DYNAMICS USE IN SCRUM RETROSPECTIVES TO PROMOTE PARTICIPATION AND INCREMENTAL IMPROVEMENTS

Kipper, L. M.¹, Assmann, C. L.¹, Nara, E. O. B.¹, Benitez, G. B.²

Universidade de Santa Cruz do Sul¹

Universidade Federal do Rio Grande do Sul²

Abstract

Currently the continuous improvement is extremely important for organizations that seek to remain competitive in the market. The practice adoption promotes the constant evolution and business processes improvement. This article presents a case study of an organization that uses the Scrum for project management's team of software development and, as expected, this framework uses the retrospective meetings to review the work progress, raising necessary adjustments and actions definition to promote continuous improvement. To ensure all team members’ participation at retrospectives meetings are used several types of dynamics. In this study were detailed dynamics already used and the obtained results through its use. Besides, it was realized a results survey of all year’s retrospective at project closure in order to verify if the identified problems still exist or the retrospectives were effective and the measures listed during these meetings were executed and achieved the goal that is the problem solving and continuous improvement.

Keywords

Agile Methods; SCRUM; Retrospective; Continuous Improvement; Dynamics.

1. Introduction

The practice of continuous improvement in organizations, nowadays, is a premise to maintain its competitiveness in the market. Occurring in a structured or not, improvement programs must provide to the enterprises the conditions for effect changes, making them flexible in the face of changes in social and economic contexts (5).

Agile methods, in the environment of Information Technology - IT companies make software developers respond quickly to changes (16). Both when it comes to changing requirements of systems and technologies during the development and when it requires work process changes looking the continuous improvement.
This study provides an introduction to agile methods, particularly the Scrum framework. Besides, brings the importance of the retrospective applications and reports, in practice, the use of dynamics in these meetings that promote the evolution of the team's work of software development.

2. Agile Methods

In 2001, a group of seventeen experienced professionals in the area of software development met to discuss what techniques to use and what actions could improve the performance of projects in this area.

During the discussion the group realized that certain techniques used and respected during the project execution contributed to these worked well. The result of this perceptions set was the agile manifest.

In this manifest were defined four values that define the agile approach in the software development:

1. **Individuals and their interactions** more than processes and tools.
2. **Software running** more than comprehensive documentation.
3. **Customer collaboration** more than negotiating contracts.
4. **Respond to changes** more than following a plan.

These rules define what should be valued in software development, but it does not mean that the other items should not exist. According to the authors of the manifest, item two, for example, does not mean that the systems will not have more documentation, this must exist, but the software in operation should be the priority.

The rules also value people and the interaction between them, bringing that the negotiations and the discussions are essential for agile development, rather than the use of tools and process definition.

The agile approach provides the customer stay close the system developing team in order to form a partnership or collaboration and leave a formal requirement of customer and seller software.

The change answers come to address something that really happens during the development project of any system: they are constant changes. Therefore agilists maintain the constant delivery of software and thus return the client also faster. So, is abandoned the idea to deliver the system to the user only when is finished, ignoring the changes and new ideas that might occur along the way. Occurring the need for changes, the team can answer these quickly from the definition of a new direction or minor adjustments in the development.
The agile manifesto (7) brings the essence of a set of good practices and values created from the experience of professionals in order to improve the performance of software development projects. So, is the definition of agile philosophy. The methods that use this philosophy are called agile methods. These methods consist of practices and specific elements that, when adopted, promote agile philosophy, for an example as pair programming, requirements definition constant, daily meetings, software pieces demonstrations in a short time to the interesting parties with the development cycles (3).

According Armony (2), in recent years companies are increasingly looking to adopt agile practices for the development management of their systems, it can be noted that was clearly established itself as the new trend for software development. The SCRUM framework is widely used for software development in Brazil and worldwide (2). This is detailed in this study.

3. SCRUM

Scrum is not a methodology, is a framework (12). This means that Scrum does not say exactly how to apply it, but expect you to adapt the process to the reality of your business (6).

A framework is a set of concepts used to solve a problem or specific area. Each component of the framework has a purpose and is essential to the success and use of Scrum (12).

In beginning Scrum was inspired by an article of Nonaka and Takeuchi (14), titled "The New New Product Development Game" that first used the name "Scrum", which is a Rugby move (11), to describe approach to developing of new products used at the time in companies in Japan and the United States.

According to Schwaber (12), Scrum assumes the premise that systems development is complex and unpredictable to be fully planned in the beginning. However, one should use the empirical and planned model in short periods rather than accomplish prescriptive detailed planning of the project.

Marçal (8) agrees with the statements of Schwaber (12) and complements that the method is also based on principles such as: small teams of seven to nine people maximum, with requirements that are little stable and short cycles which divides the development time in intervals in maximum of 30 days.

Scrum is based on empirical theories which claim that knowledge comes from experience, where decisions are made based on what is known (13). This framework predicts an incremental and iterative approach, that is, at each iteration, which may be a period of two to four weeks, one part is delivered, testable by the system user, in other words, a product increment. Three pillars support the implementation of an empirical process control: transparency, inspection and adaptation (13).
1. **Transparency**: The relevant aspects of the process should be visible to those responsible for the results with a common understanding of what is being seen. For example: there should be a common definition of "done" among those responsible for developing the product and those who give ok to the work.

2. **Inspection**: Scrum users should frequently inspect the artifacts, work progress and detecting variations, adapt.

3. **Adaptation**: Is the fact adapt, make adjustments as soon as problems are identified.

A Scrum project starts with defining the product vision that will be developed (12). The view lists the product characteristics set by the client, and if any, its assumptions and constraints.

Then the Product Backlog, or list of features that the system should have, is created together with users. Kniberg (6) complements that is where it all begins, the Product Backlog is a list of requirements, user stories (requirements are written in the terminology of the customer), use cases, things that the customer wants, or simply can be called Backlog Items.

This list should be prioritized, i.e., what is most important to the customer should be at the top. The role responsible for maintaining and updating this list is the Product Owner or "Dono do Produto" that represents the final user and it can be a person sent by users or a member of IT staff.

The beginning of each iteration is accomplished a planning meeting which aims to group definition as to what will be developed in the next two weeks. Each feature set is estimated by the team, starting from the top, i.e., the most important. And, according the estimative results of how long it will be necessary to each feature being developed are defined how many will integrate the work that will be executed in the next iteration.

Marçal (8) complements that the team collaboratively defines what may go into development of the next iteration (or sprint), considering its production capacity. To control the activities evolution of the iteration by the team, the Scrum proposes the use of a picture, where is visible the implementation stage of the iteration functions.

In Figure 1, can be seen an example of this organization table, with activities at the beginning are in column "Undone" and, as they are initiated or developed will be moved to the next column. Even in the table of Figure 1, in the right corner Burndown Chart is used to verifying the number of features that remain to be developed along the iteration, i.e., how much lack of work.

Throughout the iteration execution, the team makes a daily meeting (Daily Scrum Meeting) of no more than 15 minutes and usually in front of the table. The purpose of this meeting is to monitor the progress of the work, negotiate the remaining tasks of the iteration and schedule other
necessary meetings. At the daily meeting each team member answers three basic questions (SCRUM_GUIDE, 2011):

- What I did on the project since the last meeting?
- What will I do until the next meeting?
- What are the barriers (if any) that are in your way?

At the end of the iteration, if the team was able to achieve the goal that is the completion of tasks, is realized the review meeting (Sprint Review Meeting). During this meeting, the team presents the results achieved by the Product Owner and / or system users, for they approve or not what has been developed. If accepted the product increment, begins the testing period in the system with forecast return of possible adjustments until the end of next iteration.

The facilitator of all Scrum meetings is Scrum Master, this is also responsible for establishing and teaching the practices and rules of the framework and make involved people follow them. Its responsibility also seeks to resolve interference or impediments that the team is suffering thereby removing potential barriers to project development (3).

![Figure 1 – Table of SCRUM. Source: Berni (3).](image)
In Figure 2 can be viewed Scrum cycle, from the list of features or use cases, prioritized to enter the next iteration. The "Iteration Plan" is the result of the planning meeting which summarizes into a task list. And at the end of the iteration, the product increment occurs, i.e., is delivered to users over a piece of software that they use and, as a consequence, give the fastest return on possible adjustments.

![Figure 2 - SCRUM Cycle. Source: Schwaber (12).](image)

Scrum provides a lightweight set of practices and dynamics to project management of software development (3). A key feature of this process is continuous improvement, this occurs from retrospective meetings, held at the close of each iteration, always after the review meeting. The retrospective had not been mentioned during the Scrum presentation because it is part of the main subject of this article and it is presented in the next section in detailed order to demonstrate their importance in the improvement and success of projects of this nature.

4. The SCRUM Retrospective and Continuous Improvement

The retrospective meeting is an opportunity to the Scrum team inspect and create an improvement plan that must run during the next iteration (SCRUM_GUIDE, 2011). Armony (2) adds that this meeting is essential because it serves for the staff review the latest iteration, with connexion to its development process, and plans their adaptation, within the Scrum rules, in order to make it more effective in the next iteration. The Scrum Guide (SCRUM_GUIDE, 2011)
highlights that the Scrum Master should encourage the team to improve. It also describes in steps the hindsight purposes and details this process a little more:

• Inspect how was the last iteration with respect to people, relationships, processes and tools.
• Identify and sort the main items that were well and potential improvements.
• Create a plan for improvements to the program which the Scrum Team does.

Kniberg (6) states that teams, in general, do not appear prone to make retrospectives. However, the Scrum Master should ensure that these happen because without a stimulus, the teams do not make retrospective and move on to the next iteration.

The author points out that these meetings are extremely useful, because this is the only chance the team has to improve, the act of making the meeting is because of the team ideas and actions for improvement starting from the team, which ensures that will be better accepted rather than something idealized only by an integral.

Derby and Larsen (4) finalize conceptualizing the retrospective as a special meeting.

When we speak in retrospectives, something that comes to mind is: a special meeting where the team withdraws after finalizing and deliver the additional work to verify and adapt their methods. Retrospectives enable the team to learn to direct their actions for change and not focus only on the development process, but the team and its needs (4).

A complete Scrum cycle, with all scheduled meetings, including the retrospective meeting, promotes a process of continuous improvement. Mesquita and Alliprandini (9) consider that improvement is a process of problem solving. And complement stating that the perfect doesn't exist in practice, the essence of continuous improvement towards this quest is evolving constantly and conscious, overcoming obstacles, learning from mistakes and successes, sharing each knowledge, contributing not only for personal and individual growth, but also professional and organizational. As the Scrum framework does not indicate exactly how to apply retrospective so that they are effective, productive, and that all the team members participate, were sought and applied various dynamics in order that they happen as expected.

4.1 Dynamics used in retrospectives

Aiming to provide all team members participate actively in retrospectives, some dynamics were applied in team meetings during the project development. Each of them will be detailed in the next sections.

4.1.1 Triple nickels
Derby and Larsen (4) suggest the dynamics Triple Nickels. This was selected because it meets the requirements to provide time to the single conclusion for later group discussion. The following steps are part of this dynamic:

1. The project team forms a circle around a table, if available.
2. Each member team writes on a sheet of paper and individually, three characteristics or three events that occurred in the last iteration.
3. Then, the paper with the three facts, are exchanged with fellow in right.
4. This fellow will review the facts that another colleague wrote, adding his/her opinion. The paper sheets will circulate until they pass by all into the group.
5. In the end, each member of the group presents his/her facts with comments written by colleagues.
6. The facilitator of the meeting, in this group is the Scrum Master that will write in the frame these facts and, through team discussion, classify as positive or negative point of the last iteration.

As a meeting result, it has a list of negative and positive points. This is sent to the team shortly after the meeting. For the negatives are listed its causes and consequences. If possible, actions are defined and should be adopted by the group at the next iteration. In other cases, you do not get a consensus, or the issue cannot be resolved in a meeting or discussion. The Scrum Master is responsible for initiating or giving way to actions, studies, meetings, finally, to the actions that seek to solve the problems listed in the list as a negative point and that there are not possible to find solutions through the retrospective meeting.

4.1.2 Plan of action plan retrospective

The aim of this retrospective format is defines actions that can be implemented immediately to address problems perceived by staff. At first, each team member has ten minutes to, individually, define how many actions as possible and write them on post-its. After this first moment it will be formed pairs with the intention of sharing the thoughtful actions before individually. Each pair dismisses the actions that were not as priority and could be discarded, reaching five remaining actions.

Each pair meets another, now with the objective of defining the four most important of the ten remaining actions of each formed group. And finally, the whole group is assembled, each quartet presents their pre-defined actions. The group must reach a consensus and list three priority actions. These actions form an action plan, and for each are set a responsible. The expected completion date is the end of the next iteration.
In the next retrospective meeting of this team these actions will be reviewed by the group to check what was developed or initiated. Actions that have not been initiated are brought and discussed the reasons by the group and responsible.

This process is based on teams with a number of members multiples of two, however, with some creativity you can use with teams of all sizes. This means that you can start with trios, after the individual moment, for example, form the group of six people and then gather the whole group (1).

The differential formed from this retrospective meeting is that the consensus about the importance of the actions listed is essential that occurs and all members of the team being involved from the beginning. The goal is to define only three actions, these must be the most important, because a larger number of actions may cause loss focus in the team and cannot do what is priority for the group. Set a large list of actions to be undertaken in the next iteration is a common mistake in developing action plans (AGILE RETROSPECTIVE, 2009).

4.1.3 Top five retrospective plan

The dynamic Top Five Retrospective Plan (1) forecasts that, before starting the meeting, the facilitator, in case, the Scrum Master, request that participants bring five issues that need to be resolved to improve the work and development of the team. Besides the issues, bring to the meeting possible solutions for each problem listed.

At the beginning of the meeting, the Scrum Master writes on a blackboard or a flip chart, problems and possible solutions, creating a column or sheet for each problem and asks pairs to being formed (preferably people who do not normally work together) to analyzing each.

Each pair have a limited time at three minutes each, they have to analyze and discuss each situation and suggest new solutions or complements existing ones. This process ends when all pairs have passed through all the problems thus become familiar with all situations.

In the end, each member of the meeting may vote in three solutions, that are most interesting, appropriate and/or priority. After that, the winning solutions are highlighted and the facilitator asks a volunteer to be responsible for starting or run each.

The result of this meeting, namely the three actions chosen by the team should be visible for the group, including with the identified responsible for they can track the progress of the same during the next iteration.

4.1.4 Start, stop, continue, more of, less of wheel

This format playful retrospective meeting aims to get the quick return of the team on what needs to be continued and what needs to be adjusted in the next iterations (1).
The following steps are part of this type of meeting: the facilitator draws a large circle on a board or flip chart and divide this circle into five pieces, like a pizza.

On each piece write the words: Start=começar, Stop=parar, Continue=Continuar, More of=Fazer mais, Less of=Fazer menos. Doing the following questions, respectively:

1. What should we start doing and maybe we have not done yet?
2. What should we stop doing that is not helping and is hindering the group?
3. What is working and we should continue?
4. What should we do more?
5. What should we do less?

The next step is to encourage the meeting participants to contribute to each category. The method indicates that silence is used by the facilitator, who hopes people talking.

A variation of this used dynamic by this group was the use of roles. It was placed each "piece of pizza" on a sheet, separately. The sheets have gone through all meeting participants and, individually, each one could define what should start, continue and what needs to be adjusted.

There was a moment of discussion on each quoted item in the papers, and after the consensus of the group, the facilitator drew the circle with the pieces and added what was discussed and validated by the group. The part of the circle named "Start" contains what should be initiated. At the end of retrospective, each item of this part shall have a responsible for continuing with this and the actions for something to happen.

### 4.1.5 Questions retrospective

This dynamic is based on the format of the Triple Nickels. The difference is that questions are used to guide the three facts to be listed by each participant (1). The three questions to be written on each sheet of paper are:

1. In this iteration, which unusual things happened?
2. In this iteration, what we've done very well?
3. In this iteration, which we did and what we should avoid in the future?

Each member responds and moves to the next colleague to contribute with a comment, complementing what was written. Similarly, Triple Nickels that in the dynamic process ends when each piece of paper go through all meeting participants.

It is important to set a time when the paper changes member to member, the method suggests three and five minutes. Completed this step, each member is asked to read aloud their answers, as
well as the contributions of colleagues. Each item is discussed by the group, and the main points are written on post-its and stuck in the frame in order to give visibility to each item.

The frame should be divided into three parts, according to the initial questions. The inclusion of post-its with content already validated by the group is interesting because it preserves the initial response of each member, but also reveals the colleagues contributions.

4.1.6 Everyday retrospective

A Everyday Retrospective (1) indicates that is usually applied in a group of seven to nine people. The dynamic indicates that, to inspire the team, it is interesting to bring information about what happened in the last iteration (10). For example, the burndown graphic, the calendar, the realized or pending task list in the iteration. Ask the group to remember and analyze the last iteration, based on brought information.

The next step is dividing the whiteboard into three parts with the following titles: Good= what was good in the last iteration, Bad= which was bad and Ugly= what can be improved for future iterations.

The group is asked to write on post-its the facts that were good, bad, and what they can improve in their respective columns. If the facilitator wants to limit the time, it can be defined only in five post-its by group member, or let people bring new ideas as they arise. Each item is read aloud to the group, the duplicates are grouped and each one is discussed, which can bring new ideas that had not been thought.

Done it, the members have the right five votes each. The facilitator should make clear that the vote should be given to the most important items in the design of each. Even one same item can be voted more than once.

The facts most voted should be selected, the amount depends on the speed of time, i.e., how many actions the team can run in the next iteration, which would be two weeks. Normally, three items are selected.

For each fact most voted it should have one responsible, and based on the discussed and selected items, the team must set goals to long and short term.

Finishing the retrospective the facilitator asks to the group how the meeting was productive for them and what could be improved next time.

5. Results and Discussion

5.1 Survey and analysis of the project retrospectives
In the end of the project were sought results of all retrospective meetings realized during the year in order to classify which of these listed problems still persisted and what were resolved during the iterations development.

The period of the survey is the same of the project: from January to August 2011. A member of the team, which participates in all team activities, led this survey already discarding the problems that certainly have been resolved.

All the problems that existed some doubt or possibility of not being resolved and it would be interesting discuss in a group remain on the list. As the problems that aren't resolved.

To facilitate the analysis of the issues, these were grouped by category, such as: communication, technical issues, testing and system documentation. The list was inserted into a table, and next to the raised questions were added columns: Start, Stop, Continue and Improve, in order to each question being discussed and classified into one of these opinions again. In Figure 3 is presented part of the document containing the problems list, already categorized.

<table>
<thead>
<tr>
<th>Communication - Team Interaction</th>
<th>Start</th>
<th>Stop</th>
<th>Continue</th>
<th>Improve</th>
<th>ACTIONS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Team communication/ team interaction.</td>
<td></td>
<td></td>
<td></td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>Opine practice during the meetings (feedback).</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Share the difficulties as soon as possible in the daily meetings.</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Negotiation task.</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Exchange activities among developers when necessary.</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Technical discussions, project meetings.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Technical discussions.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Meetings about the doubts before development.</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>- Planning Meeting to justify</td>
<td></td>
</tr>
</tbody>
</table>

**Figure 3** - Part of the document summarizing the retrospective. Source: Elaborated by authors.

Besides the aforementioned columns, was added the column "Actions" in order to define new activities to seek improving a given situation. Every raised action earned a responsible, a group member. These actions must be performed in order to improve for the next projects that the team will work.

To control the development of these actions the team chose to use a board, now in electronic format, where they were registered. Thus, the actions are visible and "waiting" to be developed.

**5.2 Controlled activities by scrum team**

The found solution by the team to electronic control of the actions from the retrospective survey of the year was the Trello tool (15). This allows a visual control, which refers to the already done control of development tasks of the iteration in the physical frame.
The Trello (15) is an available tool on the Internet, very simple and aims to facilitate collaboration among members of a work team. In Figure 4 can be viewed the screen of the tool, already with the registered activities and two team members in the participants field ("Members"). To nominate a person for a job, just drag and drop the team member in the desired task. Likewise to change the task status from "In Development" to "Ready" for example, just drag the task to the required list and drop with the mouse.

![Figure 4 - Task Controls in Trello tool (15).](image)

The place is configurable, i.e., the names in the lists are defined by the team that can create new frames (Boards), depending on the tasks to be controlled. The use of the Trello tool (15) is currently being tested by the Scrum Master. So far the tool has met the criteria of simplicity, easy to use interface with drag-and-drop and customizable place according to the needs of the group.

6. Conclusion

We believe that the approach about the continuous improvement through the retrospectives meetings and use of dynamics to promote the participation of all team members were presented in a simple and practical form. The goal is that the contributions left by this study may be enjoyed by other companies that use Scrum in their software development process. Reaffirm the importance of retrospectives happen to promote the work progress and set a time for each step during the meeting to not lose focus and make it productive.
The differential of using dynamics for the retrospective meetings is that it promotes the participation of all. The members who used to only listen or contribute little need to participate, activities are distributed equally to those involved. Even members who normally participated way too much end up having a relatively equal to the others. However, it can be tiring for the group, at each iteration, using the dynamics. It is important not to have a pre-defined format and always leave members free to bring questions they deem relevant.

Most retrospective meetings are set the actions and listed on a chalkboard or flip chart. It is important that this record stays visible throughout all the next iteration and only be updated in the next iteration. Thus, the group ends up reviewing these settings and can discuss the reasons for something not being done or analyze if the results were satisfactory when some action was performed.

7. References


(15) Trello – Site da ferramenta para controle de tarefas. Available in: https://trello.com/


Correspondence
Kipper, L. M., Universidade de Santa Cruz do Sul, PPGSPI (Programa de Pós-Graduação em Sistemas e Processos Industriais). Avenida Independência, 2293 – zip code: 96815-900, Santa Cruz do Sul, Brazil. Phone: (51) 3717-7632. E-mail: liane@unisc.br. URL: http://lattes.cnpq.br/1353629457597527

Assmann, C. L., Universidade de Santa Cruz do Sul, PPGSPI (Programa de Pós-Graduação em Sistemas e Processos Industriais). Avenida Independência, 2293 – zip code: 96815-900, Santa Cruz do Sul, Brazil. Phone: (51) 3717-7632. E-mail: vipchofer@vipchofer.com.br. URL: http://lattes.cnpq.br/9679187770642146

Nara, E. O. B., Universidade de Santa Cruz do Sul, PPGSPI (Programa de Pós-Graduação em Sistemas e Processos Industriais). Avenida Independência, 2293 – zip code: 96815-900, Santa Cruz do Sul, Brazil. Phone: (51) 3717-7632. E-mail: elpidio@unisc.br. URL: http://lattes.cnpq.br/285238667149033