XXV Congresso de Iniciação Científica da Unicamp

October 18 to 20 Campinas | Brazil

2017

From Classical to Agile, Supporting the Transition: Agile Self-Assessment (ASA)

Matheus Florêncio*, Fernando Sambinelli, Marcos A. F. Borges

Abstract

Transformation from Classical Software Development to Agile Software Development is a long road. Assessment Methods that support this transformation have been created by the community academic and the industry. But they still present some gaps. It is what this research propose to solve.

Key words: *Agile Methodologies, Agile Manifesto, Continuous Improvement*

Introduction

Agile Software Development (ASD) changed the way software is developed. Countless researches deal with the subject since ASD was founded by Agile Manifesto (AM) in 2001 [1]. As increased the researches about ASD, people started to ask how to measure the transformation from classical development for ASD [2]. Thus was initiated the creation of evaluation methods (EMs) for ASD. These are: Comparative Agility (CA) [3], OPP Framework (OPP) [4], Sidky Agile Measurement Index (SAMI) [5], Thoughtworks Assessment Model (TAM) [6] and 4-D Framework (4-D) [7]. However, these methods present some alignment inconsistencies with Agile Principles (APs) [2]. Therefore, this research proposes a tool to solve these issues.

Results and Discussion

This research started with the review and analysis of EMs for ASD seeking alignment inconsistencies with APs. Thereafter, was conducted a literature review to build all criteria used in the Agile Self-Assessment (ASA). Soon was noticed that these EMs were inconsistent with some APs as also noticed in [2]. Therefore, was prototyped ASA that propose solve this issues. Below is discussed the inconsistencies and solutions for them.

Flexibility: At the analysis of EMs was observed that them obligate a team or organization (i.e applicator) to employ practices that are not appropriate in their context. The applicator need to employ all practices included in the EMs. Thus, the EMs becomes inflexible what is inconsistent to flexibility principle as also noticed by [2].

Simplicity: CA and SAMI exhibits a substantial number of questions to answer, 125 and 300 respectively. It could imply in a huge effort to the applicator. Therefore, could be induced that both are inconsistent with simplicity principle.

Technical excellence pursuit: In none of the analyzed EMs was verified an extension to create action plans and targets, both features of continuous improvement and PDCA [9]. Thus, could be induced that them are not totally aligned with technical excellence pursuit principle.

ASA proposes to solve flexibility inconsistencies setting a feature that enables a team to decide which practices employ. To solve the simplicity issue was chosen for ASA the OPP as base. OPP has less, but relevant, practices to be assessed. Also, to provide continuous improvement was created two features, one enables ASA applicator set targets and second enable creation of action plans.

ASA contains 25 practices. Each practice contains 1-3 criteria that help asses the practice extent in the applicator. Assessed, the measures generated by ASA exhibits alignment of the applicator with APs and Agile Values. Alignments and practices are provided by OPP. In the end a report present the current and target state, as also, the action plans to reach the target.

Conclusions

This research proposes to solve inconsistencies between EMs and APs. At the research was noticed inconsistencies in the flexibility, simplicity and technical excellence pursuit principles. These inconsistencies lead Agile Practitioners avoid the application of EMs [9].

ASA proposes to solve alignment inconsistencies between EMs and APs. ASA solves the flexibility inconsistency because leave the applicator to decide which practices assess, simplicity because reduce the number of points to assess and, technical excellence pursuit because enable continuous improvement through PDCA concept. ASA should be applied in case-studies and be developed as online platform as a future work.

¹ Dingsøyr, T., Nerur, S., Balijepally, V., Moe, N.B.: A decade of agile methodologies: Towards explaining agile software development. *J. Syst. Softw.* 85, 1213–1221 (2012).

² Nafchi, M.Z., Zulzalil, H., Gandomani, T.J.: On the current agile assessment methods and approaches. In: 8th. Malaysian Software Engineering Conference (MySEC). p. 251–254. IEEE, Langkawi, Malaysia (2014).

³ Williams, L., Rubin, K., Cohn, M.: Driving Process Improvement via Comparative Agility Assessment. In: 2010 Agile Conference. p. 3–10. IEEE, Nashville, TN, USA (2010).

⁴ Soundararajan, S., Arthur, J.D., Balci, O.: A Methodology for Assessing Agile Software Development Methods. In: Agile Conference (AGILE). p. 51–54. IEEE, Dallas, TX, USA (2012).

⁵ Sidky, A., Arthur, J., Bohner, S.: A disciplined approach to adopting agile practices: the agile adoption framework. *Innov. Syst. Softw. Eng.* 3, 203–216 (2007).

⁶ Thoughtworks: Thoughtworks Assessment Model, <http://www.agileassessments.com/>

⁷ Qumer, A., Henderson-Sellers, B.: An evaluation of the degree of agility in six agile methods and its applicability for method engineering. *Inf. Softw. Technol.* 50, 280–295 (2008)

⁸ Deming, E.: *Out of the Crisis*. (1986).

⁹ Fontana, R.M., Reinehr, S., Malucelli, A.: Maturing in Agile: What Is It About? In: *Agile Processes in Software Engineering and Extreme Programming. XP 2014*. p. 94–109. Lecture Notes in Business Information Processing, Springer, Cham (2014).