

A Preliminary Checklist for Capturing Baseline Situations in Studying the Impacts of Agile Practices Introduction

Indira Nurdiani

Blekinge Institute of Technology
SE-37179, Karlskrona, Sweden
indira.nurdiani@bth.se

Samuel A. Fricker

Blekinge Institute of Technology/ Fachhochschule
Nordwestschweiz
CH-5210 Windisch, Switzerland
samuel.fricker@fhnw.ch

Jürgen Börstler

Blekinge Institute of Technology
SE-37179, Karlskrona, Sweden
jurgen.borstler@bth.se

Kai Petersen

Blekinge Institute of Technology/ Hochschule Flensburg
SE-37179/ 24943 Flensburg
kai.petersen@bth.se

ABSTRACT

To assess the benefits of introducing Agile practices, it is important to get a clear understanding of the baseline situation, i.e. the situation before their introduction. Without a clear baseline, we cannot properly assess the extent of impacts, both positive and negative, of introducing Agile practices. This paper provides a preliminary guideline to help researchers in capturing and reporting baseline situations. The guideline has been developed through the study of literature and interviews with industry practitioners, and validated by experts in academia.

CCS CONCEPTS

• **Software and its engineering** → **Agile software development**;

KEYWORDS

Agile practices, checklist, baseline situation

ACM Reference Format:

Indira Nurdiani, Jürgen Börstler, Samuel A. Fricker, and Kai Petersen. 2018. A Preliminary Checklist for Capturing Baseline Situations in Studying the Impacts of Agile Practices Introduction. In *CESI'18: CESI'18/IEEE/ACM 6th International Workshop on Conducting Empirical Studies in Indus, May 28, 2018, Gothenburg, Sweden*. ACM, New York, NY, USA, 4 pages. <https://doi.org/10.1145/3193965.3193969>

1 INTRODUCTION

Agile practices are usually introduced to complement or replace existing processes and practices [6, 11, 12]. There are many empirical studies aimed at studying the benefits and limitations of introducing Agile practices [3]. However, these empirical studies report contradictory results on the impact of Agile practices [3, 8].

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

CESI'18, May 28, 2018, Gothenburg, Sweden

© 2018 Association for Computing Machinery.

ACM ISBN 978-1-4503-5736-4/18/05...\$15.00

<https://doi.org/10.1145/3193965.3193969>

It is often difficult to compare results across empirical studies because the baseline situation, i.e. situation prior to Agile practice introduction is not reported [11]. Describing the baseline situation is important to fully understand the extent of improvement [5], or the lack thereof, from introducing Agile practices.

Current guidelines do not detail which information should be captured when describing the baseline situation prior to introducing Agile practices. Potentially, all contextual information reported in the literature could be relevant. However, reporting “everything” may not be feasible as it is too effort-intensive to collect and report. In this position paper, we propose a checklist for baseline information that should be included when reporting the impacts of introducing Agile practices. The description of baseline information will make it easier to compare the effects of introducing Agile practices as a complement, or a replacement, to existing practices. The checklist has been validated by a number of experts in academia.

2 RELATED WORK

There are a number of checklists or guidelines to report contexts of empirical studies in software engineering. Kitchenham et al [5] suggest that the following contextual information is to be provided: (1) the objectives of the study, (2) the baseline that the results are compared, and (3) external constraints. However, the checklist does not detail what information needs to be captured to describe the baseline. Petersen and Wohlin proposed that contextual information pertaining to process, product, practices and tools, people, organization, and market are to be captured [10]. The checklist provides a general overview of contexts that need to be captured, but does not indicate, which information is relevant when comparing the adoption of agile practices with a particular baseline. Currently there is no guideline that supports researchers to capture the baseline situation when reporting the impacts of introducing Agile practices.

Scope of the guideline. We focus on information relevant to the internal constituents of a software development organization. Constituents are the components that make up a software organization, they are: (1) workforce – human resources and their skills and knowledge, (2) management and organization structure – deals with leadership, the organization hierarchy and roles, and coordination between the constituents, (3) process – concerted activities

that takes place in the organization, (4) infrastructure – technology platforms required to conceive, build, and release the product or service [8, 14]. The focus on internal constituents is selected because introducing Agile practices can affect these constituents [9].

3 CHECKLIST DEVELOPMENT AND VALIDATION

The checklist was developed from a literature study [9] and interviews with 11 industry practitioners. The resulting checklist was then validated with seven experts in academia, see Figure 1.

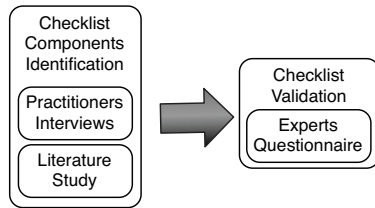


Figure 1: Overview of Research Methods

Practitioners Interviews. The interviews were done as a follow-up from a survey¹. In the survey, we inquired the respondents to indicate when (in month and year) Agile practices started being adopted, and if they are still in used. The follow up interviews inquired: (1) rationales for introducing the Agile practices at the time that they were introduced, (2) rationales for abandoning one or more practices, and (3) the impacts of adding, or abandoning Agile practices over time. From the interviews, we identified the potential components of the checklist, which was then complemented by the literature review [9].

Eleven practitioners with experience in Agile projects participated in the interviews from 11 different organizations from various domains such as financial services, telecommunication, biomedical research, etc. The roles of practitioners include developers, testers, project/ program managers, and Scrum masters. The interviewees also came from different countries, e.g., Sweden (4 interviewees), Indonesia (2 interviewees), Brazil, Ireland, Finland, USA, and Canada (each represented by one interviewee).

Literature review. The aim of the literature review was to identify flexibility attributes in inter-disciplinary literature and map them against Agile practices. The detailed description of the literature review process can be found in [9]. The literature review identified known techniques and principles in management and manufacturing that are well aligned with Agile practices and principles. This was used to complement the findings from the interviews.

Experts Questionnaire. We contacted experts in academia to validate the content of the checklist and obtain feedback for further improvements. The experts were selected given their publication activities in Agile methodology, or past and current participation as PC members in Agile conferences. We sent a summary of the checklist to the experts together with a short questionnaire, inquiring the experts' perception on the usefulness and relevance to research,

and novelty of the identified component. The questionnaire was self-administered. However, if the expert was available, a video call was also established. In total, we obtained feedback from seven experts.

4 A CHECKLIST TO CAPTURE BASELINE SITUATIONS

The checklist is structured according to the constituents described in Section 2. An overview of the checklist is shown in Table 1. In the following paragraphs, we described each component of the checklist.

Workforce. In manufacturing and management, hiring highly skilled individual is important to be flexible [9]. Four of our interviewees indicated that having good technical skills, product knowledge, and development experience is important for successful Agile practice adoption. One of them mentioned the lack of product knowledge led to the discontinuation of tracking progress. Another interviewee also mentioned that she provided training in estimation (non-technical skills) so the team members are able to do planning poker.

This raises the question of whether the constellation team members' competencies, experience, and skills could have an impact on the outcome from adopting Agile practices. If a large percentage of the team members are highly skilled with in-depth product knowledge, they are less likely to encounter problems with the introduction of Agile practices. This indicates the importance of capturing baseline context information on the **competencies, experience, and skills of the team members**. In the checklist (see Table 1) we recommend to capture the percentage (%) of team members given their levels of product knowledge and experience, and to enumerate technical and non-technical skills.

Management and Organization Structure. Management literature suggests a number of principles to be flexible, e.g., adopt a flat organization structure with self-organizing team, promote openness and recursiveness [9]. These principles are well aligned to Agile principles and practices. One interviewee indicated that the company owners exemplified openness by allowing the team to be self-organized. Meanwhile, another interviewee suggested that her team has always been self-organized even prior to Agile transformation.

This led to the question of whether the existing management policies and organization structure could have an impact on the outcome of adopting Agile practices. A team whose members are accustomed to be self-organized with a manager who promotes openness is less likely to develop friction with Agile way of working. Such a team will have better chance to succeed with Agile transformation, compared to a team that is not accustomed to being self-organized. This indicates the importance of capturing baseline context information pertaining to **management principles and organization structure** when studying the impacts of introducing Agile practices. In the checklist, we recommend different scales to indicate the information pertaining to existing management principles and organization structure.

Process. Six of our interviewees indicated that the practices that are often associated with Agile had existed prior to Agile transformation. These practices include face-to-face meeting (3

¹The survey design can be found in: rebrand.ly/bthagilesurvey2016

Table 1: Baseline Context Information Checklist

| To what extent can one judge the improvement from implementing Agile practices given the following baseline: | | Suggested scale |
|--|--|--|
| Workforce | 1. Constellation of team members' competencies, experience, and skills: 1a. Team members' product knowledge 1b. Software development experience 1c. Team members technical and non-technical skills | no/basic (%) - sufficient (%) - good (%) - in-depth (%) ^{a,b} immature (%) - little experience (%) - experienced (%) - very experienced (%) ^{a,b} enumerate (e.g., coding skills, project management skills, etc.) |
| Management and Organization structure | 2. Management principles that are in place: 2a. Organizational structure 2b. Self-organization 2c. Openness - willingness of managers to accept new ideas, or redefine roles 2d. Recursiveness - the ability of managers to switch between planning and implementation | hierarchical - flat/matrix Tuckman's stage of group development [15] Tannenbaum and Schmidt's leadership continuum [13] No - Yes |
| Process | 3. Existing practices or activities 3a. Formal/informal practices that are similar to Agile practices | enumerate (e.g., coding in pairs, improve internal code structure, etc.) |
| Infrastructure | 4. System characteristics: 4a. Existing legacy code 4b. Product components modularity 4c. Hardware component | No - Yes Monolithic - Modular (cohesion and coupling measures could be used, see [4]) Embedded systems - minimum hardware component - no hardware component |

^a The scale is adapted from [2].^b indicate the % of the team members within the respective level.

interviewees), pair programming (2 interviewees), coding standard (4 interviewees), refactoring (2 interviewees), short iteration & frequent releases, and self-organizing cross functional team (one interviewee each). For some interviewees, these practices were done as informal practices, i.e. only within the interviewees' teams or projects.

It is possible that Agile-like practices have existed in a software development team, either as formal or informal practices. This led to the question on whether the existing formal or informal practices could have an impact on the outcome of adopting Agile practices. A team that has had formal or informal practices similar to Agile has an advantage over those that do not because they may require less learning time and are used to Agile way of working. Furthermore, a number of our interviewees indicate that Agile practices need one another to work, like retrospective and continuous integration, or tracking progress and planning game. If a team has one of the practices already established prior to Agile transformation, it would be easier to introduce the other Agile practices. This indicates the importance of capturing baseline context information on **existing formal or informal practices** when studying the impacts of introducing Agile practices. In the checklist, we recommend to enumerate as many Agile-like practices as possible.

Infrastructure. Two interviewees indicate that improving the modularity of the product was needed to introduce some Agile practices, like continuous integration and Test-Driven Development (TDD). Two other interviewees indicate that legacy code prevented them to adopt some Agile practices, e.g., TDD and simple design.

With respect to hardware dependencies, two other interviewees mentioned that the hardware components in their development posed challenges. Primarily because the development cycle for the hardware was different from the software part. For example, hardware may not be ready for the start of the next sprint.

This led to the question whether the internal design of the product and hardware dependencies could have an impact on the outcome of adopting Agile practices. A team that maintains a well-designed, less complex and modular system is less likely to perform major refactoring to enable the introduction of some Agile practices. Hardware components may not prevent a team to adopt Agile, certainly it could pose challenges, like longer lead time. Thus, the extent of hardware components may influence the impacts (positive or negative) from adopting Agile practices. This indicates the importance of capturing base line context information on the existing **systems characteristics**. In the checklist, we recommend scales to indicate information pertaining to the product internal structure and hardware dependencies.

5 DISCUSSION

The individual components of the checklist are not new. It is known that team members skills and experience can impact the outcome of introducing pair programming [1]. Concerns pertaining to management support have been reported as a possible challenge when introducing Agile practices [7]. The experts whom we consulted also agree that the components are not new. However, the novelty lies in the perspective we put on the components, i.e. framing them

as baseline situation that needs to be captured when studying the impacts of introducing Agile practices. The experts also agree that aggregating the components of the checklist and making them explicit is where the novelty and usefulness lie. As we can reflect upon our own experience in conducting the interviews, the practitioners themselves were not aware of the baseline situation. We managed to identify the relevant information on the baseline situation by asking the interviewees to reflect on when Agile practices were introduced and why they were introduced at that particular time.

The issue of completeness was raised by a number of the experts. At the moment, the checklist is preliminary and we do not claim to have a complete list of information for capturing baseline situation. Similar to the checklist described in [10], this checklist is proposed to “raise awareness” on the importance of capturing baseline context information, and the possible overlap between them, a sentiment shared by some of the experts. For example, there could be an overlap between the state of product internal design and existing formal or informal practice; product modularity could be the by-product of the processes, e.g., refactoring, code review, etc. We also suggest to use this checklist to complement existing checklists, e.g., [5, 11].

A study may identify multiple agile-like states and organizations may have multiple baselines. A study may focus on two or more of them. We argue that at least two states need to be described with the support of the checklist to reason on effects. Effects may be value delivered to the customer, product quality, delivery time and cost, and customer relationships (see e.g. [12]). From a reporting point of view, we suggest to explicitly name the state/baseline, the baseline characteristics, and the desired/measured effects to make changes (deltas) explicit and traceable. Figure 2 provides an illustration for multiple states included in a study.

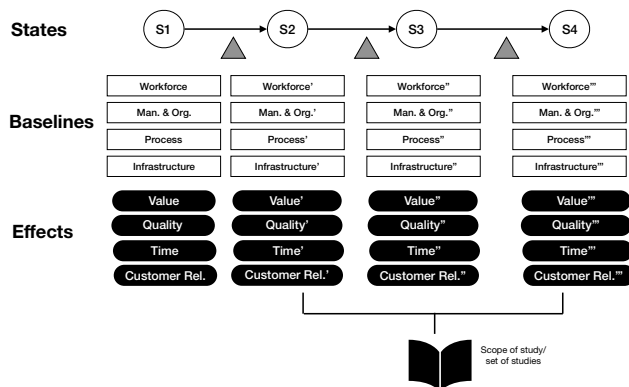


Figure 2: Checklist Usage

6 CONCLUSIONS

This paper proposes a preliminary checklist to capture baseline situations, particularly when reporting the impact of Agile practices introductions. The checklist was developed from a literature review and interviews with 11 industry practitioners. The checklist aggregates baseline information pertaining to the workforce,

management and organization structure, process, and infrastructure. The checklist could be used by other researchers to improve their study design. By reporting the baseline situation prior to the introduction of Agile practices, the readers can better understand the extent that circumstances of the organization contribute to the benefits and limitations of introducing Agile practices.

The checklist is not fully complete yet and will be extended with baseline information that could affect the introduction of Lean tools. Furthermore, currently, the checklist is only intended for empirical studies targeting Agile practices. The checklist should also be extended and customized so it could be used beyond studying the impacts of introducing Agile practices. The components of the checklist have been statically validated by seven experts in academia, but it requires further validation for its implementation in empirical studies.

ACKNOWLEDGMENTS

The authors would like to thank Prof. Jürgen Münch, Prof. Barbara Weber, Prof. Michael Felderer, Dr. Eric Knauss, Prof. Helena Holmström Olsson, Prof. Helen Sharp, and Dr. Marco Kuhrmann for their evaluations of the checklist components described in this paper.

REFERENCES

- [1] E. Arisholm, H. Gallis, T. Dybå, and D. I. K. Sjöberg. 2007. Evaluating Pair Programming with Respect to System Complexity and Programmer Expertise. *IEEE Transactions on Software Engineering* 33, 2 (Feb 2007), 65–86.
- [2] R. Britto, D. Smite, and L. O. Damm. 2016. Software Architects in Large-Scale Distributed Projects: An Ericsson Case Study. *IEEE Software* 33, 6 (Nov 2016), 48–55.
- [3] Tore Dybå and Torgeir Dingsøyr. 2008. Empirical Studies of Agile Software Development: A Systematic Review. *Information and Software Technology* 50, 91 (August 2008), 833 – 859.
- [4] Ronald Jabangwe, Jürgen Börstler, Darja Šmite, and Claes Wohlin. 2015. Empirical evidence on the link between object-oriented measures and external quality attributes: a systematic literature review. *Empirical Software Engineering* 20, 3 (01 Jun 2015), 640–693.
- [5] B. Kitchenham, L. Pickard, and S. L. Pfleeger. 1995. Case studies for method and tool evaluation. *IEEE Software* 12, 4 (Jul 1995), 52–62.
- [6] Marco Kuhrmann, Philipp Diebold, Jürgen Münch, Paolo Tell, Vahid Garousi, Michael Felderer, Kitija Trektore, Fergal McCaffery, Oliver Linssen, Eckhart Hanser, and Christian R. Prause. 2017. Hybrid Software and System Development in Practice: Waterfall, Scrum, and Beyond. In *Proceedings of the 2017 International Conference on Software and System Process (ICSSP 2017)*. 30–39.
- [7] Sridhar Nerur, Radha Kanta Mahapatra, and George Mangalaraj. 2005. Challenges of Migrating to Agile Methodologies. *Commun. ACM* 48, 5 (May 2005), 72–78.
- [8] Indira Nurdiani, Jürgen Börstler, and Samuel Fricker. 2016. The Impacts of Agile and Lean Practices on Project Constraints: A Tertiary Study. *Journal of Systems and Software* 119, Supplement C (2016), 162 – 183.
- [9] Indira Nurdiani, Jürgen Börstler, and Samuel Fricker. 2018. Literature Review of Flexibility Attributes: A Flexibility Framework for Software Developing Organization. *Journal of Software: Evolution and Process* (February 2018).
- [10] Kai Petersen and Claes Wohlin. 2009. Context in Industrial Software Engineering Research. In *Proceedings of the 2009 3rd International Symposium on Empirical Software Engineering and Measurement (ESEM '09)*. IEEE Computer Society, Washington, DC, USA, 401–404.
- [11] Kai Petersen and Claes Wohlin. 2010. The Effect of Moving From a Plan-driven to an Incremental Software Development Approach with Agile Practices. *Empirical Software Engineering* 15, 6 (2010), 654–693.
- [12] Adam Solinski and Kai Petersen. 2016. Prioritizing agile benefits and limitations in relation to practice usage. *Software Quality Journal* 24, 2 (2016), 447–482.
- [13] Robert Tannenbaum and Warren H Schmidt. 1973. *How to choose a leadership pattern*. Harvard Business Review Boston, MA.
- [14] Tommi Tapanainen, Mikko Hallanoro, Johannes Päiväranta, and Hannu Salmela. 2008. Towards an Agile IT Organisation: A Review of Prior Literature. In *Proceedings of the 2nd European Conference on Information Management and Evaluation (ECIME 2008)*. 425 – 432.
- [15] Bruce W Tuckman. 1965. Developmental sequence in small groups. *Psychological bulletin* 63, 6 (1965), 384.