Requirements Engineering for Large Scale Agile Systems Development[*]

Rashidah Kasauli

Chalmers | University of Gothenburg, Sweden
rashida@chalmers.se

Abstract. Agile practices such as continuous integration and delivery promise shorter time to market and improved quality. For this reason, such practices have been adopted in many software companies. In the context of systems development, additional constraints apply, e.g., as a result of scale or parallel development of hardware and software. Traditionally, stage-gate processes with a focus on up-front requirements analysis are common in large-scale systems engineering. However, long upfront analysis is considered anti agile and there is some friction between RE (which is often considered as a waterfall activity or phase) and agile practices. This paper describes a research agenda that plans to address this friction. The work proposes a series of empirical studies to discover the information needs and related knowledge, pertinent to product development. We expect to make a contribution by establishing guidelines and frameworks that can be used to make requirements engineering a foundation for agile systems development.

Keywords: agile requirements, requirements and continuous integration, systems engineering, software-intensive systems

1 Introduction

Large-scale systems development companies are operating in increasingly competitive and dynamic markets. These new business challenges call for a product development process which is able to respond faster to changing situations. For this reason, large systems development companies have adopted agile methods [13, 19], like scrum and XP, which offer the promise of faster time to market and improved quality [17].

Agile methods, which were originally designed for small teams, have been applied successfully but with challenges in large organizations [10]. To combat these challenges, suggestions for companies to build and tailor these methods to operate in other environments [8, 20] came up, while others suggested the combination of agile and traditional methods [11, 13, 15, 17].

At the same time Requirements Engineering (RE), which is necessary for project success [7], is seen as a traditional or plan-driven process [14]. This might be acceptable, since some claim that product development companies cannot be fully agile [9], as parts of the product development (e.g. hardware) have to remain mainly plan-driven. Although claimed beneficial, the role of RE practices in agile methods is still unclear to the software development community[7]. There is little work on how RE and agile methods interact in a software-intensive systems development environment.

Based on that, we start by exploring the area of RE with two large-scale software intensive companies that are using agile methods in their development process. We then derive, based on actual industry needs and related work, the specific research goals we have and derive a research agenda to address the friction between RE and agile methods.

2 Progress so far

As a starting point, using semi-structured interviews and workshops, we have explored the problem in one telecommunications company and one automotive company. We found that companies are using agile methods and want to adapt the new agile strategy of continuous delivery and deployment. Thus increasing the focus on value creation. In this situation, there is a pattern, where teams receive the high level requirements which they break down into user stories to be implemented in each sprint. However, there are some discrepancies that prevent the delivery of value. Through thematic analysis of the transcribed interviews, we identified several discrepancies. The following are the key results that relate to this thesis.

Gap between customer and developer: The agile methods advocate face-to-face communication between the customers and developers. However, that is practically impossible in the large-scale context as the customer base for large-scale companies is quite wide and distributed in most, if not all, cases. Developers complain that the levels between them and the customers are so many that there is so much value lost by the time the requirements get to them. Also developers complain that they are not given access to the requirements early enough, thus increasing the probability of misinterpreting the requirements. The earlier the programmers participate in requirements engineering process, the less the risk of developing the wrong product.

Information flow supporting agility: Since the direct communication between developer and customer is very complex, there is concern and therefore a need to address the way the information should flow to (and) or from and within the organization. To receive requirements, the levels within the organizations use varying tools or artifacts that cause a loss of value during conversion. Developers for instance use user stories as a way of communication among themselves which the product owners may not understand. After development, the customers are unable to try out the products in a short time for timely feedback. They still find a need to go through their long verification cycles. This could be because of the lack of avenues to give them an easy way to try out features early enough or the lack of documentation to guarantee quality of the release.

Managing documentation requirements: For software-intensive systems, the regulatory requirements are one of the primary factors. These regulations and policies, which guide the development of certain software components, require some sorts of documentation to back them up. Agile methods, however, do not provide for handling such requirements.

These challenges are all addressing the conflicts that agile ways of working have with the traditional RE practices. The far away customer as opposed to on-site customer for agile methods, the lack of emphasis on documentation as opposed to specific documents for knowledge sharing in the traditional ways all disturb the flow of information between stakeholders. So the general question for this thesis is: *In what way can RE be modified to complement the agile process of development?*

The remainder of the paper discusses related work, research aim, methodology and gives a conclusion.

3 Related Work

The applicability of agile methods in large-scale development of software intensive systems is challenging [6] as there are some misalignments between the agile methods and the already established requirements management processes [17]. This could be because the implementation of the original agile ideas is not practical in the context of large-scale companies which have already set up a traditional foundation.

RE is a process by which the services that the customer requires from a system and the constraints under which it operates are established. Traditionally, during the RE process, a requirements document was created for knowledge sharing [12]. In contrast, agile methods advocate for face-to-face communication [1] between stakeholders to reach a similar goal. Several studies [13, 15, 19] have, however, indicated the use a surrogate customer in the large-scale context which introduces requirements communication challenges [4, 11] that could also impact the quality of the product being developed [2]. Bjarnasson et al. [2] particularly pointed out low understanding of each others' roles and the scale of organization and product as some of the major causes of communication gaps.

To answer the question of scaling agile methods, Ronkainen and Abrahamsson [16] explored the possibility of using agile development techniques in a systems development environment and defined requirements for new agile methods targeted to facilitate the development of embedded software. They provide techniques to handle issues of documentation, software architecture, experimentation and test-driven development. These proposals were not tried out and there is no apparent research to show which techniques were developed. This research intends to expound on that and develop the techniques to enhance RE in the embedded systems domain.

To further try to understand agile methods in large-scale, more studies [3, 5] have been done. Eklund et al., for instance, [5] conducted a study in which they identified industrial challenges of scaling agile to embedded systems development. Among those identified was the length of the feedback loops between customers and management and handling of specific product requirements. These are consistent with our findings in Sec.2.

Despite the fact that works have come out to identify these challenges, not much has been done to address them. Some works [8, 20] have proposed modification of the agile methods to fit the environment but the actual processes or practices or frameworks for modification are not given.

4 Research Aim

The aim of this research is to design a framework that consists of RE techniques that are suitable for requirements engineers to enhance the value provided through handling documentation and regulations in software-intensive systems development while also supporting agile development, continuous integration, and continuous deployment.

4.1 Research Questions

As a starting point, we focus on the practices and processes in use by asking the following questions:

RQ A *What practices can be used to manage documentation without compromising the agile structure?*

Since documentation serves both internal for (effective information flow between stakeholders) and external purposes (quality assurance) to capture both we further break it down into:

RQ 1 *In what way can information flow be managed so that a project organization consistently maintains value and keeps a competitive edge in product?*

This can be broken down into other research questions:

RQ 1A *What processes ensure that all stakeholders get the same understanding of the requirements in order to deliver value?*

RQ 1B *What practices for managing user stories ensure that the stories contain strategic value for a given sprint?*

RQ 2 *How can agile methods be extended to cater for the quality (regulatory) requirements in software intensive systems?*

4.2 Expected Contributions

The research expects to contribute to the body of knowledge of tailoring RE and agile methods to work in the systems engineering context. The contribution will be through frameworks addressing how RE should be structured and executed to cope with the possibly unique challenges in embedded systems development. Focus will be specifically in regards to:

1. Requirements flow in the organization from customer to release — to help firms manage a unified understanding of value in the organization.
2. Documentation requirements handling — to help firms deal with the regulatory requirements in an agile setting.

5 Research Method

5.1 Reserch design and data collection

The research will follow the research approach proposed by Sein et al. [18]. Having identified the problems to address, the researcher will have extensive interviews, focus group meetings and workshops with the stakeholders in the participating companies. That will be followed with analysis of the results to devise the best framework that can work for the stakeholders.

This research will build on the foundation of related work, which have been gathered and synthesized in relation to RE [7] in the systems agile development environment [16] with special attention to providing a working solution at all levels of abstraction from the customer needs over system requirements, subsystem requirements, down to software requirements, their implementation and then up again through the different integration and testing steps while still working agile.

As Inayat et al. [7] recommended, *more empirical research is needed in agile RE to better understand the impact of the practices in dealing with quality and self-organizing teams*, the main research methods will be empirical in nature. Case studies will be used to conduct exploratory study of the current status of operation and find areas that require improvement. This exploratory study will be guided by the research questions. The units of analysis will be embedded systems software firms in the fields of telecommunications, automotive and heavy machinery. The research addresses challenges in the software development process and therefore we have identified the stakeholders in this research to be the developing teams, product owners or project managers and the testers.

Triangulation will be needed in order to establish and generalize the challenges from the different domains. Qualitative data will be derived from interviews with identified stakeholders in the different firms. A combination of in depth interviews with open ended questions, focus groups and workshops will be used when confronting the firm representatives. Document analysis will be another possible data source.

5.2 Validity

This research is part of a bigger project with many experienced researchers with varying backgrounds. We plan to have atleast two interviewers per session to reduce the risk of bias and misinterpretation. To ensure shared understanding of interview questions, we plan to start each interview with an introduction part in which we share our understanding of the subject with the interviewee. Anonymity will be guaranteed to the interviewees to ensure practitioners are not threatened to express their opinions. The results of the study may be limited to the case companies. However, we aim to expand our research to other companies dealing in different types of products. Triangulation will be used Each company's characteristics will be presented to the readers

6 Conclusion and Plan

It is important for RE to be done rightly for the projects to be successful. In the advent of agile practices like continuous integration and delivery, firms should be able to fully reap their benefits without completely getting rid of their old ways but to tailor their methods to fully support the agile ways of working. The approaches or frameworks to achieve that purpose are still lacking and need to be explored.

Currently the researcher is on the verge of completing her initial problem analysis and identification of challenges relating to agile development in the software-intensive system context and is planning for an empirical study to analyze the problem yet deeper through engagement with the companies. Once it has been done, the researcher will analyze the data collected and triangulate to devise a solution.

Acknowledgments. This research is funded by SIDA Bright project and partly supported by the Software Center (Large-Scale RE project) of Chalmers University of Technology, Sweden.

References

1. Beck, K., Beedle, M., Van Bennekum, A., Cockburn, A., Cunningham, W., Fowler, M., Grenning, J., Highsmith, J., Hunt, A., Jeffries, R., et al.: Manifesto for agile software development (2001)
2. Bjarnason, E., Wnuk, K., Regnell, B.: Requirements are slipping through the gapsa case study on causes & effects of communication gaps in large-scale software development. In: 2011 IEEE 19th international requirements engineering conference. pp. 37–46. IEEE (2011)
3. Daneva, M., Damian, D., Marchetto, A., Pastor, O.: Empirical research methodologies and studies in requirements engineering: How far did we come? Journal of systems and software 95, 1–9 (2014)
4. Dikert, K., Paasivaara, M., Lassenius, C.: Challenges and success factors for large-scale agile transformations: A systematic literature review. Journal of Systems and Software (2016)
5. Eklund, U., Olsson, H.H., Strøm, N.J.: Industrial challenges of scaling agile in mass-produced embedded systems. In: International Conference on Agile Software Development. pp. 30–42. Springer (2014)

6. Heikkila, V.T., Damian, D., Lassenius, C., Paasivaara, M.: A mapping study on requirements engineering in agile software development. In: 41st Euromicro Conf. on Softw. Eng. and Advanced Applications (SEAA '15). pp. 199–207 (2015)
7. Inayat, I., Salim, S.S., Marczak, S., Daneva, M., Shamshirband, S.: A systematic literature review on agile requirements engineering practices and challenges. Computers in human behavior 51, 915–929 (2015)
8. Kahkonen, T.: Agile methods for large organizations-building communities of practice. In: Agile Development Conference, 2004. pp. 2–10. IEEE (2004)
9. Karlström, D., Runeson, P.: Integrating agile software development into stage-gate managed product development. Empirical Softw. Eng. 11(2), 203–225 (2006)
10. Kettunen, P., Laanti, M.: Combining agile software projects and large-scale organizational agility. Software Process: Improvement and Practice 13(2), 183–193 (2008)
11. Laanti, M., Salo, O., Abrahamsson, P.: Agile methods rapidly replacing traditional methods at nokia: A survey of opinions on agile transformation. Information and Software Technology 53(3), 276–290 (2011)
12. Lucia, A.D., Qusef, A.: Requirements engineering in agile software development. Journal of Emerging Technologies in Web Intelligence 2(3), 212–220 (2010)
13. Olsson, H.H., Alahyari, H., Bosch, J.: Climbing the" stairway to heaven"–a mulitiple-case study exploring barriers in the transition from agile development towards continuous deployment of software. In: 2012 38th Euromicro Conference on Software Engineering and Advanced Applications. pp. 392–399. IEEE (2012)
14. Paetsch, F., Eberlein, A., Maurer, F.: Requirements engineering and agile software development. In: null. p. 308. IEEE (2003)
15. Rohunen, A., Rodriguez, P., Kuvaja, P., Krzanik, L., Markkula, J.: Approaches to agile adoption in large settings: a comparison of the results from a literature analysis and an industrial inventory. In: PROFES'10. pp. 77–91. Springer (2010)
16. Ronkainen, J., Abrahamsson, P.: Software development under stringent hardware constraints: do agile methods have a chance? In: XP'03. pp. 73–79. Springer (2003)
17. Savolainen, J., Kuusela, J., Vilavaara, A.: Transition to agile development-rediscovery of important requirements engineering practices. In: 2010 18th IEEE International Requirements Engineering Conference. pp. 289–294. IEEE (2010)
18. Sein, M.K., Henfridsson, O., Purao, S., Rossi, M., Lindgren, R.: Action design research. MIS quarterly pp. 37–56 (2011)
19. Sekitoleko, N., Evbota, F., Knauss, E., Sandberg, A., Chaudron, M., Olsson, H.H.: Technical dependency challenges in large-scale agile software development. In: International Conference on Agile Software Development. pp. 46–61. Springer (2014)
20. Sillitti, A., Succi, G.: Requirements engineering for agile methods. In: Engineering and Managing Software Requirements, pp. 309–326. Springer (2005)