to understand how Agile they are, this cycle time metric shows a distorted view of reality. Haug & Arlbjørn (2011) also make a case for why data quality is important and stipulate barriers to break through in order to master data quality.

The experiment showed that some categories in the dashboard (Quality, Workflow health, Workflow health current sprint), resulted in more actions by the scrum masters and their teams than other categories (Employee satisfaction, Responsiveness). While the lack of actionability for the employee satisfaction dimension has already been discussed, the Responsiveness dimension has not. It is unclear whether the Responsiveness-category in the dashboard resulted in less action taken because

of its 'looking back'-nature on how many features/user stories have been completed and their cycle time, or because the scrum masters knew the Responsiveness-visuals were not representative of reality. As previously discussed, scrum masters often forget to mark a feature as 'complete', resulting in an increased cycle time. This distortion of reality can have an impact on how the scrum masters view the actionability and usefulness of metrics when they know the data quality is compromised. For future practitioners, it is crucial to manage the data quality in order to leverage the most value and potential action from a dashboard containing Agile metrics.

Limitations and threats to validity

The generalizability of these results is limited by the small sample size of seven Agile teams. The small sample size is due to the focus of this research; it was more qualitative than quantitative (as evident from the weekly conversations with the seven scrum master in a period of eleven weeks). So even though it was a conscious effort to keep the sample size small, it does impact the generalizability of the results. However, because the seven participating teams had different levels of Agile maturity and they all worked on very different products, one could say this threat to validity is sufficiently mitigated.

The reliability of the data is slightly impacted by the fact that two of the seven Agile teams that participated in this research could or did not leverage the value from the dashboards as much as was expected. One team (Team D) was nearing the end of their project and the team barely did any work. While this was not known at the beginning of this research, it would have been better to swap out this team for a more active Agile team if this was known beforehand. The second team (Team E) had an on-hold phase for several weeks in the beginning of the experiment, due to many problems experienced by the customers. The result of this on-hold phase was that this team could not leverage that much from the dashboard in the first weeks of the experiment phase.

Besides these general threats to validity which are applicable for any type of research, there are also threats to validity which are specific to (participatory) action research. There seems to be no consensus on the threats to validity for (participatory) action research; each article identifies different threats to validity. In this section, we will discuss a few of these threats to validity.

Checkland & Holwell (1997) state that action research is unable to equal the replicability of natural science, which is characterized by "*testing hypotheses to destruction*". To compensate, Checkland & Howell (1997) state that action researchers must at least establish a research process which is recoverable by interested third-parties. This threat to validity has been partly covered by documenting extensively what has been done, why it has been done and what the results have been. For example, during the eleven weeks of the action-taking phase, a logbook was kept. Based on this logbook, main findings and learnings were formulated, which helped answer the research question.

Herr & Anderson (2014) state the fact that action research is political could be a threat to validity. Because of the action research' participatory nature, it is important to be aware of this political factor. Action research is usually conducted in institutions or organizations, which are inherently political (Herr & Anderson, 2014). Office politics is part of this political factor; office politics revolves around influence and relationships employees have with and over each other (Harvard Business Review, 2021). An action researcher may be indirectly influenced or affected by these office politics. For this particular research, this political threat to validity was mitigated due to the design of the research. Seven product teams participated in the research, all seven teams were responsible for very different products in the organization. The interactions were primarily with the seven scrum masters on an individual basis, so this political factor was not really encountered (even though it

may had been present). While the sponsor of this action research was the head of the EMEA Project Management Office, his role was limited to occasionally providing feedback.

Coughlan & Coghlan (2002) state that the principal threat to the validity for action research is the researcher's lack of neutrality. As an action researcher, one is involved in the shaping and telling of a story. The action researcher must always take into consideration to which extent the told story is an accurate representation of what took place and how the story is understood (Coughlan & Coghlan, 2002). This threat to validity is mitigated by the logbook, with which I tried to take notes about what I saw, heard and talked about with the scrum masters throughout the eleven weeks; this was done while being aware of the importance of neutrality. The main findings are based on how the logbook is understood by the action researcher; what patterns could be extrapolated and what evidence (e.g. team A in week 4 said such and such) is there for making such a claim.

Future work

Based on the limitations, future studies should consider whether participating Agile teams in such an action research are the best candidates to participate in the research. Not because of their Agile maturity, but because not each team has enough work to leverage value from a dashboard (e.g. teams that are primarily concerned with maintenance of a product instead of actively developing it).

This research demonstrated a relation between Agile maturity of teams and how they perceive and use metrics in the form of a dashboard. Future research is required to establish whether this is the case on a quantitative scale and whether Agile maturity is the main or only independent variable at play.

Future research could also take into account a longer time period than the eleven weeks used in this action research. Does the use of and value leveraged from the dashboard/metrics change over a longer period of time? Of course, this should again be done with Agile teams who are in an active development phase, as to maximize the value from this research.

7 – Conclusion

The goal of this research project was to help the case study organization one step further in their Agile transformation by researching the benefits and challenges of introducing a dashboard containing Agile metrics to Agile teams, in order for the teams to improve their work. This framework of metrics could then be used by the organization's Project Management Office as a standardized approach to help Agile teams and their scrum masters in their quest for increased productivity, efficiency and effectivity. Whether such a framework was created and tested is not that straightforward. On the one hand, the dimensions of organizational performance (Stettina et al., 2021) were used as categories in the PowerBI dashboard. Thus, claiming a framework was created is too farfetched. However, those dimensions of organizational performance were used as the main categories in the dashboard to group and organize the metrics. Thus, a framework in the form of a PowerBI dashboard, containing dimensions of organizational performance by Stettina et al. (2021) was created and tested.

Revisiting the research question

This research has attempted to answer the following research question: "*What are the benefits and challenges of introducing a dashboard containing Agile metrics to Agile teams, in order for the teams to improve their work?*". The encompassing methodology used to answer this research question was the action research method. In applying this action research method, a dashboard template was tested through an 'experiment' in the action-taking phase.

Benefits of introducing a dashboard containing Agile metrics to Agile teams, in order for the teams to improve their work

The dashboard with Agile metrics primarily helped teams with a lower Agile maturity in their Agile processes and administration. Actionable metrics (mainly found in the category 'Quality', 'Workflow Health' and 'Workflow health current sprint') helped these teams in their Agile process and administration. The Agile Manifesto states that the continuous improvement of processes is recommended (Lárusdóttir et al, 2014); the dashboard in this research assisted the teams in this endeavour. The dashboard with Agile metrics compelled the scrum master and their teams to think about what they saw and how they can improve both as a team and in their way of working.

While one of the challenges was that data quality affects the usefulness and actionability of metrics, there was also a benefit in this poor data quality for their administration. The visuals affected by poor data quality (and thus not representative of reality) helped the team understand how their administration in their project management tool impacted the usefulness and actionability of metrics in the dashboard. This triggered some of the scrum masters to take data quality in their administration more seriously.

Challenges of introducing a dashboard containing Agile metrics to Agile teams, in order for the teams to improve their work

While already mentioned as a benefit, one of the challenges in introducing a dashboard containing Agile metrics and making it useful, was the fact that some metrics were affected by poor data quality. Due to this poor data quality, some metrics were not useful or actionable for the scrum masters and their teams; this contributed to some scrum masters perceiving the dashboard as a whole as less valuable to open and use.

The second challenge is change management. Scrum masters and their teams suddenly had to use and integrate a dashboard with Agile metrics in their processes and routines they were much accustomed to. It takes time to change the behaviour of people (Bugwandeen & Ungerer, 2019). This matter was overlooked and underestimated in this action research; during the first few weeks of the

experiment, almost all scrum masters forgot to integrate the dashboard in their daily work processes. This was despite the fact that the value of the dashboard had been shown and demonstrated before the experiment started. For the future practitioner, it is important to take into consideration how to manage and facilitate this change in order to maximize the value of using metrics/a dashboard. A potential way to aid in this change management is to implement some sort of accountability; according to Bugwandeen & Ungerer (2019), accountability can help in changing the behaviour of stakeholders.

The third challenge in introducing such a dashboard is there needs to be a balance in creating a dashboard template that is useful for all/most Agile teams, while accommodating the different demands and preferences of these teams. Each team uses the project management tool in a slightly different way, which translates into changing the dashboard for each team to fit their processes. One the one hand, one does not want to spend a lot of time customizing the dashboard template for each team, especially when lots of Agile teams will be using such a dashboard. On the other hand, one needs to create a dashboard template valuable enough so it can be used and adapted quickly to each team, so the majority of teams can stand to benefit from it.

The fourth challenge is that two of the dimensions in the dashboard (customer & employee satisfaction) rely on subjective and manual input from satisfaction surveys. As previously discussed, the downside of this is the response rate/engagement decreases over time if the survey is conducted in an non-interactive manner. Should these two dimensions and their subjective data sources be included in such a dashboard? Throughout this research, it took additional time and effort to import and visualize the data (due to technical limitations), compared to the automatic data connections for the other dimensions in the dashboard. Additionally, the experiment showed that visualizing both customer and employee satisfaction barely resulted into action by the scrum masters. The answers of both dimensions stayed the same for weeks or months at the time, due to the frequency of assessing the employee or customer satisfaction. Thus, it is unclear whether the results of the employee and customer satisfaction should be included in such a dashboard, because of its subjectivity, infrequent new information and observed inaction.

The benefits and challenges have been laid out. However, it cannot be concluded that the dashboard tested and introduced in this research actually contributed to improving the work of Agile teams. Instead, it was observed and can be concluded the dashboard proved useful to support the scrum master and their team (especially teams with a lower level of Agile maturity) in their Agile process and administration.

What is the bigger picture?
Looking across the findings and this whole research project, the bigger picture shows how a measurement framework, in the form of a dashboard (i.e. the artefact) with actionable metrics could help teams with their administrative tasks and increase their efficiency and effectivity in their Agile work processes. Whether such a dashboard actually increases performance is something which cannot be answered by this research. Using satisfaction surveys are helpful in an Agile context, but both the tool and formulation of questions should be adapted to each audience in order to maximize the audience's engagement. This research showed how information in a dashboard – based on a measurement framework - can be useful, but not necessarily actionable. Actionability is an important factor to consider in creating a dashboard.

Bibliography

Abrahamsson, P., Fronza, I., Moser, R., Vlasenko, J., & Pedrycz, W. (2011, September). Predicting development effort from user stories. In *2011 International Symposium on Empirical Software Engineering and Measurement* (pp. 400-403). IEEE.

Agile Alliance. (2021, August 17). *Agile Cycle Time and Throughput Case Study | Agile Alliance*. Retrieved August 18, 2022, from https://www.Agilealliance.org/resources/experience-reports/actionable-metrics-siemens-health-services/

Agile Tools. (n.d.). *Backlog Health*. Retrieved August 2, 2022, from https://www.Agile-tools.io/library/metric/backlog-health

Ahmad, G., Rahim Soomro, T., & Raza Naqvi, S. M. (2016). An overview: merits of Agile project management over traditional project management in software development. *Journal of Information & Communication Technology*, *10*(1), 105-120.

Airfocus. (2022, April 12). *A Product in Product Management*. Retrieved July 7, 2022, from https://airfocus.com/glossary/what-is-a-product-in-product-management/

Alahyari, H., Svensson, R. B., & Gorschek, T. (2017). A study of value in Agile software development organizations. *Journal of Systems and Software*, *125*, 271-288.

Al-Heyasi, A. (2018). Individuals performance measurement in Agile software development. *Eurasian Journal of Social Sciences*, *6*(1), 1-6.

Alnaji, L., & Salameh, H. (2015). Performance-measurement framework to evaluate software engineers for Agile software-development methodology. *European Journal of Business and Management*, *7*(2), 183-190.

Alromaihi, M. A., Alshomaly, Z. A., & George, S. (2017). Job satisfaction and employee performance: A theoretical review of the relationship between the two variables. *International Journal of Advanced Research in Management and Social Sciences*, *6*(1), 1-20.

Alshmemri, M., Shahwan-Akl, L., & Maude, P. (2017). Herzberg's two-factor theory. *Life Science Journal*, *14*(5), 12-16.

Alwin, D. F. (2007). *Margins of error: A study of reliability in survey measurement*. John Wiley & Sons.

Amaratunga, D., & Baldry, D. (2002). Moving from performance measurement to performance management. *Facilities*.

Anderson, D. J. (2003). *Agile management for software engineering: Applying the theory of constraints for business results*. Prentice Hall Professional.

Anderson, D. J. (2010). *Kanban: successful evolutionary change for your technology business*. Blue Hole Press.

Anderson, E. W., Fornell, C., & Lehmann, D. R. (1994). Customer satisfaction, market share, and profitability: Findings from Sweden. *Journal of marketing*, *58*(3), 53-66.

Andrei, B. A., Casu-Pop, A. C., Gheorghe, S. C., & Boiangiu, C. A. (2019). A study on using waterfall and Agile methods in software project management. *Journal of Information Systems & Operations Management*, 125-135.

Arafeen, M. J., & Bose, S. (2009). Improving Software Development Using Scrum Model by Analyzing Up and Down Movements on The Sprint Burn Down Chart: Proposition for Better Alternatives. *International Journal of Digital Content Technology and its Applications*, *3*(3).

Armstrong, M. (2006). *Performance management: Key strategies and practical guidelines*. Kogan Page Limited.

Armstrong, M., & Baron, A. (1998). Performance management: The new realities, Institute of Personnel and Development. *New York*.

Armstrong, M., & Murlis, H. (2007). *Reward management: A handbook of remuneration strategy and practice*. Kogan Page Publishers.

Askariazad, M. H., & Babakhani, N. (2015). An application of European Customer Satisfaction Index (ECSI) in business to business (B2B) context. *Journal of Business & Industrial Marketing*.

Atlassian. (n.d.). *What is kanban?* https://www.atlassian.com/Agile/kanban

Atlassian. (n.d.-a). *Five Agile metrics you won't hate*. https://www.atlassian.com/Agile/project-management/metrics

Atlassian. (n.d.-b). *The Spotify Model for Scaling Agile*. https://www.atlassian.com/Agile/Agile-at-scale/spotify

Atlassian. (n.d.-c). *What is Agile?* https://www.atlassian.com/Agile

Avgeriou, P. C., Taibi, D., Ampatzoglou, A., Fontana, F. A., Besker, T., Chatzigeorgiou, A., ... & Tsintzira, A. A. (2020). An overview and comparison of technical debt measurement tools. *Ieee software*, *38*(3), 61-71.

Avgeriou, P., Kruchten, P., Ozkaya, I. and Seaman, C., 2016. "Managing technical debt in software engineering (dagstuhl seminar 16162)." In Dagstuhl Reports (Vol. 6, No. 4). Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik.

Aydin, S., & Özer, G. (2005). National customer satisfaction indices: an implementation in the Turkish mobile telephone market. *Marketing Intelligence & Planning*, *23*(5), 486-504.

Bahrami, M. A., Aghaei, A., Barati, O., Tafti, A. D., & Ezzatabadi, M. R. (2016). Job Motivating Potential Score and Its Relationship with Employees' Organizational Commitment among Health Professionals. *Osong. Public Health Res. Perspect*.

Bean, R. (2022). Why Becoming a Data-Driven Organization Is So Hard. *Harvard Business Review. Julkaistu*, *24*, 2022.

Beck, K., Beedle, M., Van Bennekum, A., Cockburn, A., Cunningham, W., Fowler, M., ... & Thomas, D. (2001). Manifesto for Agile software development.

Bessire, D., & Baker, C. R. (2005). The French Tableau de bord and the American Balanced Scorecard: a critical analysis. *Critical perspectives on accounting*, *16*(6), 645-664.

Bhandari, P. (2022, November 11). *What Is a Controlled Experiment? | Definitions & Examples*. Scribbr. https://www.scribbr.com/methodology/controlled-experiment/

Bin, A. S., & Shmailan, A. (2015). The relationship between job satisfaction, job performance and employee engagement: An explorative study. *Issues in Business Management and Economics*, *4*(1), 1-8.

Bodur, S. (2002). Job satisfaction of health care staff employed at health centres in Turkey. *Occupational medicine*, *52*(6), 353-355.

Boerman, M. P., Lubsen, Z., Tamburri, D. A., & Visser, J. (2015, May). Measuring and monitoring agile development status. In *2015 IEEE/ACM 6th International Workshop on Emerging Trends in Software Metrics* (pp. 54-62). IEEE.

Bokhari, M. U., & Siddiqui, S. T. (2011, March). Metrics for requirements engineering and automated requirements tools. In *Proceedings of the 5th National Conference* (pp. 10-11).

Boon, G. C., & Stettina, C. J. (2022). A Case for Data-Driven Agile Transformations: Can Longitudinal Backlog Data Help Guide Organizational Improvement Journeys?. In *International Conference on Agile Software Development* (pp. 114-130). Springer, Cham.

Boot, Renée. Technology Trend Research & Forecasting Using Text Mining as Input for Scenario Planning, Thesis Master ICT in Business and the Public Sector, LIACS, Leiden University, 2022.

Bourguignon, A., Malleret, V., & Nørreklit, H. (2001). Balanced scorecard versus French tableau de bord: beyond dispute, a cultural and ideological perspective.

Brezočnik, L., & Majer, Č. (2016, July). Comparison of Agile methods: Scrum, Kanban, and Scrumban. In *Proc. 19th Int. Multiconference Inf. Soc* (Vol. 100, pp. 1-5).

Brown, M. G. (2020). *Keeping score: Using the right metrics to drive world-class performance*. Productivity Press.

Budacu, E. N., & Pocatilu, P. (2018). Real Time Agile Metrics for Measuring Team Performance. *Informatica Economica*, *22*(4).

Bugwandeen, K., & Ungerer, M. (2019). Exploring the design of performance dashboards in relation to achieving organisational strategic goals. *South African Journal of Industrial Engineering*, *30*(2), 161-175.

Buresh, D. L. (2008). Customer satisfaction and Agile methods. *Internet: https://rs.ieee. org/images/files/Publications/2008/2008-24.pdf ,[Apr. 26, 2017]*.

Buresh, D. L. (2008). Customer satisfaction and Agile methods. *Internet: https://rs.ieee. org/images/files/Publications/2008/2008-24.pdf ,[Apr. 26, 2017]*

Calvo-Mora, A., Leal, A., & Roldán, J. L. (2005). Relationships between the EFQM model criteria: a study in Spanish universities. *Total quality management & business excellence*, *16*(6), 741-770.

Carroll, I. (2021, January 7). *How to manage blockers & impediments – 2021 update*. Solutioneers. Retrieved August 5, 2022, from https://www.solutioneers.co.uk/blockers/

Cervone, H. F. (2011). Understanding Agile project management methods using Scrum. *OCLC Systems & Services: International digital library perspectives*.

Checkland, P., & Holwell, S. (1998). Action research: its nature and validity. *Systemic practice and action research*, *11*(1), 9-21.

Çiçek, M. C., Köksal, G., & Özdemirel, N. E. (2005). A team performance measurement model for continuous improvement. *Total Quality Management & Business Excellence*, *16*(3), 331-349.

Cohn, M. (2005). *Agile estimating and planning*. Pearson Education.

Cohn, M. (2010). *Succeeding with Agile: software development using Scrum*. Pearson Education.

Costello, P. J. (2003). *Action research*. Bloomsbury Publishing.

Coughlan, P., & Coghlan, D. (2002). Action research for operations management. *International journal of operations & production management*.

Crowther, D., Davies, M. L., & Cooper, S. M. (1998). Evaluating corporate performance: a critique of Economic Value Added. *Journal of Applied Accounting Research*, *4*(2), 3-34.

Davis, C. W. (2015). *Agile metrics in action*. Manning Publications Company.

Dawes, J. (2008). Do data characteristics change according to the number of scale points used? An experiment using 5-point, 7-point and 10-point scales. *International journal of market research*, *50*(1), 61-104.

De Geuser, F., Mooraj, S., & Oyon, D. (2009). Does the balanced scorecard add value? Empirical evidence on its effect on performance. *European Accounting Review*, *18*(1), 93-122.

DeCastellarnau, A. (2018). A classification of response scale characteristics that affect data quality: a literature review. *Quality & quantity*, *52*(4), 1523-1559.

Decker, P. J., & Borgen, F. H. (1993). Dimensions of work appraisal: Stress, strain, coping, job satisfaction, and negative affectivity. *Journal of Counseling Psychology*, *40*(4), 470.

Den Hartog, D. N., Boselie, P., & Paauwe, J. (2004). Performance management: A model and research agenda. *Applied psychology*, *53*(4), 556-569.

Deng, W. J., Yeh, M. L., & Sung, M. L. (2013). A customer satisfaction index model for international tourist hotels: Integrating consumption emotions into the American Customer Satisfaction Index. *International journal of hospitality management*, *35*, 133-140.

Dick, B. (n.d.). *Action research frequently asked questions*. Aral. Retrieved July 2, 2022, from http://www.aral.com.au/resources/arfaq.html

Digital AI. (2021). 15th Annual State Of Agile Report. In *Digital AI*. Digital AI. Retrieved September 16, 2022, from https://digital.ai/resource-center/analyst-reports/state-of-Agile-report/

Dixon, M., Freeman, K., & Toman, N. (2010). Stop trying to delight your customers. *Harvard Business Review*, *88*(7/8), 116-122.

Dransfield, R. (2000). *Human resource management*. Heinemann.

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

Eilers, K., Peters, C., & Leimeister, J. M. (2022). Why the Agile mindset matters. *Technological Forecasting and Social Change*, *179*, 121650.

Engelhardt, N. (2019). Comparison of Agile and traditional project management: Simulation of process models. *Acta academica karviniensia*, *19*(2), 15-27.

Epstein, M. J., & Manzoni, J. F. (1997). *The Balanced Scorecard and Tableau de Bord: a global perspective on translating strategy into action* (Vol. 63). Fontainebleau: Insead.

ERIKSSON, P. (2017). In Search of Excellence--Exploring Real Time Performance Measurement for Driving Continuous Improvement in Agile Teams.

EURIB. (n.d.). *American Customer Satisfaction Index*. Eurib. https://www.eurib.net/acsi-american-customer-satisfaction-index/acsi-american-customer-satisfaction-index-figuur-1/

Fisher, N. I., & Kordupleski, R. E. (2019). Good and bad market research: A critical review of Net Promoter Score. *Applied Stochastic Models in Business and Industry*, *35*(1), 138-151.

Fornell, C. (1992). A national customer satisfaction barometer: The Swedish experience. *Journal of marketing*, *56*(1), 6-21.

Fornell, C., Johnson, M. D., Anderson, E. W., Cha, J., & Bryant, B. E. (1996). The American customer satisfaction index: nature, purpose, and findings. *Journal of marketing*, *60*(4), 7-18.

Forte Group. (2022, April 2). *Metrics that matter: How you should be measuring Agile*. Retrieved July 3, 2022, from https://fortegrp.com/how-you-should-be-measuring-Agile/

Furlong, K. (2019, August 16). *Burndown Chart vs. Cumulative Flow Diagram (CFD)*. Excella. Retrieved July 4, 2022, from https://www.excella.com/insights/burndown-chart-vs-cumulative-flow-diagram-cfd

Gareis, R. (2007). Management of the project-oriented company. *The Wiley Guide to Project, Program & Portfolio Management, Hoboken (NJ)*, 250-270.

Geckoboard. (n.d.). *Customer Satisfaction (CSAT) - KPI Example*. https://www.geckoboard.com/best-practice/kpi-examples/customer-satisfaction-csat/

Ghalayini, A. M., & Noble, J. S. (1996). The changing basis of performance measurement. *International journal of operations & production management*.

Grady, R. B. (1992). *Practical software metrics for project management and process improvement*. Prentice-Hall, Inc..

Grigoroudis, E., & Siskos, Y. (2004). A survey of customer satisfaction barometers: Some results from the transportation-communications sector. *European Journal of Operational Research*, *152*(2), 334-353.

Gustavsson, T. (2016). Benefits of Agile project management in a non-software development context: A literature review. In *Fifth International Scientific Conference on Project Management in the Baltic Countries, April 14-15, 2016, Riga, University of Latvia* (pp. 114-124). Latvijas Universitate.

Haaften, R. van. (n.d.). *The European Customer Satisfaction Index (ECSI)*. Van Haaften. Retrieved July 9, 2022, from https://www.van-haaften.nl/customer-satisfaction/customer-satisfaction-models/61-the-european-customer-satisfaction-index

Hancer, M., & George, R. T. (2003). Job satisfaction of restaurant employees: An empirical investigation using the Minnesota Satisfaction Questionnaire. *Journal of Hospitality & Tourism Research*, *27*(1), 85-100.

Hartmann, D., & Dymond, R. (2006, July). Appropriate Agile measurement: using metrics and diagnostics to deliver business value. In *AGILE 2006 (AGILE'06)* (pp. 6-pp). IEEE.

Harvard Business Review. (2021, October 11). *You Can't Sit Out Office Politics*. Retrieved December 21, 2022, from https://hbr.org/2021/07/you-cant-sit-out-office-politics

Haug, A., & Arlbjørn, J. S. (2011). Barriers to master data quality. *Journal of Enterprise Information Management*.

Hayes, W., Miller, S., Lapham, M. A., Wrubel, E., & Chick, T. (2014). *Agile metrics: Progress monitoring of Agile contractors*. CARNEGIE-MELLON UNIV PITTSBURGH PA SOFTWARE ENGINEERING INST.

Heigham, J., & Croker, R. (Eds.). (2009). *Qualitative research in applied linguistics: A practical introduction*. Springer.

Herr, K., & Anderson, G. L. (2014). *The action research dissertation: A guide for students and faculty*. Sage publications.

Highsmith, J. (2006). An adaptive performance management system. *Cutter Consortium Agile Project Management Executive Summary*, *7*(6), 63-74.

Hoda, R. (2011). Self-organizing Agile teams: A grounded theory.

Hoda, R., Noble, J., & Team Fhall, S. (2008, April). Agile project management. In *New Zealand computer science research student conference* (Vol. 6, pp. 218-221).

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

Hsu, S. H. (2008). Developing an index for online customer satisfaction: Adaptation of American Customer Satisfaction Index. *Expert systems with Applications*, *34*(4), 3033-3042.

Huang, H. B. (2010). What is good action research. *Action research*, *8*(1), 93-109.

Ivory, M. Y., Sinha, R. R., & Hearst, M. A. (2001, March). Empirically validated web page design metrics. In *Proceedings of the SIGCHI conference on Human factors in computing systems* (pp. 53-60).

Javdani, T., Zulzalil, H., Ghani, A. A. A., Sultan, A. B. M., & Parizi, R. M. (2013). On the current measurement practices in Agile software development. *arXiv preprint arXiv:1301.5964*.

Joshi, A., Kale, S., Chandel, S., & Pal, D. K. (2015). Likert scale: Explored and explained. *British journal of applied science & technology*, *7*(4), 396.

Justin, E., & Joy, M. M. (2022). Managing the most important asset: a twenty year review on the performance management literature. *Journal of Management History*.

Kanbanize. (n.d.). *Introducing the 6 Agile Metrics to Boost Work Delivery and Efficiency*. Kanban Software for Agile Project Management. Retrieved August 5, 2022, from https://kanbanize.com/Agile/project-management/Agile-metrics

Kanji, G. K. (2002). Business excellence: make it happen. *Total Quality Management*, *13*(8), 1115-1124.

Kaplan, R. S. (2009). Conceptual foundations of the balanced scorecard. *Handbooks of management accounting research*, *3*, 1253-1269.

Kaplan, R. S., & Norton, D. P. (2005). The balanced scorecard: measures that drive performance. *Harvard business review*, *83*(7), 172.

Kassab, M. (2015, August). The changing landscape of requirements engineering practices over the past decade. In *2015 IEEE fifth international workshop on empirical requirements engineering (EmpiRE)* (pp. 1-8). IEEE.

Kaur, A., & Bahl, K. (2015). Analysis of Agile Project Management practice and Comparison with Traditional Project Management. *International Journal of Innovative Science, Engineering & Technology*, *2*(5), 887-889.

Kemmis, S., McTaggart, R., & Nixon, R. (2014). The action research planner: Doing critical participatory action research.

Kennerley, M., & Neely, A. (2002). Performance measurement frameworks: a review. *Business performance measurement: Theory and practice*, *2*(3), 145-155.

Khurum, M., Gorschek, T., & Wilson, M. (2013). The software value map—an exhaustive collection of value aspects for the development of software intensive products. *Journal of software: Evolution and Process*, *25*(7), 711-741.

Kinicki, A. J., McKee-Ryan, F. M., Schriesheim, C. A., & Carson, K. P. (2002). Assessing the construct validity of the job descriptive index: a review and meta-analysis. *Journal of applied psychology*, *87*(1), 14.

Kišš, F., & Rossi, B. (2018, September). Agile to lean software development transformation: A systematic literature review. In *2018 Federated Conference on Computer Science and Information Systems (FedCSIS)* (pp. 969-973). IEEE.

Kissflow, Inc. (2022, August 24). *The Complete Guide to Cumulative Flow Diagram*. https://kissflow.com/project/Agile/guide-to-cumulative-flow-diagram/

Kohlbacher, M., Stelzmann, E., & Maierhofer, S. (2011, April). Do Agile software development practices increase customer satisfaction in Systems Engineering projects?. In *2011 IEEE International Systems Conference* (pp. 168-172). IEEE.

Korpivaara, I., Tuunanen, T., & Seppänen, V. (2021). Performance measurement in scaled Agile organizations. In *Proceedings of the Annual Hawaii International Conference on System Sciences*. University of Hawai'i at Manoa.

Kristensen, K., Martensen, A., & Gronholdt, L. (2000). Customer satisfaction measurement at post Denmark: results of application of the European customer satisfaction index methodology. *Total Quality Management*, *11*(7), 1007-1015.

Ktata, O., & Lévesque, G. (2010, May). Designing and Implementing a Measurement Program for Scrum Teams: What do Agile developers really need and want?. In *Proceedings of the Third C* Conference on Computer Science and Software Engineering* (pp. 101-107).

Kupiainen, E., Mäntylä, M. V., & Itkonen, J. (2015). Using metrics in Agile and Lean Software Development–A systematic literature review of industrial studies. *Information and software technology*, *62*, 143-163.

Kwak, Y. H. (2005). A brief history of project management. *The story of managing projects: An interdisciplinary approach*, 1-9.

Laanti, M. (2017, May). Agile transformation model for large software development organizations. In *Proceedings of the XP2017 Scientific Workshops* (pp. 1-5).

Lappi, T., Karvonen, T., Lwakatare, L. E., Aaltonen, K., & Kuvaja, P. (2018). Toward an improved understanding of Agile project governance: A systematic literature review. *Project Management Journal*, *49*(6), 39-63.

Lárusdóttir, M. K., Cajander, Å., & Simader, M. (2014, September). Continuous improvement in Agile development practice. In *International Conference on Human-Centred Software Engineering* (pp. 57-72). Springer, Berlin, Heidelberg.

Lebas, M. J. (1995). Performance measurement and performance management. *International journal of production economics*, *41*(1-3), 23-35.

Leffingwell, D. (2021, September 27). *Agile Teams*. Scaled Agile Framework. https://www.scaledAgileframework.com/Agile-teams/

Levison, M. (2009). What is a Good Agile Metric.

Lim, Maarten. The Impact of Agile transformations on the success of organizations, [Unpublished Thesis Master ICT in Business and the Public Sector], LIACS, Leiden University, 2022.

Lockett, M., De Reyck, B., & Sloper, A. (2008). Managing project portfolios. *Business Strategy Review*, *19*(2), 77-83.

Lohan, G., Conboy, K., & Lang, M. (2010). Beyond Budgeting and Agile software development: A conceptual framework for the performance management of Agile software development teams.

Lohan, G., Conboy, K., & Lang, M. (2010, October). Beyond budgeting: A performance management model for software development teams. In *International Conference on Lean Enterprise Software and Systems* (pp. 126-138). Springer, Berlin, Heidelberg.

Lohan, G., Lang, M., & Conboy, K. (2013). A performance management model for Agile information systems development teams. In *Building sustainable information systems* (pp. 297-308). Springer, Boston, MA.

Malina, M. A., & Selto, F. H. (2001). Communicating and controlling strategy: an empirical study of the effectiveness of the balanced scorecard. *Journal of management accounting research*, *13*(1), 47-90.

Mann, C., & Maurer, F. (2005, July). A case study on the impact of scrum on overtime and customer satisfaction. In *Agile Development Conference (ADC'05)* (pp. 70-79). IEEE.

Manning, F. J. (1991). Morale, Cohesion, and Esprit de Corps. In Gal, R. & Mangelsdorff, A. D. (Eds.), Handbook of Military Psychology (pp. 453–470). Chichester: John Wiley & Sons Ltd

Marchesi, M., & Succi, G. (2003). Extreme programming and Agile processes in software engineering. *Proceedings of XP*.

Mas, A., Mesquida, A. L., & Pacheco, M. (2020). Supporting the deployment of ISO-based project management processes with Agile metrics. *Computer Standards & Interfaces*, *70*, 103405.

Matrix Resources. (2014, July). *Cumulative Flow Diagrams: An Alternative to Burndown Charts*. MATRIX Resources. Retrieved July 4, 2022, from https://www.matrixres.com/blog/2014/07/cumulative-flow-diagrams-an-alternative-to-burndown-charts?source=google.com

Melnikovas, A. (2018). Towards an explicit research methodology: Adapting research onion model for futures studies. *Journal of Futures Studies*, *23*(2), 29-44.

Mertler, C. A. (2009). *Action research: Teachers as researchers in the classroom*. Sage.

Meyer, M. W. (2003). *Rethinking performance measurement: Beyond the balanced scorecard*. Cambridge University Press.

Microsoft Learn. (2022, October 4). *What are Analytics views? - Azure DevOps*. Retrieved July 4, 2022, from https://learn.microsoft.com/en-us/azure/devops/report/powerbi/what-are-analytics-views?view=azure-devops

Microsoft. (n.d.). *What Is Power BI? Definition and Features | Microsoft Power BI*. Retrieved September 7, 2022, from https://powerbi.microsoft.com/en-us/what-is-power-bi/

Mohpman JR, A. M., & Mohrman, S. A. (1995). Performance Management Is" Running the Business". *Compensation & Benefits Review*, *27*(4), 69-75.

Mounsey, S. M. (2017, January 6). *Backlog Health Check*. Sketch and Scrum. Retrieved August 4, 2022, from https://sketchandscrum.com/2016/12/29/backlog-health-check/

Neely, A. (1999). The performance measurement revolution: why now and what next?. *International journal of operations & production management*.

Neely, A., Adams, C., & Crowe, P. (2001). The performance prism in practice. *Measuring business excellence*.

Neely, A., Kennerley, M., & Martinez, V. (2004). Does the balanced scorecard work: an empirical investigation. *Performance measurement and management: public and private, Centre for Business Performance, Cranfield University, Cranfield*, 763-70.

Neely, A., Mills, J., Platts, K., Richards, H., Gregory, M., Bourne, M., & Kennerley, M. (2000). Performance measurement system design: developing and testing a process-based approach. *International journal of operations & production management*.

NEF Consulting. (2021, May 13). *EFQM Excellence Model*. Retrieved August 8, 2022, from https://www.nefconsulting.com/training-capacity-building/resources-and-tools/efqm-model/

Nirpal, P. B., & Kale, K. V. (2011). A brief overview of software testing metrics. *International Journal on Computer Science and Engineering*, *3*(1), 204-2011.

Nudurupati, S. S., Bititci, U. S., Kumar, V., & Chan, F. T. (2011). State of the art literature review on performance measurement. *Computers & Industrial Engineering*, *60*(2), 279-290.

Nzuve, S., & Nyaega, G. (2013). Application of balanced scorecard in performance measurement at Essar Telecom Kenya limited. *Available at SSRN 2231330*.

Otley, D. (1999). Performance management: a framework for management control systems research. *Management accounting research*, *10*(4), 363-382.

Otley, D. (2008). Performance management: a framework for analysis. In *Performance Management* (pp. 24-39). Palgrave Macmillan, London.

Oza, N., & Korkala, M. (2012). Lessons learned in implementing Agile software development metrics.

Öztayşi, B., & Uçal, Ǐ. (2009, July). Comparing MADM techniques for use in performance measurement. In *ISAHP 2009 Proceedings. The 10th International Symposium on the Analytic Hierarchy Process* (Vol. 29, pp. 53-64).

Padmini, K. J., Bandara, H. D., & Perera, I. (2015, April). Use of software metrics in Agile software development process. In *2015 Moratuwa Engineering Research Conference (MERCon)* (pp. 312-317). IEEE.

Paul, R. A., Kunii, T. L., Shinagawa, Y., & Khan, M. F. (1999). Software metrics knowledge and databases for project management. *IEEE Transactions on Knowledge and Data Engineering*, *11*(1), 255-264.

Pfeiffer, M. (2021, December 21). *12 DevOps KPIs you should track to gauge improvement*. IT Operations. https://www.techtarget.com/searchitoperations/tip/Nine-DevOps-metrics-you-should-use-to-gauge-improvement

Pignatiello, G. A., Martin, R. J., & Hickman Jr, R. L. (2020). Decision fatigue: A conceptual analysis. *Journal of health psychology*, *25*(1), 123-135.

Project Management Institute.(2013). A guide to the project management body of knowledge (PMBOK guide) (5th ed.). Newtown Square, PA: Author.

Przybyłek, A., & Kotecka, D. (2017, September). Making Agile retrospectives more awesome. In *2017 Federated Conference on Computer Science and Information Systems (FedCSIS)* (pp. 1211-1216). IEEE.

Pulakos, E. D. (2004). *Performance management: A roadmap for developing, implementing and evaluating performance management systems* (pp. 1-42). Alexandria, VA: SHRM foundation.

Pulford, K., Kuntzmann-Combelles, A., & Shirlaw, S. (Eds.). (1995). *A quantitative approach to software management: the AMI handbook*. Addison-Wesley Longman Publishing Co., Inc..

Racheva, Z., Daneva, M., & Sikkel, K. (2009, June). Value creation by Agile projects: Methodology or mystery?. In *International Conference on Product-Focused Software Process Improvement* (pp. 141-155). Springer, Berlin, Heidelberg.

Ram, P., Rodríguez, P., Oivo, M., Martínez-Fernández, S., Bagnato, A., Choraś, M., ... & Ahola, M. (2020). Actionable software metrics: an industrial perspective. In *Proceedings of the Evaluation and Assessment in Software Engineering* (pp. 240-249).

Raslan, A. T., Darwish, N. R., & Hefny, H. A. (2015). Towards a fuzzy based framework for effort estimation in Agile software development. *International Journal of Computer Science and Information Security*, *13*(1), 37.

Root-Bernstein, R. S. (1982). The problem of problems. *Journal of Theoretical Biology*, *99*(1), 193-201.

Sabol, A., & Sverer, F. (2017). A review of the economic value added literature and application. *UTMS Journal of Economics*, *8*(1), 19-27.

Santiago Rivera, D., & Shanks, G. (2015). A dashboard to support management of business analytics capabilities. *Journal of Decision Systems*, *24*(1), 73-86.

Saunders, M., Lewis, P., & Thornhill, A. (2009). *Research methods for business students*. Pearson education.

Schwaber, K., & Sutherland, J. (2011). The scrum guide. *Scrum Alliance*, *21*(19), 1.

Scott-Hill, B., Anslow, C., Ferreira, J., Kropp, M., Mateescu, M., & Meier, A. (2020, August). Visualizing progress tracking for software teams on large collaborative touch displays. In *2020 IEEE Symposium on Visual Languages and Human-Centric Computing (VL/HCC)* (pp. 1-5). IEEE.

Scribbr. (2022, July 15). *What is a Likert scale?* https://www.scribbr.com/frequently-asked-questions/what-is-a-likert-scale/

Scrum.org. (2020). The Evidence-Based Management Guide: Measuring Value to Enable Improvement and Agility. In *Scrum.org*. Retrieved July 24, 2022, from https://scrumorg-website-prod.s3.amazonaws.com/drupal/2020-12/EBM%20Guide%202020_1.pdf?nexus-file=https%3A%2F%2Fscrumorg-website-prod.s3.amazonaws.com%2Fdrupal%2F2020-12%2FEBM%2520Guide%25202020_1.pdf

Scrum.org. (n.d.). *The Scrum Framework Poster*. Retrieved July 10, 2022, from
https://www.scrum.org/resources/scrum-framework-poster

Scrum.org. (n.d.-a). *Agile Metrics: Velocity*. Retrieved July 9, 2022, from
https://www.scrum.org/resources/blog/Agile-metrics-velocity

Scrum.org. (n.d.-b). *Using Velocity as a Management Instrument? Don't Do It!* Retrieved July 4, 2022,
from https://www.scrum.org/resources/blog/using-velocity-management-instrument-dont-
do-it

SeaLights. (2019, October 15). *10 Powerful Agile Performance Metrics - and 1 Missing Metric*.
Sealights. https://www.sealights.io/software-development-metrics/10-powerful-Agile-
metrics-and-1-missing-metric/

Seymour, T., & Hussein, S. (2014). The history of project management. *International Journal of
Management & Information Systems (IJMIS)*, *18*(4), 233-240.

Sharma, S., Sarkar, D., & Gupta, D. (2012). Agile processes and methodologies: A conceptual study.
International journal on computer science and Engineering, *4*(5), 892.

Shujaat, S., Manzoor, A., & Syed, N. A. (2014). Impact of team work on employee satisfaction. *IBT
Journal of Business Studies (JBS)*, *2*(2).

Siderova, S. (2022, 29 juli). *From supermarkets to software: History of Kanban*. Nave Blog - Expert
tips and guidelines for Kanban teams. https://getnave.com/blog/kanban-history/

Silvius, G. (2021). The role of the project management office in sustainable project management.
Procedia Computer Science, *181*, 1066-1076.

Sokolov, D. (2022, October 19). *Azure DevOps Tutorial: Agile Project Management*. PPM Express.
https://ppm.express/blog/azure-devops-tutorial-Agile-project-management/

Stanton, J. M., Sinar, E. F., Balzer, W. K., Julian, A. L., Thoresen, P., Aziz, S., ... & Smith, P. C. (2002).
Development of a compact measure of job satisfaction: The abridged Job Descriptive Index.
Educational and psychological measurement, *62*(1), 173-191.

Stettina, C. J., & Heijstek, W. (2011, June). Five Agile factors: Helping self-management to self-reflect.
In *European Conference on Software Process Improvement* (pp. 84-96). Springer, Berlin,
Heidelberg.

Stettina, C. J., Els, V. V., Croonenberg, J., & Visser, J. (2021, June). The Impact of Agile
Transformations on Organizational Performance: A Survey of Teams, Programs and
Portfolios. In *International Conference on Agile Software Development* (pp. 86-102).
Springer, Cham.

Stoica, M., Ghilic-Micu, B., Mircea, M., & Uscatu, C. (2016). Analyzing Agile development-from
waterfall style to scrumban. *Informatica Economica*, *20*(4), 5.

Striteska, M., & Spickova, M. (2012). Review and comparison of performance measurement systems.
Journal of Organizational Management Studies, *2012*, 1.

Susman, G. I., & Evered, R. D. (1978). An assessment of the scientific merits of action research. *Administrative science quarterly*, 582-603.

TestMatick. (n.d.). *Defect Removal Efficiency*. Retrieved July 6, 2022, from https://testmatick.com/software-testing-glossary/defect-removal-efficiency/

The Liberators. (2022). *Start a survey*. Retrieved August 8, 2022, from https://teammetrics.theliberators.com/Sample/Create?reason=EBM

Tripp, J. F., Riemenschneider, C., & Thatcher, J. B. (2016). Job satisfaction in Agile development teams: Agile development as work redesign. *Journal of the Association for Information Systems*, *17*(4), 1.

Trivedi, D. (2021). Agile methodologies. *International Journal of Computer Science & Communication*, *12*(2), 91-100.

Um, K. H., Fang, Y., & Seo, Y. J. (2021). The ramifications of covid-19 in education: beyond the extension of the kano model and unipolar view of satisfaction and dissatisfaction in the field of blended learning. *Educational Measurement: Issues and Practice*, *40*(3), 96-109.

University of Central Missouri. (n.d.). *Guides: Action Research: Getting Started*. Retrieved August 2, 2022, from https://guides.library.ucmo.edu/c.php?g=879745

Vacanti, D. S. (2014). Actionable Agile Metrics for Predictability.

Vacanti, D., & Vallet, B. (2014). Actionable Metrics at Siemens Health Services.

Van De Weerd, I., Brinkkemper, S., Nieuwenhuis, R., Versendaal, J., & Bijlsma, L. (2006, September). Towards a reference framework for software product management. In *14th IEEE International Requirements Engineering Conference (RE'06)* (pp. 319-322). IEEE.

Verbruggen, F., Sutherland, J., van der Werf, J. M., Brinkkemper, S., & Sutherland, A. (2019, January). Process efficiency-Adapting flow to the Agile improvement effort. In *Proceedings of the 52nd Hawaii International Conference on System Sciences*.

Verwijs, C. (2020, October 15). *Agile Teams: Don't use happiness metrics, measure Team Morale*. Medium. https://medium.com/the-liberators/Agile-teams-dont-use-happiness-metrics-measure-team-morale-3050b339d8af

Verwijs, C. (2022, May 10). *In-Depth: What Makes Scrum Teams Effective? (v1.2) - The Liberators*. Medium. Retrieved August 5, 2022, from https://medium.com/the-liberators/what-makes-scrum-teams-effective-7ca9f56a82c7

Verwijs, C. L. T. (n.d.). *Scrum Team Survey*. https://questionnaire.scrumteamsurvey.org/initiate

Verwijs, C., & Russo, D. (2021). A theory of scrum team effectiveness. *arXiv preprint arXiv:2105.12439*.

Wall, T. D., Clegg, C. W., & Jackson, P. R. (1978). An evaluation of the job characteristics model. *Journal of Occupational Psychology*, *51*(2), 183-196.

Weerd, I., Brinkkemper, S., Nieuwenhuis, R., Versendaal, J. M., & Bijlsma, A. (2006). A reference framework for software product management. *Technical report UU-CS*, *2006*(014).

Weiss, D. J., Dawis, R. V., & England, G. W. (1967). Manual for the Minnesota satisfaction questionnaire. *Minnesota studies in vocational rehabilitation*.

Woodward, C. J., Cain, A., Pace, S., Jones, A., & Kupper, J. F. (2013, August). Helping students track learning progress using burn down charts. In *Proceedings of 2013 IEEE International Conference on Teaching, Assessment and Learning for Engineering (TALE)* (pp. 104-109). IEEE.

Yazdanpanah, M., & Feyzabad, F. R. (2017). Investigating Iranian farmers' satisfaction with agricultural extension programs using the American customer satisfaction index. *Journal of Agricultural & Food Information*, *18*(2), 123-135.

Zhang, B., Vos, M., Veijalainen, J., Wang, S., & Kotkov, D. (2016). The issue arena of a corporate social responsibility crisis: The Volkswagen case in Twitter. *Studies in Media and Communication*, *4*(2).

Appendix A – Mapping of Performance Management and Measurement frameworks

	Name of the framework/model ↓	Focus of the framework/model ↓	Productivity "Total output divided by total input"	Responsiveness "Quickness of response to either customer or market demand"	Quality "Measure of excellence, both product and development quality"	Workflow health "The way that work is organized, or the sequence of steps in a work process".	Employee satisfaction & engagement	Customer satisfaction	Financial
Performance Management & Measurement Frameworks	Economic Value Added (EVA)	Financial performance measure	How is Productivity covered? -	How is Responsiveness covered? -	How is Quality covered? -	-	How is Employee satisfaction & engagement covered? -	How is customer satisfaction covered? -	The EVA is a calculation of financial performance.
	Balanced Score Card (BSC)	Q1: To achieve our vision, how should we appear to our customers? Q2: To satisfy our shareholders and customers, what business processes must we excel at? Q3: To achieve our vision, how will we sustain our ability to change and improve? Q4: To succeed financially, how should we appear to our shareholders?	How is Productivity covered?	How is Responsiveness covered?	How is Quality covered?	How is workflow health covered?	How is Employee satisfaction & engagement covered?	How is customer satisfaction covered?	How is Financial covered?
	Software Value Map (SVM)	Value Perspectives (similar to BSC); each perspective has a cluster of value aspects (VA); each VA has multiple subvalue aspect (SVAs); each SVA has end nodes called value components (VCs)	How is Productivity covered?	How is Responsiveness covered?	How is Quality covered?	How is workflow health covered?	How is Employee satisfaction & engagement covered?	How is customer satisfaction covered?	How is Financial covered?
	Cicek Team performance model (CTPM)	Structure, Input, Process and Output that affect team performance	How is Productivity covered?	How is Responsiveness covered?	How is quality covered?	How is workflow health covered?	How is employee satisfaction covered?	How is customer satisfaction covered?	How is Financial covered?
	Keegan's perf. Measurement matrix (KPMM)	Categorizes measures as being 'cost' or 'non-cost', and 'external' or 'internal'. Allows organizations to plot measures and identify possible adjustment in measurement focus.	How is Productivity covered?	How is Responsiveness covered?	How is Quality covered?	How is Workflow health covered?	How is employee satisfaction & engagement covered?	How is customer satisfaction covered?	How is Financial covered?
	Fitzgerald framework of per. Measurement (FPPM)	Measures classified as results (financial performance, competitiveness) and determinants (quality, flexibility, resource utilisation, innovation)	How is productivity covered?	How is Responsiveness covered?	How is quality covered?	How is workflow health covered?	How is employee satisfaction & engagement covered?	How is customer satisfaction covered?	How is Financial covered?
	SMART Pyramid (SMARTP)	Management control system with performance indicators to define and sustain success. Four distinct hierarchies in which are components one can measure (e.g. financial and market measures)	How is productivity covered?	How is responsiveness covered?	How is quality covered?	How is workflow health covered?	How is employee satisfaction & engagement covered? Not covered.	How is customer satisfaction covered?	How is Financial covered?
	Brown's Macro Process Model (BMPM)	Five stages in a business process with respective measures of their performance: Input measures, Process measures, Output measures, Outcome Measures, Goal	How is Productivity covered?	How is Responsiveness covered?	How is Quality covered?	How is workflow health covered?	How is employee satisfaction & engagement covered?	How is customer satisfaction covered?	How is Financial covered?
	Performance Prism (PP)	Stakeholders are central to performance measurement. Several areas affect stakeholder contribution: Strategies, Processes, Capabilities	How is Productivity covered?	How is Responsiveness covered?	How is Quality covered?	How is workflow health covered?	How is employee satisfaction & engagement covered?	How is customer satisfaction covered?	How is Financial covered?
	EFQM Business Excellence Model (EFQMBEM)	Self-assessment framework for measuring the strengths and possible areas for improvement of an organization. There are enablers (leadership, people, policy & strategy, partnerships & resources, processes) and results (people results, customer results, society results, key performance results).	How is Productivity covered?	How is Responsiveness covered?	How is Quality covered?	How is workflow health covered?	How is employee satisfaction & engagement covered?	How is customer satisfaction covered? One of the eight fundamental concepts of the model is called 'customer focus'.	How is Financial covered?
	Kanji Business Excellence Measurement System (KBEMS)	This framework is used for the measurement of performance from the point of view of internal stakeholders and performance from external stakeholder perspective. Internal stakeholders (leadership, customer focus, management by facts, improvement of process, people-based management, people performance, continous improvement, culture of continous improvement, performance excellence) and external stakeholders (organizational values, process excellence delight the stakeholders, performance excellence)	How is Productivity covered?	How is Responsiveness covered?	How is Quality covered?	How is workflow health covered?	How is employee satisfaction & engagement covered?	How is customer satisfaction covered?	How is Financial covered?
	Evidence Based Management Guide (EBMG)	EBMG specifies four key value aspects (KVA): Time to Market, Ability to Innovate, Unrealized Value, Current Value. Each KVA has multiple key value measures	How is Productivity covered?	How is Responsiveness covered?	How is Quality covered?	How is Workflow health covered?	How is employee satisfaction & engagement covered?	How is customer satisfaction covered?	How is Financial covered?

Figure 27: Mapping of Performance Management and Measurement Frameworks

Appendix B – Agile metrics & visualizations

Overview of the explored Agile metrics

Multiple metrics or measurement practices are explained in detail using the following structure:

- **What does it show?:** A short explanation of what a specific measure signifies and how it is used in practice by Agile teams.
- **Merits**: Explains what the benefits are of using this metric, how could the team potentially use this to e.g. improve their performance.
- **Pitfalls & considerations**: Just because a metric could be very useful, does not mean it should always be used. There are risks to be considered, for example wrongly interpreting results or using a metric to compare teams which is discouraged.

Velocity

What does it show?

Velocity is commonly used instead of productivity in Agile contexts (Cohn, 2005; Javdani et al., 2013). Velocity is defined as "*an indication of the average amount of product backlog turned into an increment of a product during a sprint by a scrum team*" (Scrum.org, n.d.-a). As one can see, velocity does not mean productivity, but instead it is used to gauge the productivity of an Agile team in practice.

Merits

One of the goals of productivity is that a constant (or improving) pace is maintained by an individual or the team. The maintaining of a constant pace is usually measured with velocity (Davis, 2015). Velocity can also be used to predict the team's output in future sprints (SeaLights, 2019). Velocity can also be used to compare the committed and completed work in sprints, as seen in figure 28 (Atlassian, n.d.-a):
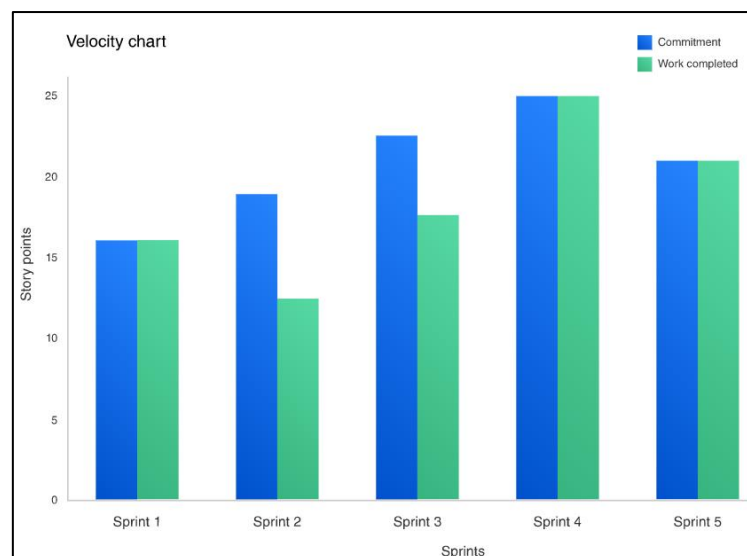
Figure 28: Velocity chart (Atlassian, n.d.-a)

Pitfalls & considerations

The pitfall of using velocity as a measure of a team's productivity is that it may lead to comparisons of velocity between teams. Each team has its own unique composition of team members, expertise, experience and objectives. All these factors are reflected in the team's velocity, making it unique to that specific team. (Atlassian, n.d.-a)

Even when velocity is not used to compare different teams, velocity remains an empirical observation and is not an estimation or target to aim for it. (Scrum.org, n.d.-b)

One possible way to control for varying velocity of each sprint, is to use the coefficient of variation (= (standard deviation / average*100). This provides information about the stability of the velocity; an indication the team found a stable velocity. (Hayes et al., 2014)

Throughput

What does it show?

Throughput signifies the (average) number of work items that are completed per unit of time (e.g. the amount of user stories finished per sprint). This metric is different from velocity; velocity measures the story points per iteration, while throughput measures the number of work items finished per unit of time (Agile Alliance, 2021).

Merits

The challenge in using velocity is the fact that story points are based on the team's unique dynamic to 'guesstimate' story points to user stories. If the size and complexity of the user stories or tasks are relatively the same, throughput provides an alternative view of how productive the team is.

Pitfalls & considerations

A consideration would be that not all user stories are created the same; some user stories are guesstimated to be 5 story points and others 3 story points. Looking at purely the number of work items, it will display a skewed view of how productive a team is, since not all work items (e.g. user stories) are of the same complexity and size.

Lead time

What does it show?

One of the ways responsiveness can be measured is with the metric 'lead time per feature' (Boon & Stettina, 2022). Lead time is defined as "the amount of time that passed from a request to fulfilling the request" (Budacu & Pocatilu, 2018). Kišš & Rossi (2018) use a similar definition. Lead time is also one of the most prominent metric in the Kanban method. (Budacu & Pocatilu, 2018)

Merits

Lead time provides the Agile team with information on how responsive they are; if a customer asks for a certain feature to be implemented, does it remain in the backlog for a year before it will be added to the sprint backlog? While cycle time is a component of the lead time, cycle time focuses on when the work on a certain feature actually starts and ends, without taking into account when the task was created in the backlog (Budacu & Pocatilu, 2018). Thus, lead time is a measure of responsiveness, while cycle time is a measure of productivity. Additionally, lead time may assist teams in understanding how Agile they actually are (Verbruggen et al., 2019).

Pitfalls & considerations

Lead time is useful and provides the team with a zoomed out picture on how long the team takes to go from 'task created' to 'work completed'. However, each user story is unique and has a different

story point estimation assigned to it. For example, table 20 is partly sourced from (Budacu & Pocatilu, 2018):

Table 20: Story Point Estimation vs. Cycle Time (Budacu & Pocatilu, 2018)

Story point estimate	User stories count	Avg. cycle time (work days)
1	36	12.5
3	43	22.39
5	34	22.32
8	25	28.76
13	8	48

When only looking at the average cycle time of all the user stories, the team would see a cycle time of 29.79 (average of the last column). While perhaps interesting to know, it does not provide the full picture. When a sprint contains only relatively difficult user stories with story points estimates of 8-13 (with corresponding average cycle times of 28-48), the team may get discouraged because of the cycle time of the sprint is far above the average of 29.79. So it is important consider the cycle time in combination with story points to produce a more accurate view of how long a user story will take to complete.

Cycle time

What does it show?

While cycle time was briefly mentioned when discussed as a component of lead time, it is a relevant metric to explore in the context of productivity. Cycle time is defined as "the amount of time that passed from when work actually started to fulfilling the request" (Budacu & Pocatilu, 2018). When cycle time is measured, only the time spent actually working on a task counts (Verbruggen et al., 2019). It is a given that teams must keep track of when they start to work on a user story and when it is finished.

Merits

A historical view of cycle time per user story can be used when the customer or management makes the inquiry of how long on average it takes for a certain feature to be implemented if the team would start immediately. Understanding the cycle time will enable the team to communicate more effectively with their stakeholders and manage expectations of the team and stakeholders better. It also provides a starting point to continuously improve upon; long cycle times may facilitate discussion on why cycle times are so long and especially how the team can bring this number down.

A possible visualization of the cycle time can be done through the control chart. This chart in figure 29 (Atlassian, n.d.-a) focuses on the cycle time of bugs; it is an easy way to improve processes of the team. (Atlassian, n.d.-a)
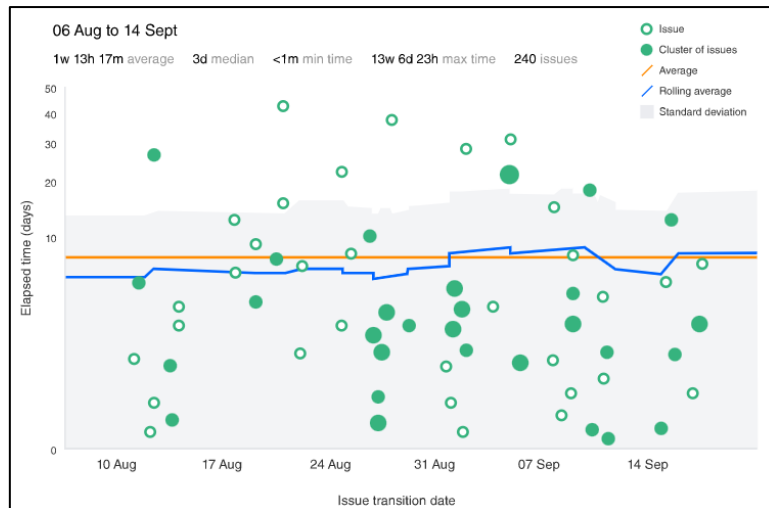
Figure 29: A Control Chart for bugs (Atlassian, n.d.-a)

Pitfalls & considerations

As already mentioned, cycle time is measured in working hours. Because Agile team members are not immune to interruptions such as meetings and answering e-mails, it may prove difficult to track the actual hours worked on a certain task. If a developer is constantly interrupted and loses focus, would (s)he count this as working hours?

Another consideration would be hours spent is not synonymous to being productive. Someone working 8 hours on a user story with half focus would perhaps accomplish as much as someone who would work 3 hours on the same user story with their full attention and focus.

While this is inevitable and part of their work, this cycle time should not be viewed as absolute and more as an estimate in order to prevent the aforementioned pitfalls.

Figure 30 (Budacu & Pocatilu, 2018) highlights the difference between cycle time and lead time. Lead time starts from when the request has been submitted to the product backlog, and finishes when the request has been finished/fulfilled. Contrarily, cycle time only starts when the actual work is begun on the request/feature, and ends when the request is fulfilled. Figure 30 thus shows how cycle time is part of the lead time, but lead time also includes potential idle time since a request could stay in the product backlog for a while before it gets prioritized into the sprint backlog.
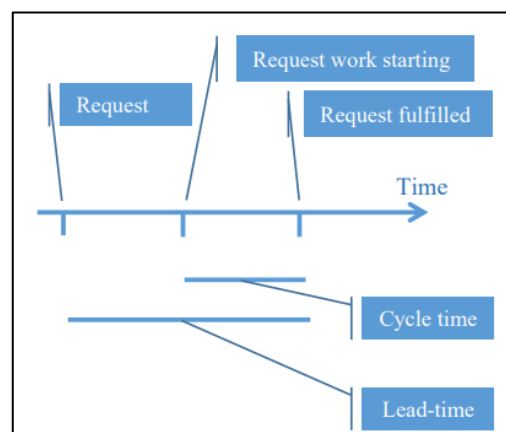
Figure 30: The difference between lead time and cycle time (Budacu & Pocatilu, 2018)

Flow efficiency

What does it show?

Flow efficiency is calculated based on the amount of time a task spends in 'working' statuses and the amount of time it spends in 'waiting' statuses. The formula to calculate this is (Work time / (work time + wait time)) * 100%. A flow efficiency is deemed as good from 40% and upwards (Mas et al., 2020).

Merits

This metric is useful understand the team's efficiency. It creates awareness within the team about interruptions and disruptions that prevent them from working on tasks. (Mas et al., 2020)

Pitfalls & considerations

To optimally and correctly use this metric, the team has to monitor when they are working on value-added tasks and when no active work is done. These variables are cumbersome to track and result in administrative work which can make the team adverse to using this metric.

Time to remove impediment/mean time to repair

What does it show?

The time to remove an impediment/mean time to repair is the average amount of time it takes from when an error is detected to when it is fixed (Scrum.org, 2020). This is similar to the lead time, however the lead time in the previous section was focused on features and user stories, and this is focused on the impediments or bugs.

Merits

Using the mean time to repair helps reveal the efficiency of a team to fix an error. This is of course directly related to customer satisfaction; how long does a customer have to wait from raising a trouble report to having it resolved? Perhaps there are impediments identified months ago but never made it to the sprint backlog due to forgetfulness. This metric will help teams improve their efficiency to fix an error.

Pitfalls & considerations

"Not all problems are created equal" (Root-Bernstein, 1982); just because a trouble report has been identified months ago, does not mean automatically this is cause for concern. Each problem identified has its own risk, severity and urgency. Thus, it is important to consider these aspects when visualising the time to remove impediments in an Agile team.

Open defect severity

What does it show?

Not all defects are created equal. To help the team with the prioritization of fixing the open defects, it is useful to classify the open bugs with some sort of categorization. This way, the team has a better understanding of how many open bugs are critical and how many are more related to quality of life.

Merits

A visualization of the open defects and their severity provides teams with the ability to prioritize if there are too many defects that require corrective work. Using categorizations (e.g. from a scale of 1 to 5), the team will be able to see how many critical defects are still in the backlog or how many moderate defects are waiting for their respective solution to be approved by stakeholders.

Pitfalls & considerations

Providing Agile teams with such an overview has many benefits outlined above. Since the categorization will be used to create a sense of urgency, one should be cautious when assigning this prioritization to the defects. For example, who will decide whether an identified defect has a severity level 4 instead of 3 or 5? This should always be communicated with the customer and the team to manage the expectations of stakeholders and workload of the team.

Burndown/up chart

What does it show?

The burndown chart is a tool to help in planning and monitoring of the work progress (Scrum.org, n.d.-a); it is used to present the amount of remaining work (Cohn, 2005). Three types of burndown charts are commonly used (Cervone, 2011):

- Sprint burndown chart; documents the progress of the sprint;
- Release burndown chart; documents the progress of the release;
- Product burndown chart; documents the product progress on an overall level.

These three types of burndown charts are represented in terms of time (x-axis) and duration (y-axis) (Cervone, 2011). In the ideal situation, the actual burndown performance would have zero deviation from the ideal one, both in a positive and negative sense.

Merits

A comparison between estimated work (ideal burndown) and remaining work together could help Agile teams to make decisions, for example adding or dropping user stories in case the project is ahead or behind schedule. For example, Arafeen & Bose (2009) have tried to provide indicators and a roadmap on how to analyse burn down chart deviation. This can then be used to create a predictable band that would could potentially improve iteration planning. Burndown charts can also help teams understand if tasks are being completed too quickly or if tasks are too large, causing them to be completed too sluggishly (Scott-Hill et al., 2020).

Pitfalls & considerations

While burndown charts have numerous benefits in identifying a variety of issues (i.e. scope creep, poor estimation, there is a limitation on how many user stories or tasks can be added or removed based on the positive or negative deviation of the sprint burndown chart. A typical sprint lasts from 1 to 4 weeks. The question can then be asked; how useful is it if a team is working with 3-week sprints, to adjust the work in the second week when the burndown chart does not show the desired 'burndown'.

Contrarily, the burndown chart is perhaps more useful for tracking release and product progress. These have longer timelines than a sprint and thus allow for more bandwidth to add or remove work from the original plan. One such application in literature is helping students tracking their learning progress using burndown charts, in a period of 14 weeks (Woodward et al., 2013). This timeline allows for more possible adjustments and actually observing whether an adjustment has an effect. With sprints of 1 to 4 weeks, this is less feasible and useful.

As a team becomes more mature[10], one can make the assumption these teams also become better at estimating the work that is needed to complete user stories/tasks. What should follow is the fact

---

[10] Mature teams have experience in working with multiple Agile projects and/or have more than 12 months of experience using Agile methods (Hoda, 2011)

that the burndown chart will almost look like an ideal one. The fact that this actual burndown is different from the ideal one, will mostly due to unforeseen circumstances rather than bad planning or estimating the effort needed to complete a user story/task. That is why a burndown chart would be useful to use, but not necessarily test through an experiment. This estimation effort versus actual expended effort is then better captured with the metric 'Estimation Health', which will highlight how well an Agile product team is able to correctly guesstimate the effort needed per user story/task.

Cumulative flow diagram (CFD)

What does it show?

While on a first look, the cumulative flow diagram seems to present the same information as a burndown/up chart, the cumulative flow diagram shows additional information. The CFD was introduced by Anderson (2003) as a replacement for the burnup chart. It tracks the stages of the work items, while the burndown chart tracks completion (i.e. counting down the number of work items to do until zero) (Kissflow, 2022). The CFD is a stacked area chart; each time interval shows the number of work items in each stage (e.g. backlog, in progress, review, done). (Furlong, 2019)

Merits

Using a CFD is useful to show the work in progress (WIP); it may lead to increasing the throughput and reducing lead time. WIP could be used to predict and estimate the completion date of the work item. The CFD helps teams to measure their efficiently and can guide them as to where to focus their improvement endeavours. (Javdani et al., 2013)

CFDs are also a useful tool to highlight possible bottlenecks in the process; it may be the tasks are accumulating in a particular state (e.g. a lot of user stories have to be reviewed and they are accumulating quickly in that review state). (Matrix Resources, 2014)

Not only can it be used as a way to track the overall progress of an iteration, it can also be used to track the bug backlog. (Hayes et al., 2014)

Whether the CFD is used as a way to track the overall progress of an iteration or the issues in the defect backlog, the way the different categories in a CFD are displayed reveals interesting and actionable information. One of which is the fact that there should only be a small number of work items in the WIP stage. (Anderson, 2010).

Additional information from the CFD are the cycle times for work items in progress. While it may be unreasonable to expect Agile teams to calculate the cycle time from looking at the CFDs themselves, patterns could show if team members are remaining idle while waiting for other stakeholders or steps in the process. (Hayes et al., 2014)

Pitfalls & considerations

While the CFD allows the user to visualize the flow and extract valuable information, it can be confusing as to how one should read and interpret this diagram. This would then be the first consideration to keep in mind; teaching the team members and scrum master to correctly read and interpret this diagram in order to maximize the value of this measurement practice. Since it could prove difficult in reading the average cycle time and other metrics from this diagram, it could be supplemented with calculated metrics such as the average cycle time as a number in addition to the CFD.

Another possible pitfall would be showing all workflows of all work item types in the same chart. This added complexity will make it almost impossible to gain precise insights. A solution to mitigate this would be to design CFDs on the portfolio, project, epic, feature and product backlog item levels.

Overall, the literature is quite positive on how the CFD can be used to help teams improve their workflow health & time to market, and identify issues early on.

Estimation health/accuracy of estimation

What does it show?

The health/accuracy of estimation shows the team how close their estimations are to reality (Davis, 2015). For example, a mean value of 6 means on average tasks take six days longer than the team's original estimation. This metric could be shown through a number, but another way to historically visualize this is in a graph. This way, the team is able to see if their accuracy of estimation increases or decreases over time.

Merits

The merits of this metric are numerous. First of all, this metric provides teams with insight on how accurate they are with regards to the estimated time that will be spent on user stories. Secondly, facilitating teams with increasing their estimation accuracy will probably have an effect on customer satisfaction. The more accurate the estimation, the less over- or under promising to the customer occurs. Third, it has the chance to increase employee satisfaction; better planning because of this metric will most probably result in less stress (i.e. unrealistic expectations are less likely to occur with regards to time spent on user stories).

Pitfalls & considerations

While a very useful metric to use and adapt to, this estimation accuracy will unlikely ever be 0. During the team's work, it is almost guaranteed something unexpected will come up which changes the scope or requirements of the initial user story. The goal would not necessarily be to have the perfect estimation health, but to facilitate and enable discussion with the team and find ways to better estimate user stories. After all, an estimate remains an estimate.

Release stabilization period

What does it show?

The release stabilization period shows the time developers spent on correcting issues between the point the developers claim it is ready to release and the point where the corrective changes are actually released. (Scrum.org, 2020)

Merits

This metric can help the Agile product teams identify if there is a delay in releasing completed features/user stories. It may the case user stories are completed, but it will not be shipped in the next release. One can then ask the question why it was worked on during this sprint at all, if it does not provide immediate value to the customer.

Pitfalls & considerations

If the team already knows each completed feature automatically is present in the next release/demo, it may not be the most useful metric to incorporate. Likewise, if the product team knows there is a specific reason why a certain feature is not immediately released, this metric would again not be the most useful.

Time in status

What does it show?

This metric shows the average time spent a work item spends in a particular status, such as the status 'in progress' (Mas et al., 2020).

Merits

This metric can be used to discover patterns in how long work items remain in certain statuses. If tasks are waiting to be reviewed for a long time, this may be a symptom the team does not have enough capability to review all tasks before a deadline. Patterns of time in status can be observed; this provides input for discussion between the team members to look for a root cause or explanation. (Mas et al., 2020)

Pitfalls & considerations

Using time in status may not be very actionable for the team. Just because the average (or median) time for a task to be moved out of 'review' is fifty days, does not provide the team with much actionable information. To remedy this, one would have to provide a historical overview of how the average (or median) time each category developed; does the time spent in review increase over time? This may again not be that actionable, but it will provide input for conversation if the team identifies this as a problem that requires to be solved.

Issue/defect/trouble report count

What does it show?

It simply shows the amount of current issues/defects of the product. While not as complex as some other metrics or measurement practices, it is a very effective way to indicate a product's quality. In the paper of Boon & Stettina (2022), the amount of external trouble reports is an indicator of quality in the context of large-scale Agile transformations.

Merits

Just displaying the number of issues would not be a very effective way for the team to adjust their work. What could provide merit with regards to this subject is an overview of the identified, in progress and solved issues. This provides teams with more understanding of how many issues there are in total, while providing an overview of how many and which are currently being worked on.

Pitfalls & considerations

When using an absolute count of issues/defects, this number may not show the team how severe or easy to fix certain issues are. If there is one issue open and fifty are closed, this may seem positive. But if that one issue is critical and those fifty closed issues were small fixes related to user experience, one can ask how useful it is to use this absolute number of trouble reports.

Technical debt

What does it show?

Technical debt is the "design or implementation construct that is expedient in the short term, but sets up a technical context that can make a future change more costly or impossible" (Avgeriou et al., 2020; Avgeriou et al., 2016)). Technical debt rears its head when a team makes a short-term decision which has negative long-term effect on the product or codebase.

Merits

Providing insight in the technical debt of a team's codebase will help make the team better see the

negative consequences of systemically taking shortcuts. When the team observes numerous problems such as code duplication and unnecessary coupled code, they might make better long-term decisions in their work to prevent this technical debt from piling up.

Pitfalls & considerations
As Avgeriou et al. (2020) state in their research on technical debt, every tool or method on the market uses different metrics, indices and quality models to assess technical debt. Besides this lack of a unified methodology, not all Agile product teams are software development teams. Additionally, demands from the customers and management may make it inevitable to incur technical debt, even if a metric shows this technical debt is increasing.

Productivity index/time spent context-switching
What does it show?
This index shows how much time is lost by the team, for example to interruptions caused by meetings and alternating between various (unrelated) tasks. It is measure of how much time is actually spent by a team member on a certain task. (Scrum.org, 2020)

Merits
This productivity index shows how much time is spent productively on the tasks that are relevant for the Agile team. When certain delays or impediments occur, a low productivity index could be a possible explanation. After all, the less time a team member can spend on a task in total, the less time this team member can spend it efficiently.

Pitfalls & considerations
This metric is not very practical and actionable for two reasons. The first is that team members usually do not control which meetings they have to attend and which off-topic tasks they have to perform. Secondly, the administrative work that is needed to accurately show and use this metric is too cumbersome. If team members were to keep track of this, it would lead to cumbersome manual administrative work, since they would have to keep track of when they get distracted and when they do not have time to work on their assigned work in general.

Defect removal efficiency
What does it show?
One of the aspects with which one can determine a product's quality is the number of issues/defects. If the customer uses the product and reports no issues when using it or no occurred defects, one can conclude this specific aspect of quality is sufficient. Of course, that would be an ideal situation; when developing a software product, it is almost guaranteed defects, issues or errors will eventually occur.

The Defect Removal Efficiency (DRE) represents the amount bugs detected by the team, compared to the amount of bugs that were detected by the users of the product. The DRE's formula is (number of defects found internally / number of defects found internally + number of defects found externally) * 100%. (TestMatick, n.d.)

Merits
This DRE-metric indicates whether the team is producing sufficient quality in their value creation for the customer. For example, team A has a DRE of 90% and team B has a DRE of 50% while both have 25 issues identified in total. This metric indicates whether the issues are identified by the team itself

before the customer communicates or even observes this issue in their product. Team A is producing work of higher quality because more defects were found internally (and hopefully fixed) compared to team B which had half of their defects identified by their customers.

This metric makes teams more aware of quality as a product aspect; from an implicit to an explicit aspect. This metric pushes and incentivizes the team to look more carefully at the quality of their releases as to minimize the possibility of defects found by the customer.

Pitfalls & considerations
Like most metrics, the possible pitfall would be focusing too much on the percentage itself instead of what it actually signifies. In an effort to drive up the percentage, a possible solution of the team would be to not classify issues as issues, but as regular items in the backlog instead. The DRE-metric should help teams become aware of the internal and external perspective on the product's quality, which the teams need to keep in mind.

Another consideration would be that the percentage does not show/indicate whether the total number of defects/issues found is relatively high or low. If team C has a RDE of 95%, but the number of defects found internally is 95 and externally 5, one can beg the question whether the DRE of 95% is an appropriate indicator of quality. If RDE were to be used, this would have to be combined with the total number of defects as well in order to prevent a distorted view of the product's quality.

Backlog size & health
What does it show?
Many authors recommend a good product backlog should be kept DEEP (Detailed appropriately, emergent, estimated, prioritized) (Agile Tools, n.d.; Mounsey, 2017). The backlog health shows the team how many user stories in the product backlog are in the 'ready' state. This 'ready' state signifies that the user story can be picked up and executed in the next sprint. The backlog health is deemed positive if there are enough user stories with a cumulative number of story points greater than the average velocity of the team. Thus, the calculation for the backlog health would be (cumulative amount of story points of ready user stories / average velocity). An optimal value would lie between 2 and 3; any more than this indicates too much planning upfront and negates the emergent and iterative nature of the product backlog and Agile in general. (Agile Tools, n.d.)

Merits
A healthy product backlog helps the team increase the value of their work (Agile Tools, n.d.). To ensure the team has enough user stories for the upcoming sprints, it is necessary to maintain a healthy product backlog. It also shows if the team needs to place more emphasis on refinement/grooming/elaboration of user stories (Mounsey, 2017), since this is necessary to have a user story in the product backlog to be classified as 'ready'.

Pitfalls & considerations
One of the considerations of using the metric would be that each team has its own definition of when a user story is classified as 'ready' in the product backlog. A team may perform the refinement and elaboration of the user story at the beginning of the sprint, while another team will perform this task before the user story moves from product backlog to sprint backlog. Since the calculation of the backlog health is relative for each team, this will most likely not pose any challenges in applying this metric.

Innovation rate

What does it show?

Innovation rate is the percentage of effort or money spent of new capabilities of a product, divided by total product effort or cost. The innovation rate shows the capacity of the team or organization to deliver new product features and capabilities. (Scrum.org, 2020)

If a team is busy with the maintenance of existing products, there is less room to innovate, learn and carve out a competitive advantage (Forte Group, 2022).

Merits

Since delivering value to the customer is important and central to the Agile way of working, the innovation rate shows how much effort and time is spent on 'keeping the lights on' and how much time they (should) have to focus on creating new features that provide value (Forte Group, 2022). This can facilitate discussion with stakeholders to for example open up more room for initiatives and innovation instead of corrective maintenance. Root causes of why so little time is spent of innovation and developing new value-driven features can be discovered by displaying this innovation rate.

Pitfalls & considerations

The first pitfall would be that it is difficult to measure. Sometimes corrective work needs to be performed simultaneous with developing innovative solutions. Keeping track of this could be seen as annoying and as too much administrative work. The next pitfall is the fact that it is especially difficult to measure when the team is servicing an internal business; how would one calculate the effort and cost of developing new value-driven features when an Agile team only has internal customers (i.e. customers within the same organization) which won't pay or buy directly for features as external customers would? While the latter is not included in the formula, it is necessary to know this in order to gauge whether the innovation was successful or not.

Impediments & blocker overview

What does it show?

Perhaps not as quantitative as other discussed metrics, but still as relevant is an overview of impediments and blockers. A blocker is defined as "something that stops work from continuing on a work item" and an impediment is defined as "something that slows down a work item". (Carroll, 2021)

This could be as easy as showing a list of the impediments/roadblocks the team is experiencing. One example would be 'too many meetings or distractions' or 'technical environment issues'.

Merits

When a team experiences an impediment or a roadblock, it might happen that it will be mentioned during the sprint ceremonies such as the daily stand-up or the refinement session. But making these impediments and blockers explicit by showing them in a simple list will remind the team what prevents them from performing at the highest level and will actively spur the scrum master (Trivedi, 2021) or the team members in action to remove these impediments or blockers.

Pitfalls & considerations

Just showing a list or overview of the various impediments and blockers will probably not immediately spur the team into action. Ownership is important to encourage resolving these impediments and blockers; assigning someone (whether it is a scrum master or another team

member) to follow-up on this impediment/blocker will create ownership and this may stimulate the resolving of the specific impediment/blocker by this 'owner'.

Defects solved

What does it show?

While we already discussed an overview of all the open defects with their associated severity, it does not show the team if there are any patterns in the resolving of these defects. An overview of the various solved defects with their associated severity and duration could provide interesting insights to the team.

Merits

By looking at historical patterns of the amount of defects solved with their severity and how long it took, the team is potentially be able to discern patterns of how long each category of defects on average takes. This could then be used to refine the categories themselves and better plan for when an issue will be resolved when a new defect is identified.

Pitfalls & considerations

Defects could be multi-faceted. Inferring predictions for the future based on past averages will not always work. Thus, it should be used as a guideline and recommendation. Moreover, one can question how actionable an overview of the solved issues are, since each issue has the potential to be completely unique with its own context that influences the time it takes to solve it.

Work Item Age

What does it show?

Work item age is the time that passes between a task that was started and the current time of the task. (Kanbanize, n.d.)

Merits

This work item age metric shows how tasks are progressing through the stages of the workflow process. It shows the team in which stages of the process work items remain most frequently and for how long on average they remain there. (Kanbanize, n.d.)

Pitfalls & considerations

While it is useful to see which tasks in the process take longer than expected or are outliers compared to the average, there is usually a valid reason for this. Perhaps a developer is waiting for an impediment to be removed before he/she can continue. The fact that this task is an outlier is in that scenario not that actionable, unless the impediment gets escalated to higher management. It does provide the scrum master and the team with a valuable overview of the flow and the WIP-times, but it is possible the anomalies are not always that actionable.

Customer Usage Index

What does it show?

The Customer Usage Index is a measure that shows the degree to which customers are using the product and whether their usage meets the expectations of the team or business. (Scrum.org, 2020)

Merits

This measurement can provide insight into which feature is the most used; this can help in

prioritizing user stories and resource allocation within the team and it can help in communicating with the customer. At the end of the day, an Agile team wants to provide value to the customer. It may happen that a team spent multiple sprints on developing a certain feature, but with this customer usage index it shows this specific feature has not been used in the past two months. This observation can spark new discussions and help prioritize tasks and allocate resources better.

Pitfalls & considerations

There are some aspects to consider. First of all, this can be difficult to measure. Not every application has this built-in functionality to track how many times a certain feature is used. Not only that, this creates an additional layer of complexity; with each new release, the customer usage index should be programmed for and applied on this new feature. This not only takes up more time, but increases the likelihood of bugs/issues occurring. Second, while this measurement seems very beneficial to use in theory; how useful would this be in practice. If a customer does not use a feature for a few weeks, it does not mean the developed feature is/was not important.

Number of releases

What does it show?

The idea behind these metrics is that these will be used to increase performance and value of the team and to the customer. The number of releases could be used to show the business value delivered by the team (Boon & Stettina, 2022). Although a simple metric, if the team is using the other metrics to improve their workflow health, one of the possible outcomes would be more frequent (and of course, better) releases to the customer.

Merits

If sprints are solely used to correct defects or solve impediments, this would signify these defects or impediments were not identified appropriately earlier by the scrum master or product owner. Using the various proposed metrics will help the team focus more on delivering value and spotting potential negative trends earlier on. This in turn will translate into a higher frequency of releases, which can be used to show the business value to various stakeholders.

Pitfalls & considerations

Just because the number of releases is increasing does not mean more business value is delivered to the customer or organization as a whole. Quantity is not equal to quality, this should be kept in mind when the number of releases is changing for the better or worse.

Stakeholder satisfaction & trust

What does it show?

Besides the customer and the team members, there are also other stakeholders involved such as business managers and the people funding Agile teams. It is important that teams are aware of the sentiments of stakeholders (Zhang et al., 2016) and potentially act upon this. There are numerous stakeholder satisfaction models available in the scientific literature and each has their own merits and applicability.

Merits

It is important to understand the various sentiments of the stakeholders that are involved directly and indirectly. If it is the case a certain important stakeholder who partly funds the team is dissatisfied because he doesn't get as involved as he would like to, it is a sign for the team to include

this particular stakeholder more often. If the team is informed on the various stakeholders and their (dis)satisfaction, they are able to take action if necessary.

Pitfalls & considerations
Measuring stakeholder satisfaction, trust and sentiment could prove challenging. One the one hand you don't want to overburden the stakeholders with surveys and pointless meetings, while on the other hand you want to be informed on their view of the team and the project. It is a delicate balance, and perhaps a separate field of research entirely.

Customer/user satisfaction gap
What does it show?
The customer satisfaction gap shows the difference between a customer's expected experience and their actual experience. (Scrum.org, 2020)

Merits
Looking at this more qualitative measure, it helps team identify opportunities regarding the customer experience. The consideration of both current value and unrealized value provides teams with the option to position present and feature benefits (Scrum.org, 2020). Just because a customer does not complain, does not mean the desired customer experience is currently met.

Pitfalls & considerations
Unfortunately, this measurement is difficult to quantitively capture. One can of course ask the customer what their current experience is on a scale from 1 to 7 (or 10), but the same cannot be asked for their desired experience. This would have to be captured in a more qualitative way, such as feedback sessions with the customer. To remedy this, this could be incorporated into the design of the customer satisfaction survey.

Sprint completion percentage
What does it show?
This metric shows the team how much work was completed each sprint (Bahrami et al., 2016). In addition to the estimation health mentioned previously, this metric can help the teams realize whether they are completing the planned work or not, regardless of over- or underestimating the required work.

Merits
This sprint completion percentage will not necessarily immediately improve the productivity of the team, but it will spark and facilitate discussions on why the allocated work for the sprints are not at 100%. Certain root causes may be identified that can then be tackled in order to improve this sprint completion percentage.

Pitfalls & considerations
This sprint completion percentage does not tell the full story. The name 'Agile' already implies the teams welcome change if it benefits the value of the product or customer. There may be unexpected changes in the amount of work that is needed to complete a certain user story; team members may get sick or overloaded with unnecessary meetings which in turn has an effect on how much of a sprint the team is able to complete. Thus, this percentage should always be considered in the context of the team's circumstances.

Enhancement Rate

What does it show?

The enhancement rate shows the throughput of user stories in an Agile project. The difference between regular throughput and enhancement rate is the fact that the enhancement rate is based on status and effort estimation. The enhancement rate shows for each sprint what percentage of the previous sprint has gone into production or acceptance in the current sprint. (Boerman et al., 2015)

Merits

The enhancement rate can be tracked to understand how much work on average is pushed to production or acceptance of the previous sprint. The team can use this metric to keep a consistent pace or they can use it as a proxy-measure of the development effectiveness. If the team sees the enhancement rate dropping, they can investigate why it has dropped below a certain threshold.

Pitfalls & considerations

Every sprint is different; different tasks require different expertise, a team member may be sick or the infrastructure may experience outages. All these variables at play can have an effect on the enhancement rate. One cannot always control the enhancement rate; it is a product of the previous sprint's circumstances. A drop in the enhancement rate is thus not necessarily bad; one should look at the enhancement rate over a longer period of time.

Scope prognosis

What does it show?

The scope prognosis is the average of the enhancement rate, combined with the remaining project size and the last expected iteration. It shows an indication of the anticipated degree of functional completeness. (Boerman et al., 2015)

Merits

A product owner could use this metric to understand in which sprint, what percentage of the scope would be completed (Boerman et al., 2015). If the scope prognosis shows a percentage lower than expected in a certain sprint, the product owner and the scrum master could for example agree to reduce the scope of the project.

Pitfalls & considerations

The scope prognosis is what it says; a prediction. One should use this as a supportive tool, instead of thinking this is how the project will exactly unfold in the upcoming sprints. It will help with making decisions about the scope of the project, but the practitioner should always consider this metric does not represent reality.

Project size remaining

What does it show?

A relatively simple metric, the remaining project size shows the amount of work that is remaining for the project (i.e. total work for the project – completed work already). (Boerman et al., 2015)

Merits

On its own, the benefit of using remaining project size is straightforward; one knows how much of the project is already completed and still remaining. This can be used for planning or scope changes.

The other benefit is stipulated in the work of Boerman et al. (2015), where adjusting the remaining project size can simulate cope change, and thus change the scope prognosis visualization.

Pitfalls & considerations

Since it's a relatively simple metric, no pitfalls and considerations could be thought of.

Changed PBIs

What does it show?

The metric 'changed PBIs' tracks the descriptions which have been changed of existing PBIs/user stories. It shows if the description of a PBI is different from the description of the PBI in the previous sprint. It can be shown as a percentage; the percentage of changed PBIs compared to the total number of PBIs. (Boerman et al., 2015)

Merits

Changing the descriptions of PBIs can have an impact on the effort needed to complete the PBI. The benefit of using this metric is that the scrum master or product owner knows which PBIs have changed and can check if the original estimated effort is still the same when the new description is taken into account. (Boerman et al., 2015)

Pitfalls & considerations

Changing the description of a PBI does not always mean a completely different scope for the PBI. It may happen that a spelling or grammar mistake is fixed. Thus, when implementing this metric, it should be considered that just because a description has changed, does not mean the scope or the PBI has changed as well. Perhaps an additional metric should be used to know for sure if the altered description has an impact on the scope of the PBI.

Added PBIs

What does it show?

Added PBIs indicates the percentage of PBIs which are added to the backlog of a sprint, which were not present when the iteration started. (Boerman et al., 2015)

Merits

It could be used to keep the 'scope creep' at bay. It helps the team keep track of how and when PBIs are added. It could be used to solve a deeper-rooted problem if (almost) every iteration results in added PBIs during the iteration.

Pitfalls & considerations

Sometimes, the priorities of the business change during an iteration. A scrum master or product owner does not always have enough influence to reject added PBIs. It may happen that something important comes up which takes priority; in that case the metric 'added PBIs' would not be the most useful or interesting metric to use.

Rejected PBIs

What does it show?

The number or percentage of PBIs which have been rejected from the backlog compared to the project size. (Boerman et al., 2015)

Merits
Instead of just deleting a PBI from the backlog, it is better to keep track of these rejected PBIs. This way, the team gains greater insight of their backlog process and knows how many and which PBIs have been rejected. (Boerman et al., 2015)

Pitfalls & considerations
Similar to the metric 'Added PBIs', the metric 'Rejected PBIs' could be the result of management or other stakeholders with influence simply deciding that a PBI is not relevant anymore. The value of this metric becomes less, because in those cases the team has no say in which PBIs are rejected or not.

Project size
What does it show?
The size of the backlog. This can either be done by using the total effort estimation (i.e. summation of all PBI's effort estimation) or simply counting the number of PBIs. (Boerman et al., 2015)

Merits
Can help with the communication towards other stakeholders about the size, complexity and difficulty of the project. The team has a better picture of the work that is needed to complete this project from start to finish. Another benefit is that it can be used to request (additional) budget, if the effort estimation is known and accurate.

Pitfalls & considerations
If the effort estimation is not known for all or the majority of PBIs, one needs to use the number of unique PBIs. However, this does not mean anything, since PBIs can vary from e.g. 50 story points to 1 story point. It is thus important to use this if the effort estimations are known for the PBIs. Furthermore, effort estimations are just that; estimations. One always needs to consider the reality can be different from expectations; this should be kept in mind when using this metric.

Time prognosis
What does it show?
The time prognosis shows in which iteration the product is likely to be finished. (Boerman et al., 2015)

Merits
The time prognosis can be used to change either the scope or the likely end-iteration if it seems the current likely end-iteration is not achievable for the team. It can also be used in communication with stakeholders when they would like to know when the project is likely to be finished.

Pitfalls & considerations
The nearer the practitioner is to the likely end-iteration, the more accurate the time prognosis becomes. However, at the start of a project that will take a long time to complete, the time prognosis should always be considered as an estimation, since risks play a bigger role if the project is in its early iterations.

Priority shift

What does it show?

Priority shift simply shows if a PBI was assigned a different estimation in the backlog (Boerman et al., 2015). If a team uses the priority levels 1 through 5 (with 1 being the highest priority); a PBI can change from priority 2 to priority 4. This means the priority shift is 2, since the priority shift is the relative changes in priority (Boerman et al., 2015).

Merits

The priority shift-metric can help the team to track when and how priorities of PBIs change. If priorities are constantly changing for all PBIs, the team can perhaps reconsider how they are assigning PBIs in the first place.

Pitfalls & considerations

The team does not always have full control or influence of the shifting priorities. Thus, it would be good to consider or track PBIs with shifted priorities due to team insight or PBIs with shifted priorities due to external influence.

Effort at risk

What does it show?

Before a PBI is pushed to production and adds value to the software product, there is a risk that any effort made is lost, for example due to changing priorities or budget cuts. Thus, the effort at risk metric shows the effort that has not yet been converted to product value. This metric has three categories: low risk, medium risk and high risk. As the PBI moves from the design-phase to the production-phase, the risk decreases. (Boerman et al., 2015)

Merits

Visualizing effort at risk, as Boerman et al. (2015) do in their paper, helps the teams to quantify the risk they are taking every sprint. If the category 'high risk' for a certain sprint is deemed excessive, the scrum master can take action, together with the product owner or sponsor.

Pitfalls & considerations

The scrum master does not always have the ability to take actions based on the 'effort at risk'-metric. Even though it is visualized, there is always an inherent risk in 'wasting' effort on PBIs due to changing circumstances.

Estimation shift

What does it show?

The estimation shift shows the difference (positive or negative) in the estimated effort of a sprint compared to the previous sprint; it returns the sum of these. For example, if a sprint has a total decrease of 75 story points, and a total increase in the same sprint of 50 story points, the estimation shift would be -25. (Boerman et al., 2015)

Merits

The benefit of using this metric is shows both in a positive and negative sense what how the estimation has changed. Patterns can be discerned from this; if a consistent negative estimation shift is observed, the scrum master can review the estimation practices of the team.

Pitfalls & considerations

When visualized this metric, one should take care to split the negative and positive estimation shift. Otherwise there is a risk of the negative and positive estimation shifts cancelling each other out; this would provide an deceitful representation of the estimation shift. (Boerman et al., 2015)

Appendix C – Introductory interview list of questions

1. Introduction myself, purpose of my research and purpose of this specific meeting. Ask for permission to record the meeting.

2. Product
   a. Could you tell me about the [Product]?
   b. Why was this product team started; What problem does your product solve for the customer(s)?
   c. Is there a demo or training material available so I can get a better understanding of the product?
   d. Optional: scope of your responsibility with regards to the product development?

3. Customers
   a. Who are the customers of your product? (internal or external)
   b. How do you communicate with the customer with regards to requirements and feedback loops?
   c. How do you assess the customer satisfaction of each iteration?

4. Product team
   a. How many members and FTE are part of the product team?
      i. How does this impact the quality and quantity of what is worked on?
   b. What is each team member's role and expertise? Are there technical (i.e. software engineers) in the team?
   c. How long has the team been together and is it stable with regards to the members?

5. Metrics
   a. Do you have experience with using metrics, measurement practices, KPIs or OKRs in an Agile environment/team? If yes, which ones?
   b. Do you use metrics to measure success/performance?
   c. What was your experience; did it work well or not? Why (not)? How much time did it take to implement and get used to these metrics?
      i. Did you notice a positive effect on customer satisfaction when acting upon these metrics in the past?
   d. How extensively do you use Azure DevOps as a project mgmt. tool? Any other tools?

6. Agile methodology
   a. You are using the Agile Scrum methodology, do you deviate from this in any way?
   b. How mature do you feel the team is with regards to the Agile methodology?
   c. What is the cadence of your product team (when does a sprint start, when R&R, when customer demo, etc.)?
   d. Are there any current (or past) pain points or challenges experienced by the team with regards to Agile working?
   e. Is there anything that the team wants to implement/do (i.e. KPIs); why are they not doing this specific thing?

Appendix D – Agile maturity survey

Table 21 shows the questions asked in the Agile maturity survey. All questions, except for the first question, had a 5-point bipolar Likert scale. The possible answers were: 'Strongly disagree', 'Somewhat disagree', 'Neither agree nor disagree', 'Somewhat agree' and 'Strongly agree'.

Table 21: The questions asked in the Agile maturity assessment survey

Questions asked in the Agile maturity survey
Looking at the Agile maturity model of Laanti (2017), please use this model to determine your team's Agile maturity. Please note that each stage builds upon the next stage, from left to right. For example, when you think your team is at the 'advanced' level, all the practices of the stages 'beginner', 'novice' and 'fluent' should be in place.
We can decide for ourselves how we achieve a work goal.
We are generally good at organizing ourselves.
We are encouraged to learn new skills that help to handle changes.
Mistakes are used as a chance to adjust our approach.
We have the courage to take on new tasks for which we do not yet know all the requirements.
We can adjust to changes.
Through direct conversation, we try to find out what the customer needs.
We talk to our customers regularly.
We try to find out what is most important for the customer.
During our work, we frequently think about how my job helps customers.
We try to reach our goals by satisfying customers.
We come up with new ideas to better complete our tasks.
We like to exchange views with others about the challenges of reaching our goal.
It is important to us to always learn something new.
We enjoy exploring new situations.
We solve difficult challenges best when working together with others in teams.
Our work is transparent for others.
Different perspectives within our team are appreciated.
We like to support each other in our team.
We regularly peer review our approaches.