What are the most important challenges associated with transitioning to Scrum for game development?

COMP150 - Agile Development Practice

1804356

November 11, 2018

Project managers are often tasked with implementing Scrum to teams that are either unfamiliar with or simply against Agile Practices. This paper discusses some of the more common issues raised by teams and supervisors alike and attempts to provide some insight into possible solutions

1 Introduction

Today's software industry is faced with ever greater challenges, varying from increased pressure on build release frequency all the way to client interaction and feedback. Naturally, the need for new development practices arose and as a consequence more and more companies are transitioning (wholly or partially) to Agile Development Practices, especially Scrum. The purpose of this paper is to identify what the greatest challenges to team productivity are during this very transition period and suggest feasible solutions that

can be implemented by project managers to dampen the negative effects the transition can have on a team.

2 Coaching the masses

Regardless of the degree of transition to Scrum, however, it is most often the case that teams will encounter several setbacks in the process. In other words, one could state that Tuckman's model of Team Transformation loses its linearity in more practical situations [1].

2.1 A new mindset

One of the greatest challenges project managers face when implementing Agile is overwriting the Waterfall mindset of the team with an entirely different one. However, this is no easy task, as it is very common for members of the teams to simply discard the changes Agile or even actively resist the transition [2] This is especially true when dealing with multidisciplinary teams composed of designers, artists, programmers, etc. As it was observed, the issue stems from the fact that the benefits of Agile practices cannot be observed immediately or rather, while the teams agree on their benefits, "the team members develop cold feet when they really had to practice them"[3]. It is generally agreed upon that the best course of action in such cases is to implement the practices gradually, in small incremental steps, so team members can familiarize themselves with the new practices[1], [4]. Additionally, any conflicts that arise can be resolved at this stage, before any progress-impeding escalations.

2.2 Productivity impact

As previously stated, the linearity of Tuckman's model of Team Transformation (Figure 1) is lost in practice. The reason is that in spite of the team managers' best efforts, teams will inevitably oscillate between different phases. Additionally, it should be noted that the greatest variations are usually encountered during the Storming and the Norming phases, according to several case studies [1], [3]. These changes can be attributed to either the team or the project managers pushing for changes that need time to get get used to. In other words, the transition period can be considered as an "experimental" phase in which each team adopts and adapts to changes at its own pace. As it should be expected, some practices prove to be beneficial for the team, while others prove to be detrimental and the team has to roll back the changes.

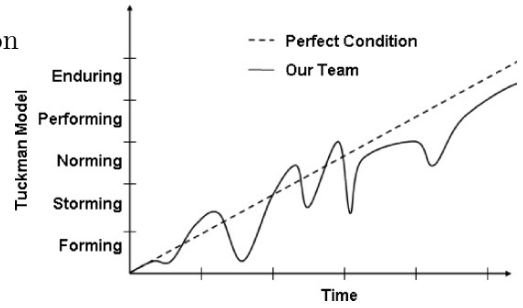

Figure 1: practice vs. theory

Problems arise, however, when a discrepancy exists between the members of each team, such as, but not limited to: physical location (separate rooms, departments, countries etc.), which in turn leads to the use of web-based meetings, language barriers and even individual personalities (as is the case in most game studios). In such situations, several approaches can be taken: some studies or papers suggest that teams should be encouraged to sort such problems by themselves [1], [3], [4] while others put forward the argument that they should also receive some external guidance (such as taking a "gamified" approach) to reduce the impact of such conflicts on productivity [5], [6] or improving upon the use of distributed teams. Each of these approaches has been proven to be effective in certain contexts, and therefore project managers should make a decision

based on their own circumstances. It should be noted, however, that trial-and-error is likely the best way to find suitable practices.

3 Evening out

During the later stages of adopting Agile practices, several papers note the introduction (and the demand for) self-organization within teams [1], [3], [6]–[8], which can take several forms: leader roles that change by rotation, agile coaching, scrum masters and coordinators. In the case of large scale teams, where Scrums-of-Scrums are implemented, teams will develop their own approach to communicating between each other and will likely require a minimum amount of supervision, while maintaining a certain level of individuality. Project managers should welcome this change and should try to not impose strict limitations on inter-team interaction (such as separation of members or supervision from someone unfamiliar with the team). Another aspect that should be tracked is the addition or removal of team members, which can easily create tensions within the team. This means that teams ought to be well planned and future changes in the team structure, as well as implementing distributed teams, have to be taken into consideration beforehand.

4 Case study: Guild Wars 2

During a GDC conference in 2016, the Developers of the popular MMO Guild Wars 2, ANet, held a presentation regarding the challenges encountered during their iterative development process of their latest (at the time) expansion pack, Heart of Thorns. Several challenges are addressed in the presentation such as the number of feedback sources involved in the development loop, the balancing of adding new features and fixing old bugs and adapting the scope of each development loop.

4.1 Breaking it down

The team was able to take an interesting approach to breaking down each development loop (iteration) by breaking it down into 3 "Levers" that can be pulled to maximize product value: Time, Product Size and Feedback, as seen in Figure 2. This way, each issue can be visualized by scaling each of the levers to represent its nature. This approach is especially effective in pinpointing issues that are hindering progress and encouraging prompt action from the teams.

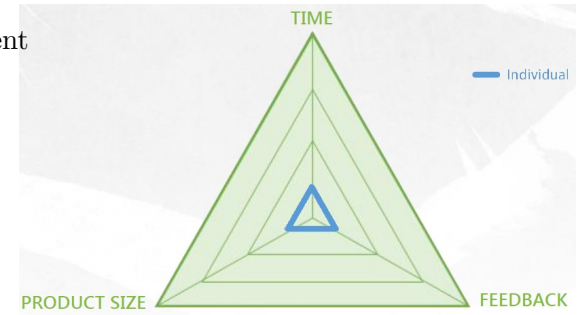

Figure 2: the three levers

4.1.1 Remarks

This approach proved to be especially effective in the context of keeping the client(s) involved in the development loop, as the number of feedback sources kept increasing as the development continued. Furthermore, the team continued to create release versions until the profit margin was broken, ensuring both a playable (and mostly bug-free) experience, and a time and cost effective release of the product.

5 Conclusion

The basis for any transition to Scrum ought to be centered around the teams' interests and should be tailored by its members, for its members. While controlling the transition should be avoided, coaching it is recommended, and encouraging the team to form its own version of Agile practices is a must.highlighting key points, limitations, and outstanding questions. It should not introduce any new content or information.

References

[1] E. C. Lee, "Forming to performing: Transitioning large-scale project into agile", in *Agile 2008 Conference*, Aug. 2008, pp. 106–111. DOI: 10.1109/Agile.2008.75.

[2] *Four types of Resistors when adopting Agile by Mike Cohn*, https://www.mountaingoatsoftware.com/blog/four-types-of-resistors-when-adopting-agile/comments, Accessed: 2018-11-11.

[3] K. Sureshchandra and J. Shrinivasavadhani, "Moving from waterfall to agile", in *Agile 2008 Conference*, Aug. 2008, pp. 97–101. DOI: 10.1109/Agile.2008.49.

[4] R. Jochems and S. Rodgers, "The rollercoaster of required agile transition", in *Agile 2007 (AGILE 2007)*, Aug. 2007, pp. 229–233. DOI: 10.1109/AGILE.2007.59.

[5] L. Benedicenti, F. Cotugno, P. Cianfrini, A. Messina, W. Pedrycz, A. Sillitti, and G. Succi, "Applying scrum to the army - a case study", in *2016 IEEE/ACM 38th International Conference on Software Engineering Companion (ICSE-C)*, May 2016, pp. 725–727.

[6] M. Paasivaara, C. Lassenius, and V. T. Heikkilä, "Inter-team coordination in large-scale globally distributed scrum: Do scrum-of-scrums really work?", in *Proceedings of the 2012 ACM-IEEE International Symposium on Empirical Software Engineering and Measurement*, Sep. 2012, pp. 235–238. DOI: 10.1145/2372251.2372294.

[7] R. M. Parizi, T. J. Gandomani, and M. Z. Nafchi, "Hidden facilitators of agile transition: Agile coaches and agile champions", in *2014 8th. Malaysian Software Engineering Conference (MySEC)*, Sep. 2014, pp. 246–250. DOI: 10.1109/MySec.2014.6986022.

[8] R. T. Hans, "Work in progress - the impact of the student scrum master on quality and delivery time on students' projects", in *2017 International Conference*

on Learning and Teaching in Computing and Engineering (LaTICE), Apr. 2017, pp. 87–90. DOI: 10.1109/LaTiCE.2017.22.