Emma Karttunen

Retrospectives: Optimizing Workflow in Agile Teamwork Environment

Tradenomi
Tietojenkäsittely
Syksy 2018

KAMK • University of Applied Sciences

Tiivistelmä

Tekijä(t): Karttunen Emma

Työn nimi: Retrospektiivit: Työskentelymetodien optimointi tiimityöskentelyssä

Tutkintonimike: Tradenomi (AMK), peliala

Asiasanat: Agile, Kanban, Scrum, iteraatio, sprintti

Tässä esseessä käyn läpi mitä retrospektiivit ovat, mihin niitä käytetään ja kuinka ne tulisi järjestää. Nykyään työympäristöissä Agile metodit, jotka tähtäävät tiimityön joustavuuteen, valtaavat alaa ja osana niitä ovat tulleet myös retrospektiivit. Nykyisessä työssäni retrospektiivit ovat olennainen osa työympäristöä.

Ensimmäisenä käyn läpi hieman retrospektiivien historiaa ja kuinka ne tulivat osaksi IT yritysten arkea. Retrospektiivit ovat osa Agile metodologian metodeja, joilla pyritään korjaamaan tiimityöskentelyn ongelma kohtia projektin aikana, eikä vasta sen jälkeen. Retrospektiivien tarkoitus on sovittuina aikaväleinä tarkastella projektin historiaa ja siinä käytettyjä työtapoja. Tavat joko pidetään, muutetaan tai hylätään, riippuen kuinka tarpeellisiksi tiimi ne kokee. Myös uusien tapojen lisääminen kuuluu retrospektiiviin.

Retrospektiivejä on monenlaisia ja tässä esseessä keskityn Kerthin, Loeffrelin, sekä Derbyn ja Larsenin malleihin. Kerthin mukaan retrospektiivit tulisi suunnitella päämärän mukaisesti ja erinäistä rakennetta ei tarvita. Päämäärä voi esimerkiksi olla tiimin ongelmien korjaaminen. Derby ja Larsen esittävät mallin, jossa on viisi vaihetta ja antavat näin enemmän työkaluja retrospektiivien järjestämiseen. Loeffrel taas lisää malliin vielä yhden askeleen, joka keskittyy juurikin taaksepäin katsomiseen ja vanhojen tapojen tarkasteluun. Derby ja Larsen keskittyvät enemmän uusien ongelmien löytämiseen.

Lopussa käyn läpi omakohtaisia kokemuksia retrospektiivien järjestämisestä firmassa, jossa tällä hetkellä työskentelen. Annan esimerkkejä niin suuremmassa kuin pienemmässäkin tiimissä työskentelystä. Lisäksi kuvaan tuntemuksiani oppimisprosessistani retrospektiivien maailmaan.

Abstract

Author(s): Karttunen Emma

Title of the Publication: Retrospectives: Optimizing Workflow in Agile Teamwork Environment

Degree Title: Bachelor of Business Administration, Game Industry

Keywords: Agile, Kanban, Scrum, iteration, sprint

In this essay I will go through what retrospectives are, what they are for, and how they should be organized. Nowadays Agile methods, which aim for flexibility in a teamwork environment, are rising in popularity and retrospectives are part of that philosophy. Retrospectives have a vital role in my current work environment.

First I will go through the history of retrospectives and how they became part of the routine in IT and software companies. Retrospectives belong to Agile methodology, which aim to resolve any issues in a teamwork environment during the project rather than after. The reason retrospectives exist is that they let us look into the past and assess the pipelines and workflows we use. Any practices used can be kept, changed, or abandoned based on the current needs of the team. Adding new habits to the team's workflow is also part of retrospectives.

There are many types of retrospectives but this time I will focus on Kerth's, Loeffrel's, and Derby's and Larsen's models. According to Kerth retrospectives should be designed based on a preset goal and no specific structure is needed. Derby and Larsen argue that a five-step model works well with retrospectives and give them a framework to work in. Loeffrel adds one more step to Derby's and Larsen's model which focuses on looking back and assessing old habits. Derby and Larsen focus more on resolving new underlying issues.

At the end of the essay I will go though my own experiences as a retrospective facilitator in the company I currently work in. I give examples of both, smaller and bigger projects. I also describe my learning experience in the world of retrospectives.

Sisällys

1 Introduction 1

2 History and definition 2

3 Basic Structure and why it is used 5

3.1 Scrum Model 5

3.2 Derby’s and Larsen’s Model 6

3.3 Loeffrel’s Model 7

3.4 Kerth’s Model 10

4 Facilitation 13

4.1 Basics of Facilitation 13

4.2 Dynamic Facilitation and Structure 14

4.3 Ideologist 15

4.4 Leader Skills 16

5 Designing my first retrospective 21

5.1 Purpose 21

5.2 Attendees 21

5.3 Place and time 21

5.4 Theme 21

5.5 Cycle 22

5.6 Activities 22

5.7 Break 22

6 Example 1: Tourist Retrospective 23

6.1 Brief and Role 23

6.2 Basic Structure 23

6.3 Preparations 24

6.4 Activities 25

6.4.1 Set the stage: Quick Questions 25

6.4.2 Gather data: Spring Wedding 26

6.4.3 Generate insights: Fish Bowl 27

6.4.4 Define experiment: Maximize follow through 29

6.4.5 Closing: Pleased/Surprised 29

6.5 What I learned 30

Example 2: Stormdivers Retrospective 31
6.6 Brief and Role 31
6.7 Basic Structure 31
6.8 Preparations 31
6.9 Activities 32
6.9.1 Set the stage: Weather 32
6.9.2 Check hypothesis: Looking Back 32
6.9.3 Gather data: Mad, Sad, Glad and Dot Voting 33
6.9.4 Generate insights: Cause and Cure 33
6.9.5 Define experiment: Who, What, When 33
6.9.6 Closing: One Word 34
6.10 What I learned 34

7 Example 3: Retrospective Workshop 35
7.1 Brief and Role 35
7.2 Basic Structure 35
7.3 Preparations 35
7.4 Activities 36
7.4.1 Set the Stage: Expectations 36
7.4.2 Gather Data: On the Boat 36
7.4.3 Generate Insights: 5 Whys 36
7.4.4 Define Experiment: Who, What, When 37
In “Who, What, When” activity I first asked the participants to use “Me, We, Us” method to find solutions to the root causes we collected in the previous activity. I specified that they should find as concrete solutions as possible. We then put those solutions on the whiteboard and the participants could voluntarily choose a solution they would like to make as their task. We only briefly discussed about the time boxing since these tasks weren’t meant to been see through. The main point was just to introduce the retrospective framework to the students. 37
7.4.5 Closing: One Sentence 37
7.5 What I learned 37

8 Conclusion 38

Resources 40

Liitteet

List of Terms

Retrospective	Looking back to workflows and pipelines the team has been using.
Agile	Project management philosophy that allows scalability and iteration.
Pipeline	Technical requirements and steps that an asset needs to pass before completion.
Scrum	And Agile method that relies on sprints and metric information of its processes.
Open day	Team building day that has many of the same elements as retrospective.
Sprint	A time boxed iteration loop that Agile methods often use, for example a two-week period.
Specialist	A person who has a lot of knowledge of one specific area of the production.
Generalist	A person who has some knowledge of many areas of the production.
Experiment	A process or system that the team agrees should be tried out and iterated.
Kanban	And Agile method that relies on constant flow of tasks and visual management.
Kanban board	Kanban methods way to organize project management visually.
Postmortem	Done after project to document what was learned from it, what kind of processes and systems should be taken to the next project.
Scope	Describes the scale and workload of a project.
Bot	An automated system that can take over some of the tasks in the pipeline.
Alpha/Beta	Alpha and beta are stages that a product goes though before completion.
ADT	Attention deficit trait, a self-developed attention deficit that is also caused by the environment, multitasking is the main cause if this deficit.

1 Introduction

Nowadays IT and software companies tend to prefer Agile methods to Waterfall model. Waterfall model designs the project pipelines before the project starts and doesn’t support flexibility. This is where Agile comes in. Agile methods aim for a flexible teamwork environment that can evolve with the team and the project. Part of the Agile methods are retrospectives, a way to look into the past and assess old team traditions.

Retrospectives allow us to look into what actually makes the team motivated, angry or focused. The best part is, all the ideas come from the team, from the trenches. As well as optimizing the teamwork processes, retrospectives change the workflows and pipelines to work for a specific team. All teams are unique and applying the same practices to all of them might, in the worst-case scenario, end in mutiny.

Retrospectives shouldn’t be underestimated as a project management tool. Many seem to think that they are time consuming and unnecessary. Let me try to prove you wrong.

2 History and definition

As Marc Loeffler mentions in his book “Improving Agile Retrospectives” (Loeffrel, Marc (2018) Improving Agile Retrospectives: Helping Teams Become More Efficient. United States of America: Pearson Education, Inc.), the word retrospective comes from the Latin word “retrospectare” which means “look back”. He explains that retrospective, as we use the concept nowadays, is a review. Every time you think back to, let’s say, events of the previous day, reminisce a holiday, or go through memories from the last year, that’s a retrospective. Loeffler also describes retrospective as collecting a community together to discuss its values and practices. Loeffler uses a family new year’s tradition as an example in which all the family members go through the previous year’s memories and choose topics to discuss. He has even implemented a retrospective activity into this celebration called 5-Whys. For our purposes this example is a bit too far from the actual topic which is why it’s better to focus on the general purpose that Loeffler suggest retrospectives should happen in a team work environment.

According to Loeffler, retrospectives focus on positive future action that will change the work environment effectively. This is how retrospectives differentiate themselves from postmortems that are usually meant for going through past action and the consequences of those after a project has been finished. Retrospectives aim for continuous improvement of processes and workflow. Unlike postmortems, retrospectives are held multiple times during the project rather than at the end when it would be too late to change the practices in the team. In Agile environments retrospectives are traditionally held in between sprints. Loeffler argues that the other reason why Agile teams adopt retrospectives in their routine is the Agile manifesto. The manifesto consists of four pairs of values and 12 principles that define the core standards of an Agile team. The last of the principles is “At regular intervals, the team reflects on how to become more effective, then tunes and adjusts its behaviour accordingly” (Loeffrel 2018, 3). Retrospectives offer a framework to embrace this principle in an Agile work environment.

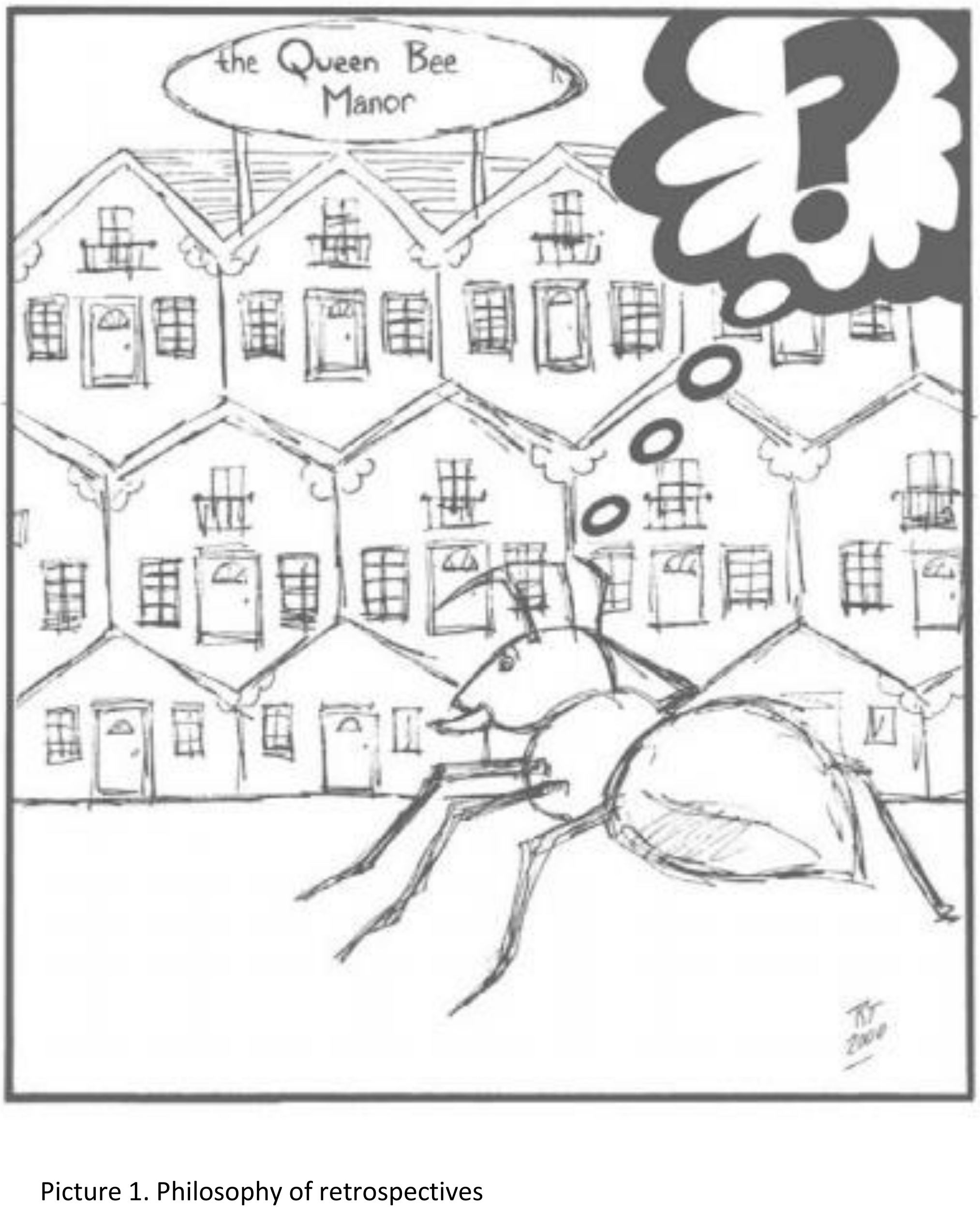
To really understand why retrospectives have been adopted by companies, especially ones in the software industry, it’s important to learn a bit about Agile philosophy. Pia-Maria Thoren (Thoren, Pia-Maria (2017) Agile People: A Radical Approach for HR and Managers (That Leads to Motivated Employees). New York: Lioncrest Publishing) explains that the core of the Agile philosophy is the idea of adapting to ones surrounding conditions. She compares Agile to Darwin’s theory of evolution. The species that can adapt in a changing world are the ones that survive. The corporate world works under the same law of the jungle. Thoren suggests that the software industry's traditional waterfall model doesn't allow adapting to new situations since it is a linear and sequential design process. In the waterfall model, the last phase needs to be completed before the next stage can begin and no overlapping of the phases is allowed. Agile takes an incremental approach in which work is completed in batches, or sprints, after which every task is tested and evaluated. This allows the fixing of practices that clearly don't work and errors in the product before the end of the project. In the early days of Agile the focus was more on the bugs and errors of the product, where as now it has moved on to more human practices and behaviour side of the project work environment.

According to Thoren, most used Agile project management methods are Scrum, Scrumban, and Kanban. All these share the core values of Agile philosophy, continuous improvement and adapting to changes, but they all do it in very different ways. In Scrum, the team is protected by the Scrum Master from the outside forces, for example from customers or the publisher. The team itself is small, from three to nine members, and it works in one to four-week sprints. The team usually has a specific area of the product to develop but in some cases even the whole product if the scope is small enough. The key idea of Scrum is that customer's expectations change which means the team needs to adapt to those changes quickly. The Scrum Master communicates the changes, that the customer wants to make, to the team since then the team can focus solely on developing the product. Kanban focuses more on human systems than product requirements.

Kanban originates from Lean manufacturing that was developed by Toyota car factories (Walters, Chad (2012) Toyota Way Principle #3: "Pull" Systems. Leanblitzconsulting.com/2012/05/Toyota-way-principle-3-pull-systems/. Haettu 10.8.2018). Work is rather pulled according to capacity from the backlog rather that pushed which means the owner of the work decides when the task is accepted on the production line. Kanban tries to avoid bottlenecks and multitasking by keeping up a constant workflow and limiting tasks of team members. Kanban doesn't work in sprints, but it does have feedback loops which retrospectives support.

Scrumban was developed as a hybrid of Scrum and Kanban methods. It is used by teams that take Scrum as their chosen project management method and the use Kanban methods to support continuous improvements. Unlike Kanban, Scrumban teams work in smaller units and use time-boxed iteration loops such as sprints. Unlike Scrum though, Scrumban uses management rather than self-organized teams, and relies on a continuous workflow. These are just very basic definitions for all these project management methods but for our purposes they are enough.

In his GDC talk Clinton Keith (Keith, Clinton (2007) Agile Game Development. https://www.gdcvault.com/play/639/Agile-Game. Haettu 1.8.2018) talks about retrospectives in relations to Scrum and its history. Scrum is one of the older Agile methods. Keith focuses on the product side and giving the team the tools to improve its productivity. Since Scrum is one of the older methods of the Agile family, it was developed by programmers for programmers and therefore in game development the Scrum practices are a bit outdated. After all, we work in multidiscipline teams. This is one of the reasons why retrospectives were adapted to the game development industry, to change practices so that there is one project management method for the whole team. Kanban and Scrumban were born from the changes made to the original Scrum model and they keep evolving to this day. Retrospectives were created for exactly this need to change practices to fit a changing environment, such as team composition.



Picture 1. Philosophy of retrospectives

3 Basic Structure and why it is used

3.1 Scrum Model

Clinton Keith (Keith, Clinton (2007) Agile Game Development. https://www.gdcvault.com/play/639/Agile-Game. Haettu 1.8.2018) talks about the traditional Scrum structure of a retrospective in his GDC talk. Scrum works in predetermined time-boxes that are called sprints which are often week or two long. This means that the time allocated for retrospectives is short. Scrum also focuses on product management rather than people systems which makes the whole approach more systematic.

The traditional Scrum retrospective structure is simple: “What has worked”, “What hasn’t worked”, and “What should we start doing”. “What has worked” collects practices in the pipeline and workflow that have worked during the last sprint. “What hasn’t worked” collects practices that need to be abandoned or changed for the next sprint. “What should we start doing” is for new practice experiments that the team would like to try and evaluate in the next retrospective.

Since the time slot is short, the Scrum retrospective doesn’t really use different activities to accomplish these goals, just simple facilitation methods that we will discuss in a later chapter. An example of this practices could be using the “Me, We, Us” method that gets the whole team to participate in a conversation by first collecting individual thoughts and then group thoughts. Post-its are often used to make the process more visually pleasing and simple.

Keith argues that retrospectives’ key point is to maximize productivity and minimize waste. By this he means that in a retrospective the team should find ways to optimize the pipeline processes in a way that there is a minimum amount of unnecessary work done by the team members. This often means adding automatization and removing steps from workflow that are not vital. This also leads to increased productivity. The changes can be as complicated as creating a new bot taking care of beta access distributions or as simple as removing meetings that take over half an hour.

The focus, according to Keith, is to evolve to faster and faster product development. That is why it’s important to incorporate metrics into retrospectives. Every change needs to be able to be measured by numbers. For example, if beta access distribution was 200 a week when done manually, a goal might be to increase that to 1000 when an automatized bot takes over. At the next retrospective the team would then see if this goal has been reached. The idea is that when we have metrics in use, it is easier to measure the amount and effectiveness of change. This can be done in a product focused goals but it's much harder when considering human systems that might even measure effectiveness through feelings. Keith suggests that iteration of changes need to happen after every retrospective to fit them into a routine. During sprint the team doesn't deviate from the plan made at the last retrospective since this would break the flow. Experiments in pipelines and workflows also need time to show their true potential therefore abandoning them after one try is not advisable.

3.2 Derby's and Larsen's Model

Esther Derby and Diana Larsen (Derby, Esther & Larsen, Diana (2007) Agile Retrospectives: Making Good Teams Great! Washington, D.C.: Pragmatic Bookshelf.) introduce a more complex retrospective structure that is often used in bigger team. Kanban teams are therefore more suited for this approach. Derby and Larsen suggest that there should be five stages in a retrospective: Set the Stage, Gather Data, Generate Insight, Decide What to Do, and Close the Retrospective. Each of these stages support different purposes of a retrospective.

”Set the Stage”, according to Derby and Larsen, is the part where the supervisor of the retrospective reviews the goal and the agenda. Derby and Larsen suggest that the goal should be predetermined. It works as a sort of a theme for the whole session and is decided by the supervisor. He/she decides the theme based on his/her knowledge about the team's issues and working habits. The agenda simply tells all the participants the schedule and activities of the retrospective. It lets people know what to expect and how long each phase has been planned to last. After the review the supervisor goes through working practices for the sessions. This means the rules of conduct during the retrospective, such as do not use mobile. This is done to minimize distractions and to make sure people are mentally present during the session.

“The Gather Data” part is for collecting subjective and objective information from the team. Derby and Larsen point out that information needs to come from everyone point of view. Therefore, there needs to be methods to involve even the shyest members of the group. The main goal is to have a shared picture of the project and all the issues it might have. If only the loudest people get to talk, the view of the project and work environment becomes very specific to those individuals. It is important to acknowledge various temperaments in the group.

”Generate Insight” is the phase where the group analyses the picture they created together in the previous stage. Now it is the time to look at the deeper issues and discuss them as a team. Activities for this session should encourage people to talk and share their ideas and thoughts about the subject. The challenge in this phase too, is to get everyone involved.

”Decide What to Do” is for deciding what the team would like to try as an experiment in their current project. The supervisor collects the insights that the team has gathered in the previous stage. Then he/she prioritizes the issues that have surfaced based on his/her knowledge of the team’s workflow, project goals, and work environment. It is time to figure out solutions for problems that the team faces. These problems could be such as “the office is too noisy”, and the solution for that could be “introducing silent hours”. As well as generating solutions, or as we call them experiments, the team should choose who oversees what. Also, scheduling is a good idea so that those solutions can be effectively tested out within sprint.

”Close the Retrospective” is a short phase in which the supervisor summarizes what has been agreed during the session and repeats the experiments the team is planning to try out. This serves as a reminder for the team of what they have committed to. The supervisor then thanks the team and ask for feedback. He/she can hold a mini-retrospective about the retrospective, for example by having an activity with the team in which he/she asks everyone to tell their feelings in one sentence. This is just one way to collect feedback as an activity.

3.3 Loeffrel’s Model

Derby’s and Larsen’s retrospective works well for a big team and introduces a more complicated structure. For a small team, however, the structure might mean more work than it produces effective experiments. In this case, the basic Scrum retrospective works well. Marc Loeffrel (Loeffrel, Marc (2018) Improving Agile Retrospectives: Helping Teams Become More Efficient. United States of America: Pearson Education, Inc.) introduces an even more structured retrospective that builds on Derby’s and Larsen’s architecture. It has more stages and more details that need to be considered. Loeffrel’s retrospective is more for teams who have a dedicated coach acting as a supervisor. Coaches specialize in Agile methods and all that comes with those, including retrospectives. It might be too big of a task for a producer or other team members to get familiar with Loeffrel's ideas since they have their own areas of specialization.

"Set the Stage" is the first step. According to Loeffrel, the set the stage should be mainly for bringing everyone present mentally. Even if people are physically present, it doesn't mean they have left their tasks, work stress, or other pressing things out of their minds for this session. Set the stage tries to minimize the risk of somebody being so lost in their thoughts they forget to contribute to the conversations. Establishing a safe work environment is also important. Loeffrel uses the saying "What happens in Vegas, stays in Vegas" to describe the atmosphere that should be reached. All that is said at the session, stays in that session. Part of the safe working environment is also something that Loeffrel calls "team character". Esther Derby and Diana Larsen introduced this idea too, though their version was simply about establishing the rules of the session. To Loeffrel, team character is something that is built for many purposes and is present at all meetings and sessions the team has. It defines the rules of team work in any situation. Team character is usually created outside a retrospective session since it will take time. Communication, rules of conduct, length of meetings and sessions, and sometimes even software tools to be used, are all defined in the team character document. If the team is large, it is advisable to develop a team character. In smaller teams, that are by nature more agile, this might be a bit more difficult since people are constantly busy and might work within multiple disciplines. Derby's and Larsen's version would be more suitable for smaller teams since it is created quickly just before the session and no one needs to be in charge updating and revising the documentation. In Loeffler's retrospective the theme or topic has also been predetermined by the lead, but it can also change during the set the stage step if the situation demands it, based on the failed experiments.

"Check Hypothesis" step is for checking the experiments of the previous retrospective. This is an important step for continuity and commitment. The lead goes through all the experiments decided on in the previous retrospective and asks the team what they have done to carry those out. Therefore, it would be good to have metric data available. If, for example, the team decided at the previous retrospective that they should have more outings together, it would be good to have the number of outings per month or week they have had previously. Only then the improvement can be clearly seen. In case some of the experiments failed to be carried out or otherwise didn't make a difference to the situation, the next part of the retrospective should be dedicated to answering the question why. There might be some issues with the team's working practices that need to be fixed.

“Gathering Data” is a crucial part of retrospectives. Loeffrel argues that both hard facts and personal feelings should be collected during this step. The method could be either letting people ponder for a while by themselves or discuss the topic as a group. Or even a mixture of both, which we often use in the game industry. Collecting personal feelings makes the people more invested and gives hard facts notes perspective. The overall mood of the team is something that affects the whole process and should be acknowledged. The team can just discuss the theme, but visualization is always a good idea. This way the whole team sees the notes and can easily refer to them during the conversation. It also helps people to remember what was important after the session. It’s important to thank the whole team after this step since it is often hard to talk about personal feelings in a work space.

Teams often, without realizing it, only treat the symptoms of problems, not the causes. This can lead to a cycle of problems renewing themselves and people using quick fixes to avoid confronting deeper issues in the team work. The “Generate Insights” step is for collecting causes of problems which means asking the question “why”. The team needs to establish why certain things are happening to uproot them and get rid of them once and for all. “5 Whys” is a method that Loeffrel suggests in which the team chooses a problem and then asks, “why is this happening”. Then when they find an answer, they ask why again until this cycle has been completed five times. This is a good activity for a small team, but bigger teams usually need a bit more guidance.

“Define Experiment” is the phase that leads to progression. According to Loeffrel, the word “experiment” is important to use during this step. It gives the team the impression that the change suggested is not necessarily a permanent or mandatory part of working practices. It is just a test to see if it worked. This way the team has more sense of control when they enter back into everyday workflow. If the experiment doesn’t work, in the next retrospective the team will decide what happens to it. It can be abandoned all together or it can be tried again. It is just important that the experiment goes through the “gather insight” step and it is found out why it failed. The word is also important because it takes a more scientific approach than, let's say, “trying out”. There should be some way to measure the changes in workflow between retrospectives and then make notes about any improvements. If nothing is documented and the experiment are “just a bit of fun”, they won’t help the team. The lead should keep track of the documentation and share it with the team.

“Closing” step should be a short ceremony that collects the feedback of the retrospective and lets everyone leave with a smile on their face. Or at least with a bit of hope that the workflow will now have improvements. Loeffrel argues that the results of the retrospective could be gathered in metric form into a ROTI chart. A ROTI chart has an x and a y axis and numbers one to five on both. One means that the session was a total waste of time and five means that the session was a success. The team then places an x marking their mood position on the chart. The team should use the closing to celebrate achievements made during it and thank each other for the time spent. The lead then thanks everyone and asks for more feedback after the retrospective.

3.4 Kerth's Model

Norman L. Kerth (Norman L. Kerth (2001) Project Retrospectives: A Handbook for Team Reviews. New York, NY, USA: Dorset House Publishing.) focuses more on the goals of retrospectives rather than a specific structure. He is more interested in who are the people you invite, how you manage time, and how you guide your team to achieving a specific goal. Kerth gives most value to getting to know the team beforehand. The supervisor needs to establish why the team wants and needs to have a retrospective. These two don't necessarily go hand in hand. It might be that the team is more focused on small details or single processes that are bothering them, when in fact a retrospective tries to collect more of a wide picture. That is why the supervisor should be skilful in organizing retrospectives. He/she needs to know what to ask from the team before the actual sessions starts. According to Kerth, good questions could be "Why do you want to have a retrospective?", "Who the learning is intended for", and "Can you tell me the most important details about the project?". Those questions help finding an underlying goal for the retrospective that is predetermined by the supervisor. Kerth gives a few example goals that could guide the direction of the session.

"Capture Effort Data" works best at the end of a project. It quantifies the efforts made during the project. Metrics are important which means all data is translated to numbers. All data from the whole project is gathered and used when designing the next project. In the game industry we don't really use this goal since it is more important for us to stay agile during the whole process. Instead we have postmortems (Wawro, Alex (2015) 10 seminal game postmortems every developer should read. http://www.gamasutra.com/view/news/238773/10_seminal_game_postmortems_every_developer_should_read.php. Haettu 16.8) in which we reminisce about all the ups and downs of a project.

The "Get the Story Out" goal is meant for big projects in which it is almost impossible to know all that is happening in each area of the process. Even in smaller teams and projects everyone rarely knows each story that is happening. The goal would be getting all the most important stories heard from each area of the project so that every field synchronizes at least during retrospectives. It also helps to put all the pieces in context and understanding other disciplines that might be involved in the project. It is also important to tell people why certain decisions were made and acted upon. Often the team members might only see the consequences without knowing the reasons and this might cause tension in the group.

To Kerth, ”Improving Workflow and Management” is just one goal among others. In the game industry this goal is the priority of most retrospectives. To achieve this goal, according to Kerth, the facilitator of the retrospective needs to start by reflecting what has occurred during the project and then the team works with him to see what could be done better. This type of goal supports the Agile philosophy perfectly since it aims to maximize productivity and minimize waste. The project workflow is optimized a bit further after every retrospective and the experiments are team specific. Every team is different which means every team needs its own working methods. Kerth warns about the “We need to fix this specific person” attitude. The goal should never blame individuals for problems. Instead, it is supposed to focus on specific workflow and process issues that individuals are just part of. It is never good for the work culture to try to find culprits, especially not in a group situation, where it would feel like an attack.

”Capture Collective Wisdom” is another goal that Kerth introduces that is more like a postmortem in the game industry. Kerth suggests that since many companies don’t have long term teams that projects are given to but rather have teams built around new projects, the wisdom from previous projects must somehow get passed on. If there’s no documentation, it often leads to individuals forgetting the lessons learned from the past. This knowledge doesn’t need to be metric. Well done documentation makes sure that the knowledge stays in the company and then individuals can take their share of it to their new team. Of course, this team needs to establish its own work culture and workflows, but they need to start from somewhere. If they don’t need to reinvent the wheel, they can focus on optimizing their processes through retrospectives.

According to Kerth, many companies do not use this goal since they don’t think it’s appropriate to discuss feelings at the workplace. They also might not have safe ways to deal with those feelings. However, humans are humans, and it is very likely that all sorts of feelings surface at the office whether those are formally discussed or not. Kerth argues that retrospectives could be used to repair the team if tension or grudges have taken their toll. He thinks that there are many ways to do this safely without damaging the working environment even more. To him, retrospectives are an opportunity to recognize underlying tension and then find ways to fix issues with the team.

In simple terms, this goal is for clearing the air. Kerth doesn't offer any examples for safe ways to deal with feelings. He insists that often high performing teams can find the ways by themselves.

If the team only focuses on solving the next problem, something that Kerth calls “post-project blues” might set in. Kerth argues that most people working in the software industry tend to be problem-solvers and often focus on issues in the processes to optimize them. Even though this is natural for them, it doesn’t mean they don’t need to be reminded about their accomplishments occasionally. An ”Enjoy the Accomplishments” - type of retrospective works best when it’s held a few times in the middle of the projects and at the end. However, most of the retrospectives should follow some other goal. Celebrating also helps the team move onto the next project with a sense of pride. After all, every finished project is a victory.

4 Facilitation

4.1 Basics of Facilitation

According to Pepe Nummi (Pepe Nummi (2007) Fasilitaattorin käsikirja: Tarina siitä miten Ykä Hirvi vie ryhmän tuskasta tulokseen. Helsinki: Edita), various facilitation techniques are used in many organizations that have a lot of specialized teams working in them. Facilitation enables gathering information from different specialized areas and collecting them into a neat bundle. Facilitation comes from the Latin word *facil*, which means easy. Many problems in team work are caused by broken workflows and processes, and facilitation has been invented to fix those problems. To completely understand facilitation, one must first make a separation between working processes and the topic of discussion. A facilitator's job is to guide the team through an efficient process to get the results they are looking for in their area of discussion, but to be neutral about the topic. Nummi argues that facilitation should only be used when the team already has the knowledge to solve the problems they are facing. They only need someone to guide them through the pit falls. However, a facilitator can join conversations by asking guiding questions if the conversation seems to get stuck.

Pepe Nummi divides facilitation into three categories: the first generation, the second generation, and the third generation. According to Nummi, the first generation was introduced by Kurt Lewin who was invited to help arguing ship builders during world war two. Back then, psychologists worked with the team to see how they could improve their working processes. The second generation was created to support cultural aspects of the team. These teams were driven by communal decision making which is typical for Agile teams. The second generation also made a difference between idea creation and decision making which meant that it favoured activities such as World Café and facilitated conversations. For example, in World Café ideas are collected first as a group, and only after are those ideas prioritized. Then decisions can be made on the highest priority ideas. The third-generation facilitation methods support complex decision making and strive for better understanding of group dynamics. Since nowadays many teams and companies have specialists from various fields, it is important that those fields can communicate with each other and with other teams. Main methods that the third-generation uses are Open Space, Dynamic Facilitation, and Ideologist. I will focus mainly on the practices of the third-generation, since those are the most used today.

4.2 Dynamic Facilitation and Structure

You only need a few things to use dynamic facilitation at a retrospective session: a whiteboard, post-it notes, and sharpies. The pens need to leave a mark that can be seen across the room, that is why sharpies work well. First, the whiteboard needs to be divided into four sections: facts, worries, core question, and solutions. During a typical dynamic facilitation meeting, anyone can shout out their idea at any time. The facilitator then writes the idea on a post it and sticks it on the board under a category where the idea fits most. The main priority is to collect the ideas into a visual presentation. Categories are there to help organize the post-its, but it is not advisable to start debating where a post-it should go if it's not clear which type it is. Putting the ideas in the right boxes is only a secondary objective.

The facts category is reserved for post-its that consist of facts related to the topic of discussion. Facts, as in everyday life as well, are something that cannot be debated about. At least not in a logical sense. A fact could be such as "the money we have now for this project will last till October if we are working with a team this size" or "our hardware cannot support these visual effects".

Worries are the second category. Every project has issues to deal with and often these issues are accompanied by worries. Worries could be such as "what do we do if the money runs out" or "how do we design good looking visual effects without the proper hardware". Worries are often the ones that need solutions for the boat to stay afloat.

Core questions are something that cannot be answered at a single session. They often need a whole other meeting for themselves to find any meaningful answers. Actually, "what do we do if the money runs out" could be a core question too since it needs a lot of strategizing. This is a good example of a post-it that might fit into two categories. Another core question could be "how do we find a budget for new hardware".

The solutions category lists answers for worries and even core questions. Core questions cannot usually be resolved during one session, especially if this one session is reserved for another topic. Solutions can be discussed and prioritized afterwards and made into experiments in the case of retrospectives.

The great thing about dynamic facilitation is that it doesn't stop at the first idea or solution but keeps evolving the ideas. It also gives the team an opportunity to paint a wide picture of the project and a chance to voice any worries that might have gone unnoticed before. Dynamic facilitation does have its weaknesses though. It might take a long time to get to the root of some of the questions, even multiple meetings. People usually need time to evolve their ideas and get inspiration from others before they can think about the problem by themselves. The method also might leave the shyest or quietest people outside of the conversation. Anyone can add to the conversation anytime they want which leads to the loudest ones being the centre of attention most of the time. Therefore, retrospectives often use the "Me, We, Us" method.

4.3 Ideologist

The "ideologist" method is similar to the "Me, We, Us" method, often used in the game industry, but is a bit more complex. First is the individual work step in which the participant writes their own ideas on paper. Next, the team divides itself into groups of three. In these groups, all members steal the best ideas from each other on their own paper. At this point a short walk break is in order and then new groups of three are created. Stealing continues and this cycle with the walking breaks is repeated about two to four times. In the last group of three, the members choose the best four ideas and write them on a piece of paper. The collected ideas are then brought up in a group discussion, but not introduced any further, since the ideas are already familiar to the teams. The last step is then to discuss the ideas, prioritizing and evolving them.

According to Nummi, the "ideologist" method removed the problem of some people not contributing to the discussion. Unlike in the dynamic facilitation method, in the ideologist method the team members had the time to write their own ideas down without anyone else stealing the stage. In my experience, some of the louder team members might find this method a bit too slow, but I still think it removes one of the most tedious problems in collective decision making. I still haven't found a better method to do this than giving the time for everyone to work in silence on their own first.

4.4 Leader Skills

Norman L. Kerth (Norman L. Kerth (2001) Project Retrospectives: A Handbook for Team Reviews. New York, NY, USA: Dorset House Publishing.) argues that there are two factors you should consider before leading a retrospective: your facilitation skill level and your understanding of the project you are reviewing with the team. According to Kerth, the facilitator should always be an outsider. Since it is the facilitators job to observe the process and the environment, summarize outcomes, and encourage team members to explore ideas, it is crucial that the facilitator can remain neutral. Best way to ensure this is to have an outsider as a lead. The facilitator also needs to be technically competent to understand the projects architecture and vocabulary among other things. This way he/she can confidently lead the discussion with the team and have an understanding where the discussion is going. People skills are also vital. The lead needs to be able to solve conflicts, make sure that the situation is safe to talk about all kinds of ideas and feelings, encourage everyone to express themselves, and help make unconstructive feedback constructive. Kert suggests though that if you cannot find one person who has both technical and people skills, you could have two facilitators.

Marc Loeffrel (Loeffrel, Marc (2018) Improving Agile Retrospectives: Helping Teams Become More Efficient. United States of America: Pearson Education, Inc.) focuses more on the learning of how to become a skilled retrospective facilitator. We all need to start from somewhere. He describes a few first steps that a beginning retrospective leader should take to support his/her first time on the driver's seat.
Loeffrel holds visual retrospectives in high regard since it helps the team to establish a common language. He describes how he has often seen teams discussing a topic but haven't been able to see the whole picture until it is visualized. Loeffrel argues that the leader, or anyone else from the team, doesn't have to be a great artist to participate a visual retropective. If your handwriting is clear, your picture present quite cleraly what they should, and you love different colors of post-its, you are good to go. Mike Rodhe shows in his book, *The Sketchnote Handbook: The Illustrated Guide to Visual Note Taking*, fun ways to make visual notes. He also argues that sketch notes help to clarify hierarchy, create a relaxed atmosphere, and produce a clear visual mapping.

Unlike Kerth, Loeffrel doesn't think that the facilitator needs to be from outside the team. However, he does acknowledge that there are differences between internal and external leaders. The team needs to decide first which one is better suited for its purposes. Loeffrel also argues that external leaders are usually better since they are more likely be neutral and do not have a pre-established role in the team. Motions might fly high during the retrospective and it is easier for an external leader not to take part in them. He/she is not part of the group dynamics which means he/she is also not bind by them. Internal leader might also want to play his/her role in the team even though, as a retrospective facilitator, he/she should be just an observer. Sometimes it's better to have a professional who has much experience leading retrospectives. Sometimes the company might even have one already hired, and in this case he/she is probably working outside the team. Internal leader might be the better option though if the company is trying out retrospectives for the first time and doesn't want to invest in an external facilitator. Sometimes the team also wants to have more control over their own retrospectives and already has a seasoned veteran in their troops. The most important thing for an internal leader is to try remaining neutral. This is probably quite impossible but there's no hurt in trying. Loeffrel describes techniques such as reminding yourself which role you are playing, and rotating retrospectives from one team member to another. It might also be a good idea to visualize your role by, for example, writing it on a whiteboard. This way even if you change your role at some point, team knows the role you are playing. Rotating makes sure that you get to play your own role in the team most of the time. It is easier for the lead if he/she knows that they will get their turn to participate as well as observe the retrospectives.

To Loeffrel, retro cycle should be interconnected in which case it can easily be goal-orientated. The point of retrospectives is to optimize the workflow processes which means it cannot start from zero every time. Experiments should be discussed and re-evaluated every time. Documenting and changing the goals into a metric form help supporting the interconnected cycle. Lead is the one responsible that such documentation exists and is available and easy to access to all the participants. Photographing the process is also vital. This helps the team to remember what was discussed and is big part of visual retrospectives.

Esther Derby and Diana Larsen (Derby, Esther & Larsen, Diana (2007) Agile Retrospectives: Making Good Teams Great! Wash-ington, D.C.: Pragmatic Bookshelf.) take a more management approach to retrospective leading. They focus on what areas the leader should managed before and during a session. To them changing the agenda of retrospectives is also important since it is the nature of the human mind to get bored.

Managing activities is, of course, an important step before starting one. This step comes to us quite naturally but there are some pitfalls we need to be aware of. Explaining the purpose of the activity is also something the team is often interested in since people like to often know what kind of territory they are walking on and that is worth crossing in the beginning with. When you are introducing an activity for the first time, write down what you are about to say and practice it at home. It is important that you learn to be very clear with your instructions. After the activity, it should be debriefed. This helps the participants to see what they get out of the practices you have chosen. Derby and Larsen introduce a simple four step method to debrief an activity.

1. Sensory input. Ask the question “What did you see and hear?”
2. Ask the team how they responded to the inputs and if they were challenged or surprised by them.
3. Ask questions like “What insight you have about this?” and “Does this tell you something about the project”.
4. In the last step you have established a link between the activity and the project which means that you have completed the whole learning cycle. Now it is time to ask, “What is something you might do differently”.

The above debriefing example follows the same flow as the structure of the whole retrospective introduced by Derby and Larsen: set the stage, gather data, generate insights, decide what to do, and close the retrospective. It is the basic learning cycle of the human mind. The debriefing part should take 50% to 100% of the time of the activity.

To Derby and Larsen, managing group dynamics means making sure that everyone has a safe space to talk. The leader needs to observe who might be talking over others or stealing the spotlight, who hasn’t said a word yet, and when the conversation gets stuck. To get the quieter people to talk, you can ask questions such as “We haven’t heard from Samantha yet. Would you like to add anything Samantha?”. To the louder ones you can try to talk to before the session and ask them to tone down. If they still won’t realize that they are taking over the whole meeting, gently try to ask other people to join the conversation and ask the louder person to listen for a bit. It is important to stay calm all the time. The leader doesn’t need to create more tension when there might already be some in the room. Derby and Larsen suggest asking three types of questions in the case that the conversation gets stuck. Derby and Larsen also suggest asking other people, especially those who have been thinking more than talking, what other kind of questions we could ask to make things move more quickly.

- What have we tried before? What happened? What would you like to happen differently?
- If we had that, what would we gain?
- Have you ever tried this a different way? What happened?

Derby and Larsen also talk about something called the “I”-language. For example, if someone says “You were late which is why I couldn’t make the build!”, the leader can ask the person to use “I” language in which the person talking is not allowed to use the word you and only talks about his/her point of view. For example, the previous sentence could change into a form “I noticed that this commit was late so I couldn’t make the build”. This way all the unnecessary blaming is avoided. However, sometimes blaming and tears are unavoidable. Derby and Larsen give basic instructions to handle human emotions.

- *Tears:* Give a box of tissues and ask: “What happened? Can you share it with a group?”
- *Shouting:* Hold a stop sign with you hand and ask: “I would really like to hear what you have to say but I can’t when you are shouting. Can you please lower your voice?”
- *Stomping out:* If a person leaves, let he/she go. Then ask the team: “What happened?” The person may re-enter to the session, but it might be a good idea to talk about what happened briefly.
- *Inappropriate laughter or clowning:* Ask the team: “It seems that every time I touch this topic someone makes a joke. Do you know why this is?”
- *Uncharacteristic silence:* Ask the team: “You seem a bit tired and quiet. May I ask why is that?”
- *Currents beneath the surface:* Again, it is best to ask team what is going on if they seem fidgety and uneasy.

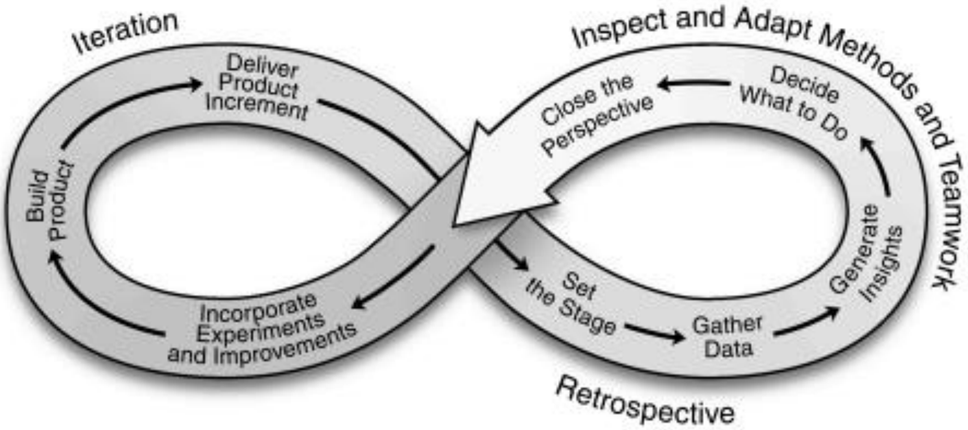
Managing time is often quite har when you have a big group that is full of energy. If the team seems to get really excited about a certain activity, just keep an eye on the clock and how much time you have booked for that activity. When the time runs out, ask the group what they would like to do. Explain to them that the overtime used on this activity will take up some time from the next. However, if they consider this activity be more important than the next, the current activity can be continued. As a leader you might need to change the next activity to a shorter one though.

During the session it is vital that you are managing you as well as your team. If you get anxious during the retrospective, let the team go for a break. Then ask yourself a few questions.

- “What just happened?”
- “How much was inside me and how much was outside me?”
- “How did the group get here?”
- “Where does the group need to go next?”
- “What are three options I have for next steps?”
- “What will I offer the group?”

These steps remind you that you have a strategy and you can get back to the session easily. It also helps you to find the causes of the current situation and treat them. Then you might not get overwhelmed again.

Picture 2. Cycle of retrospectives



5 Designing my first retrospective

5.1 Purpose

The purpose of my first retrospective was to get me used to organizing them. I was not that familiar with the structure yet, but I was about to take over the reprospectives from my more experienced colleague. For the team that participated the retrospective, the purpose was to find out how the team was feeling after going through some tough times. After establishing the issues in the team, we could then focus on fixing the team dynamic.

5.2 Attendees

My colleague asked me to organize a retrospective for our smaller, Stormdivers, team. I limited the attendees just to the development team. For example, marketing was excluded since the team felt that the discussion wouldn't have stayed as relevant for them if the marketing department participated.

5.3 Place and time

I chose one of our more relaxing meeting rooms as the place for the session. Retrospectives should be had outside of the normal office space and have a place that supports a safe atmosphere. Two hours was the time limit for the retrospective since longer than that made people exhausted. The cycle was already set to six weeks which meant the date was set to six weeks after the last retrospective.

5.4 Theme

The theme was "Sanity". The team had had trouble with funding and schedules which had cause variety of issues. We tried to find out why the milestones weren't met in time and then find solutions.

5.5 Cycle

The cycle had been set to six weeks before I started at the company. The team followed Agile project management philosophy’s Kanban method which meant they didn’t have sprints like in Scrum. The cycle length sometimes changed but that wasn’t ideal, routines should be set and followed. I started to enforce the six-week cycle.

5.6 Activities

I picked the activities I used from Loeffrel’s book Improving Agile Retrospectives (Loeffrel, Marc (2018) Improving Agile Retrospectives: Helping Teams Become More Efficient. United States of America: Pearson Education, Inc.). He had a good framework for a first retrospective and I only changed couple of activities that suited better for the team’s needs. I borrowed some activities from my colleague that I found useful.

5.7 Break

My colleague taught me that every retrospective should have something nice served during its break. Housemarque’s traditional treat had been doughnuts and I ordered those as well. Two hours is a long time which is why a break in between was vital. People needed time to cool down.

6 Example 1: Tourist Retrospective

6.1 Brief and Role

This was my first retrospective at Housemarque, which is why I was only observing and participating in the activities. My colleague, who already had years of experience organizing retrospectives and other team events, was leading the activities. At university we had only organized very classic Scrum style retrospectives and I didn't know anything about the underlying structure or its purpose. Therefore, I didn't learn as much as I should have during this retrospective. The retrospective was held for one of teams called Tourist. The team was quite big, around 40 people, which meant that the session could be no shorter than two hours. The theme my colleague had chosen was "Empathy".

6.2 Basic Structure

The basic structure in the classic Scrum style retrospective is very different from the newer, more team focused structure. During my observation of this retrospective I didn't even realize how the underlying structure worked. I focused on the activities and how data was collected through them. The main goal was the same as in every retrospective, to find issues and habits that work in the team's workflow. Agile teams are constantly trying to optimize their workflow and make the team more productive without sacrificing the team's wellbeing.

My colleague had the basic structure running underneath all the activities even if I didn't realize it. He followed it trough in the classic order without repeating any stages: setting stage, gathering data, generating insights, defining the next experiment, and closing. In every retrospective he has organized, he has always skipped the check hypothesis stage. He explained that he likes to give each retrospective a specific theme and not to tie it in with the previous ones. In the classic Scrum style retrospective, it's vital that the retrospectives are aware of each other and follow up developments between retrospectives. My colleague's approach to retrospectives is a good example how nowadays the focus is more on the team's wellbeing and current mood rather than the optimization of the product's pipelines. The themes are human experience focused, such as empathy, stress control, and communication.

6.3 Preparations

We had had a company wide open day just before the Tourist team's retrospective. The goal of the open day is to make everyone in the company aware of news and issues that have been circulating inside the project teams. Since there are multiple teams working on various projects and in various office spaces, it's important to make sure that at least once in a month all the important information is synchronized. The team members can suggest topics that they would like to talk about that day and then each is given a timeslot that might be shared with few other topics. People then go to a conversation group of which topics they find interesting.

On that open day one of the members of Tourist team wanted to talk about empathy and motivation since he felt that there were some issues within the team in that department. The conversation went overtime which led to the idea of taking the topic to the next retrospective.

The tourist team had grown a lot during the past half a year which meant that some of the team members felt they were losing the homely feeling of a small team. Before the growth everyone used to know each other, and the sense of team spirit was stronger. Now was the time to try to bring that same feeling to the bigger team, or at least something similar. It's natural for people to divide themselves into smaller groups but we needed to do something to maintain the wellbeing of the team.

The agenda my colleague had devised was playful but stayed in the basic structure: setting stage, gathering data, generating insights, defining the next experiment, and closing. Going through all the points with a team of around 40 people would take around two hours. He had scheduled a 15-minute coffee break at the 50-minute mark.

For setting stage, he had chosen quick questions round in which all the team members wrote on post-its what they would like to get out of this retrospective. Next step, spring wedding, was about collecting all the good and bad of the last sprint or, in other words, the time between this and the previous sprint. Fish bowl was reserved for the discussion about the central theme and maximizing follow through defined all the changes the team wanted to implement to their workflow and habits before the next retrospective. Closing was done by asking each team member if they were surprised or pleased about the revelations of this retrospective.

My colleague had collected as many sharpies and post-it notes as he could since those were needed for quick questions, spring wedding, and maximize follow through activities. The spring wedding also required different colors of post-its. He had also collected five chairs for the fish bowl activity. He only used two white boards, one for the activities and the other for the agenda which was visible during the whole retrospective.

The room was the one we use during some of the more relaxed meetings and workshops. It has a collection of beanbag chairs and comfy sofas. Retrospectives should be held somewhere else than the office where the team works most of their time. It offers a welcome change to one's surroundings and makes us think differently. Also, the more relaxing the space is, the better. People need to be in the right mind set to open-up about their ideas and thoughts.

Traditionally our company has ordered doughnuts for retrospective coffee breaks since those are a rare treat. On everyday basis we have fruit and protein bars at the office since those are healthier options. My colleague ordered us a nice assortment of doughnuts and made us some coffee.

6.4 Activities

6.4.1 Set the stage: Quick Questions

The ”Quick Questions” activity sets up the mood and gathers people in the same space not only physically but also mentally. My colleague had chosen “describe in one sentence, what would you like to get out of this retrospective” topic for the quick questions round. Each person could only write down one goal.

Each team member wrote down their personal goal of this retrospective on a post-it note and placed it on the whiteboard. When everyone was done, my colleague read aloud each post-it. If any of the goals was unclear, he asked the person who wrote the note to clarify its message. The goals were mostly related to the central theme of empathy and motivation, such as “getting to know the new team members”, “finding out why we are stressed”, and “reducing tension”.

The “Quick Questions” activity can be used in the setting up the stage of the retrospective easily since it only takes a few minutes and the supervisor of the retrospective can define the question or the topic him/herself. If the question is tied to the central theme, it immediately makes people think about the topic and everything that has happened in the team related to it. Everyone should be mentally present at this activity and not thinking about the task they have waiting on their desk. This activity can also give the retrospective supervisor an idea of what the major concerns in the team are and maybe even allow him/her to change the agenda to accommodate those if needed.

6.4.2 Gather data: Spring Wedding

In the “Spring Wedding” activity, the board is divided into four categories: “Something New”, “Something Old”, “Something Borrowed”, and “Something Blue”. “Something New” stands for habits and processes that the team would like to try, “Something Old” are the habits the team have found useful in the past and would like to keep, “Something Borrowed” is for habits the team has seen other teams using and would like to try, and “Something Blue” is any issues that the team has found in their current workflow.

My colleague chose to use the “Me, We, Us” method to gather data in each of the categories. The point of this method is that it made it comfortable for even the shyest team members to voice their ideas. In the “Me” stage the team members write down on post-its what they think should be under each category. During this phase people are working on their own, writing down only their personal ideas and feelings. In the “We” stage the team is divided into small groups, from 3 to 4 people in each. The group then shares the ideas they wrote in the previous stage and discusses about them. As group they choose which ideas are taken to the next and final stage. The final stage, “Us”, is when each group shares their ideas on the whiteboard. Each team takes a turn on the board to stick their notes on it and to explain the through the process behind those. Now each category should have notes in them.

The Tourist team found that two of the categories were very similar and there was no reason to have both. Those categories were “Something New” and “Something Borrowed”. They said that all the ideas under something new are most likely borrowed from somewhere since nowadays not many ideas are completely unheard of. You could also say that it doesn’t really matter if the ideas are new or borrowed since they are all new to the team. They all could be under the “Something New” category. On the other hand, the categories are tied to a theme of a wedding and therefore are supposed to make the activity a bit more fun. Just having categories of “New”, “Old”, and “Issues” would take away from the playful atmosphere which also contributes to the more relaxed structure of the retrospective. Some people also found the first stage, “Me”, a bit stressful since they had just started working in the team and didn’t know much about its culture or workflow yet. They thought it would have been easier to start from the group stage. However, the danger with this is that some people won’t contribute at all to the conversation. The aim of retrospectives is to get the whole team to talk and skipping the “Me” stage fights against this idea.

For the preparation of the next activity, the team chooses the most interesting post-its from the board. This can be done in many ways but this time my colleague chose to pick the ones that had most in common with the main theme. He only chose three central ideas since more than that would have made the next activity confusing and too long for the time slot.

6.4.3 Generate insights: Fish Bowl

Explaining the Fish Bowl activity to a new team is never simple but so far, we have found it productive when the team realizes its full potential. Five chairs are placed in a circle in the middle of the room. This will be the fish bowl. The idea is that only four people can sit in the circle at the same time, leaving one of the chairs empty. People sitting on the chairs discuss a predetermined topic, which in this case has been chosen from the outcomes of the pervious activity. If all the four chairs are full and someone from outside the circle wants to join the conversation, he/she can sit on the fifth chair. Since one chair always needs to be empty, someone needs to leave the circle. Usually this is someone who has already sat in the circle for a while and has contributed a lot to the conversation already. We have always had people willingly leaving a chair when someone else joins in. We have never had to intervene as supervisors.

The facilitator works as a recorder during the activity. He writes down all the good ideas and solutions the team comes up with in the conversation. Those are written again on post-its and put on the board. He/she also helps the team if the conversation suddenly stops. He/she can ask guiding questions or introduce a new topic if no one from the team does it. Otherwise the supervisor doesn’t participate to the discussion.

The first topic that we had chosen for this activity was “how to increase empathy at the work place”. We never actually got to the other topics since the team got really excited about this one. The conversation started to flow from the start and the people in the “Fish Bowl” changed frequently. The team wanted more activities together, silence in the office at least at certain times, personal space, their special needs considered when designing the open office organization, polls about the stressfulness of the work, and more time to work within their own discipline. The most problematic areas were the open office solutions and tasks that came from the outside of the team members' own expertise areas. In the game industry we tend to have creative minds that are quite different from each other. Some need silence, some need to be able to talk about their ideas as often as possible. Some are distracted by movement around them, some like to walk around when they are trying to solve a problem. The office was too small for the growing team which meant the personal space got smaller. While there were no perfect solutions for these problems the team decided to put up separating walls for those who couldn't work with people moving around them all the time. All the discussions could be had outside the office, in the kitchen. This way the office stayed a silent workspace. An idea about colourful cubes was introduced but so far not implemented. Those cubes would have worked like a traffic light: green means you can talk to me, yellow means only bother in emergencies, and red means please no distractions. Unfortunately, the team still has problems with the tasks and those not being from their expertise area. When the company was smaller it was up to anyone who could to grab the next task but now the team members were more specialists rather than generalists. Also, the communication hadn't caught up with the growing team size so people were chasing information they didn't know where to get. The only experiment we could think for this problem was that every time someone gets a task that disrupts their own work, they can put it on the Kanban board. Then it's up to the producer to see who should have that task.

Some of the team members criticized the Fish Bowl because it excluded the people outside the circle from the discussion. Only the people in the chairs can talk during the activity and if you have something to add, you need to sit on a chair. Some felt that if they only had a sentence or two to say, they should be allowed to comment from the sidelines. The main purpose of the Fish Bowl is to control a conversation in a big team. If 40 people talk at the same time, the chances are that the room gets divided to smaller groups anyway since people are commenting from all sides. It also makes it hard to join since there are many loud members in the team. Fish Bowl limits the chance for off topic comments and shouting on top of other voices. Overall though, the Fish Bowl has been popular in our teams.

6.4.4 Define experiment: Maximize follow through

During the last activity the supervisor has collected ideas and solutions to the problems that the team has been discussing. To make sure that the solutions are implemented, there needs to be the definition of the next experiment. A popular method for this phase is the “Who, What, When”.

The supervisor divides the whiteboard into three categories: who, what, and when. He/she then places the solutions gathered from the last activity in the “What” column. The team members then have a chance to pick a solution they would like to oversee. For example, “putting up room dividing walls” is usually taken up by a member of the QA department since office changes are part of their area in our company. However, people are volunteering for these tasks and anyone who would like to have that task can take it. They write their name in the “Who” column next to the task and then define the time period they need to finish the task. When all the tasks have been assigned to someone, the next experiment has been outlined. The experimental solutions are then run till the next retrospective to see if they work. If they don’t, it is the aim of the next retrospective to either fix them or just quit these experiments. If they do, they are added to the team’s routine and workflow.

It is important to assign responsibility for each solution experiment to a specific person. If the solutions are just left for the whole team to take care of, nobody will make sure they are implemented. Everyone is expecting someone else to take the initiative or they think someone already has. The solutions are also easily forgotten if they are owned by no one. People are not doing this because they are lazy or don’t think these issues are important. It just clarifies and supports communication if everyone knows who oversees what.

We have used the “Who, What, When” method at every retrospective since it’s clear and easy to explain. It also allows people to volunteer for tasks, which make them more invested. It would be good to try other methods as well since it keeps the retrospectives more interesting.

6.4.5 Closing: Pleased/Surprised

The ”Pleased or Surprised” activity is very simple. In turn every team member says if they were pleased or surprised by the outcomes of this retrospective. This gives some idea to the supervisor how the team feels. More detailed feedback is collected afterwards since that is not the point of the retrospective and there is no reason to keep the team away from their actual work for longer than necessary. Last thing the supervisor does is thank the team and ask them to give him feedback either in written or verbal form later. A retrospective should always end on a positive note and the team should leave the room happy, or at least hopeful. This time everything seems to have gone according to plan.

6.5 What I learned

My experience of retrospectives was quite limited at this point. I learned a lot about controlling the activities and making the retrospectives more fun. I had never thought that these kinds of activities could be used as a framework to collect data from the team. In the previous retrospectives I had been part of the only aim was to see what went well in the last sprint, what didn't work, and what new things we should try. It was very productivity focused and didn't really concern itself with the general feel of the workplace. People weren't as eager to voice their opinions and all the retrospective repeated the same formula. Now I learned that changing the agenda occasionally and pacing the retrospective itself with different kinds of activities, keeps people awake and interested.

Example 2: Stormdivers Retrospective

6.6 Brief and Role

My colleague had organized a retrospective for the bigger team in our company. He left the smaller team, Stromdivers, for me. This was my first time leading a retrospective at Housemarque, and the first time I used Loeffler's structure. I later noticed that there were things I didn't realize I should prepare for and that is why the introduction to activities was sometimes confusing. The theme of this retrospective was "Sanity". We tried to find out how the team was feeling about the project since they had gone through some tough times.

6.7 Basic Structure

I used Loeffrel's (Loeffrel, Marc (2018) Improving Agile Retrospectives: Helping Teams Become More Efficient. United States of America: Pearson Education, Inc.) structure even though my colleague favored Derby's and Larsen's structure. I saw it as an important step to use Check Hypothesis – phase. Retrospectives need to be looking back since that is why there where created, to evaluate the past actions of the team. This also helped to see if the team was committed to try changes that were agreed at the last retrospective.

6.8 Preparations

I booked a space for the retrospective that had a relaxed atmosphere and made sure I had enough pencils and post-its. I made sure I understood how to visualize as much of the retrospective as possible. I also practiced introducing the activities since I had trouble performing in front of an audience.

6.9 Activities

6.9.1 Set the stage: Weather

The first activity, “Weather”, was about figuring out the mood of the team. First, I introduced the game. I drew five symbols on the board that portrayed different weather conditions: sunny, overcast, cloudy, rainy, and stormy. The team then translated their feelings into these weather conditions and drew the symbol on a post-it. Everyone did this individually, not as a group. The ready post-its were stuck on the whiteboard. In this case, a lot of people felt either overcast or cloudy. This could mean that the project was doing fine despite of a few problems, though more likely people avoided showing extreme emotions. It’s not uncommon for people to feel that showing your true feelings, happy or sad, makes them vulnerable. This might happen even without realizing it since not many people want to be the only one to place their post-it to sunny or stormy, though there were a few of those as well. Middle ground seems like a safe option.

6.9.2 Check hypothesis: Looking Back

When my colleague held retrospectives, he often left out the check hypothesis stage. I haven’t asked why this was, even though it would be a good idea. I can only guess that is because of the time limit or because his retrospectives focus more on general welfare of the team. If we could ask Norman L. Kerth’s (Norman L. Kerth (2001) Project Retrospectives: A Handbook for Team Reviews. New York, NY, USA: Dorset House Publishing.) opinion, he would probably say that my colleagues set the “repair damage to the team goal”. I, however, decided that looking back to the previous experiments and seeing if the team had fulfilled their commitments, would be an important step. I had learned that from the Scrum retrospectives we had had in smaller teams. I knew this step couldn’t be too long which is why we just quickly went through with the team what had been agreed as experiments last time and if those had been completed. There were two unfinished experiments which meant we needed to step up our game. If the experiments are not planned to be carried out, those can be left out of the commitments. Still, I was happy with what the team had done during the past six weeks. We agreed that all the experiments should be adopted to our pipeline and workflow.

6.9.3 Gather data: Mad, Sad, Glad and Dot Voting

Mad, Sad, Glad is an activity in which the whiteboard is divided into four areas: things you feel mad about, things you feel sad about, things you feel glad about, and one box is reserved for thank you notes. I chose to use “Me, We, Us” method for this activity that I had learned from my colleague. I explained to the team that first we would write on post-its our own notes that fit in the four categories we had on the whiteboard. It would be great if those notes tied in with the theme, but all post-its that had anything to do with the project were accepted. The first part was done as individuals and I gave everyone five minutes to write down their thoughts. After five minutes the team asked for extra time, which is why I gave them five more minutes. I usually give less but this time it seemed that five minutes for the whole task was way too short. After the first step we moved to the “We” group part of the activity. Now I gave the team 10 minutes to discuss as a group which four notes they would like to put on the board, one per each category. I had forgotten to mention to the team how many notes they should have written in the previous “Me” part which cause a bit of a problem. People had too many notes to share with their group and this took a lot of time. I had to give the group session a few extra minutes.

6.9.4 Generate insights: Cause and Cure

I divided the post-its in mad and sad categories into themes and then asked the team why these things are happening. I let people shout out ideas and then collected those on post-its. The only problem with this method was that the shy team members didn’t get to speak. Luckily the team was small, and most people got to talk in the end. When there were enough ideas in the cause section, we moved on to finding solutions to those problems in the same method.

6.9.5 Define experiment: Who, What, When

After dividing the board into who, what and when sections, I placed the solutions listed in the previous activity in the “What” column. All the participants then had a chance to choose which one they would like to be responsible. They also needed to time box their tasks since this made it easier to follow.

6.9.6 Closing: One Word

For closing, I asked the participants to describe the retrospective in one word. This proved to be quite challenging, which is why we change it to one sentence. The feedback was mostly positive but there was some confusion about the post-it note calculation.

6.10 What I learned

I learned that there are many little things to consider when it come to organizing a retrospective. From this point on I always gave a note limit to each task and made it scalable for different team sizes. I also noticed that all questions should be clear and rather just asking one question at a time.

7 Example 3: Retrospective Workshop

7.1 Brief and Role

My colleague at work had promised to hold a workshop at Kajaani University of Applied Sciences and asked me to go with him. I was still technically studying at the university and still had to do my thesis presentation. Therefore, we decided that I should run the workshop for the students and get my presentation done at the same time. My role at the retrospective workshop was a facilitator but also a teacher. I was well prepared but a bit nervous of how my teaching style would work with people I don't know personally. However, it was easier to stay neutral and objective when I was outside the student circle. I chose ADT, attention deficit trait (Huotilainen, Minna & Peltonen, Leeni (2017) Tunne Aivosi. Helsinki: Otava), as our topic since it has been talked about in the news lately. ADT, in short, is a attention deficit disorder that people develop when multi-tasking at work a lot.

7.2 Basic Structure

I was using Derby's and Larsen's (Derby, Esther & Larsen, Diana (2007) Agile Retrospectives: Making Good Teams Great! Washington, D.C.: Pragmatic Bookshelf.) structure since this time there was no need of looking back in time. The students were from different teams and didn't have a project together. We made up an imaginary project and, of course, that had no history to refer to.

7.3 Preparations

Since this was the first time for me teaching a class, I made sure I'd taken everything into account. I made the plan scalable because I didn't know how many people would be attending. For example for note calculations I made a plan A and a plan B, depending on how big the group would be. The workshop organizers got me the space I needed and all the equipment. The whole workshop was six hours long which is why I scheduled two coffee breaks and a lunch break. Usually retrospectives last less than two hours, and I noticed that some of the activities couldn't be played with a medium size group in that time frame. I also had to involve theory which I communicated through slides at the very end.

7.4 Activities

7.4.1 Set the Stage: Expectations

In the beginning of the workshop I asked people to write on post-its what they are expecting from this retrospective. My colleague had introduced this activity to me at the first retrospective I was part of. After everyone had written their expectations down, the post-its were put on the white-board and I read them aloud. This got everyone mentally ready for the retrospective.

7.4.2 Gather Data: On the Boat

"On the Boat" activity was used to collect data. I drew a boat on the whiteboard that had sails and an anchor. The sails represented things that help the team to concentrate at work and the anchor represented things that disrupt our concentration. I asked the team to use "Me, We, Us" method to collect thoughts on post-its and place them on the whiteboard. I set the note limit on three since we were going to play " 5 Whys " next and for that only couple of the notes could be chosen. This game was chosen because it is very simple and it suited well with the ADT theme.

7.4.3 Generate Insights: 5 Whys

I learned "5 Whys" activity from Loeffrel's book Improving Agile Retrospectives (Loeffrel, Marc (2018) Improving Agile Retrospectives: Helping Teams Become More Efficient. United States of America: Pearson Education, Inc.). We chose four notes from the last activity for this one, two from the positive side and two from the negative side. It ended up being too many and therefore I would recommend choosing only two, one from positive and one from negative side. We used "Fish Bowl" activity as a framework and discussed about the chosen notes. We first took one of the notes and asked "Why is this happening? Why is this the way it is?". The idea was to do 5 why cycles with each of the notes and trying to find the root causes. It is better to treat those causes rather than just symptoms since any problems are less likely to reappear this way. I let the participants just talk and wrote down all the ideas they came up with.

7.4.4 Define Experiment: Who, What, When

In “Who, What, When” activity I first asked the participants to use “Me, We, Us” method to find solutions to the root causes we collected in the previous activity. I specified that they should find as concrete solutions as possible. We then put those solutions on the whiteboard and the participants could voluntarily choose a solution they would like to make as their task. We only briefly discussed about the time boxing since these tasks weren’t meant to been see through. The main point was just to introduce the retrospective framework to the students.

7.4.5 Closing: One Sentence

We only had little time left after all the other activities which meant I tried to get the feedback of the session as quickly as possible. I asked everyone to say in one sentence what they thought about this retrospective. The feedback was very positive and for all the participants this was a completely new way to organize retrospectives.

7.5 What I learned

Before the workshop, I had always been an internal retrospective facilitator. This time I was external and I really noticed the benefit of it. It was much easier to concentrate just on the facilitating and not the topics that were discussed. I would recommend external organizer to all teams if the resources allow this.

8 Conclusion

Retrospectives should be part of any work environment that considers its employees valuable. It gives employees a chance to build their own project management style and support democratic leadership. IT- and software industries are already using new ways to ensure their employees stay happy and content.

Smaller teams only require a simple Scrum retrospective structure, whereas bigger teams need more complex structures to manage all the communication channels they have. Complex structures also able us to dig deeper and find the root causes of problems. It is important to evaluate those since it gives us a chance to get rid of the issues once and for all.

Retrospective facilitator needs to be objective and the easiest way to achieve this is to get someone from outside the team to lead. As an insider you know more about the team, but it is quite hard to stay objective. Therefore, I would recommend trying both, insider and outsider, to see the difference it makes.

Overall retrospectives are a great tool for team building and finding new innovative ways to lead a project. They also create a work environment which feels safe for everyone. Nowadays when companies aim for more and more transparency, retrospectives offer an easy tool to achieve such goal.

Resources

Books:

Norman L. Kerth. (2001). Project Retrospectives: A Handbook for Team Reviews. New York, NY, USA: Dorset House Publishing.

Esther Derby and Diana Larsen. (2007). Agile Retrospectives: Making Good Teams Great! Washington, D.C.: Pragmatic Bookshelf.

March Loeffrel. (2018). Improving Agile Retrospectives: Helping Teams Become More Efficient. United States of America: Pearson Education, Inc.

Pia-Maria Thoren. (2017). Agile People: A Radical Approach for HR and Managers (That Leads to Motivated Employees). New York: Lioncrest Publishing

Pepe Nummi. (2007). Fasilitaattorin käsikirja: Tarina siitä miten Ykä Hirvi vie ryhmän tuskasta tulokseen. Helsinki: Edita

Minna Huotilainen & Leeni Peltonen. (2017). Tunne Aivosi. Helsinki: Otava

Internet pages:

Chad Walters (2012, 8. toukokuuta). Toyota Way Principle #3: “Pull” Systems. Haettu 10.8.2018, sivustolta LeanBlitz internetosoite: leanblitzconsulting.com/2012/05/Toyota-way-principle-3-pull-systems/.

Clinton Keith (2007) Agile Game Development. Haettu 1.8.2018 sivustolta GDC Vault internetosoite: https://www.gdcvault.com/play/639/Agile-Game. Haettu 1.8.2018

Alex Wawro (2015) 10 seminal game postmortems every developer should read. Haettu 16.8 sivustolta Gamasutra internetosoite: http://www.gamasutra.com/view/news/238773/10_seminal_game_postmortems_every_developer_should_read.php.

Liitteet